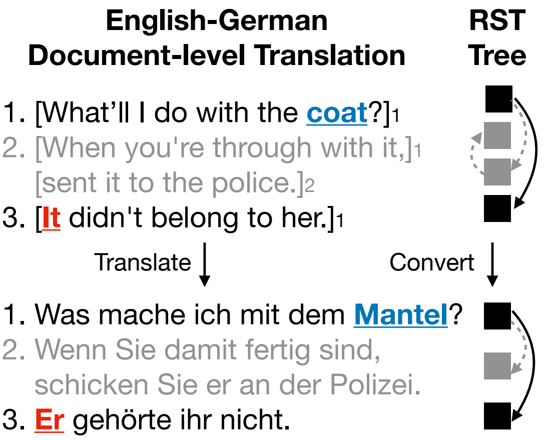
Document-level Neural Machine Translation Using Dependency RST Structure

Anonymous ACL submission

Abstract

Document-level machine translation (MT) extends the translation unit from the sentence to the whole document. Intuitively, discourse structure can be useful for document-level MT for its helpfulness in long-range dependency modelling. However, few efforts have been paid on leveraging discourse information for document-level neural machine translation(NMT). In this paper, we propose a dependency Rhetorical Structure Theory (RST) tree enhanced NMT model, RST-Transformer. The model only needs to encodes the dependency RST tree of the source document via the attention mask, and can enhance both the encoder and the decoder. Experiments on English-German datasets in both non-pretraining and pretraining settings show that our discourse information enhanced approach outperforms the current state-of-the-art document-level NMT model.



1. [What'll I do with the coat?]1
2. [When you're through with it,]1 [sent it to the police.]2
3. [It didn't belong to her.]1

1. Was mache ich mit dem Mantel?
2. Wenn Sie damit fertig sind, schicken Sie er an der Polizei.
3. Er gehörte ihr nicht.

Figure 1: Illustration of our RST-Transformer handling the cross-sentence dependency. Other Document-level NMT models need to encode sentences 1-3 to model the coreference between "coat" and "it" in sentence 1 and 3. The proposed discourse-aware model RST-Transformer selects text span {1-1, 3-1}. The right side of the figure illustrates the dependency RST tree we use, where the tree of target document can be converted from the tree of source.

1 Introduction

As the neural machine translation (NMT) (Bahdanau et al., 2014; Vaswani et al., 2017) gets close to human performance on the sentence-level translation, the mistakes at the document-level become more obvious (Kim et al., 2019). Previous work (Voita et al., 2019; Ma et al., 2021) shows these mistakes could be reduced by introducing contexts into context-agnostic NMT models.

Previous methods that explored to integrate context information into NMT models can be broadly divided into two categories. The first category takes fix scope sentences as context and modeling the context-aware representation by extra context encoder (Miculicich et al., 2018; Wang et al., 2017a) or unified encoder (Ma et al., 2020; Zheng et al., 2020). The second category takes global context into consideration by using translation memory (Kuang et al., 2018; Tu et al., 2018), hierarchical model (Tan et al., 2019; Maruf et al., 2019) or reinforcement learning (Kang et al., 2020). However, exploring structured information for document-level NMT has so far received relatively little attention (Xiaomian and Chengqing, 2020; Chen et al., 2020; Xu et al., 2020b). Structured information is proven be helpful for sentence-level MT in enforcing meaning preservation (Marcheggiani et al., 2018), handling data sparsity (Song et al., 2019) and modeling long-range dependencies (Zhang et al., 2019; Wu et al., 2017). In another aspect, the discourse structure helps many other document-level tasks like summarization (Xu et al., 2020a), sentiment analysis (Huber and Carenini, 2020), translation evaluation (Guzmán et al., 2014; Joty et al., 2017).

In this paper, we present RST-Transformer, a discourse-aware document-level NMT model build upon Transformer (Vaswani et al., 2017). To shorten long-range dependencies and mask unrelated sentences, we take dependency RST tree (Mann and Thompson, 1987), a structure describing how text spans in a document related to each other, as the indicator for guiding document-level MT. Figure 1 shows an example for cross-sentence dependencies, where the "It" in sentence 3 refers to "coat" in the sentence 1. In German, every noun has a gender (masculine, feminine, neuter) and the pronoun referring to it also use the same gender. For example, If the masculine noun "coat" is changed to a feminine noun, like "pants", the corresponding translation of "It" should be changed from "Er" to "Sie". Guided by the dependency RST tree, the RST-Transformer can discard redundant context and obtain additional capacity to include more long-range dependencies, producing fewer document-level mistakes.

More specifically, the RST-Transformer has two different attention modules: Sentence attention [1] (SentAttn) and RST attention (RSTAttn), which have sentence attention mask and RST attention mask, respectively. The sentence attention mask is used to differentiate the current sentence and its context. It makes the SentAttn focus on local sentences. The mask in RSTAttn is converted by the dependency RST tree of the source document. It makes the RSTAttn focus on spans related to the current span in the dependency RST tree. To fusion the sentence-level information encoded in SentAttn and document-level information encoded in the RSTAttn, we investigate three fusion methods: Serial, Parallel and Mix.

We evaluate our model on three commonly used document MT benchmark datasets for English-German translation. The results show that the RST-Transformer outperforms Transformer and current state-of-the-art document-level NMT models both on non-pretraining and pretraining settings. Further, we demonstrate that the RST-Transformer can still surpass other models without introducing the dependency RST tree in the decoding process.

2 Problem Definition

Formally, we denote $\mathbf{X} = \{X_1, X_2, \dots X_N\}$ as the source document with N sentences and $\mathbf{Y} = \{Y_1, Y_2, \dots Y_N\}$ as the target document with the same number of sentence. We assume that each source sentence X_i is aligned with the target sentence Y_i, where $i \in [1, N]$.

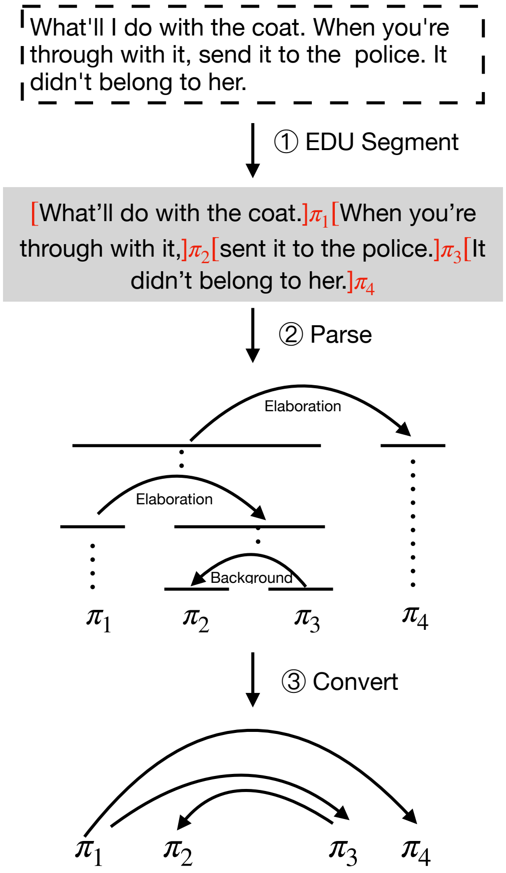

What'll I do with the coat. When you're through with it, send it to the police. It didn't belong to her.
① EDU Segment
[What'll do with the coat.]π_1[When you're through with it,]π_2[sent it to the police.]π_3[It didn't belong to her.]π_4
② Parse
Elaboration
Elaboration
Background
π_1
π_2
π_3
π_4
③ Convert
π_1
π_2
π_3
π_4

Figure 2: Example of discourse segmentation and RST tree conversion. The original document is segmented into 4 EDUs (π_1-π_4) in step ① and then parsed into an RST discourse tree in step ②. The dependency RST tree is converted from RST tree in step ③. Arcs in RST tree and dependency RST tree indicate modification.

Given the source document to translate, we assume a pair of source and target dependency RST trees (defined in Section 2.1) $\mathbf{T_X}$ and $\mathbf{T_Y}$ to help generate the target document. Therefore, the translation probability from $\mathbf{X}$ to $\mathbf{Y}$ can be represented as:

$$\hat{\mathbf{Y}} = \operatorname{argmax} P(\mathbf{Y}|\mathbf{X}, \mathbf{T_X}, \mathbf{T_Y}) \qquad (1)$$

2.1 Dependency RST Tree

In the RST framework, the discourse structure of the text is represented in a constituent tree structure. The dependency RST tree is the dependency perspective of the RST tree (Li et al., 2014; Morey et al., 2018), which is helpful in text summarization (Xu et al., 2020a; Hirao et al., 2013). The way to get the dependency RST tree of the input document is shown in Figure 2. In step ①, the document is split into contiguous, adjacent and non-overlapping text spans called Elementary Discourse Units (EDUs), a sub-sentence phrase unit originating from RST. In step ②, non-terminal

[1] It is analogous to GroupAttn in G-Transformer (Bao et al., 2021). G-Transformer is the state-of-the-art document-level NMT model, which encode the local and global attention by GroupAttn and GlobalAttn respectively.

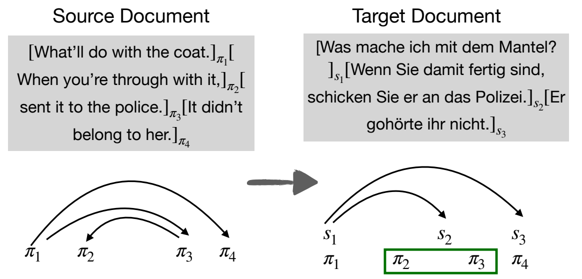

Source Document
[What'll do with the coat.]$_{\pi_1}$[When you're through with it,]$_{\pi_2}$[sent it to the police.]$_{\pi_3}$[It didn't belong to her.]$_{\pi_4}$
Target Document
[Was mache ich mit dem Mantel?]$_{s_1}$[Wenn Sie damit fertig sind, schicken Sie er an das Polizei.]$_{s_2}$[Er gohörte ihr nicht.]$_{s_3}$
π_1 π_2 π_3 π_4
s_1 s_2 s_3
π_1 π_2 π_3 π_4

Figure 3: Case of converting the dependency RST tree of the source document to the tree of the target document by merging EDUs.

nodes are composed of two or more adjacent EDUs merged upwards to form an RST tree. When merging, the more semantically important unit is called the "nucleus", and the other units that modify the "nucleus" are called "satellite". The arc from π_3 to π_2 means π_2 is a "satellite" modifying the "nucleus" π_3. The "nucleus-satellite" relationship can be further refined into a variety of rhetorical relationships such as elaboration and background. Finally, in step ③, the RST tree is converted to its dependency perspective (Li et al., 2014).

2.2 Trees for source and target documents

We use third-party tools to get the dependency RST tree for the source document. The details can be found in Section 4.2. However, the target document cannot obtain the dependency RST tree during decoding. The one-to-one correspondence between sentences in the source and target document makes the inter-sentence relationship between the two documents is consistent. Thus we can obtain a coarse-grained dependency RST tree for the target document. As shown in Figure 3, the tree for the target document can be converted by the tree of the source document by merging the EDUs in the same sentence and ignoring the inner-sentence relations. Zhang et al. (2021) empirically shows no EDU across sentences, and all EDUs in the same sentence should be exactly covered by a complete subtree. This ensures that leaf nodes could be sentences and the target document tree still forms a tree.

3 RST-Transformer

Our RST-Transformer extends from the Transformer NMT (Vaswani et al., 2017) architecture. Figure 4(a) provides an overview of the proposed model. The encoder and decoder are composed of N_a Sentence layers and N_b RST layers, respectively. According to Jawahar et al. (2019), lower layers of Transformer catch more local syntactic relations, while the higher layers represent long-distance relations. Based on this finding, we use the RST layer on the top layers for cross-sentence interaction and the Sentence layer on the lower layers for sentence-level information. Section 3.1 and Section 3.2 describes the Sentence and RST layers, respectively. Section 3.3 introduces three proposed methods to infuse local and global information in the RST layer.

3.1 Sentence Layer

Our sentence layer is the Transformer (Vaswani et al., 2017) considering sentence boundary. Roughly speaking, Transformer consists of two sublayers: self-attention network and feed-forward network. We integrate the sentence boundary information into the self-attention module. The attention function discussed maps a query and a set of key-value pairs to an output. The self-attention module produces representations by applying the attention function to each pair of tokens from the input sequence. Given a text representation $H \in \mathcal{R}^{T \times d}$:

$$Q = HW_Q,\ K = HW_K,\ V = HW_V \qquad (2)$$

$$\mathrm{Attn}(Q, K, V) = \mathrm{softmax}\left(\frac{QK^T}{\sqrt{d_k}}\right)V \qquad (3)$$

where the projection matrices $\{W_Q, W_K, W_k\} \in \mathcal{R}^{d \times d_k}$ are trainable parameters, T denotes the length of input tokens and d, d_k are the embedding size and hidden size, respectively.

The self-attention network can capture global contextual features. But too much information may cause training failure on the document-level translation task (Bao et al., 2021). Thus, we update Equation 3 by sentence boundary guiding, naming sentence attention (SentAttn). In addition to input Q, K and V, the sentence mask $M_s \in \mathcal{R}^{T \times T}$ is also involved:

$$\mathrm{SentAttn}(Q, K, V) = \mathrm{softmax}\left(\frac{QK^T}{\sqrt{d_k}} + M_s\right)V \qquad (4)$$

where the sentence mask M_s is an indicator that detects whether two tokens are in the same sentence. Specifically, M_s gives negative infinity for the token pairs not in the same sentence to make $softmax$ close to 0. Note that the SentAttn is same as the GroupAttn in Bao et al. (2021). For every two tokens x_i, x_j in the source document, we have:

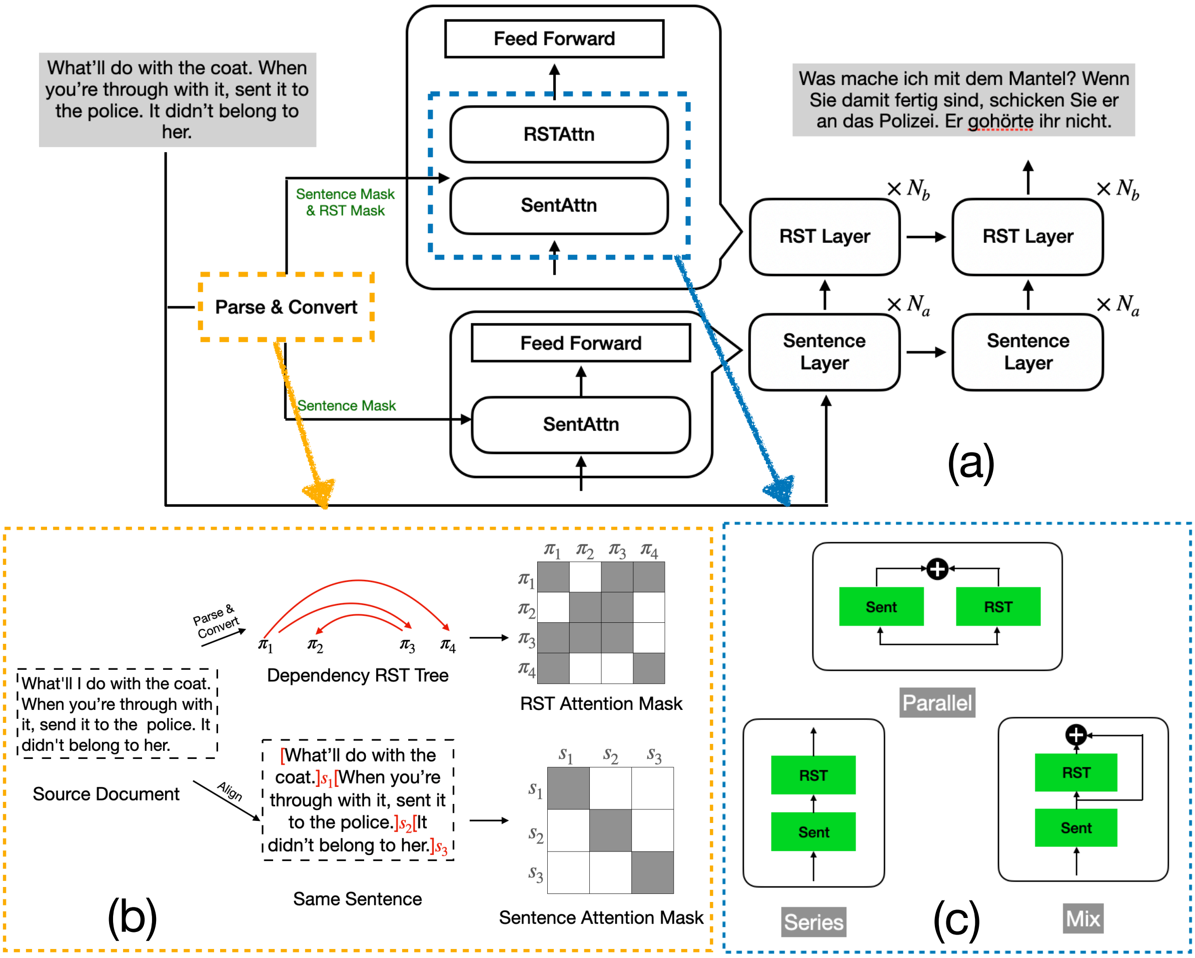

What'll do with the coat. When you're through with it, sent it to the police. It didn't belong to her.
Feed Forward
RSTAttn
SentAttn
Sentence Mask & RST Mask
Parse & Convert
Feed Forward
Sentence Mask
SentAttn
Was mache ich mit dem Mantel? Wenn Sie damit fertig sind, schicken Sie er an das Polizei. Er gohörte ihr nicht.
RST Layer
RST Layer
× N_b
× N_b
Sentence Layer
Sentence Layer
× N_a
× N_a
(a)
Parse & Convert
π_1 π_2 π_3 π_4
Dependency RST Tree
RST Attention Mask
What'll I do with the coat. When you're through with it, send it to the police. It didn't belong to her.
Source Document
Align
[What'll do with the coat.]s_1[When you're through with it, sent it to the police.]s_2[It didn't belong to her.]s_3
Same Sentence
s_1 s_2 s_3
Sentence Attention Mask
(b)
Sent
RST
Parallel
RST
Sent
Series
RST
Sent
Mix
(c)

Figure 4: (a) Model architecture of RST-Transformer. (b) The schematic of converting sentence boundary and dependency RST tree to attention masks. The white part in masks means the value is -∞ and the grey part means 0. They mean there is a constrain on attention or not, respectively. (c) The architecture of different fusion methods.

$$M_s(i,j) = \begin{cases} 0 & x_i, x_j \text{ in same sentence,} \\ -\infty & \text{otherwise.} \end{cases} \tag{5}$$

The lower half of Figure 4(b) shows an example of M_s. Due to the alignment assumption in Section 2, we can also get the sentence mask for the target document during training and decoding.

3.2 RST Layer

Our RST layer has two attention modules and one feed-forward module. One attention module is SentAttn defined in Equation 4, which captures local information. The other is RST attention (RSTAttn), used to capture the discourse structure enhanced global information.

Similarly, we update Equation 3 using the dependency RST tree. We involve RST mask $M_{rst} \in \mathcal{R}^{T\times T}$ into it:

$$\begin{aligned} &\mathrm{RSTAttn}(Q,K,V) = \\ &\qquad \mathrm{softmax}\left(\frac{QK^T}{\sqrt{d_k}} + M_{rst}\right)V \end{aligned} \tag{6}$$

where the RST mask M_{rst} is an indicator. If the EDUs where two tokens x_i and x_j located are directly related (parent or child) in the dependency RST tree or same, $M_{rst}(i,j)$ is set to 0 and negative infinity otherwise:

$$M_{rst}(i,j) = \begin{cases} 0 & x_i,\ x_j \text{ in same EDU} \\ & \text{or related,} \\ -\infty & otherwise. \end{cases} \tag{7}$$

The upper half of Figure 4(b) shows an example of M_{rst}. The dependency RST tree of the source and target document can be get from the Section 2.1.

Dataset	#Sent	#Documents
TED	0.21M/9K/2.3K	1.7K/92/23
News	0.24M/2K/3K	6K/81/155
Europarl	1.67M/3.6K/5.1K	118K/239/360

Table 1: Statistics of three datasets.

3.3 Fusion Methods for SentAttn and RSTAttn

To investigate what kind of structure is better for information fusion between SentAttn and RSTAttn, we propose three different RST-Transformer structures: Serial, Parallel and Mix. Their structural diagram is shown in Figure 4(c).

Given the source document representation $H \in \mathcal{R}^{T\times T}$, we can get the Q, K, V same as Equation 2:

- **Serial** connects SentAttnt and RSTAttn and in series:

$$H'_s = \text{SentAttn}(Q, K, V), \tag{8}$$

$$\{Q'_s, K'_s, V'_s\} = H'\{W'_Q, W'_K, W'_V\} \tag{9}$$

$$H' = \text{RSTAttn}(Q'_s, K'_s, V'_s), \tag{10}$$

where the projection matrices $\{Q'_s, K'_s, V'_s\}$ are trainable parameters.

- **Parallel** uses a gate-sum module (Zhang et al., 2016; Tu et al., 2017) to combine RSTAttn and SentAttn:

$$H'_{rst} = \text{RSTAttn}(Q, K, V), \tag{11}$$

$$g = \text{sigmoid}([H'_s, H'_{rst}]W + b), \tag{12}$$

$$H' = H'_s \odot g + H'_{rst} \odot (1 - g), \tag{13}$$

where the H'_s obtained by Equation 8, which is same as Serial, W and b are linear projection parameters, and $\odot$ denotes element-wise multiplication.

- **Mix** integrates the Parallel and Serial structures. We obtain H'_s and H'_{rst} by Equation 8-10, which are same as Serial. Then we combine H'_s and H'_{rst} to obtain the H' by a gate-sum module (Equation 12-13), which is same as Parallel.

4 Experiments

We compare our RST-Transformer with sentence-level Transformer and previous document-level NMT models on both non-pretraining and pretraining settings. We conduct experiments on three English-German (En-De) datasets. Following Liu et al. (2020), we calculate case-sensitive sentence BLEU (s-BLEU) and document BLEU (d-BLEU) as the metrics.

4.1 Datasets

Following previous work (Maruf et al., 2019), we use three En-De datasets as the benchmark to evaluate our method, which comes from three different domains: **TED** is transcriptions of TED talks from IWSLT 2017, **News** comes from News Commentary v11, **Europarl** is extracted from Europarl v7. The statistic of these datasets can be found in Table 1. We tokenize and truecase the sentences with MOSES (Koehn et al., 2007) tools, applying BPE (Sennrich et al., 2016) with 32000 merging operations. We split documents into instances with up to 512 tokens and ensure the integrity of the sentence.

4.2 Implement Details

We train our models on 4 GPUs of A100.

Hyper-parameters We use the Adam (Kingma and Ba, 2015) optimizer to train the models. The training strategy is the same as Bao et al. (2021), except for 1) the learning rate is $3e$-4, 2) the batch size is limited to 4096, 8192, 8192 for TED, News and Europarl, respectively. We search the batch size in [4096, 8192] and the learning rate in [$1e$-4, $3e$-4, $5e$-4]. We determine the number of updates/steps automatically by early stop on validation set. N_a is set to 4 and N_b is set to 2 for all experiments. Following previous work (Bao et al., 2021; Miculicich et al., 2018), we train the models in two stages. First we optimize the parameters for a Transformer model with sentence-level data. Then we use the Transformer to initialize our RST-Transformer and continuous train on the document-level data.

RST Discourse Parsing We first apply NeuralEDUSeg (Wang et al., 2018) to obtain EDUs. Then, we use StageDP (Wang et al., 2017b) to parse the segmented document. After obtaining the RST tree, we apply the algorithm proposed by Li et al. (2014) to convert the RST tree to its dependency perspective.

4.3 Baselines

We compare our model with the following NMT models:

Transformer is the base configuration Transformer NMT model in Vaswani et al. (2017) trained on

	TED		**News**		**Europral**	
	s-BLEU	d-BLEU	s-BLEU	d-BLEU	s-BLEU	d-BLEU
Transformer	24.80	-	25.10	-	31.37	-
HAN	24.58	-	25.03	-	28.60	-
SAN	24.42	-	24.84	-	29.75	-
Hybrid Context	25.10	-	24.91	-	30.40	-
Flat-Transformer	24.87	-	23.55	-	30.09	-
G-Transformer	25.12	27.17	25.47	27.08	32.39	34.08
RST-Transformer-Serial	25.55	27.75↑	25.83	27.38	32.43	34.16
RST-Transformer-Parallel	25.57	27.75↑	26.05↑	27.68↑	32.64	34.33
RST-Transformer-Mix	**25.61**	**27.84**↑	**26.43**↑	**28.00**↑	**32.87**↑	**34.64**↑
Fine-tune on mBART						
BART fine-tuned on sentence	27.78	-	29.90	-	31.87	-
BART fine-tuned on doc	-	28.29	-	30.49	-	34.00
G-Transformer + BART	28.06	30.03	30.34	31.71	32.74	34.31
RST-Transformer-Serial + BART	27.96	29.95	30.35	31.60	32.94	34.49
RST-Transformer-Parallel + BART	**28.29**	**30.26**	30.71	32.01	33.01	34.59
RST-Transformer-Mix + BART	28.26	30.23	**30.89**↑	**32.29**↑	**33.14**	**34.63**

Table 2: Case-sensitive BLEU scores on En-De translation. "↑" indicates statistically significant (Koehn, 2004) over the state-of-the-art G-Transformer at $p < 0.05$.

sentence-level data. And we use the model to initialize our RST-Transformer.

HAN (Miculicich et al., 2018) uses a hierarchical attention mechanism with two levels (word and sentence) of abstraction to incorporate context information from both source and target documents.

SAN (Maruf et al., 2019) considers both source and target documents by selecting relevant sentences as contexts from a document.

Hybrid Context (Zheng et al., 2020) uses a relative self-attention module to encode context at encoder and decoder.

Flat-Transformer (Ma et al., 2020) uses a unified encoder encode context and source sentence at same time and only encode source at top encoder layer.

G-Transformer (Bao et al., 2021) uses two attention modules to encode full document and local sentence at the top two layers. It is a special case of our RST-Transformer-Parallel structure when all EDUs connect to each other.

4.4 Results

Table 2 shows the overall results on three datasets. We find that our three fusion methods outperform the base model and previous document-level NMT models. Especially the RST-Transformer-Mix achieves the best results at all three datasets. the RST-Transformer-Mix can achieve improvements of 1.18 average s-BLEU scores over the context-agnostic Transformer. Therefore, the Mix is the most effective method of information fusion. It is worth noting that our RST-Transformer-Parallel model is better than G-Transformer on average by 0.63 s-BLEU scores and 0.71 d-BLEU scores, respectively. As mentioned above, G-Transformer can be regarded as the RST-Transformer-Parallel model without dependency RST tree. It proves that the RST structure can provide effective information.

There is an active topic about document-level MT using pretraining. We use mBART25 (Liu et al., 2020) to initialize our RST-Transformer and finetune it with learning rate $3e$-4. Taking advantage of sequence-to-sequence pretraining, the result of RST-Transformer-Mix+BART is 2.46 s-BLEU scores better than the RST-Transformer-Mix on average. Compared to the document-level state-of-the-art pretrained model G-Transformer+BART, our best model gives 0.55 s-BLEU score improvement on the News dataset. It shows the discourse structure information also enhances performance although in well-pretrained settings.

5 Analysis

In this section, we investigate our RST-Transformer to reveal its strengths and weaknesses in terms of (1) training without the Transformer initialization, (2) decoding without dependency RST tree, (3) changes in document phenomena of translations, (4) a case study. We use the Mix structure for

Size	Method	TED	News	Europarl
Base	Trans	0.42	0.45	30.40
	G-Trans	23.53	23.55	32.18
	our-Mix	**24.28**	**24.12**	**32.33**
Big	Trans	0.72	0.33	27.33
	G-Trans	23.29	22.22	32.04
	our-Mix	**23.99**	**23.97**	**33.25**
Large	Trans.	0.23	0.31	1.27
	G-Trans	6.23	13.68	31.51
	our-Mix	**13.22**	**20.04**	**31.99**

Table 3: s-BLEU on different model size. "Trans." is Transformer, "G-Trans" is G-Transformer, "our-Mix" is our RST-Transformer-Mix.

analysis because of its best performance.

5.1 Training From Scratch

Most of the current document-level NMT models are initialized by a pretrained sentence-level Transformer because this strategy can improve the performance. Bao et al. (2021) also shows the current transformer model prefers to stick around local minima without the initialization. The "Trans" rows in Table 3 show the s-BLEU score of Transformer trained on document-level datasets in different model sizes. Here the Base, Big model size is the same as the base, large configurations in (Vaswani et al., 2017) and the Large size is the same setting of BART large model (Lewis et al., 2020). As the table shows, we have a better performance on different model sizes. The performance drops slightly when increasing the model size from Base to Big. Further to the Large model, there is no sharp drop either. On the Large setting, the performance gap between ours and the G-Transformer reaches +6.99/+6.36 on the two small datasets (TED and News). The results indicate that our model can better prevent training failure.

5.2 Decoding without RST Tree

When decoding using RST-Transformer, an additional step is required to obtain the dependency RST tree of the source document. This is not expected in practical application. In order that our model can still model long-distance dependencies without the help of discourse information, we added a hyper-parameter $drop_{rst}$ in the training process. It means the RST attention mask M_{rst} is set to all zero with probability $p = drop_{rst}$. Figure 5 shows the results of RST-Transformer-Mix training with $drop_{rst} \in [0, 0.3, 0.5, 0.7, 1.0]$.

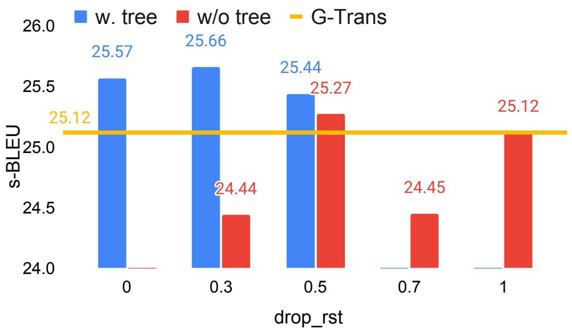


Figure 5: s-BLEU on the TED testset at different $drop_{rst}$ and decoding settings. "w. tree" means decoding with dependency RST trees and "w/o tree" is decoding without trees. The empty column, like the red column at position "0" means the result is lower than the 24.0 s-BLEU score.

When $drop_{rst} = 0$, our RST-Transformer-Mix can reach the best result with the help of the dependency RST tree. But it is failed without the tree. Once considering the absence of dependency RST tree during training ($drop_{rst} > 0$), our model can quickly adapt to this situation. Especially when $drop_{rst} = 0.5$, the RST-Transformer-Mix is better than the current state-of-the-art model even without the help of discourse structure. That means we only need to integrate discourse information during the training phase and the model could learn to model long-dependency itself.

5.3 Discourse Phenomena

We examine whether our models can capture discourse phenomena by evaluating our model on the consistency testsets (Voita et al., 2019). The consistency testsets evaluate the discourse phenomena include deixis, lexicon consistency, ellipsis (inflection and verb phrase) in English-Russian. Each testset contains contrastive examples consisting of a positive translation with correct discourse phenomenon and negative translations with incorrect phenomena. The goal is to determine whether a model is more likely to generate a correct translation than the incorrect variation. We follow the training setting proposed by Voita et al. (2019) and use both 6M sentence pairs and 1.5M document pairs from OpenSubtitles2018 (Lison et al., 2018) to train our model. As the results are shown in Table 5, our RST-Transformer-Mix obtains the best results.

5.4 Case Study

Table 4 shows an example of the deixis problem in En-De document-level MT, similar to Figure 1.

Context	S_{-2}: Two twin domes, two radically opposed design cultures.
	S_{-1}: One is made of thousands of steel parts, the other of a single silk thread.
Source	S_0: One is synthetic, **the other** organic.
	S_1: One is imposed on the environment, **the other** creates it.
	S_2: One is designed for nature, **the other** is designed by her.
Reference	R_0: die eine ist synthetisch, **die andere** organisch.
	R_1: eine wird der Umwelt auferlegt, **die andere** erschafft diese.
	R_2: die eine ist für die Natur entworfen, **die andere** wird durch sie erschaffen.
Transformer-sent	T_0: eine ist synthetisch, **das andere** organische.
	T_1: die eine wird auf die Umwelt auferlegt, **die andere** schafft es.
	T_2: man wird für die Natur gestaltet, **das andere** wird von ihr entworfen.
RST-Transfotmer-Mix	T_0: die eine ist synthetisch, **die andere** organische.
	T_1: die eine wird auf die Umwelt auferlegt, **die andere** schafft sie.
	T_2: die eine ist für die Natur gestaltet, **die andere** wird von ihr entworfen.

Table 4: An example of deixis problem in En-De. Given the context S_{-1} and S_{-2}, the phrase "the other" in S_{1-3} should be translate to "die andere" as shown in T_{1-3} because the "domes" it refers to is a feminine word in German.

Method	deixis	lexical	Ellipsis
sent	50.0	45.9	40.7
concat	83.5	47.5	76.4
MCN	61.3	46.1	48.3
G-Transformer	89.9	**83.6**	-
RST-Transformer-Mix	**91.7**	**83.6**	**79.1**

Table 5: Accuracy (%) of consistency testsets.

Given the context $S_{-1,-2}$, we need to translation three sentences S_{0-2}. As the Reference R_{0-2} shows, the pronoun phrase "the other" in the Source should be translated to the correct gender "die andere" because the noun "domes" it refers to is feminine. The Transformer-sent cannot consider the cross-sentence context and translate these pronouns into neutral "das andere" in T_0 and T_2. But our RST-Transformer-Mix keeps translating "the other" into "die andere" correctly, suggesting an effective capability of handling long-range dependencies.

6 Related Work

Structure-aware NMT is a well-studied topic in the sentence-level translation and the structure information is proven to be useful. Many approaches claim performance improvements by using treebank syntax (Sennrich and Haddow, 2016; Eriguchi et al., 2016; Bastings et al., 2017; Aharoni and Goldberg, 2017). Other approaches incorporate syntactic information in NMT models relatively indirectly (e.g. multi-task learning (Luong et al., 2015; Nadejde et al., 2017; Eriguchi et al., 2017; Hashimoto and Tsuruoka, 2017)). Unlike these syntactically-aware NMT methods mentioned above, Marcheggiani et al. (2018) and Song et al. (2019) investigate semantic role labeling and abstract meaning representation on NMT by GCN (Kipf and Welling, 2016) and GRN (Zhang et al., 2018) respectively.

On the contrary, it is still the mainstream practice to treat documents as plain text in document-level machine translation (Voita et al., 2018; Wang et al., 2019; Miculicich et al., 2018; Maruf et al., 2019; Xu et al., 2020b; Ma et al., 2020; Zheng et al., 2020; Liu et al., 2020; Bao et al., 2021). Only little work has done with structured information. Xu et al. (2020b) proposed a graph-based approach where graphs are constructed according to inter-sentential (adjacency, dependency) and intra-sentential (lexical consistency, coreference) relations. Compared with their work, we use the dependency RST tree supported by linguistic theory. and introduce the tree by attention mask. Xiaomian and Chengqing (2020) and Chen et al. (2020) also enrich input word embeddings with path embeddings based on RST trees. Unlike their methods, we integrate RST trees into the encoder and decoder of the conventional transformer model via an attention mask.

7 Conclusion

In this paper, we present a novel discourse structured information enhanced approach for document-level NMT. We modify the Transformer architecture to integrate the dependency RST tree of the source document by attention mask in encoder and decoder. Experiments show that our method gives state-of-the-art results compared to existing models under pretraining and non-pretraining settings. Our further analysis also shows that our model achieves nearly the same results even when decoding without the discourse information.

References

Roee Aharoni and Yoav Goldberg. 2017. Towards string-to-tree neural machine translation. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 132–140, Vancouver, Canada. Association for Computational Linguistics.

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

Guangsheng Bao, Yue Zhang, Zhiyang Teng, Boxing Chen, and Weihua Luo. 2021. G-transformer for document-level machine translation. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 3442–3455, Online. Association for Computational Linguistics.

Jasmijn Bastings, Ivan Titov, Wilker Aziz, Diego Marcheggiani, and Khalil Sima'an. 2017. Graph convolutional encoders for syntax-aware neural machine translation. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, pages 1957–1967, Copenhagen, Denmark. Association for Computational Linguistics.

Junxuan Chen, Xiang Li, Jiarui Zhang, Chulun Zhou, Jianwei Cui, Bin Wang, and Jinsong Su. 2020. Modeling discourse structure for document-level neural machine translation. *arXiv preprint arXiv:2006.04721*.

Akiko Eriguchi, Kazuma Hashimoto, and Yoshimasa Tsuruoka. 2016. Tree-to-sequence attentional neural machine translation. *arXiv preprint arXiv:1603.06075*.

Akiko Eriguchi, Yoshimasa Tsuruoka, and Kyunghyun Cho. 2017. Learning to parse and translate improves neural machine translation. *arXiv preprint arXiv:1702.03525*.

Francisco Guzmán, Shafiq Joty, Lluís Màrquez, and Preslav Nakov. 2014. Using discourse structure improves machine translation evaluation. In *Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 687–698.

Kazuma Hashimoto and Yoshimasa Tsuruoka. 2017. Neural machine translation with source-side latent graph parsing. *arXiv preprint arXiv:1702.02265*.

Tsutomu Hirao, Yasuhisa Yoshida, Masaaki Nishino, Norihito Yasuda, and Masaaki Nagata. 2013. Single-document summarization as a tree knapsack problem. In *Proceedings of the 2013 Conference on Empirical Methods in Natural Language Processing*, pages 1515–1520, Seattle, Washington, USA. Association for Computational Linguistics.

Patrick Huber and Giuseppe Carenini. 2020. Unsupervised learning of discourse structures using a tree autoencoder. *arXiv preprint arXiv:2012.09446*.

Ganesh Jawahar, Benoît Sagot, and Djamé Seddah. 2019. What does bert learn about the structure of language? In *ACL 2019-57th Annual Meeting of the Association for Computational Linguistics*.

Shafiq Joty, Francisco Guzmán, Lluís Màrquez, and Preslav Nakov. 2017. Discourse structure in machine translation evaluation. *Computational Linguistics*, 43(4):683–722.

Xiaomian Kang, Yang Zhao, Jiajun Zhang, and Chengqing Zong. 2020. Dynamic context selection for document-level neural machine translation via reinforcement learning. *arXiv preprint arXiv:2010.04314*.

Yunsu Kim, Duc Thanh Tran, and Hermann Ney. 2019. When and why is document-level context useful in neural machine translation? *arXiv preprint arXiv:1910.00294*.

Diederick P Kingma and Jimmy Ba. 2015. Adam: A method for stochastic optimization. In *International Conference on Learning Representations (ICLR)*.

Thomas N Kipf and Max Welling. 2016. Semi-supervised classification with graph convolutional networks. *arXiv preprint arXiv:1609.02907*.

Philipp Koehn. 2004. Statistical significance tests for machine translation evaluation. In *Proceedings of the 2004 Conference on Empirical Methods in Natural Language Processing*, pages 388–395, Barcelona, Spain. Association for Computational Linguistics.

Philipp Koehn, Hieu Hoang, Alexandra Birch, Chris Callison-Burch, Marcello Federico, Nicola Bertoldi, Brooke Cowan, Wade Shen, Christine Moran, Richard Zens, Chris Dyer, Ondřej Bojar, Alexandra Constantin, and Evan Herbst. 2007. Moses: Open source toolkit for statistical machine translation. In *Proceedings of the 45th Annual Meeting of the Association for Computational Linguistics Companion Volume Proceedings of the Demo and Poster Sessions*, pages 177–180, Prague, Czech Republic. Association for Computational Linguistics.

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*, pages 596–606, Santa Fe, New Mexico, USA. Association for Computational Linguistics.

Mike Lewis, Yinhan Liu, Naman Goyal, Marjan Ghazvininejad, Abdelrahman Mohamed, Omer Levy, Veselin Stoyanov, and Luke Zettlemoyer. 2020. BART: Denoising sequence-to-sequence pre-training for natural language generation, translation, and comprehension. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 7871–7880, Online. Association for Computational Linguistics.

Sujian Li, Liang Wang, Ziqiang Cao, and Wenjie Li. 2014. Text-level discourse dependency parsing. In *Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 25–35, Baltimore, Maryland. Association for Computational Linguistics.

Pierre Lison, Jörg Tiedemann, and Milen Kouylekov. 2018. Opensubtitles2018: Statistical rescoring of sentence alignments in large, noisy parallel corpora. In *Proceedings of the Eleventh International Conference on Language Resources and Evaluation (LREC 2018)*.

Yinhan Liu, Jiatao Gu, Naman Goyal, Xian Li, Sergey Edunov, Marjan Ghazvininejad, Mike Lewis, and Luke Zettlemoyer. 2020. Multilingual denoising pretraining for neural machine translation. *Transactions of the Association for Computational Linguistics*, 8:726–742.

Minh-Thang Luong, Quoc V Le, Ilya Sutskever, Oriol Vinyals, and Lukasz Kaiser. 2015. Multi-task sequence to sequence learning. *arXiv preprint arXiv:1511.06114*.

Shuming Ma, Dongdong Zhang, and Ming Zhou. 2020. A simple and effective unified encoder for document-level machine translation. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 3505–3511, Online. Association for Computational Linguistics.

Zhiyi Ma, Sergey Edunov, and Michael Auli. 2021. A comparison of approaches to document-level machine translation. *CoRR*, abs/2101.11040.

William C Mann and Sandra A Thompson. 1987. *Rhetorical structure theory: A theory of text organization*. University of Southern California, Information Sciences Institute Los Angeles.

Diego Marcheggiani, Jasmijn Bastings, and Ivan Titov. 2018. Exploiting semantics in neural machine translation with graph convolutional networks. *arXiv preprint arXiv:1804.08313*.

Sameen Maruf, André F. T. Martins, and Gholamreza Haffari. 2019. 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)*, pages 3092–3102, Minneapolis, Minnesota. Association for Computational Linguistics.

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

Mathieu Morey, Philippe Muller, and Nicholas Asher. 2018. A dependency perspective on RST discourse parsing and evaluation. *Computational Linguistics*, 44(2):197–235.

Maria Nadejde, Siva Reddy, Rico Sennrich, Tomasz Dwojak, Marcin Junczys-Dowmunt, Philipp Koehn, and Alexandra Birch. 2017. Predicting target language ccg supertags improves neural machine translation. *arXiv preprint arXiv:1702.01147*.

Rico Sennrich and Barry Haddow. 2016. Linguistic input features improve neural machine translation. *arXiv preprint arXiv:1606.02892*.

Rico Sennrich, Barry Haddow, and Alexandra Birch. 2016. 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)*, pages 1715–1725, Berlin, Germany. Association for Computational Linguistics.

Linfeng Song, Daniel Gildea, Yue Zhang, Zhiguo Wang, and Jinsong Su. 2019. Semantic neural machine translation using amr. *Transactions of the Association for Computational Linguistics*, 7:19–31.

Xin Tan, Longyin Zhang, Deyi Xiong, and Guodong Zhou. 2019. Hierarchical modeling of global context for document-level neural machine 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)*, pages 1576–1585, Hong Kong, China. Association for Computational Linguistics.

Zhaopeng Tu, Yang Liu, Zhengdong Lu, Xiaohua Liu, and Hang Li. 2017. Context gates for neural machine translation. *Transactions of the Association for Computational Linguistics*, 5:87–99.

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*, 6:407–420.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. In *Advances in neural information processing systems*, pages 5998–6008.

Elena Voita, Rico Sennrich, and Ivan Titov. 2019. When a good translation is wrong in context: Context-aware machine translation improves on deixis, ellipsis, and lexical cohesion. *arXiv preprint arXiv:1905.05979*.

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 (Volume 1: Long Papers)*, pages 1264–1274, Melbourne, Australia. Association for Computational Linguistics.

Longyue Wang, Zhaopeng Tu, Andy Way, and Qun Liu. 2017a. Exploiting cross-sentence context for neural machine translation. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, pages 2826–2831, Copenhagen, Denmark. Association for Computational Linguistics.

Xinyi Wang, Jason Weston, Michael Auli, and Yacine Jernite. 2019. Improving conditioning in context-aware sequence to sequence models. *arXiv preprint arXiv:1911.09728*.

Yizhong Wang, Sujian Li, and Houfeng Wang. 2017b. A two-stage parsing method for text-level discourse analysis. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 184–188, Vancouver, Canada. Association for Computational Linguistics.

Yizhong Wang, Sujian Li, and Jingfeng Yang. 2018. Toward fast and accurate neural discourse segmentation. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 962–967, Brussels, Belgium. Association for Computational Linguistics.

Shuangzhi Wu, Ming Zhou, and Dongdong Zhang. 2017. Improved neural machine translation with source syntax. In *Proceedings of the Twenty-Sixth International Joint Conference on Artificial Intelligence, IJCAI-17*, pages 4179–4185.

KANG Xiaomian and ZONG Chengqing. 2020. Fusion of discourse structural position encoding for neural machine translation. *CHINESE JOURNAL OF INTELLIGENT SCIENCE AND TECHNOLOGIE*, 2(2):144.

Jiacheng Xu, Zhe Gan, Yu Cheng, and Jingjing Liu. 2020a. Discourse-aware neural extractive text summarization. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 5021–5031, Online. Association for Computational Linguistics.

Mingzhou Xu, Liangyou Li, Derek Wong, Qun Liu, Lidia S Chao, et al. 2020b. Document graph for neural machine translation. *arXiv preprint arXiv:2012.03477*.

Liwen Zhang, Ge Wang, Wenjuan Han, and Kewei Tu. 2021. Adapting unsupervised syntactic parsing methodology for discourse dependency parsing. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 5782–5794.

Meishan Zhang, Zhenghua Li, Guohong Fu, and Min Zhang. 2019. Syntax-enhanced neural machine translation with syntax-aware word representations. 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)*, pages 1151–1161, Minneapolis, Minnesota. Association for Computational Linguistics.

Meishan Zhang, Yue Zhang, and Duy-Tin Vo. 2016. Gated neural networks for targeted sentiment analysis. In *Thirtieth AAAI conference on artificial intelligence*.

Yue Zhang, Qi Liu, and Linfeng Song. 2018. Sentence-state lstm for text representation. *arXiv preprint arXiv:1805.02474*.

Zaixiang Zheng, Xiang Yue, Shujian Huang, Jiajun Chen, and Alexandra Birch. 2020. Towards making the most of context in neural machine translation. *arXiv preprint arXiv:2002.07982*.