Routing Based Context Selection for Document-Level Neural Machine Translation

Weilun Fei[1], Ping Jian[1,2(✉)], Xiaoguang Zhu[1], and Yi Lin[1]

[1] School of Computer ScienceTechnology, Beijing Institute of Technology, Beijing 100081, China
[2] Beijing Engineering Research Center of High Volume Language Information Processing and Cloud Computing Applications, Beijing Institute of Technology, Beijing 100081, China
{wlfei,pjian,xgzhu,1120180804}@bit.edu.cn

Abstract. Most of the existing methods of document-level neural machine translation (NMT) integrate more textual information by extending the scope of sentence encoding. Usually, the sentence-level representation is incorporated (via attention or gate mechanism) in these methods, which makes them straightforward but rough, and it is difficult to distinguish useful contextual information from noises. Furthermore, the longer the encoding length is, the more difficult it is for the model to grasp the inter-dependency between sentences. In this paper, a document-level NMT method based on a routing algorithm is presented, which can automatically select context information. The routing mechanism endows the current source sentence with the ability to decide which words can become its context. This leads the method to merge the inter-sentence dependencies in a more flexible and elegant way, and model local structure information more effectively. At the same time, this structured information selection mechanism will also alleviate the possible problems caused by long-distance encoding. Experimental results show that our method is 2.91 BLEU higher than the Transformer model on the public dataset of ZH-EN, and is superior to most of the state-of-the-art document-level NMT models.

Keywords: Natural Language Processing · Document-Level Neural Machine Translation · Routing Algorithm.

1 Introduction

With the development of deep learning methods, neural machine translation(NMT) has made remarkable progress in most language pairs. However, the standard NMT methods are first designed for sentence-level [1–3], which may bring some document-level errors, such as document inconsistency [4–9]. In order to reduce the errors caused by sentence-level NMT when translating discourses, a large number of document-level NMT methods have been proposed to improve the translation performance by using context outside a single sentence.

The most recent context-aware methods take the context of the current sentence as inputs of NMT model, and attach another input stream in parallel [4, 5, 10]. Therefore, most researchers tend to reform the mature NMT models to merge the representation from previous sentences as context [4, 5, 11–14] into every layer of the encoder or decoder to consider the information from cross sentences. To improve the comprehension of the current text, people can combine with the future context. It is very common for us, not to mention the neural network lacking prior knowledge and common sense. Consequently, context is not necessarily limited to the sentences before the current sentence, it can also come from the future, which is ignored but effective. However, if we simply and roughly expand the scope of sentences as inputs without filtering them, it may bring burden to the model. According to [15], information in the context is not always useful. We are supposed to increase the content of context selectively.

This paper draws lessons from a routing method [16] of multilingual NMT (MNMT), and puts forward a document-level NMT routing method based on this algorithm. In MNMT, researchers find that using a mix of shared and language-specific parameters can help the models obtain a great improvement in exploring universal MNMT, but keep the question of when and where language-specific capacity matters most. This is similar to what kind of context is the most useful in document-level NMT. According to [15], we can assume that every word in the context contains different levels of document-aware information. In order to filter redundant information of context, we use routing algorithm, which helps the model select words whose document-level information is more important as context automatically. On the one hand, we avoid long-distance encoding. On the other hand, redundant contextual information is filtered out.

In our experiments, we choose the sentence before the current sentence and the sentence next to the current sentence as context. The results show that the changes we made improve the performance of document-level NMT. Compared with the methods which utilize the whole document as context [9, 17], our method still has competitiveness, especially on the dataset of ZH-EN.

2 Related Work

With the latest development and performance improvements of neural networks, people are more interested in document-level MT and textual context also shows its importance to machine translation. Based on the encoder-decoder NMT framework, existing works mainly use the following three methods to introduce document-level information:

Single-Encoder Approach. This kind of method expands the range of sentences when inputting them into the model, such as [6, 18–20], which has done a lot of research about the input of model, including the expansion of encoder input and decoder input. This kind of method is relatively rough for the application of context, which is the earliest attempt of encoder-decoder framework. These attempts proved that not only the previous context but also the future context can improve the translation effect, which is gradually ignored in later

studies. In addition, the method of fusing context at the encoder side contributes more than the method of fusing context at the decoder side. Because fusion at the decoder side may lead to error propagation.

Multi-Encoder Approach. According to when and where to fuse the output of the multi-encoder inside the decoder (see Fig. 1), [13] or outside the decoder (see Fig. 2), [13, 21, 22]. Reference [23] divides the multi-encoder method into inside multi-encoders [4, 19, 24, 25] and outside multi-encoders [5, 9, 17, 26]. The Multi-encoder method mainly adopts two fusion methods: 1) Some methods use attention mechanisms to encode context statements into the encoder or decoder, for example, Reference [4] inserts a context-attention layer into the model; 2) the others use the gate mechanism to aggregate context, thus learning anaphora resolution. These methods are similar in that they all add Transformer models with additional context-related modules.

Post Processing. Reference [27] uses deliberation network, which adds another decoder after Transformer, and employs reward teacher to model coherence for document-level machine translation. Reference [8] uses another method called document-level repair, which makes full use of monolingual document-level data in the target language.

Inspired by previous works, we add an extra context module to the Transformer model to extract context information. Reference [15] suggests that in document-level NMT, sometimes context is too long to simplify calculations, and in fact, a lot of information in the context is actually unnecessary. They retain the most likely words of the context, such as named entities and special words like POS. Combined with the above points, we use the routing method in multilingual NMT, and hope that the model itself can combine the input sentence to determine which words are useful for forming context, not just named entities and POS, so as to improve the translation effect.

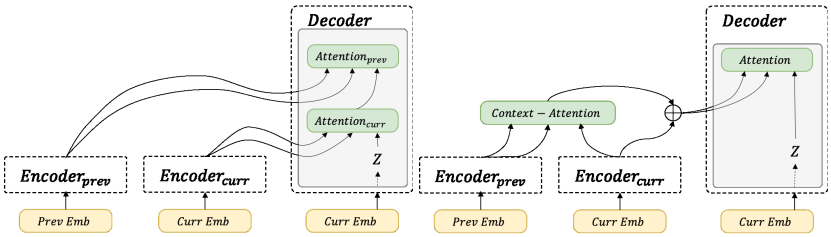

Fig. 1. Fusion inside the decoder

Fig. 2. Fusion outside the decoder

3 Background

3.1 Document-Level NMT

Compared with sentence-level NMT, document-level NMT considers contextual information. We assume that $(X, Y) \in \mathbb{C}$ where X represents source sentences

and Y represents target sentences. We use x^i to express the i-th sentence in X. y^i denotes the i-th sentence in Y. In order to generate target sentence y^k, document-level NMT is supposed to make full use of the contextual information of source sentence x^k. As the input of encoder, x is converted into the hidden state H. We define set $X^{<>k}$ as context of x^k, and then we can approximate the document-level translation probability as:

$$P(y^k|x^k;\theta) = \prod_{i=1}^{n} p(y_i^k|y_{<i}, H^k, X^{<>k};\theta) \tag{1}$$

3.2 Transformer

Encoder-Decoder architecture composed of sequence models, like RNN or LSTM, has made great improvement in NMT [2, 3]. However, Transformer [28], which relies entirely on attention mechanism, has surpassed most previous models. Considering the above points, we choose Transformer as our basic model.

To avoid gradient vanishing or explosion, the following residual normalization structure is used for the Transformer block:

$$z = LayerNorm(h + f(h)) \tag{2}$$

where h represents the output from the last block, z is the output of this block, $LayerNorm(\cdot)$ means Layer Normalization and $f(\cdot)$ can be MultiHead Attention or Feed-Forward Network. The encoder of Transformer includes Multi-Head Self-Attention and Feed-Forward Network. Though the decoder has similar sub-layers, another sub-layer called Encoder-Decoder Attention is inserted between these layers. With the help of MultiHead Attention, the model can pay attention to information from different representation subspaces:

$$Output = MultiHead(h, h, h) \tag{3}$$

$$Output = MultiHead(z, E_{out}, E_{out}) \tag{4}$$

where E_{out} represents the encoder output, h and z come from last block. When fed into the first layer of model, h represents the word embedding of sentence. Eq. (3) and Eq. (4) stand for the calculation of $Self\,Attention$ and $Enc\,Dec\,Attention$ respectively.

3.3 Conditional Language-Specific Routing (CLSR)

From the perspective of the mapping between language pairs, the MNMT model has three strategies: many-to-one, one-to-many and many-to-many. Reference [29] raises the question that just using specific language signs is not enough to explore the features of specific language. To make a thorough inquiry of when and where language specific modeling matters most in MNMT, reference [16] introduces conditional language-specific routing (CLSR), a method that keeps

the balance between language-specific path and shared path as controlled by the gates. Eq. (2) can be modified as follows:

$$z = LayerNorm(h + CLSR(f(h))) \tag{5}$$

CLSR learns a gate $g(\cdot)$ for each input token, which helps blocks in Transformer selectively route information through language-specific path h^{lang} or shared path h^{shared}:

$$CLSR(f(h)) = g(h) \odot h^{lang} + (1 - g(h)) \odot h^{shared} \tag{6}$$

$$h^{lang} = f(h)W^{lang},\ h^{shared} = f(h)W^{shared} \tag{7}$$

where W^{shared} represents the trainable parameters shared across languages and W^{lang} is the trainable parameters for specific languages. The gate $g(\cdot)$ is computed from a two-layer feed-forward network $G(\cdot)$, and zero-mean Gaussian noise is used to discretize it during training:

$$g(h) = \sigma(G(h) + \alpha(t)\mathcal{N}(0, 1)) \tag{8}$$

$$G(h) = Relu(hW_1 + b)W_2 \tag{9}$$

where $\sigma(\cdot)$ is the logisitic-sigmoid function, and W_1 as well as W_2 is trainable parameters. $\alpha(\cdot)$ is a linearly function and increases with training step t. When inferencing, $g(h)$ is replaced with a decision rule: $g(h) = \delta(G(h) > 0)$, where $\delta(\cdot)$ is a Dirac measure.

4 Method

In this section, we will introduce how we apply the aforementioned routing algorithm to selecting words as context automatically for document-level NMT in detail. Before that, we will introduce the symbols used in the model.
Assuming X and Y represent the source and target sentences in corpus $\mathbb{C}$. We define that $c^l_{k_{-1}}, c^l_{k_1}$ are outputs from the l-th Prev Encoder Layer and Post Encoder Layer. $x^l_{k_{-1}}, x^l_{k_1}$ and x^l_k are the input of the l-th encoder layer. When $l = 0$, $x^l_k = x_k$, it's the same as $x^l_{k_{-1}}$ and $x^l_{k_1}$. c^l_k means the l-th layer context hidden state, which is got by gate aggregation. $x^l_{k,self-attn}$ is used to represent the output from self-attention layer of l-th layer. We can see the details of the model in Fig. 3.

4.1 Inputs of Our Model

Considering the differences between sentence-level NMT and document-level NMT, it's necessary to introduce the inputs composition of our model. In Transformer, researchers use sine and cosine functions to calculate position embedding, which helps the attention mechanism pay attention to the word position information added to the word embedding. While in document-level NMT, the information of sentences order has its significance. We refer to the idea of [10],

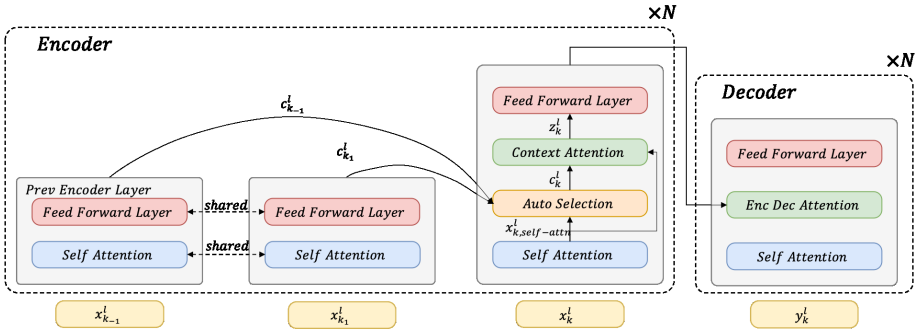

Fig. 3. The main architecture of our model. The two context-encoders share parameters. *Auto-Selection* Layer takes $x^l_{k,self-attn}$, $c^l_{k_{-1}}$ and $c^l_{k_1}$ as input to compute c^l_k which represents context information. To help $x^l_{k,self-attn}$ attend over all positions in the input context, Context Attention Layer takes $x^l_{k,self-attn}$ as query and c^l_k as key and value, in which case, we can get z^l_k.

in which way, we add the segment embedding to the position embedding and the word embedding. In Fig. 4, we take x^0_k as an example, which is the input of the first layer of our model, and we add different segment embedding to $x^0_{k_{-1}}$ and $x^0_{k_1}$(0 indicates the previous context, 2 indicates the future context).

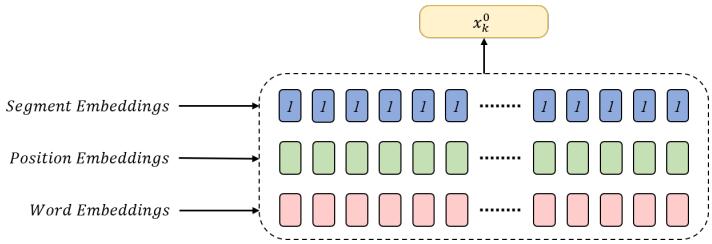

Fig. 4. The details about the composition of the inputs of the first layer of encoders, taking x^0_k as example, segment embeddings are set as 1 to represent the current sentence.

4.2 Context Attention

First of all, we explain how the model integrates context into the translation sentence. Between self-attention layer and feed-forward layer of the encoder for x_k, we insert *context attention* layer, which is defined as follows:

$$h_{AS} = MultiHead(x^l_{k,self-attn},\ c^l_k,\ c^l_k) \tag{10}$$

$$z_k^l = LayerNorm(x_{k,self-attn}^l + h_{AS}) \tag{11}$$

where z_k^l is the output of this layer and context hidden state c_k^l is computed by the algorithm following, which will be introduced in detail.

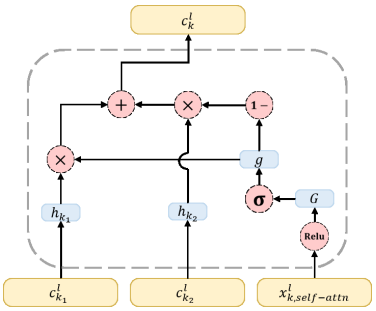

Fig. 5. The detail of *Auto-Selection* Layer. Gate is computed by $x_{k,self-attn}^l$

4.3 Auto-Selection

Since we choose the sentences around x as context, we must filter out the information that may bring unnecessary noise from context. Inspired by CLSR, we hope that our model can help x keep the balance between the information from different sentences and filter out noise, just as CLSR helps MNMT models to decide when and where to use language-specific parameters or shared parameters4:

$$c_k^l = g(x_{k,self-attn}^l) \odot h_{k-1} + (1 - g(x_{k,self-attn}^l)) \odot h_{k_1} \tag{12}$$

$$with \quad h_{k_{-1}} = c_{k_{-1}}^l W_{k_{-1}},\ h_{k_1} = c_{k_1}^l W_{k_1} \tag{13}$$

$W_{k_{-1}}$ as well as W_{k_1} is trainable parameters.

$$G(x_{k,self-attn}^l) = Relu(x_{k,self-attn}^l W_{-1} + b)W_1 \tag{14}$$

$$g(x_{k,self-attn}^l) = \sigma(G(x_{k,self-attn}^l)) \tag{15}$$

we can see Fig. 5 for details.

Compared with CLSR, we git rid of the zero-mean Gaussian noise, and totally let $x_{k,self-attn}^l$ itself to design its context. About other aspects of computation for $g(\cdot)$, following the configuration of CLSR, we apply a two-layers feed-forward network and use $Relu(\cdot)$ and $\sigma(\cdot)$ as activation function.

Now, we can summarize our training process as follows:

– The inputs $x_{k-1}^l, x_{k_1}^l$ are fed into the Prev/Post Encoder Layer respectively. The Prev Encoder Layer shares parameters with the Post Encoder Layer.

- The outputs of the Prev/Post Encoder Layer are sent to the encoder of x_k^l and x_k^l is used to calculate gate, which helps the model integrate more useful information from context to calculate the output of the encoder.
- The steps above are repeated for N times, i.e. the number of layers. Then, the outputs of encoder are sent to the decoder.
- Training the model. Continue the decoding process until meeting the end token.

5 Experiments

We mainly conduct experiments on Chinese → English and English → German task to verify our model, the details of datasets are as follows and listed in Table 1.

Table 1. The number of sentences in the datasets

Datasets		Training	Dev	Test
ZH-EN	TED	0.20M	0.88K	5.47K
EN-DE	TED	0.20M	8.96K	2.26K
	NEWS	0.22M	2.16K	2.99K
	Europarl	1.66M	3.58K	5.13K

5.1 Datasets

For fair comparison, we choose four widely used document-level parallel datasets, one Chinese → English dataset and three English → German datasets:

- TED (ZH-EN, TED). The Chinese → English datasets are from IWSLT 2015, where we mainly conduct our experiments. Following the work of [9], we take dev2010 as development set and tst2010-2013 as test set.
- TED (EN-DE, TED). According to [21], we choose IWSLT17 [30] as datasets for training. Tst2016-2017 is test set and the rest is the development set.
- News-Commentary (EN-DE, NEWS). Following [9] and [21], we obtain News Commentary v11 for training, WMT newstest 2015 for developing and WMT newstest2016 for testing.
- Europarl (EN-DE, Europarl). Train set, development set and test set are extracted from the Europarl v7 [31]. Details are mentioned in [21].

For TED ZH-EN dataset, we first use jieba for word segmentation. In all translation tasks, we tokenize the data with MOSE tokenizer [32] and apply byte-pair-encoding (BPE) algorithm [33] to encode words with sub-word units. We also use tools offered by Fairseq [34] to preprocess all dataset, in which form that model can accept.

5.2 Training Detail

On the basis of source code provided by Fairseq [34], we show detailed strategies for training the model. Adam [35] is the optimizer of the network with ($\beta_1 = 0.9, \beta_2 = 0.98$). $Warmup-updates$ is set 8000, $dropout$ is 0.1, where $warmup-init-lr$ is 10^{-7}. We set the batch size to 25,000 per batch and limit sentence length to 150 BPE tokens. For models on TED Zh-En, hidden dimension is $d_z = 256$, and the feed-forward dimension is $d_{ffn} = 512$. We use 4 layers in the encoder and decoder, each layer has 8 heads of attention. For the reset datasets, the hidden dimension and feed-forward dimension are set to 512/2048 respectively. Note that the above hyper-parameter settings are the same as those used in the baseline models.

5.3 Main Results

To make the results fair, we follow the work of [9] and [21] who use sacrebleu [36] to evaluate the translation quality. In addition to the baseline Transformer, we also compare our model with five state-of-the-art document-level NMT models including:

- Document-aware Transformer(DocT, [4]). Introducing context information by adding context sub-layers at each encoder and decoder layer.
- Hierarchical Attention NMT(HAN, [13]). Capturing the context in a structured and dynamic manner.
- Selective Attention NMT(SAN, [21]) Using sparse attention to selectively focus on relevant sentences.
- Query-guided Capsule Network(QCN, [22]). Clustering context information into different perspectives from which the target translation may concern.
- Arbitrary Context NMT(ACN, [9]). Being able to deal with documents containing any number of sentences.

Table 2. BLEU results on four datasets. The score in parentheses represents the BLEU of their baseline.

#	Models	ZH-EN	EN-DE		
		TED (baseline)	TED (baseline)	NEWS (baseline)	Europarl (baseline)
1	DocT(2018)[4]	n/a	24.00(23.28)	23.08(22.78)	29.32(28.72)
2	HAN(2018)[13]	17.90(17.00)	24.58(23.28)	25.03(22.78)	28.60(28.72)
3	SAN(2019)[21]	n/a	24.42(23.28)	24.84(22.78)	29.75(28.72)
4	QCN(2019)[22]	n/a	25.19(23.28)	22.37(21.67)	29.82(28.72)
5	ACN(2020)[9]	19.10(17.00)	25.10(23.10)	24.91(22.40)	30.40(29.40)
	Ours				
6	Transformer(2017)[28]	17.11	23.20	23.13	29.49
7	**Our Model**	20.02	25.01	24.03	29.87

As shown in Table 2, the proposed model improves the BLEU scores of the aforementioned datasets by 2.91, 1.81, 0.90 and 0.38 points compared with the baseline of sentence-level Transformer. Especially on TED ZH-EN, our model makes a significant improvement and surpasses the best model that we know by 1 point, showing its outstanding performance. Although our model is not the best on datasets of EN-DE, it is still capable of competing with other outstanding document-level NMT models, like SAN [21] and QCN [22]. We make the following analysis of the reasons for these listed results:

- Firstly, apposite translation requires more context, while document information is mainly used for semantic disambiguation. Therefore, using the whole document as context, like ACN [9] may perform better. However, after analyzing the translation results, we find that our method which uses word-level automatic routing has more advantages in structured information modeling. Besides, the proposed method is significantly improved on the datasets of TED, which may contains more structured information than the others. See Section 5.5 for details.
- Secondly, when we reproduced the experiment of DocT [4], we found that the improvement brought by training strategy is little. Considering the phenomenon above, we do not take the two-step training strategy. But we will keep following it in the future.
- Finally, the context-encoder and the module of *auto-selection* are just updated by the back propagation of the loss between the label and the predicted value. Due to the lack of other supervision, the larger the dataset is, the easier the model overfits the label. Therefore, it can be understood that our method is not significantly improved on the dataset Europarl.

Table 3. RESULTS OF ABLATION STUDY

		ZH-EN
#	**Models**	TED
1	Transformer[28]	17.11
2	DocT[4]	18.82
3	DocT+AS	19.72
4	Ours(online)	19.50
5	Ours(offline)	20.02

5.4 Ablation Study

We list our results of ablation in Table 3. We mainly produce our ablation study for the following aspects:

Offline vs. Online Document MT SAN [21] divides the source of context into two cases: *offline* context is both the context of the past and the context

of the future; *online* context is only the context of the past. In this part, we compare the result of offline and online document-level MT settings on TED ZH-EN. From the Table 3, we can find that the result of *offline*(row 4) is close to that of *online*(row 5) settings. It is quite self-explanatory that the post sentence as part of context really works in our methods. The proposed method can be extended to the full text as well, but it has achieved impressive performance even if only the previous and the post context sentences are considered, or even only the previous one. Moreover, usually the discourse structure information of local context is usually more meaningful to translation, so we mainly use pre-context and post-context sentences in our experiments.

Universality According to the experiment of *online* document-level MT, we make an assumption that whether we can apply our methods in other document-level MT methods. To test our intuition, we reproduce the model of DocT[4], whose results are listed in the row 2 and row 3 of Table 3. Compared with the original model, this result achieves an improvements of+0.9 BLEU. The main difference is that our approach of *auto-selection* helps the model to filter some redundant information and focus on the words that are really useful to improve the quality of document-level MT. But we only implement our method on a similar model to ours. We will carry out more experiments in the future to study the universality of our method.

5.5 Analysis

Table 4. Counts of conjunctions

	Ref.	**baseline**	**DocT**	**DocT+AS**	**Ours(online)**	**Ours(offline)**
and	3251	2569	2702	3027	3055	**3210**
but	561	590	587	**606**	594	590
or	233	183	201	**206**	186	197
because	285	295	314	316	307	**337**
so	853	487	497	526	519	**563**
yet	27	9	2	10	**15**	12
then	186	98	84	107	**161**	140

In order to analyze our model's ability of capturing structured information between sentences, we list some common conjunctions that can express the relationship of sentences in Table 4, such as *and*, *but*, *because*. According to the statistical results, we find that the document-level MT models tend to generate more conjunctions to capture the structured information. From the comparison of the statistical results of DocT [4] and *online*, we can find that the addition of *auto-selection* allows DocT [4] to add more related words than the original version which is also reflected in the *online* of our model. According to the results

of *offline*, we find that with the addition of future context, *offline* tends to add words that express the coordination or causality between sentences.

In order to prove our analysis aforementioned, we list an example in Table 5. The sentences in the source language express both coordination and causality. Among the listed models: baseline, DocT, DocT+AS, online and offline, only the offline model using automatic selection and future context information shows coordination and causality in translation results, which is helpful to prove the effectiveness of our methods.

Table 5. Results of baseline and document-level NMT models

Src: **因为** 音乐 可以 帮 他 将 他 的 思维 妄想， 转换 成形 通过 他 的 想象力 和 创造力 变成 现实
Ref: **because** music allows him to take his thoughts and delusions **and** shape them through his imagination and his creativity , into reality .
Baseline: **because** music can help him think of his thinking , transform his imagination through his imagination and creativity .
DocT: **because** music can help him think of his thoughts , change their imagination through his imagination and creativity .
DocT+AS: **because** music can help him think of his mind as a delusion , through his imagination and creativity .
Ours(online): **because** music can help him turn his mind into a delusion of his imagination and his creativity .
Ours(offline): **because** music can help him think of his delusions , **and** turn it into his imagination and his creativity into reality .

6 Conclusion and future work

In this paper, we expand the source of context, and integrate the future context with the sentence to be translated, which is beneficial to the document-level NMT. In order to filter redundant information, we study the routing algorithm in MNMT, and propose a document-level NMT routing algorithm based on this algorithm. With *auto-selection*, the model together with the input is capable of deciding which words to use as context. According to the results of experiments, our *online* model achieves +1.39 BLEU improvement compared with the baseline on TED ZH-EN, which proves the effectiveness of *auto-selection* in document-level NMT; Combined with the future context, our model improves the BLEU by another 0.52 points, which proves that document-level NMT benefits from future contextual information. In addition, we also transplant our method to the previous document-level NMT work, which proves the universality of our method.

We still have a lot of work to do. For example, we do not achieve the expected results on the EN-DE datasets. These problems have already been mentioned

above. The lack of other supervision methods and information after the integration of deep coding are the key points that need to be solved in our future work. Besides, we will continue to study the universality of our method in other document-level NMT methods.

References

1. K. Cho, B. van Merriënboer, C. Gulcehre, D. Bahdanau, F. Bougares, H. Schwenk, and Y. Bengio, "Learning phrase representations using RNN encoder–decoder for statistical machine translation," in *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, (Doha, Qatar), pp. 1724–1734, Association for Computational Linguistics, Oct. 2014.
2. D. Bahdanau, K. Cho, and Y. Bengio, "Neural machine translation by jointly learning to align and translate," *arXiv preprint arXiv:1409.0473*, 2014.
3. M.-T. Luong, H. Pham, and C. D. Manning, "Effective approaches to attention-based neural machine translation," in *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*, pp. 1412–1421, 2015.
4. J. Zhang, H. Luan, M. Sun, F. Zhai, J. Xu, M. Zhang, and Y. Liu, "Improving the transformer translation model with document-level context," in *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pp. 533–542, 2018.
5. E. Voita, P. Serdyukov, R. Sennrich, and I. Titov, "Context-aware neural machine translation learns anaphora resolution," in *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 1264–1274, 2018.
6. R. Agrawal, M. Turchi, and M. Negri, "Contextual handling in neural machine translation: Look behind, ahead and on both sides," in *21st Annual Conference of the European Association for Machine Translation*, p. 11.
7. L. Guillou, C. Hardmeier, E. Lapshinova-Koltunski, and S. Loáiciga, "A pronoun test suite evaluation of the english–german mt systems at wmt 2018," *WMT 2018*, p. 570, 2018.
8. E. Voita, R. Sennrich, and I. Titov, "Context-aware monolingual repair for neural machine translation," in *EMNLP/IJCNLP (1)*, 2019.
9. Z. Zheng, X. Yue, S. Huang, J. Chen, and A. Birch, "Towards making the most of context in neural machine translation," in *IJCAI*, 2020.
10. E. Voita, R. Sennrich, and I. Titov, "When a good translation is wrong in context: Context-aware machine translation improves on deixis, ellipsis, and lexical cohesion," in *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pp. 1198–1212, 2019.
11. S. Jean, S. Lauly, O. Firat, and K. Cho, "Does neural machine translation benefit from larger context?," *arXiv preprint arXiv:1704.05135*, 2017.
12. L. Wang, Z. Tu, A. Way, and Q. Liu, "Exploiting cross-sentence context for neural machine translation," in *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, pp. 2826–2831, 2017.
13. L. Miculicich, D. Ram, N. Pappas, and J. Henderson, "Document-level neural machine translation with hierarchical attention networks," in *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pp. 2947–2954, 2018.

14. Z. Tu, Y. Liu, S. Shi, and T. Zhang, "Learning to remember translation history with a continuous cache," *Transactions of the Association for Computational Linguistics*, vol. 6, pp. 407–420, 2018.
15. Y. Kim, D. T. Tran, and H. Ney, "When and why is document-level context useful in neural machine translation?," in *Proceedings of the Fourth Workshop on Discourse in Machine Translation (DiscoMT 2019)*, pp. 24–34, 2019.
16. B. Zhang, A. Bapna, R. Sennrich, and O. Firat, "Share or not? learning to schedule language-specific capacity for multilingual translation,"
17. S. Maruf and G. Haffari, "Document context neural machine translation with memory networks," in *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 1275–1284, 2018.
18. J. Tiedemann and Y. Scherrer, "Neural machine translation with extended context," in *Proceedings of the Third Workshop on Discourse in Machine Translation*, pp. 82–92, 2017.
19. P. Koehn and R. Knowles, "Six challenges for neural machine translation," *ACL 2017*, p. 28, 2017.
20. S. Sukhbaatar, É. Grave, P. Bojanowski, and A. Joulin, "Adaptive attention span in transformers," in *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pp. 331–335, 2019.
21. S. Maruf, A. F. Martins, and G. Haffari, "Selective attention for context-aware neural machine translation," in *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pp. 3092–3102, 2019.
22. Z. Yang, J. Zhang, F. Meng, S. Gu, Y. Feng, and J. Zhou, "Enhancing context modeling with a query-guided capsule network for document-level translation," in *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pp. 1527–1537, 2019.
23. B. Li, H. Liu, Z. Wang, Y. Jiang, T. Xiao, J. Zhu, T. Liu, *et al.*, "Does multi-encoder help? a case study on context-aware neural machine translation," in *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pp. 3512–3518, 2020.
24. Q. Cao and D. Xiong, "Encoding gated translation memory into neural machine translation," in *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pp. 3042–3047, 2018.
25. S. Kuang and D. Xiong, "Fusing recency into neural machine translation with an inter-sentence gate model," in *Proceedings of the 27th International Conference on Computational Linguistics*, pp. 607–617, 2018.
26. S. Jiang, R. Wang, Z. Li, M. Utiyama, K. Chen, E. Sumita, H. Zhao, and B.-l. Lu, "Document-level neural machine translation with inter-sentence attention," *arXiv preprint arXiv:1910.14528*, 2019.
27. H. Xiong, Z. He, H. Wu, and H. Wang, "Modeling coherence for discourse neural machine translation," in *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 33, pp. 7338–7345, 2019.
28. A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, L. Kaiser, and I. Polosukhin, "Attention is all you need," in *NIPS*, 2017.
29. N. Arivazhagan, A. Bapna, O. Firat, D. Lepikhin, M. Johnson, M. Krikun, M. X. Chen, Y. Cao, G. Foster, C. Cherry, *et al.*, "Massively multilingual neural machine translation in the wild: Findings and challenges," *arXiv preprint arXiv:1907.05019*, 2019.

30. M. Cettolo, C. Girardi, and M. Federico, "WIT3: Web inventory of transcribed and translated talks," in *Proceedings of the 16th Annual conference of the European Association for Machine Translation*, (Trento, Italy), pp. 261–268, European Association for Machine Translation, May 28–30 2012.
31. P. Koehn, "Europarl: A parallel corpus for statistical machine translation," *Mt Summit*, vol. 5, 2008.
32. P. Koehn, H. Hoang, A. Birch, C. Callison-Burch, M. Federico, N. Bertoldi, B. Cowan, W. Shen, C. Moran, R. Zens, C. Dyer, O. Bojar, A. Constantin, and E. Herbst, "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*, (Prague, Czech Republic), pp. 177–180, Association for Computational Linguistics, June 2007.
33. R. Sennrich, B. Haddow, and A. Birch, "Neural machine translation of rare words with subword units," in *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 1715–1725, 2016.
34. M. Ott, S. Edunov, A. Baevski, A. Fan, S. Gross, N. Ng, D. Grangier, and M. Auli, "fairseq: A fast, extensible toolkit for sequence modeling," in *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics (Demonstrations)*, pp. 48–53, 2019.
35. D. Kingma and J. Ba, "Adam: A method for stochastic optimization," *Computer Science*, 2014.
36. M. Post, "A call for clarity in reporting bleu scores," in *Proceedings of the Third Conference on Machine Translation: Research Papers*, pp. 186–191, 2018.