Improving Context-Aware Neural Machine Translation with Source-side Monolingual Documents

Linqing Chen[1] , **Junhui Li**[1*] , **Zhengxian Gong**[1] , **Xiangyu Duan**[1] , **Boxing Chen**[2] ,
Weihua Luo[2] , **Min Zhang**[1] and **Guodong Zhou**[1]
[1]School of Computer Science and Technology, Soochow University, Suzhou, China
[2]Alibaba DAMO Academy
{lijunhui, zhxgong, xiangyuduan, minzhang, gdzhou}@suda.edu.cn,
linqingchen21@gmail.com, {boxing.cbx, weihua.luowh}@alibaba-inc.com

Abstract

Document context-aware neural machine translation (NMT) remains challenging due to the lack of large-scale document parallel corpora. To make full use of source-side monolingual documents for context-aware NMT, we propose a **P**re-training approach with **G**lobal **C**ontext (PGC). In particular, we first propose a novel self-supervised pre-training task, which contains two training objectives: (1) reconstructing the original sentence from a corrupted version; (2) generating a gap sentence from its left and right neighbouring sentences. Then we design a universal model for PGC which consists of a global context encoder, a sentence encoder and a decoder, with similar architecture to typical context-aware NMT models. We evaluate the effectiveness and generality of our pre-trained PGC model by adapting it to various downstream context-aware NMT models. Detailed experimentation on four different translation tasks demonstrates that our PGC approach significantly improves the translation performance of context-aware NMT. For example, based on the state-of-the-art SAN model, we achieve an averaged improvement of 1.85 BLEU scores and 1.59 Meteor scores on the four translation tasks.

1 Introduction

Document context-aware machine translation aims at translating each sentence in a document under the guidance of the global context, with expectations to obtain more coherent and less ambiguous translations. Due to the availability of document-level parallel datasets, recent years have witnessed great progress in context-aware neural machine translation (NMT) with extensive attempts at leveraging document-level context, from sentence concatenation [Tiedemann and Scherrer, 2017], and mechanisms with multiple encoders [Jean *et al.*, 2017; Wang *et al.*, 2017; Zhang *et al.*, 2018; Bawden *et al.*, 2018; Voita *et al.*, 2018; Miculicich *et al.*, 2018; Maruf *et al.*, 2019; Yang *et al.*, 2019], to cache and memory-based NMT [Tu *et al.*, 2018; Kuang *et al.*, 2018; Maruf and Haffari, 2018]. However, large-scale document parallel corpora are costly to build as they require specialized expertise. Conversely, monolingual document data is much easier to find. In this paper, our goal is to leverage large-scale source-side monolingual documents to improve context-aware NMT performance.

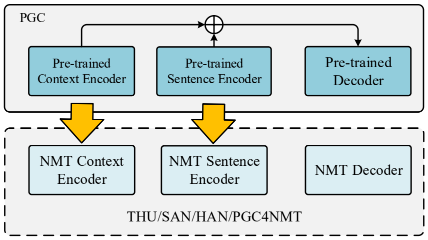

Figure 1: Illustration of our pre-trained model (upper) and downstream context-aware NMT models (lower).

There have been several attempts to boost context-aware translation performance in the scenarios where the document-level parallel corpora are small/middle-scale, or even not available. On the one hand, sentence-level parallel data is a natural resource to use. For example, [Zhang *et al.*, 2018] propose a two-stage training strategy for context-aware NMT by pre-training the model on a sentence-level parallel dataset. Such or similar training strategies are widely applied in related studies [Miculicich *et al.*, 2018; Tan *et al.*, 2019; Voita *et al.*, 2019b; Miculicich *et al.*, 2018; Maruf *et al.*, 2019]. On the other hand, monolingual target language document data could be used to increase the coherence of document translation. For example, [Voita *et al.*, 2019a] propose DocRepair trained on target-side monolingual documents to correct the inconsistencies in sentence-level translation. [Yu *et al.*, 2020] train a document-level language model to re-rank N-best translation outputs. To the best of our knowledge, [Junczys-Dowmunt, 2019] is the only work that leverages large-scale source-side monolingual documents, in which they simply concatenate sentences within a document into a long sequence and explore multi-task training via the BERT-objective [Devlin *et al.*, 2019] on the encoder.

As typical context-aware NMT models usually contain a component of capturing global context, in this paper we aim at greatly enhancing the capability of capturing use-

[*] Corresponding Author

$\mathcal{S}$	$\mathcal{T}$
S_1: $\widehat{X_1}$	T_1: X_1
S_2: $\widehat{X_2}$	T_2: X_2
S_3: X_2 [MASK1] X_4	T_3: X_3
S_4: $\widehat{X_4}$	T_4: X_4
S_5: X_4 [MASK1] X_6	T_5: X_5
S_6: $\widehat{X_6}$	T_6: X_6

(a) Document-Level input S; (b) Document-Level output T;

Figure 2: Illustration of the proposed pre-training task. Note that we assume that the original document contains 6 sentences, i.e., $\mathcal{X} = (X_1, \cdots, X_6)$ while X_3 and X_5 are gap sentences.

ful global context by exploring large-scale source-side documents. Specifically, to achieve this goal, we propose a novel self-supervised pre-training task on monolingual documents, **P**re-training with **G**lobal **C**ontext, i.e., PGC. We find that it is beneficial to mask entire selected sentences from a document and generate these gap-sentences from their neighbouring sentences as a pre-training objective.

As shown in Figure 1, our PGC model consists of a context encoder, a sentence encoder and a decoder, similar to various context-aware NMT models [Zhang *et al.*, 2018; Miculicich *et al.*, 2018; Maruf *et al.*, 2019; Zheng *et al.*, 2020]. Therefore, the PGC model could easily and effectively adapt to downstream context-aware NMT models. We initialize the downstream NMT models' two encoders with the counterparts of the pre-trained PGC model and then finetune the NMT models. Extensive experimental studies on four different translation tasks suggesting that our approach has strong generality to various downstream models.

Overall, this paper makes the following contributions:

- To leverage document-level monolingual corpora, we first propose a novel self-supervised pre-training task, in which sentence generation can benefit from document-level context.
- We evaluate the effectiveness and generality of our approach by adapting it to various downstream context-aware NMT models on four translation tasks.

2 Pre-training with Global Context (PGC) on Monolingual Documents

In this section, we first propose our pre-training task with global context (PGC). Then we present the details of our PGC model which leverages global context.

2.1 PGC Task

To be consistent with context-aware NMT, our pre-training task is sentence generation augmented by the document-level context. Our pre-training objectives are inspired by both gap sentence objective [Zhang *et al.*, 2020] and masked language model objective [Devlin *et al.*, 2019].

Context-Aware Gap Sentence Generation (CA-GSG)

Given a document with N sentences, we randomly select M sentences as gap sentences and replace them with a mask token `[MASK1]` to inform the model. For each selected gap sentence, we use its left and right neighbours as input while the gap sentence serves as output.

Context-Aware Denoisying Auto-Encoder (CA-DAE)

Given a sentence X, we follow BERT and randomly select 15% tokens in it. The selected tokens are (1) 80% of time replaced by a mask token `[MASK2]`, or (2) 10% of time replaced by a random token, or (3) 10% of time unchanged. For a sentence, we use its masked $\widehat{X}$ as input while the original X serves as output.

Combination of CA-GSG and CA-DAE

Both CA-GSG and CA-DAE are applied simultaneously in our pre-training task. For convenience of presentation, we use a concrete example to illustrate the input and output of our pre-training task. As shown in Figure 2, let assume that the original document $\mathcal{X}$ contains 6 sentences and the third and fifth sentences (i.e., X_3 and X_5) are selected as gap sentences while the others are not. On the one hand, for a sentence which is not selected as gap sentence, e.g., X_1, we use its masked version (e.g., $\widehat{X_1}$) as input while try to predict its original sentence (e.g., X_1). On the other hand, for a gap sentence, e.g., X_3, we concatenate its left and right neighbouring sentences with separator `[MASK1]` and try to predict the gap sentence (e.g., X_3). As shown in Figure 2, sentences from S_1 to S_6 constitute document-level input $\mathcal{S}$ while sentences from T_1 to T_6 make up output $\mathcal{T}$. Note that we do not include either gap sentences themselves or their masked version in $\mathcal{S}$, in case the document context contains obvious hints for generating gap sentences.

Overall, the pre-training task is to predict target output $\mathcal{T}$ by giving source input $\mathcal{S}$, which is the same as the task of context-aware translation, except that in our pre-training task $\mathcal{S}$ and $\mathcal{T}$ are in the same language while in the latter the two are in different languages.

2.2 PGC Model

We define some notations and describe our model of pre-training. Given a document-level source input $\mathcal{S} = (S_1, \cdots, S_N)$ and target output $\mathcal{T} = (T_1, \cdots, T_N)$ with N sentence pairs, we assume each source sentence $S_i = (s_{i,1}, \cdots, s_{i,n})$ consists of n words. We use d_m as the size of embedding and hidden state through the entire model.

We now design model to generate $\mathcal{T}$ by given $\mathcal{S}$. To be consistent with our purpose that our model of pre-training could easily adapt to various downstream context-aware NMT models, we propose a universal context-aware model, rather than a sophisticated one.

Figure 3 shows our model for the pre-training task. It contains two parts, namely global context encoder and a seq2seq model augmented by context representation. Note that in our pre-training, we take documents as input units.

Global Context Encoder

For the i-th input sentence S_i in a document, the global context encoder aims to extract useful global context for every

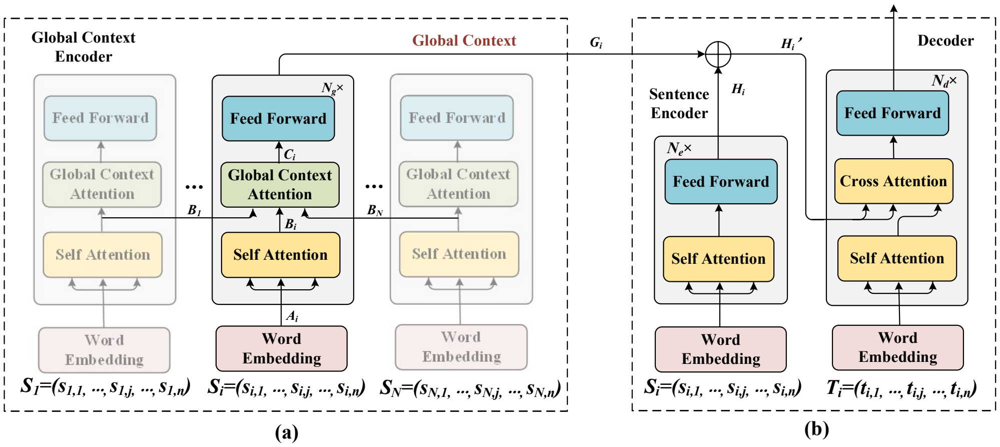

Figure 3: Illustration of our PGC model. For simplicity, we omit residual connection and layer normalization in each sub-layer.

word $s_{i,j}$ in it. As shown in Figure 3(a), the encoder consists of a stack of N_g identical encoder layers. Each encoder layer consists of three major sub-layers: a self-attention sub-layer, a global context attention sub-layer and a feed-forward neural sub-layer.

In the k-th encoder layer, the self-attention sub-layer takes $A_i^{(k)} \in \mathbf{R}^{n\times d_m}$ as input and computes a new sequence $B_i^{(k)}$ with the same length via multi-head attention function:

$$B_i^{(k)} = \text{MultiHead}\left(A_i^{(k)}, A_i^{(k)}, A_i^{(k)}\right), \tag{1}$$

where the output $B_i^{(k)}$ is in the shape of $\mathbb{R}^{n\times d_m}$.[1] For the first encoder layer, $A_i^{(1)}$ is the addition of S_i's word embedding and its position embedding while for other layers, $A_i^{(k)}$ is the output of the proceeding encoder layer.

We denote $\mathcal{B}^{(k)} \in \mathbf{R}^{(N\cdot n)\times d_m}$ as the stacking result of $\left(B_1^{(k)}, \cdots, B_N^{(k)}\right)$. Note that $\mathcal{B}^{(k)}$ is at document-level and represents the global context. The global context attention sub-layer extracts useful global context for $s_{i,j}$ in S_i. This is also done via multi-head attention function:

$$C_i^{(k)} = \text{MultiHead}\left(B_i^{(k)}, \mathcal{B}^{(k)}, \mathcal{B}^{(k)}\right), \tag{2}$$

where the output $C_i^{(k)}$ is in the shape of $\mathbb{R}^{n\times d_m}$.

Finally, the feed-forward sub-layer is applied to each position separately and identically by two linear transformations with a ReLU activation in between.

$$D_i^{(k)} = \max\left(0, C_i^{(k)} W^{F1} + b^{F1}\right) W^{F2} + b^{F2}, \tag{3}$$

where $W^{F1}, W^{F2} \in \mathbb{R}^{d_m\times d_m}$, and $b^{F1}, b^{F2} \in \mathbb{R}^{d_m}$ are model parameters.

We denote $G_i \in \mathbf{R}^{n\times d_m}$ as the final output of the global context encoder, i.e., $G_i = D_i^{(N_g)}$. That is to say, G_i represents the **context representation** for sentence S_i.

Sentence-Level Seq2Seq Model Augmented by Context Representation

As shown in Figure 3 (b), the sentence-level seq2seq model is almost same as the standard Transformer, except that it is now equipped with context representation obtained by the global context encoder. For sentence S_i, we denote the sentence encoder output as $H_i \in \mathbf{R}^{n\times d_m}$. To leverage its context representation G_i, we define a gate to linearly combine the two kinds of representation via:

$$H_i' = \lambda H_i + (1 - \lambda) G_i, \tag{4}$$

where the gating weight is computed by

$$\lambda = \text{sigmoid}\left([H_i; G_i] W^G\right), \tag{5}$$

where $W^G \in \mathbb{R}^{2d_m\times d_m}$ are model parameters.

Then we use H_i' to replace H_i as the input to the decoder.

3 Applying Pre-trained Model to Downstream Context-Aware NMT models

To test if our pre-trained model is helpful for context-aware NMT, we select the following four typical NMT models:

- DocT [Zhang *et al.*, 2018], takes two previous sentences as context. As a Document-aware Transformer, the context representations are fed into both sentence encoder and decoder.
- HAN [Miculicich *et al.*, 2018], leverages all previous source and target sentences as context and proposes a Hierarchical Attention Network to capture the context in a structured and dynamic manner. The context representations are then fed into the decoder.
- SAN [Maruf *et al.*, 2019], further uses the whole document as context. It uses Sparse Attention Network to selectively focus on relevant sentences and then attends to key words in those sentences.
- MCN [Zheng *et al.*, 2020], use a encoder builds local and global context from the entire document to understand the inter-sentential dependencies towards making the Most of Context.

[1] The actual output of this sub-layer is $LayerNorm(B_i^{(k)} + A_i^{(k)})$, where $LayerNorm$ is the layer normalization function. For simplicity, we do not include the residual addition and layer normalization functions in our sub-layers.

Meanwhile, we also provide the following contrastive NMT model to test if it is necessary for the downstream NMT models to have same structure as the pre-trained model.

- PGC4NMT, is similar to our model of pre-training. When use PGC for context-aware NMT, the only difference lies in that for NMT we use two different vocabularies for the source and target sides.

All the above context-aware NMT models consist of a context encoder, a sentence encoder and a decoder. Moreover, all the context and sentence encoders are Transformer-based with similar or same structure to the standard Transformer encoder [Vaswani *et al.*, 2017].

Applying pre-trained model to these context-aware NMT models. We load generic parameters from the context encoder and the source encoder of the pre-trained model to initialize the downstream context-aware NMT models. That is to say, if a sub-layer that does not exist in the pre-trained model, it will be randomly initialized, otherwise it will be initialized by the pre-trained model.

Fine-Tuning Strategy. We use a *two-step fine-tuning strategy* to fine-tune the downstream context-aware translation models. In the first step, as shown in Figure 1, generic parameters from the context encoder and the sentence encoder of the pre-trained model are loaded to initialize the counterparts in the translation model. Then we freeze the generic parameters and update the remaining parameters of the translation model. In the second step, we train all model parameters with a small number of iterations.

4 Experimentation

To test the effect of our approach in leveraging source-side monolingual documents, we conduct experiments on various tasks, including Chinese-English (ZH-EN), English-Spanish (EN-ES), and English-German (EN-DE) translation.

4.1 Experimental Settings

Pre-Training Data Settings. We pre-train two models, one for Chinese and the other for English based on the following two corpora, respectively:

- Chinese Gigaword (LDC2009T27), consists of eight distinct international sources of Chinese newswire. We convert traditional Chinese into simplified ones. Then all sentences are segmented by Jieba.[2]
- English Gigaword (LDC2012T21), consists of six distinct international sources of English newswire. We perform sentence segmentation, and then tokenize and truecase all sentences by Moses scripts.[3]

For both Chinese and English, we segment words into subwords by a BPE model with 30K operations. For efficient training, we split long documents into sub-documents with at most 30 sentences. We have 2.6M (7.3M) sub-documents with 24M (102M) sentences in total for Chinese (English). On the monolingual documents, we prepare training instances for the pre-training task and set gap sentence ratio to 20%.

Context-Aware NMT Data Settings. For ZH-EN, the document-level parallel corpus of training set include 41K documents with 780K sentence pairs.[4] We use the NIST MT 2006 dataset as the development set, and the NIST MT 02, 03, 04, 05, 08 datasets as test sets. The Chinese sentences are segmented by Jieba while the English sentences are tokenized and lowercased by Moses scripts. For EN-ES, the training set is from IWSLT 2014 and 2015 while the development set is dev2010 and the test set is test2010, test2011, and test2012.[5] For EN-DE (TED), the training set is from IWSLT 2017. We use test2016 and test2017 as our test set while the other as the development set. For EN-DE (News), the training set is News Commentary v11 corpus,[6] while the development set is news-test2015 and the test set is news-test2016. We tokenize and truecase all the datasets by Moses scripts. For all the translation datasets, we segment the source sentences with the corresponding BPE model learned from the pre-training data while all the target sentences are segmented by the BPE model with 25K operations learned on the corresponding target-side data. Meanwhile, we split long documents into sub-documents with at most 30 sentences.

Model Settings. We use OpenNMT[7] as the implementation of Transformer and extend it to capture context. We also re-implement DocT, HAN, and SAN as to better evaluate our pre-trained PGC models.[8] For all pre-trained and translation models, the numbers of layers in the context encoder, sentence encoder and decoder (i.e., N_g, N_e, and N_d in Figure 3) to 4, 6, 6, respectively. In inferring, we set beam size to 5.

Evaluation We report BLEU score as calculated by the multi-bleu.perl script and Meteor score. To be consistent with related studies [Zhang *et al.*, 2018; Miculicich *et al.*, 2018; Maruf *et al.*, 2019; Yang *et al.*, 2019], here we report case-insensitive BLEU score for ZH-EN and case-sensitive BLEU score for other translation tasks.

4.2 Experimental Results

Table 1 shows the BLEU and Meteor scores on the four translation tasks. It shows that on the one hand, even without pre-training, all context-aware NMT models outperform sentence-level Transformer. On the other hand, by leveraging source-side monolingual documents, our approach significantly improves the translation performance for all the four context-aware NMT models, with 0.92 to 1.96 improvement of BLEU scores and 1.04 to 1.85 improvement of Meteor scores, suggesting the effectiveness and generality of our approaching in leveraging source-side documents.

Taking SAN as representative, we compare its performance to that of PGC4NMT, and observe that although the structure of SAN is different from PGC4NMT and our pre-trained model, the degree to which SAN can benefit from pre-trained

[2] https://github.com/messense/jieba-rs

[3] http://www.statmt.org/moses/

[4] It consists of LDC2002T01, LDC2004T07, LDC2005T06, LDC2005T10, LDC2009T02, LDC2009T15, LDC2010T03.

[5] https://wit3.fbk.eu

[6] http://www.casmacat.eu/corpus/news-commentary.html

[7] https://github.com/OpenNMT/OpenNMT-py

[8] Although our experimental settings have subtle differences from theirs, e.g., not sharing vocabulary, the performance of our re-implemented systems is comparable to theirs.

Model	ZH-EN		EN-ES		EN-DE (TED)		EN-DE (News)	
	BLEU	Meteor	BLEU	Meteor	BLEU	Meteor	BLEU	Meteor
Transformer	39.64	27.56	35.50	34.60	23.02	43.66	22.03	41.37
DocT [Zhang *et al.*, 2018]	40.35	27.91	37.07	36.16	24.00	44.69	23.08	42.40
+ our approach	41.52	28.96	38.12	37.31	24.95	45.26	24.33	43.51
HAN [Miculicich *et al.*, 2018]	40.83	28.49	37.35	36.50	24.18	45.05	24.55	43.74
+ our approach	42.47	29.53	38.81	37.95	25.33	46.14	25.81	44.91
SAN [Maruf *et al.*, 2019]	41.01	28.37	38.11	37.09	24.37	45.26	24.77	44.17
+ our approach	**42.93**	29.75	**40.06**	**38.94**	**26.18**	**46.97**	**26.49**	45.58
MCN [Zheng *et al.*, 2020]	40.90	28.39	37.44	36.51	24.25	45.00	24.47	45.38
+ our approach	42.55	29.69	38.89	38.01	25.74	46.59	26.20	**46.95**
PGC4NMT	40.96	28.61	37.80	36.62	24.20	45.07	24.55	43.85
+ our approach	42.74	**30.05**	39.70	38.42	25.83	46.71	26.15	45.50

Table 1: Performance (BLEU and Meteor scores) on test sets. Significance test [Koehn, 2004] shows that the improvement achieved by our approach is significant at 0.01 for all above context-aware NMT models.

Language	#Para	#Epoch	Time
Chinese	95M	3.0	70h
English	89M	1.2	75h

Table 2: Statistics on our two pre-trained PGC models

model is similar to or even more than PGC4NMT. This further demonstrates the generality of our pre-trained model.

4.3 Result Analysis

Next we analyze how the pre-trained PGC models affect downstream translation performance.

Statistics on Our Pre-trained PGC Models

Table 2 presents statistics on our two pre-trained PGC models. With 500K training steps, we complete 3.0 and 1.2 passes over the pre-training data within 70 and 75 hours for Chinese and English, respectively.

In total, this batch size and number of steps corresponds to pre-training on $500K \times 8196 \approx 4.1B$ tokens. To better leverage pre-training dataset, it is a common practice to eke out additional performance by using an ensemble of models. We leave this in our future work.

Effect of Different Pre-Training Objectives

As shown in Figure 2, in this paper we combine both CA-GSG and CA-DAE objectives in our pre-training. To investigate the effect of CA-GSG, we use CA-DAE as the only objective in our pre-training task. In this way, the S_3 and S_5 in Figure 2 (b), for example, will be $\widehat{X_3}$ and $\widehat{X_5}$, respectively. Figure 4 compares the performance when the pre-training task is of CA-DAE objective or combination of CA-GSG and CA-DAE. For better illustration, for each translation task we scale the BLEU scores by viewing the performance of DocT·DAE as 1.0 and the performance of other systems as the ratio of their BLEU scores over DocT·DAE score.

From the figure, we see that in 19 out of 20 cases, combining CA-GSG and CA-DAE achieves better performance than CA-DAE. This result suggests pre-training benefits from the CA-GSG, which is more challenging than CA-DAE.

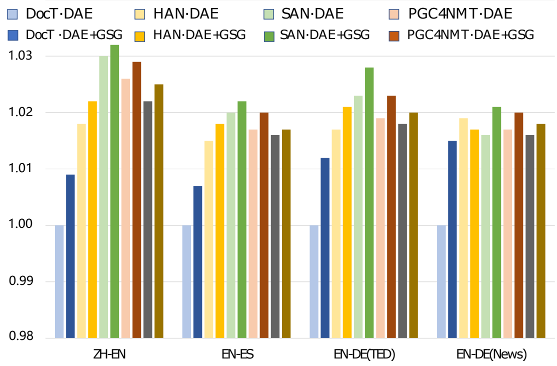

Figure 4: Translation performance comparison when the pre-training task is of CA-DAE (·DAE) or combination of CA-GSG and CA-DAE (·DAE+GSG). For each translation task we scale the performance of DocT as 1.0 for better illustration.

Effect of Gap Sentence Ratios

A significant hyper-parameter in designing the pre-training task is the gap sentence ratio. A low ratio makes the pre-training less challenging while choosing gap sentences at a high ratio makes the global context have more overlapped. For example, X_4 and its masked version $\widehat{X_4}$ in Figure 2(a) appear three times on the source input $\mathcal{S}$ due to that both its left and right neighbouring sentences X_3 and X_5 are gap sentences. We compare three variants of gap sentence ratio (10%, 20%, and 30%). As shown in Figure 5, we see that the best performance always appears at the ratio of 20%.

Effect of Pre-Trained Global Context Encoder

Next we examine whether the proposed global context encoder actually learns to effectively extract global context from the entire document. To this end, we change our per-training task to be context-agnostic with a standard seq2seq framework. As a result, only the pre-trained sentence encoders are applied to downstream translation models. From Table 3 we observe that pre-trained global context encoder helps context-aware translation, suggesting that our pre-trained context encoder is indeed capable of extracting useful global context from the entire document.

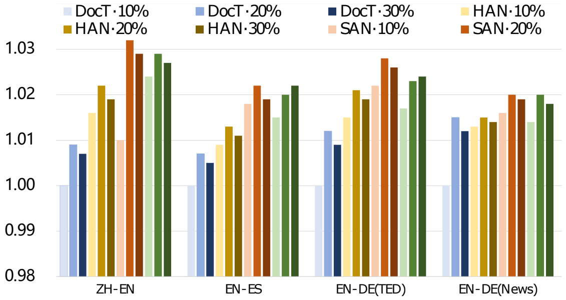

Figure 5: Translation performance comparison when use different gap sentence ratios.

Model	deixis	lex.c.	ell.infl.	ell.VP
Transformer	50.0	45.3	52.0	27.3
PGC4NMT	60.1	46.0	61.1	35.5
+ Our approach	65.0	46.3	63.7	51.9

Table 3: Accuracy(%) of discourse phenomena.

Discourse Phenomena

Following [Voita *et al.*, 2019a; Zheng *et al.*, 2020], we use contrastive test sets for the evaluation of discourse phenomena for English-Russian. There are four test sets in the suite regarding deixis, lexicon consistency, ellipsis (inflection), and ellipsis (verb phrase). Each test set contains groups of contrastive examples consisting of a positive translation and negative translations. The goal is to figure out if a model is more likely to generate a correct translation compared to the incorrect variation. We summarize the results in Table 3, which shows that our approach is better at resolving discourse consistencies compared to both the context-agnostic baseline and our proposed context-aware approach without pre-training.

5 Related Work

5.1 Context-Aware NMT

According to the scope of context being modeled, we group the related studies into two categories: (1) those which use partial context and (2) those which employ global context.

Among the first line, great efforts have been made in modeling local context. Concatenation method proposed by [Tiedemann and Scherrer, 2017] is an early attempt in RNN-based NMT. Then mechanisms with multiple encoders become promising in both RNNSearch and Transformer NMT[Jean *et al.*, 2017; Wang *et al.*, 2017; Zhang *et al.*, 2018; Bawden *et al.*, 2018; Voita *et al.*, 2018; Voita *et al.*, 2019b; Yang *et al.*, 2019]. Cache/Memory-based approaches [Tu *et al.*, 2018; Kuang *et al.*, 2018; Maruf and Haffari, 2018] also fall in this line because the cache stores word/translation in previous sentences.

Another strand of context-aware NMT takes the whole document as a translation unit and dynamically extracts useful global knowledge for every sentence in the document. The global context can be either source-side [Maruf and Haffari, 2018; Mace and Servan, 2019; Maruf *et al.*, 2019; Tan *et al.*, 2019; Zheng *et al.*, 2020; Kang *et al.*, 2020] or target-side [Xiong *et al.*, 2019].

Global Context Enc.	DocT	HAN	SAN	MCN	PGC4NMT
Not Pre-trained	41.07	41.33	41.80	41.69	41.82
Pre-trained	41.52	42.47	42.93	42.55	42.74

Table 4: Translation performance comparison on ZH-EN translation task when the global context encoder is pre-trained or not.

To make translations within a document more coherent, [Voita *et al.*, 2019a] propose DocRepair trained on monolingual target language document corpora to correct the inconsistencies in sentence-level translation while [Yu *et al.*, 2020] train a context-aware language model to re-rank sentence-level translations. Finally, [Junczys-Dowmunt, 2019] use source-side monolingual documents to explore multi-task training via the BERT-objective on the encoder. They simply concatenate sentences within a document into a long sequence, which is different from our approach.

5.2 Pre-training on Document-Level Corpora

Recently, pre-trained models with large corpora built with Transformer have made great success. Among them, BART, mBART and PEGASUS are built on document-level corpora.

BART [Lewis *et al.*, 2020] and mBART [Liu *et al.*, 2020] use a (multilingual) denoising auto-encoder which learns to reconstruct the original document from a corrupted version. BART is effective for machine translation by achieving significant improvement over a back-translation system while mBART produces significant performance gains across a wide variety of translation tasks.

PEGASUS [Zhang *et al.*, 2020] is a Transformer-based encoder-decoder model with a self-supervised objective tailored for abstractive text summarization. In PEGASUS, important sentences are removed/masked from a document then generated together as one output sequence from the remaining sentences, similar to an extractive summary.

BART, mBART and PEGASUS all fall into a standard seq2seq framework by viewing a document (or masked sentences) as a sequence. This is very different from ours, where we use an additional global context extractor for context-aware sentence generation.

6 Conclusion

To leverage monolingual document data, in this paper we have proposed a PGC model with two different training objectives. PGC model consists of a global context encoder, a sentence encoder and a decoder, which is similar to various context-aware NMT models. We have evaluated the effectiveness and generality of our pre-trained PGC model by adapting it to various downstream context-aware NMT models. Experimental results on various translation tasks demonstrate the effectiveness and generality of our approach.

Acknowledgments

This work was supported by the National Natural Science Foundation of China (Grant No. 61976148, 61876120 and 62076175).

References

[Bawden *et al.*, 2018] Rachel Bawden, Rico Sennrich, Alexandra Birch, and Barry Haddow. Evaluating discourse phenomena in neural machine translation. In *Proceedings of NAACL*, 2018.

[Devlin *et al.*, 2019] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. Bert: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of NAACL*, 2019.

[Jean *et al.*, 2017] Sebastien Jean, Stanislas Lauly, Orhan Firat, and Kyunghyun Cho. Does neural machinetranslation benefit from larger context? *Computing Research Repository*, arXiv:1704.05135, 2017.

[Junczys-Dowmunt, 2019] Marcin Junczys-Dowmunt. Microsoft translator at wmt 2019: Towards large-scale document-level neural machine translation. In *Proceedings of WMT*, 2019.

[Kang *et al.*, 2020] Xiaomian Kang, Yang Zhao, Jiajun Zhang, and Chengqing Zong. Dynamic context selection for document-level neural machine translation via reinforcement learning. In *Proceedings of EMNLP*, 2020.

[Koehn, 2004] Philipp Koehn. Statistical significance tests for machine translation evaluation. In *Proceedings of EMNLP*, 2004.

[Kuang *et al.*, 2018] Shaohui Kuang, Deyi Xiong, Weihua Luo, and Guodong Zhou. Modeling coherence for neural machine translation with dynamic and topic caches. In *Proceedings of COLING*, 2018.

[Lewis *et al.*, 2020] Mike Lewis, Yinhan Liu, Naman Goyal, Marjan Ghazvininejad, Abdelrahman Mohamed, Omer Levy, Ves Stoyanov, and Luke Zettlemoyer. Bart: Denoising sequence-to-sequence pre-training for natural language generation, translation, and comprehension. In *Proceedings of ACL*, 2020.

[Liu *et al.*, 2020] Yinhan Liu, Jiatao Gu, Naman Goyal, Xian Li, Sergey Edunov, Marjan Ghazvininejad, Mike Lewis, and Luke Zettlemoyer. Multilingual denoising pre-training for neural machine translation. *Computing Research Repository*, arXiv:2001.08210, 2020.

[Mace and Servan, 2019] Valentin Mace and Christophe Servan. Using whole document context in neural machine translation. In *Proceedings of IWSLT*, 2019.

[Maruf and Haffari, 2018] Sameen Maruf and Gholamreza Haffari. Document context neural machine translation with memory networks. In *Proceedings of ACL*, 2018.

[Maruf *et al.*, 2019] Sameen Maruf, André F. T. Martins, and Gholamreza Haffari. Selective attention for context-aware neural machine translation. In *Proceedings of NAACL*, 2019.

[Miculicich *et al.*, 2018] Lesly Miculicich, Dhananjay Ram, Nikolaos Pappas, and James Henderson. Document-level neural machine translation with hierarchical attention networks. In *Proceedings of EMNLP*, 2018.

[Tan *et al.*, 2019] Xin Tan, Longyin Zhang, Deyi Xiong, and Guodong Zhou. Hierarchical modeling of global context for document-level neural machine translation. In *Proceedings of EMNLP*, 2019.

[Tiedemann and Scherrer, 2017] Jörg Tiedemann and Yves Scherrer. Neural machine translation with extended context. In *Proceedings of the Third Workshop on Discourse in Machine Translation*, 2017.

[Tu *et al.*, 2018] Zhaopeng Tu, Yang Liu, Shuming Shi, and Tong Zhang. Learning to remember translation history with a continuous cache. *Transactions of ACL*, 6, 2018.

[Vaswani *et al.*, 2017] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N.Gomez, Lukasz Kaiser, and Illia Polosukhin. Attention is all you need. In *Proceedings of NIPS*, 2017.

[Voita *et al.*, 2018] Elena Voita, Pavel Serdyukov, Rico Sennrich, and Ivan Titov. Context-aware neural machine translation learns anaphora resolution. In *Proceedings of ACL*, 2018.

[Voita *et al.*, 2019a] Elena Voita, Rico Sennrich, and Ivan Titov. Context-aware monolingual repair for neural machine translation. In *Proceedings of EMNLP*, 2019.

[Voita *et al.*, 2019b] Elena Voita, Rico Sennrich, and Ivan Titov. When a good translation is wrong in context: Context-aware machine translation improves on deixis, ellipsis, and lexical cohesion. In *Proceedings of ACL*, 2019.

[Wang *et al.*, 2017] Longyue Wang, Zhaopeng Tu, Andy Way, and Qun Liu. Exploiting cross-sentence context for neural machine translation. In *Proceedings of EMNLP*, 2017.

[Xiong *et al.*, 2019] Hao Xiong, Zhongjun He, Hua Wu, and Haifeng Wang. Modeling coherence for discourse neural machine translation. In *Proceedings of AAAI*, 2019.

[Yang *et al.*, 2019] Zhengxin Yang, Jinchao Zhang, Fandong Meng, Shuhao Gu, Yang Feng, and Jie Zhou. Enhancing context modeling with a query-guided capsule network for document-level translation. In *Proceedings of EMNLP*, 2019.

[Yu *et al.*, 2020] Lei Yu, Laurent Sartran, Wojciech Stokowiec, Wang Ling, Lingpeng Kong, Phil Blunsom, and Chris Dyer. Better document-level machine translation with bayes rule. *Transactions of ACL*, 8, 2020.

[Zhang *et al.*, 2018] Jiacheng Zhang, Huanbo Luan, Maosong Sun, Feifei Zhai, Jingfang Xu, Min Zhang, and Yang Liu. Improving the transformer translation model with document-level context. In *Proceedings of EMNLP*, 2018.

[Zhang *et al.*, 2020] Jingqing Zhang, Yao Zhao, Mohammad Saleh, and Peter J. Liu. Pegasus: Pre-training with extracted gap-sentences for abstractive summarization. In *Proceedings of ICML*, 2020.

[Zheng *et al.*, 2020] Zaixiang Zheng, Xiang Yue, Shujian Huang, Jiajun Chen, and Alexandra Birch. Towards making the most of context in neural machine translation. In *Proceedings of IJCAI*, 2020.