Context-aware Neural Machine Translation with Mini-batch Embedding

Makoto Morishita[1,2], **Jun Suzuki**[2], **Tomoharu Iwata**[1], **Masaaki Nagata**[1]
NTT Communication Science Laboratories, NTT Corporation[1]
Tohoku University[2]
{makoto.morishita.gr, tomoharu.iwata.gy,
masaaki.nagata.et}@hco.ntt.co.jp
jun.suzuki@ecei.tohoku.ac.jp

Abstract

It is crucial to provide an inter-sentence context in Neural Machine Translation (NMT) models for higher-quality translation. With the aim of using a simple approach to incorporate inter-sentence information, we propose **m**ini-**b**atch **e**mbedding (MBE) as a way to represent the features of sentences in a mini-batch. We construct a mini-batch by choosing sentences from the same document, and thus the MBE is expected to have contextual information across sentences. Here, we incorporate MBE in an NMT model, and our experiments show that the proposed method consistently outperforms the translation capabilities of strong baselines and improves writing style or terminology to fit the document's context.[1]

1 Introduction

Current standard neural machine translation (NMT) models (Sutskever et al., 2014; Bahdanau et al., 2015; Luong et al., 2015; Vaswani et al., 2017) translate sentences in a sentence-by-sentence manner. However, some have argued that it is critical to consider the inter-sentence context in handling discourse phenomena (Hardmeier, 2012), which include coherence, cohesion, coreference (Bawden et al., 2018; Nagata and Morishita, 2020), and writing style (Yamagishi et al., 2016). To correctly translate these linguistic features, some works provide additional context information to an NMT model by concatenating the previous sentence (Tiedemann and Scherrer, 2017), applying a context encoder (Bawden et al., 2018; Miculicich et al., 2018; Voita et al., 2018), or using a cache-based network (Tu et al., 2018; Kuang et al., 2018).

Most of the previous studies have considered only a few previous context sentences. Several methods, such as the cache-based network, consider long-range context but heavily modify the standard NMT models and require additional training/decoding steps. Our goal is to make a simple but effective context-aware NMT model, which does not require heavy modification to standard NMT models and can handle a wider inter-sentence context. To this end, we propose a method to create an embedding that represents the contextual information of a document. To create this embedding, we focused on the mini-batch, which is commonly used in NMT training and decoding for efficient GPU computation. We modified the mini-batch creation algorithm to choose sentences from a single document and created an embedding that represents the features of the mini-batch. We call this embedding **mini-batch embedding (MBE)** and incorporate it in the NMT model to exploit contextual information across the sentences in the mini-batch.

Our main contributions can be summarized as follows: (i) We introduce mini-batch embedding to represent the features of sentences in a mini-batch. (ii) We incorporate mini-batch embedding in NMT to achieve simple context-aware translation and find that our approach improves translation performance by up to 1.9 BLEU points.

2 Neural Machine Translation

The current NMT model $f(\cdot)$ generates a sequence of target sentence tokens $\boldsymbol{y} = (y_1, \ldots, y_t)$ given a sequence of source sentence tokens $\boldsymbol{x} = (x_1, \ldots, x_s)$: $\boldsymbol{y} = f(\boldsymbol{x}; \theta)$, where θ is a set of model parameters and s and t are the numbers of source and target sentence tokens. The model parameters are trained by minimizing the loss function:

$$\mathcal{L}_{\mathrm{NMT}}(\theta) = -\sum_{(\boldsymbol{x},\boldsymbol{y})\in\mathbb{D}} \log P(\boldsymbol{y}|\boldsymbol{x};\theta), \quad (1)$$

[1]Our implementation is publicly available: https://github.com/nttcslab-nlp/mbe-nmt

Proceedings of the 16th Conference of the European Chapter of the Association for Computational Linguistics, pages 2513–2521
April 19 - 23, 2021.

where $\mathbb{D}$ is a set of bilingual sentence pairs. Since the model only uses a single sentence as its input, it does not consider the inter-sentence context.

3 Context-aware NMT with Mini-batch Embedding

To exploit the inter-sentence context in NMT with a simple modification, we propose **m**ini-**b**atch **e**mbedding (MBE) to represent the features of sentences in the mini-batch. Figure 1 shows an overview of how we create mini-batch embedding and incorporate it in the NMT model.

3.1 Mini-batch Embedding

Let $\boldsymbol{B} = \{(\boldsymbol{x}_1, \boldsymbol{y}_1), \ldots, (\boldsymbol{x}_n, \boldsymbol{y}_n)\}$ be a mini-batch, where $(\boldsymbol{x}_i, \boldsymbol{y}_i)$ is a pair of source/target sentences. Normally, we randomly select them from all of the training data to create a mini-batch $\boldsymbol{B}$. However, for our method, we choose sentence pairs from the same document to create a mini-batch.

Let $g_{\text{enc}}(\cdot)$ be a single Transformer encoder layer. We first compute sentence-wise contextualized embeddings $\boldsymbol{E}_i = (\boldsymbol{e}_{i,1}, \ldots, \boldsymbol{e}_{i,s_i})$ as $\boldsymbol{E}_i = g_{\text{enc}}(\boldsymbol{x}_i; \phi)$, where s_i is the number of tokens in $\boldsymbol{x}_i$ and ϕ are the model parameters. MBE $\boldsymbol{z} \in \mathbb{R}^e$ is computed:

$$\boldsymbol{z} = \frac{1}{n}\sum_{i=1}^{n} \boldsymbol{v}_i, \quad \boldsymbol{v}_i = \frac{1}{s_i}\sum_{j=1}^{s_i} \boldsymbol{e}_{i,j}, \tag{2}$$

where e is a hidden dimension of the NMT model. We use mean pooling[2] to make both sentence embeddings $\boldsymbol{v}_i$ and MBE $\boldsymbol{z}$. By adopting this procedure, we expect MBE $\boldsymbol{z}$ to have inter-sentence context features, which is desirable for a context-aware NMT.

Note that we ignore the order of sentences in a document. This is a beneficial trait because this method is also applicable to corpora with document boundaries but without in-document sentence order, such as ParaCrawl (Esplà et al., 2019).

3.2 Learning NMT with Mini-batch Embedding

To use inter-sentence information, we modify the NMT model by adding MBE to the input:

$$\boldsymbol{y} = f(\boldsymbol{x}, \boldsymbol{z}; \theta). \tag{3}$$

[2]This approach was inspired by Reimers and Gurevych (2019), who successfully created sentence embeddings from BERT embeddings (Devlin et al., 2019) by mean pooling.

We concatenated MBE to the input word embeddings, and the model uses MBE as the first input token (Fig. 1). Now the encoder/decoder takes $s+1$ and $t+1$ embeddings.

The Transformer encoder layer for MBE was jointly trained with the NMT model by modifying the loss function in Eq. (1):

$$\mathcal{L}_{\text{NMT}}(\theta, \phi) = -\sum_{\boldsymbol{B} \in \mathbb{D}'} \sum_{(\boldsymbol{x}, \boldsymbol{y}) \in \boldsymbol{B}} \log P(\boldsymbol{y}|\boldsymbol{x}, \boldsymbol{B}; \theta, \phi), \tag{4}$$

where $\mathbb{D}'$ is a set of mini-batches created from $\mathbb{D}$.

3.3 Mini-batch Embedding Gate

The MBE may degrade the translation performance when the NMT model does not need any context information to translate the mini-batch or the MBE fails to contain important information for translation. To deal with such cases, we aim to make the model estimate how important MBE is for each mini-batch. Thus we added a mini-batch embedding gate to determine MBE's importance.

In this setting, we prepared two types of mini-batches for training: (i) sentences from the same document and (ii) sentences from different documents. Then we trained a binary classifier that predicts whether the sentences in the mini-batch are selected from the same document:

$$P(d|\boldsymbol{z}) = \text{softmax}(\boldsymbol{W}\boldsymbol{z}), \tag{5}$$

where $\boldsymbol{W} \in \mathbb{R}^{2\times e}$ is a parameter matrix and d is a binary value that takes 1 if the sentences in the mini-batch are selected from the same document.

To train the classifier, we minimize the loss function:

$$\mathcal{L}_{\text{MB}}(\psi) = -\sum_{(d, \boldsymbol{B}) \in \mathbb{D}'} \log P(d|\boldsymbol{B}; \psi), \tag{6}$$

where ψ is a set of parameters for the classifier. For training, we mix the two types of mini-batches at the same ratio.

Concretely, we jointly minimize the NMT and the classifier loss functions:

$$\mathcal{L}(\theta, \phi, \psi) = \mathcal{L}_{\text{NMT}}(\theta, \phi) + \lambda \mathcal{L}_{\text{MB}}(\psi), \tag{7}$$

where λ is a hyperparameter used to control the weight of the classifier loss. We use the value predicted by the classifier as a gate. Our new weighted MBE is

$$\tilde{\boldsymbol{z}} = \alpha \boldsymbol{z}, \tag{8}$$

where $\alpha = P(d = 1|\boldsymbol{z})$, and we change $\boldsymbol{z}$ in Eqs. (3) to $\tilde{\boldsymbol{z}}$.

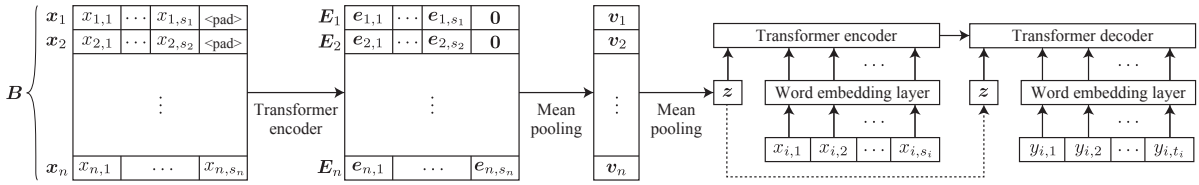

Figure 1: Overview of context-aware NMT with mini-batch embedding. $\boldsymbol{x}_i$ is a sequence of source tokens, $\boldsymbol{B}$ is a mini-batch that has n sentences, $\boldsymbol{E}_i$ is sentence-wise contextualized embeddings computed by a Transformer encoder, $\boldsymbol{v}_i$ is a sentence vector, and $\boldsymbol{z}$ is mini-batch embedding. We pad short sentences with a special <pad> token to adjust their length to the longest sentence in the mini-batch.

4 Experiments

4.1 Compared Models

We used four settings as baselines:

Baseline 6 Enc-Layers is the original Transformer NMT model with six encoder-decoder layers.

Baseline 7 Enc-Layers resembles Baseline 6 Enc-Layers, but the number of encoder layers was changed to seven. Since our MBE model requires an additional Transformer encoder layer, this model has a comparable number of parameters as the following MBE models.

2-to-1 is the context-aware translation model proposed by Tiedemann and Scherrer (2017) that translates a pair of previous and current source sentences into a target sentence. Two source sentences are concatenated with a special sentence boundary token. This method is known as a strong baseline for context-aware NMT (Bawden et al., 2018; Voita et al., 2018). Other settings are identical to those of Baseline 6 Enc-Layers[3].

DocRepair is another recent context-aware translation model, which uses two-step decoding (Voita et al., 2019). The first step generates 1-best translation with a sentence-level NMT model given a single sentence. The second step generates document-level translation given 1-best translations of four consecutive sentences concatenated with a special token.

We compared our proposed methods with the following settings:

MBE Enc resembles Baseline 6 Enc-Layers but uses MBE in the encoder.

MBE Enc w/o Gate resembles MBE Enc, but it does not use the MBE gate described in Section 3.3.

MBE Dec uses MBE in the decoder.

MBE Enc/Dec uses MBE in both the encoder and the decoder.

4.2 Experimental Settings

Datasets/Evaluation We trained Japanese-English NMT models. As training data, we used the JParaCrawl corpus (Morishita et al., 2020). JParaCrawl was created by crawling the web and aligning parallel sentences, and each sentence-pair has a URL from which the sentences were taken. In this experiment, we regarded the sentences from the same URL as a document.

We used several test sets with document boundaries: (i) scientific paper excerpts (ASPEC (Nakazawa et al., 2016)), (ii) news (newsdev2020 from WMT20 news translation shared task[4]), and (iii) TED talks (tst2012 from IWSLT translation shared task (Cettolo et al., 2012)). As a dev set to tune the NMT model, we used the ASPEC dev split. See Section A.1 in the Appendix for corpus statistics and detailed preprocessing steps.

To evaluate the translation performance, we used `sacreBLEU`[5] (Post, 2018) and report the BLEU scores (Papineni et al., 2002).

Model Configurations We used the Transformer model as an NMT model (Vaswani et al., 2017). Our hyperparameters were based on the "big" settings defined by Vaswani et al. (2017). For the MBE experiments, we set λ in Eq. (7) to 1.0. We set the mini-batch size to 3,000 tokens. If the tokens in a document were larger than this size, we

[3]Since our training data have document boundaries but the in-document sentence orders were shuffled, we randomly selected one in-document sentence and used it as previous context. For dev/test sets, we used the original sentence order.

[4]We used newsdev2020 as a test set because no official test set for English-Japanese was available at the time of writing. Since we did not use newsdev2020 for tuning the model, there is no problem with using it as a test set.

[5]The signature is `BLEU+case.mixed+lang.en-ja +numrefs.1+smooth.exp+tok.ja-mecab-0.996-IPA+version.1.4.9`

Model	ASPEC	WMT	IWSLT
Baseline 6 Enc-Layers (Vaswani et al., 2017)	26.2	18.4	12.0
Baseline 7 Enc-Layers (Vaswani et al., 2017)	26.9 (+0.7)	18.7 (+0.3)	11.9 (−0.1)
2-to-1 (Tiedemann and Scherrer, 2017)	27.0 (+0.8)	19.2 (+0.8)	12.9 (+0.9)
DocRepair (Voita et al., 2019)	27.9 (+1.7)	19.3 (+0.9)	12.3 (+0.3)
MBE Enc	28.0 (+1.8)	19.9 (+1.5)	12.2 (+0.2)
MBE Enc w/o Gate	28.0 (+1.8)	19.4 (+1.0)	13.0 (+1.0)
MBE Dec	**28.1 (+1.9)**	19.9 (+1.5)	**13.8 (+1.8)**
MBE Enc/Dec	**28.1 (+1.9)**	**20.0 (+1.6)**	13.4 (+1.4)

Table 1: BLEU scores for test sets: Values in brackets show score differences to "Baseline 6 Enc-Layers" model. The highest score in each test set is highlighted in bold.

split the document into several mini-batches[6]. If the tokens in a document are smaller, we put all tokens into a single mini-batch[7]. See Section A.2 in the Appendix for detailed hyperparameters and training settings.

4.3 Experimental Results and Analysis

Translation Performance Table 1 summarizes the model performance on several test sets. See Table 3 in the Appendix for the dev set performance. The results show that the scores of the proposed methods surpass the baseline as well as the stronger baselines that used seven encoder layers or the existing context-aware models.

Translation Examples Figure 2 shows an example translation of a sentence from the scientific paper excerpts (ASPEC test set). In this example, the word "mentions" is translated in two ways. The baseline system translated the word as "言及しています", which is a colloquial expression. In contrast, the proposed method translated it as "述べる", which suits usage in scientific papers. This shows that MBE could change the writing style to one that is more appropriate for scientific papers compared to the baseline.

Figure 3 shows another example, which is from TED talks (tst2012). This example shows how our model could change the translation of the word "you". Our method translated this word as "君", which is friendlier than the baseline output "あなた". In this document, "he" is a friendly old man, and thus the MBE output is more appropriate for this context.

These examples show that our method improved the writing style to fit the context and chose the appropriate word for the context. This indicates that MBE helped the NMT model by providing context information across the mini-batch.

Effect of Decoding Batch-size In the previous section, we discussed the translation performance given a document, which means that the sentences in the entire document are in a mini-batch. However, in practice, we sometimes have to translate a part of the document. To check the robustness of the model in such situations, we decoded the test set by limiting the number of sentences in a mini-batch.

Figure 4 shows the experimental results. The baseline model scores are identical to those in Table 1, since the model is immune to mini-batch size. Our MBE models achieve better performance when given a larger context. It reach comparable or better scores than the baseline model when given a single sentence or a smaller context. However, the model without using MBE gate (MBE Enc w/o Gate) showed a drastic drop in performance when translating a single sentence. This shows that the gate properly works to weigh the importance of MBE and improve performance.

5 Related Work

Context-aware NMT Tiedemann and Scherrer (2017) proposed a 2-to-1 (or 2-to-2) method that concatenates two source sentences and generates one (or two) target sentences. This is a simple model, but it only considers a previous sentence, while our method can make use of larger contexts. Junczys-Dowmunt (2019) extended the 2-to-2 method to document-to-document by concatenating all sentences in a document. Although they showed that the method is effective, it requires heavy computational cost since the NMT model

[6]We sorted the sentences in a document by their length when splitting the document into several mini-batches to maintain the training efficiency. Since the method focuses more on writing style and wording, we do not keep the original order of the sentences.

[7]In this case, the mini-batch size could be smaller than 3,000 tokens, since we did not want to mix up the sentences from different documents.

Source	The paper mentions the reliability assurance test and application technologies.
Reference	信頼性保証試験と適用技術を述べた
Baseline 6 Enc-Layers	この論文では、信頼性保証テストとアプリケーション技術について言及しています。
MBE Enc/Dec	本論文では,信頼性保証試験と応用技術について述べる.

Figure 2: Example translation of a sentence from scientific paper excerpts (ASPEC test set).

Source	He said, “I’m so proud of you.”
Reference	「君をすごく誇りに思うよ」
Baseline 6 Enc-Layers	彼は、 “私はあなたをとても誇りに思います” と言いました。
MBE Enc/Dec	彼は「君をとても誇りに思う」と言いました。

Figure 3: Example translation of a sentence from TED talks (tst2012).

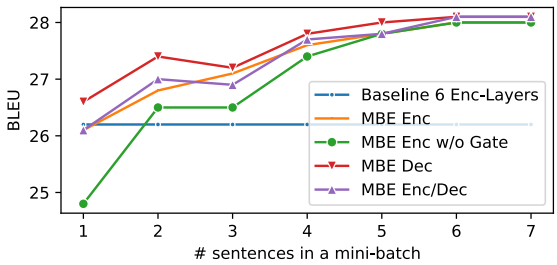

Figure 4: Relationship between the number of sentences in a mini-batch and BLEU scores on ASPEC test set.

has to process very long context. Miculicich et al. (2018) proposed a model that uses a hierarchical attention network to use previous context embeddings. However, their work can only use a few previous sentences as context, in contrast to our work that can use a larger context. Tu et al. (2018) and Kuang et al. (2018) proposed a cache-based approach to store longer context, while our work uses a much simpler architecture. Voita et al. (2019) proposed a method called DocRepair, one of the most recent context-aware NMT methods, that employs two decoding steps. It first translates a sentence by sentence-level NMT, and then the concatenated output is fed to a document-level model that outputs document-level translation. Although this is a promising method, it requires training of three sequence-to-sequence models to translate a single direction and needs two decoding steps, which slows down the translation. Our method has an advantage in that it only trains a single model and uses single-step decoding, which requires only a small computational cost.

NMT with Tags We used an MBE as the first input of the encoder/decoder. Our approach is similar to the work that uses special tags to control or provide additional information to NMT (Johnson et al., 2016; Takeno et al., 2017; Caswell et al., 2019). Johnson et al. (2016) added tags to a source sentence for indicating the target language in multilingual NMT models. Takeno et al. (2017) proposed a method that controls the target length or the domain by adding a tag to the decoder inputs. Caswell et al. (2019) used a tag to indicate the synthetic corpus (Sennrich et al., 2016). Our work, which automatically generates a tag (MBE) with the sentence in a mini-batch and uses a gate to control the importance of MBE, is different from the previous studies.

6 Conclusion

We proposed mini-batch embedding (MBE), which is a simple but effective method to represent contextual information across documents. We incorporated MBE in the NMT model, which enabled it to outperform competitive baselines. We found that our NMT model could choose the appropriate word and writing style to match the document context. An analysis showed that our model’s performance improves with a large context, but it still achieves comparable or even better performance than that of the baseline when translating a single sentence. Our future work includes applying MBE to other applications and improving the method to generate embeddings from a mini-batch.

Acknowledgments

We thank the four anonymous reviewers for their insightful suggestions.

References

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 (ICLR)*.

Rachel Bawden, Rico Sennrich, Alexandra Birch, and Barry Haddow. 2018. Evaluating discourse phenomena in neural machine translation. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (NAACL HLT)*, pages 1304–1313.

Isaac Caswell, Ciprian Chelba, and David Grangier. 2019. Tagged back-translation. In *Proceedings of the 4th Conference on Machine Translation (WMT)*, pages 53–63.

Mauro Cettolo, Christian Girardi, and Marcello Federico. 2012. WIT3: web inventory of transcribed and translated talks. In *Proceedings of the 16th Annual Conference of the European Association for Machine Translation (EAMT)*, pages 261–268.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (NAACL HLT)*.

Miquel Esplà, Mikel Forcada, Gema Ramírez-Sánchez, and Hieu Hoang. 2019. ParaCrawl: Web-scale parallel corpora for the languages of the EU. In *Proceedings of Machine Translation Summit XVII Volume 2: Translator, Project and User Tracks*, pages 118–119.

Christian Hardmeier. 2012. Discourse in statistical machine translation. a survey and a case study. *Discours. Revue de linguistique, psycholinguistique et informatique. A journal of linguistics, psycholinguistics and computational linguistics*, (11).

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. 2016. Google's multilingual neural machine translation system: Enabling zero-shot translation. *arXiv preprint arXiv:1611.04558*.

Marcin Junczys-Dowmunt. 2019. Microsoft translator at WMT 2019: Towards large-scale document-level neural machine translation. In *Proceedings of the 4th Conference on Machine Translation (WMT)*, pages 225–233.

Diederik Kingma and Jimmy Ba. 2015. Adam: A method for stochastic optimization. In *Proceedings of the 3rd International Conference on Learning Representations (ICLR)*.

Shaohui Kuang, Deyi Xiong, Weihua Luo, and Guodong Zhou. 2018. Modeling coherence for neural machine translation with dynamic and topic caches. In *Proceedings of the 27th International Conference on Computational Linguistics (COLING)*.

Taku Kudo. 2018. Subword regularization: Improving neural network translation models with multiple subword candidates. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (ACL)*, pages 66–75.

Taku Kudo and John Richardson. 2018. SentencePiece: A simple and language independent subword tokenizer and detokenizer for neural text processing. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 66–71.

Minh-Thang Luong, Hieu Pham, and Christopher D. Manning. 2015. Effective approaches to attention-based neural machine translation. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 1412–1421.

Paulius Micikevicius, Sharan Narang, Jonah Alben, Gregory F. Diamos, Erich Elsen, David García, Boris Ginsburg, Michael Houston, Oleksii Kuchaiev, Ganesh Venkatesh, and Hao Wu. 2018. Mixed precision training. In *Proceedings of the 6th International Conference on Learning Representations (ICLR)*.

Lesly Miculicich, Dhananjay Ram, Nikolaos Pappas, and James Henderson. 2018. Document-level neural machine translation with hierarchical attention networks. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 2947–2954.

Makoto Morishita, Jun Suzuki, and Masaaki Nagata. 2020. JParaCrawl: A large scale web-based English-Japanese parallel corpus. In *Proceedings of the 12th International Conference on Language Resources and Evaluation (LREC)*, pages 3603–3609.

Masaaki Nagata and Makoto Morishita. 2020. A test set for discourse translation from japanese to english. In *Proceedings of the 12th International Conference on Language Resources and Evaluation (LREC)*, pages 3704–3709.

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 10th International Conference on Language Resources and Evaluation (LREC)*.

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: Human Language Technologies (NAACL HLT), pages 48–53.

Myle Ott, Sergey Edunov, David Grangier, and Michael Auli. 2018. Scaling neural machine translation. In *Proceedings of the 3rd Conference on Machine Translation (WMT)*, pages 1–9.

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 (ACL)*, pages 311–318.

Matt Post. 2018. A call for clarity in reporting BLEU scores. In *Proceedings of the 3rd Conference on Machine Translation (WMT)*, pages 186–191.

Nils Reimers and Iryna Gurevych. 2019. Sentence-BERT: Sentence embeddings using Siamese BERT-networks. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 3982–3992.

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 (ACL)*, pages 86–96.

Nitish Srivastava, Geoffrey Hinton, Alex Krizhevsky, Ilya Sutskever, and Ruslan Salakhutdinov. 2014. Dropout: A simple way to prevent neural networks from overfitting. *Journal of Machine Learning Research*, 15:1929–1958.

Ilya Sutskever, Oriol Vinyals, and Quoc V. Le. 2014. Sequence to sequence learning with neural networks. In *Proceedings of the 28th Annual Conference on Neural Information Processing Systems (NeurIPS)*, pages 3104–3112.

Shunsuke Takeno, Masaaki Nagata, and Kazuhide Yamamoto. 2017. Controlling target features in neural machine translation via prefix constraints. In *Proceedings of the 4th Workshop on Asian Translation (WAT)*, pages 55–63.

Jörg Tiedemann and Yves Scherrer. 2017. Neural machine translation with extended context. In *Proceedings of the 3rd Workshop on Discourse in Machine Translation*, pages 82–92.

Zhaopeng Tu, Yang Liu, Shuming Shi, and Tong Zhang. 2018. Learning to remember translation history with a continuous cache. *Transactions of the Association for Computational Linguistics (TACL)*, 6:407–420.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Lukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. In *Proceedings of the 31st Annual Conference on Neural Information Processing Systems (NeurIPS)*, pages 6000–6010.

Elena Voita, Rico Sennrich, and Ivan Titov. 2019. Context-aware monolingual repair for neural machine translation. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 877–886.

Elena Voita, Pavel Serdyukov, Rico Sennrich, and Ivan Titov. 2018. Context-aware neural machine translation learns anaphora resolution. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (ACL)*, pages 1264–1274.

Hayahide Yamagishi, Shin Kanouchi, Takayuki Sato, and Mamoru Komachi. 2016. Controlling the voice of a sentence in Japanese-to-English neural machine translation. In *Proceedings of the 3rd Workshop on Asian Translation (WAT)*, pages 203–210.

Data	Usage	Sentences	Documents
JParaCrawl v2.0	train	10,120,013	24,156
ASPEC (dev)	dev	1,790	400
ASPEC (test)	test	1,812	400
WMT (newsdev2020)	test	1,998	140
IWSLT (tst2012)	test	1,670	15

Table 2: Number of sentences and documents in train/dev/test sets

A Detailed Experimental Settings

In this section, we describe more detailed experimental settings.

A.1 Data/Evaluation

The number of sentences and documents contained in the train/dev/test sets are shown in Table 2. We tokenized the sentences into subwords with `sentencepiece` (Kudo, 2018; Kudo and Richardson, 2018) and set the vocabulary size to 32k for each language. For the training set, we removed sentences whose length exceeded 250 subword tokens. For the DocRepair method, we used the JParaCrawl corpus as data for both monolingual and bilingual document-level application.

A.2 Model Configurations

We used the Transformer model as an NMT model (Vaswani et al., 2017). Our hyperparameters are based on "big" settings defined by Vaswani et al. (2017) and have six encoder/decoder layers, 16 attention heads, and 1,024 dimensions for all of the hidden states except the feed-forward network hidden states that have 4,096 dimensions. We used a dropout with a probability of 0.3 (Srivastava et al., 2014). As an optimizer, we used Adam with $\alpha = 0.001$, $\beta_1 = 0.9$, and $\beta_2 = 0.98$ (Kingma and Ba, 2015). A root-square decay learning rate schedule was used with a linear warm-up of 4,000 steps (Vaswani et al., 2017). We clipped the gradients to avoid exceeding their norm of 1.0. For the MBE experiments, we set λ in Eq. (7) to 1.0 and set the per-GPU-batch-size to 3,000 tokens. Since large-batch training can reduce training time (Ott et al., 2018), we accumulated about 280k tokens for an update. Based on dev set perplexity, we trained the model for 24,000 iterations. We saved the model every 200 iterations and averaged the last eight model parameters for decoding. We normalized the candidate translation scores by dividing their length and carried out a beam search with a size of six. Our implementation is based on `fairseq` (Ott et al., 2019). We used mixed-precision training (Micikevicius et al., 2018) to reduce memory consumption and training time. All experiments were run on eight NVIDIA Tesla V100 GPUs with 32-GB memory. Since we did not conduct a hyperparameter search, almost all of the settings were borrowed from (Morishita et al., 2020).

DocRepair requires the training of three sequence-to-sequence models: (1) an NMT model that translates language X to Y; (2) an NMT model that translates in reverse direction to make round-trip translation; and (3) a sequence-to-sequence model that converts 1-best translations to document-level translation. We used "Baseline 7 Enc-Layers" models for both (1) and (2), and newly trained the Transformer model for (3).

B Additional Experimental Results

Table 3 shows the number of parameters for each model, training speed, and BLEU scores on the dev set. The scores show the same tendency as the test set (Table 1).

The DocRepair method requires two translation models (English-to-Japanese and Single-to-Document), and thus the number of model parameters is larger than that for the other models. Although it also requires a Japanese-to-English translation model for creating round-trip translation data for training, these model parameters are not included in the table.

Since our MBE implementation was still in the experimental phase, the training speed was slower than that of the baselines, which were fully optimized by `fairseq` developers. We can further improve our implementation for faster computation, but we leave this for future work.

C Links to Data and Software

C.1 Data

JParaCrawl `https://www.kecl.ntt.co.jp/icl/lirg/jparacrawl/`

ASPEC `http://orchid.kuee.kyoto-u.ac.jp/ASPEC/`

newsdev2020 `http://www.statmt.org/wmt20/translation-task.html`

tst2012 `https://wit3.fbk.eu/`

C.2 Software

fairseq `https://github.com/pytorch/fairseq`

Model	Parameters	wps	hours for training	BLEU (ASPEC dev)
Baseline 6 Enc-Layers (Vaswani et al., 2017)	274M	187k	9.7	26.7
Baseline 7 Enc-Layers (Vaswani et al., 2017)	287M	167k	10.2	27.2 (+0.5)
2-to-1 (Tiedemann and Scherrer, 2017)	274M	125k	15.2	28.1 (+1.4)
DocRepair (Voita et al., 2019)	555M	236k	26.8	27.3 (+0.6)
MBE Enc	287M	93k	21.1	27.9 (+1.2)
MBE Enc w/o Gate	287M	82k	24.2	27.4 (+0.7)
MBE Dec	287M	93k	21.0	28.0 (+1.3)
MBE Enc/Dec	287M	92k	21.3	28.3 (+1.6)

Table 3: Number of parameters, training speed (words per sec, wps), required hours for training, and BLEU scores for the dev set.

sacreBLEU https://github.com/mjpost/sacreBLEU

sentencepiece https://github.com/google/sentencepiece