Clause Attention based on Signal Words Division

Anonymous ACL submission

Abstract

Clause Attention (CA) is very important for long sentences processing. We build and label datasets for signal word training. According to the position of signal word, the long sentences are divided into clauses which are assigned to additional block attention. The original sentence is mapped and fed into the shared encoder to learn the extraneous representation of words in its clause sentences. We use attention with prior to balance global attention with local attention. It improves the quality of long sentence processing in NER and NMT task.

1 Introduction

Long sentence translation still remains challenging for NMT due to various reasons (Cho et al., 2014). Directly inputting for translation is prone to insufficient length of long text, while extracting the key content of long text will lose some information. To deal with this problem, some of the existing machine translation models segment each long sentence into several parts in the processing (Cho et al., 2014; Pouget-Abadie et al., 2014; Kuang and Xiong, 2016). Memory compressed transformer (Liu* et al., 2018; Rae et al., 2019; Dubois et al., 2020; Tan et al., 2021) is one of the early attempts to make transformer better handle long sequences. It mainly modifies two parts: attention to location range and attention to memory compression. The former aims to divide the input sequence into modules with similar length, and run the self attention mechanism in each part, so that the attention cost of each part remains unchanged, and the activation times can be linearly scaled according to the input length. The latter uses step convolution to reduce the size of attention matrix and the amount of attention calculation, which depends on the stride length. Unfortunately, each part with similar length will break the clause structure, challenging to encode clause semantic details of a sentence into a fixed-size vector.

In the aspect of artificial translation, many scholars have put forward some skills for the translation of long and difficult English sentences. First of all, the main of sentence should be found. Besides, each sentence component is divided into clause. Finally, sentence components is combined in translation. In the process of manual translation of long sentences, the context words in some cohesive sentences are gradually classified as signal words by experience. Signal words are used as the symbol of breaking sentences, which is conducive to clarify the structure, effectively distribute attention, and finally integrate the clauses to achieve translation.

We compile human translation techniques into natural language processing. It is proposed to split the long sentences to form clauses based on signal words. We design clause attention(CA) model on the original model. Clauses are assigned to the additional mechanism. The original sentence and its each clause are shared encode for word embedding, location, attention mechanism and other coding combined with the label to decode. Finally we improve the outputting results of method in the quality of long sentence processing.

2 Background

2.1 Attention Mechanism

Attention mechanism is first proposed in NMT (Bahdanau et al., 2015), fully used in Transformer (Vaswani et al., 2017) and reviewed in a survey of Transformer (Lin et al., 2021). It is hot spot and common methods (Dai et al., 2019; Radford et al., 2019; Devlin et al., 2019). It can be seen that the development of attention mechanism is very fast. This success is partly due to the self-attention component which enables the network to capture contextual information from the entire sequence (Su et al., 2018). In this paper, we implement our method based on Transformer encoder-decoder framework, where the encoder first maps the input

sequence into a sequence of continuous representations and the decoder generates an output sequence from the continuous representations. The encoder and decoder are trained jointly to maximize the conditional probability of target sequence given a source sequence. The scaled dot-product attention used by Transformer is given in Equation (1).

$$\text{Attention}(Q,K,V) = \text{softmax}\left(\frac{QK^\top}{\sqrt{D_k}}\right)V \quad (1)$$

where $Q \in \mathbb{R}^{N\times D_k}$,$K \in \mathbb{R}^{M\times D_k}$,$V \in \mathbb{R}^{N\times D_k}$; N and M denote the lengths of queries and keys (or values); D_k and D_v denote the dimensions of keys (or queries) and values; softmax is applied in a row-wise manner. The dot-products of queries and keys are divided by $\sqrt{D_k}$ to alleviate gradient vanishing problem of the softmax function.

2.2 Block Local Attention

Self-attention plays an important role in Transformer. In the standard self-attention mechanism, every token needs to attend to all other tokens. In position-based sparse attention (Parmar et al., 2018; Tay et al., 2020), the attention matrix is limited according to some pre-defined patterns. Although these sparse patterns vary in different forms, we find that some of them can be decomposed into some atomic sparse patterns. This class of attention segments input sequence into several query blocks, each of which is associated with a local memory block. All the queries in a query block attend to only the keys in the corresponding memory block.

This class of attention segments input sequence into several query blocks, each of which is associated with a local memory block. All the queries in a query block attend to only the keys in the corresponding memory block. We then partition the length into query blocks Q of length l_q, padding with zeroes if necessary. We partition the input tensor with positional encoding into rectangular query blocks contiguous in the original sentence. We generate one query block after another, ordering the blocks in order. Within each block, we generate individual positions.

2.3 Attention with Prior

Attention mechanism generally outputs an expected attended value as a weighted sum of vectors, where the weights are an attention distribution over the values. However, it is observed that for the trained Transformers the learned attention matrix is often very sparse across most data points. Therefore, it is possible to reduce computation complexity by incorporating structural bias to limit the number of query-key pairs that each query attends to. Under this limitation, we just compute the similarity score of the query-key pairs according to pre-defined patterns in Equation (2).

$$\begin{aligned}\text{Attention}(Q_f,K_f,V_f) = \text{softmax}\left(\frac{Q_pK_p^\top}{\sqrt{D_{k_p}}}\right)V_p \\ \oplus \text{softmax}\left(\frac{Q_gK_g^\top}{\sqrt{D_{k_g}}}\right)V_g\end{aligned} \quad (2)$$

Where Q_g, K_g, V_g is calculated by the vector query value, key value, extraction value for global attention; Q_p, K_p, V_p is calculated by the vector query value, key value, extraction value for prior attention; Q_f, K_f, V_f is calculated by the vector query value, key value, extraction value for final attention; D_{k_g} is the dimension of K_g; D_{k_p} is the dimension of K_p.

3 Method

3.1 Signal Words Training

Inspired of NER model, we input the data into similar model for training, so that the model can automatically label signal words in long sentences.

Signal words dictionary. According to experience, we set signal words dictionary including punctuation marks (corresponding to Chinese commas, semicolon commas), conjunctions (and, or, but yet, for, when, as, since, out, before, after, because, although, so that, ...), relationship words (who, who, whose, whoever, what, why, why, where, how, why, ...), prepositions (in, on, with, of, to, ...), infinitive symbols (to), modified participle (past participle, now participle) and so on.

Signal words label. Due to the diversity of words and the influence of context, we can not match the text content alone, but should label the signal words in the data set in combination with part of speech, syntactic dependence and context. We use the English NER model(spaCy, Stanza and so on) to mark the part of speech of each word in the long sentence. Introducing the signal word dictionary, we judge the signal word in the sentence according to the word text and part of speech. We mark labels types [PUNCT], [SCONJ], [PRON] and [ADP] for

main of signal words. We also manually modify some labels that are difficult to be marked or easy to be wrong.

Signal words model. Signal words are function words while named entities are content words. A new fusion data set is constructed from the mutual exclusion of signal words and named entities. From the data distribution, there is a close relationship between signal words and named entities. We can transfer the signal word recognition model for named entity recognition or migrating named entity recognition model for signal word recognition. We can also do fusion model recognition.

Signal words division. We can think of a long sentence as multi-lever sequences of clauses and use signal word recognition model to separate long sentence into clause. We Split long sentences x into x_k^l clauses at l levels according to signal words. The split parts are relatively complete clauses, including main and subordinate sentences, compound sentences, prepositional guided clauses, etc. Each clause x_k^l has different length.

$$x_k^l = \{[x_1^1;x_2^1;...;x_{K_1}^1][x_1^2;x_2^2;...;x_{K_2}^2]..[x_1^l;x_2^l;...;x_{K_l}^l]\} \quad (3)$$

3.2 Self-attention Compute

To compute self-attention on the resulting long sentences, each clause can be encoded in a block. Each character assigns to self-attention by characters in its local block instead of characters in the whole sentence.The self-attention of global sentence and each clause is shown as Figure 1.

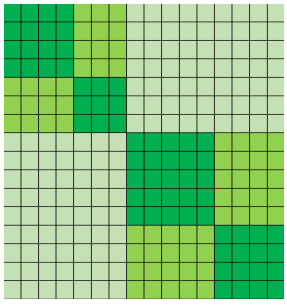

Figure 1: Self-attention of clause split.

In Figure 1, the attention weight assigned to each part is different. Signal words, as antecedents, pay more attention to the modified components of the guided clauses. The integration of each part integrates and distributes attention by means of subject-slave, juxtaposition, preposition and object according to the different signal words. Punctuation marks are split as the first level, followed by conjunctions, relational words, prepositions, infinitive symbols and modifier segmentation. We mark the attribute of each clause and replace the "signal word" in the sentence with "split level flag" + "signal word", so as to realize multi-level splitting into clauses.

For character embedding, we use BERT in NER task while Transformer in NMT task. The characters are trained to get vector x_i for global attention. The self-attention of each character in global is g_i which is calculated from x_i by Equation (1). In each clause block, the characters are trained to get vector $(x_i)_k^l$ for local attention. The self-attention of each character in global is $(b_i)_k^l$ which is calculated from $(x_i)_k^l$ by Equation (1). According to Equation (2), we take the value of the attention with prior as the final probability Y.

$$Y = g \oplus (b^1 \oplus (b^2 ... \oplus (b^{l-1} \oplus b^l)...)) \quad (4)$$

Where ${b_i}^l = (b_i)_1^l \cup (b_i)_2^l \cup ...(b_i)_{K_l}^l$, for attention of each block in each division of sentence; $b^l = {b_1}^l \cup {b_2}^l \cup ...b_{K_l}^l$, for attention of each division; $g = (g_1, g_2, ..., g_n)$, for global attention.

The clause attention is incorporated from local to global. Each incorporation $\oplus$ in Equation (4) is similarly shown in details in Equation (5).

$$b^{l-1} \oplus b^l = (1 - p_l) * b^{l-1} + p_l * b^l \quad (5)$$

Where $p_l \in [0, 1]$ is a calculated probability, which balances the probability of global attention and each local attention.

3.3 Design Clause Attention Model

We design the model in reference to existing model (Shaw et al., 2018; Zhang et al., 2020; Gao et al., 2020; Chen et al., 2020; Zhu et al., 2020). In the model, each clause is added to the encoder as an add-on module. It learns the order and connection from the input original sentence. Clauses are mapped to the original sentence which is fed into a shared encoder to learn the additional source representation of words. Then it introduces a multi-head attention module into the decoder to learn the context vector. The encoder and decoder are trained jointly to maximize the conditional probability of target sequence given a source sequence. The model is shown in Figure 2.

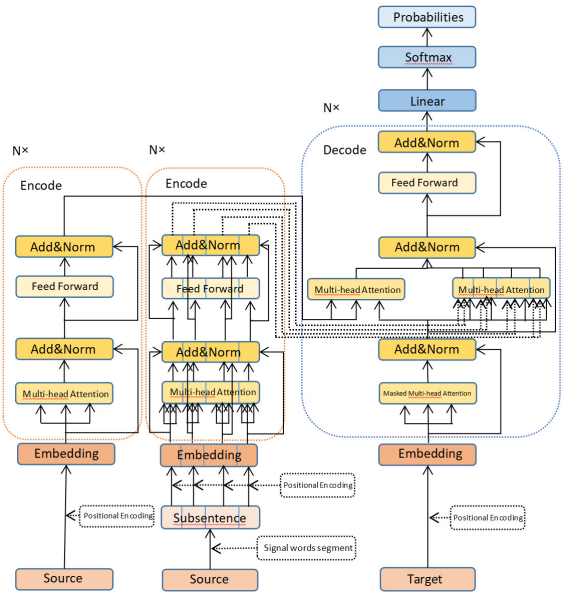

Figure 2: Clause attention model.

Encode. The original sentence and the split sentences are entered into the split model. The words are embedded to obtain the word vector, and then the position coding of each word is carried out. The attention mechanism for each split sentence focuses the attention of the words in the split sentence and reduces the interference of long sentences, maintaining a holistic connection to the attention mechanism of the original sentence. The key values K and the value V are eventually compiled for each word.

Decode. The label of the original sentence is entered into the split model. The words are embedded to obtain to get the word vector and position of each label. The query value Q of the label is obtained by using the attention mechanism. Each word output is decoded by Q key values K and numeric V, and then converted to output probability by softmax, which enables natural language processing.

4 Experiment

4.1 Setup

Datasets. We evaluate CA mainly on CoNLL-2003 for NER task and NiuTrans English-Chinese for NMT task. Besides, we also provided results on the WMT16 English-German and WMT15 English-Vietnamese task. The sentence pairs for NMT are shown in Table 1.

Evaluation. We use P, R and F1 to evaluate signal words training and our performance on CoNLL-2003 for NER task respectively. We use valid accuracy and BLEU to evaluate translation quality of English-Chinese and English-German, English-Vietnamese task respectively. We performer samples for significance test in English-Chinese task.

Task	Train	Valid	Test
NiuTrans English-Chinese	80k	10k	10k
WMT16 English-German,	20k	4.5k	4.5k
WMT15 English-Vietnamese	100k	10k	10k

Table 1: Sentence pairs for NMT.

Task	Batch Size	Epoch
NiuTrans English-Chinese	200	380
WMT16 English-German,	256	1600
WMT15 English-Vietnamese	160	168

Table 2: Batch size and epoch for NMT.

Model settings. For signal words training and NER model, we adopted similar settings as BERT-NER (Devlin et al., 2019). We download the specified pretrained BERT model provided by huggingface. We use BERT-Base for English task. For NMT model, we adopted similar settings as Transformer (Vaswani et al., 2017). The Batch Size and Epoch for NMT are shown in Table 2.

4.2 Results on CoNLL-2003

4.2.1 Results on Signal Word Training

We set labels of signal words on CoNLL-2003 datasets, comparing with different models as Transformer, Parallel RNN(Žukov-Gregorič et al., 2018) and BERT. Results are shown in Table 3.

Table 3 indicates our fusion model have a improvement over BERT on P, R and F1. The fusion model is based on multi-labels of signal words and name entities.

4.2.2 Compatibility with BERT in NER Task

We compare CA with BERT on CoNLL-2003 datasets. Results are shown in Table 4.

Table 4 indicates that the CoNLL-2003 data suppport 5598 tags incluing 2752 LOC, 1257 ORG,

Model	P	R	F1
Transformer	90.19	86.13	88.12
Žukov-Gregorič et al., 2018	89.10	87.94	88.51
BERT	**90.97**	**90.70**	**90.83**
Fusion model	**91.07**	**91.10**	**91.08**

Table 3: Results on signal words training.

Tag	P_{BRET}	P_{CA}	R_{BRET}	R_{CA}	$F1_{BRET}$	$F1_{CA}$	Support
LOC	93.73	92.07	92.93	93,90	93.33	92.97	1656
ORG	86.23	90.94	90.80	89.67	88.46	90.03	1642
PER	96.59	94.96	95.14	96.20	95.86	95.58	1606
MISC	79.64	78.39	82.85	82.45	81.21	80.36	694
avg / total	90.60	90.87	91.69	91.89	91.13	91.37	5598

Table 4: Our performance on CoNLL-2003 comparing with BERT-based.

Model	Valid accuracy	BLEU
Transformer	51.96	39.02
Wang et al. 2018	51.98	39.04
Beltagy et al. 2020	52.06	39.41
CA with rule	52.15	39.46
CA with study	52.19	40.03

Table 5: Our performance on NiuTrans English-Chinese comparing with Transformer and others.

1349 PER and 694 MISC. We find that, for tags ORG , CA+BERT can have a improvement over BERT on F1. But for others, it is opposite. Above all, the avg/total of CA+BERT can have a improvement over BERT on P, R and F1.

4.3 Results on English-Chinese Translation

Table 5 shows the translation results of different systems. CA with rule is splitting sentences by signal words dictionary and part of speech without training. CA with study is splitting sentences by signal words training model. Our outperforms have improvements over Transformer and other models (Wang et al., 2018; Beltagy et al., 2020).

Tables 6-9 show our performances on short, relative short, relative long and long sentences comparing with Transformer and Baidu. We find that, for short and relative short sentences, CA has a slight improvement, while for long and relative long sentences, CA has a significant improvement. However, some words in CA translation are inappropriate like "强烈的(strongly)" in Table 6. It is inappropriate to modify "互补性(complementary)". Transformer is worse for missing the translation of "complementary". Baidu does better in this sentence. Our performances are gradually improved with increasing the length of sentence in Table 7-9.

4.4 Results on English-German and English-Vietnamese Translation

From Table 5 and Table 10-11, we find that, for the same language family of English like German, CA has a slight improvement, while for different language families of English like Chinese or Vietnamese, CA has a significant improvement. The translation depends on the structure of language.

4.5 Discussion

We discuss CA model in three points with our experiments.

CA reduces sparse attention. For long sentence, the self-attention in Transformer is sparse and unbalanced of each character. We divide long sentence into several parts to keep the more important main attention of character with context and remove the unimportant sparse attention. It can be trained fast and reduce the parameters in model.

CA protects structure of clause. During the segment of long sentence, we provide a experienced way to locate the clause boundary. It simplifies the model to learn structure from large data. We also classify the types of signal words which is expediently encoded in the input.

CA enhances the input. With the additional attention of clause, the model can fast and better learn the structure of long sentence. The input enhances with relative short clause and balances the length or information of each part in long sentence.

5 Conclusion

In this work, we have presented a clause attention model(CA) depending on language structure for NER and NMT systems. Through signal words training and multiple splitting, CA enables clause attention information to guide what should be passed or suppressed from the encoder layer so as to make the learned distributed representations appropriate for high-level tasks. Our model is simple to implement and flexible to train. Experiments on CoNLL-2003 NER task and NiuTrans Chinese-English, WMT16 English-German and WMT15 English-Vietnamese translation tasks demonstrate the effectiveness of our model in improving both the name entities recognition and translation quality of long sentences.

References

Dzmitry Bahdanau, Kyunghyun Cho, and Yoshua Bengio. 2015. Neural machine translation by jointly learning to align and translate. In *3rd International Conference on Learning Representations, ICLR 2015, San Diego, CA, USA, May 7-9, 2015, Conference Track Proceedings*.

Model	Sentence	BLEU
Source	cross - strait scientific and technological exchanges and cooperation are strongly complementary .	-
Reference	海峡 两岸 在 科技 交流 合作 上 有 很 强 的 互补性 .	-
Transformer	两岸 科技 交流 与 合作 是 十分 强 的 .	0.70
Baidu	两岸 科技 交流 与 合作 互补性 强 .	0.78
CA	两岸 科技 交流 与 合作 具有 强烈 的 互补性 .	0.80

Table 6: Our performance on short sentence comparing with Transformer and Baidu.

Model	Sentence	BLEU
Source	if both sides of the strait can join hands to open up an international market in this field , fruitful achievements will surely be made .	-
Reference	如果 两岸 携手 在 这 方面 共同 开拓 国际 市场 , 定 有 丰硕 成果 .	-
Transformer	海峡 两岸 都 可以 共同 开放 国际 市场 , 这 方面 的 成果 必将 取得 丰硕 的 成果 .	0.58
Baidu	两岸 若能 携手 开拓 国际 市场 ， 必将 取得 丰硕 成果 .	0.67
CA	两岸 在 这 方面 可以 共同 开放 国际 市场 , 必将 取得 丰硕 成果 .	0.69

Table 7: Our performance on relative short sentence comparing with Transformer and Baidu.

Model	Sentence	BLEU
Source	the common prosperity of cross - strait academic , scientific and technological , economic , and cultural circles will benefit the people on both sides of the taiwan strait .	-
Reference	海峡 两岸 学术 科技 经济 文化 的 共同 繁荣 , 将会 使 两岸 人民 受益 .	-
Transformer	两岸 学术 , 科技 , 经济 , 文化 , 文化 各界 的 共同 繁荣 , 将 有利 於 两岸 人民 .	0.51
Baidu	两岸 若能 携手 开拓 国际 市场 ， 必将 取得 丰硕 成果 .	0.54
CA	两岸 学术界 科技界 经济界 文化界 的 共同 繁荣 ， 将 造福 两岸 人民 .	0.75

Table 8: Our performance on relative long sentence comparing with Transformer and Baidu.

Model	Sentence	BLEU
Source	we also deeply hope that young scientists on both sides of the strait will work hand in hand based on their same feelings toward china and the nation , create a competitive superiority in the international arena , and create a better future for the chinese people on both sides of the taiwan strait .	-
Reference	更 深切 期望 海峡 两岸 青年 科学家 能 在 中国 心 民族 情 的 共同 基础 上 , 心 手 相连 一起 打 拼 , 创造 在 国际间 竞争 优势 , 为 两岸 中国 人 开创 更 美好 的 未来 .	-
Transformer	我们 也 衷心 希望 两岸 青年 团结 起来 , 为 中国 和 民族 的 感情 , 在 国际 舞台 上 创造 竞争力 , 为 海峡 两岸 的 中国 人 创造 更好 的 未来 .	0.55
Baidu	我们 也 深切 希望 两岸 青年 科学家 基于 对 中国 、对 民族 的 共同 感情 ， 携手 合作 ， 在 国际 舞台 上 创造 竞争 优势 ， 为 两岸 中国 人民 创造 更加 美好 的 未来 .	0.56
CA	我们 也 衷心 希望 两岸 青年 科学家 携手 努力 , 以 中华 民族 为 中心 , 为 国际 创造 竞争 优势 , 为 两岸 中华 民族 创造 更好 的 未来 .	0.61

Table 9: Our performance on long sentence comparing with Transformer and Baidu.

Model	Valid accuracy	BLEU
Transformer	51.04	27.30
CA with rule	52.26	27.46
CA with study	52.29	27.49

Table 10: Our performance on English-German comparing with Transformer.

Model	Valid accuracy	BLEU
Transformer	56.73	23.03
CA with rule	57.03	23.34
CA with study	57.10	24.10

Table 11: Our performance on English-Vietnamese comparing with Transformer.

Iz Beltagy, Matthew E. Peters, and Arman Cohan. 2020. Longformer: The long-document transformer. *CoRR*, abs/2004.05150.

Kehai Chen, Rui Wang, Masao Utiyama, and Eiichiro Sumita. 2020. Content word aware neural machine translation. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 358–364, Online. Association for Computational Linguistics.

KyungHyun Cho, Bart van Merrienboer, Dzmitry Bahdanau, and Yoshua Bengio. 2014. On the properties of neural machine translation: Encoder-decoder approaches. *CoRR*, abs/1409.1259.

Zihang Dai, Zhilin Yang, Yiming Yang, Jaime G. Carbonell, Quoc V. Le, and Ruslan Salakhutdinov. 2019. Transformer-XL: Attentive language models beyond a fixed-length context. In *ACL*.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 4171–4186, Minneapolis, Minnesota. Association for Computational Linguistics.

Yann Dubois, Gautier Dagan, Dieuwke Hupkes, and Elia Bruni. 2020. Location Attention for Extrapolation to Longer Sequences. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 403–413, Online. Association for Computational Linguistics.

Yingqiang Gao, Nikola I. Nikolov, Yuhuang Hu, and Richard H.R. Hahnloser. 2020. Character-level translation with self-attention. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 1591–1604, Online. Association for Computational Linguistics.

Shaohui Kuang and Deyi Xiong. 2016. Automatic long sentence segmentation for neural machine translation. volume 10102, pages 162–174.

Tianyang Lin, Yuxin Wang, Xiangyang Liu, and Xipeng Qiu. 2021. A survey of transformers. *CoRR*, abs/2106.04554.

Peter J. Liu*, Mohammad Saleh*, Etienne Pot, Ben Goodrich, Ryan Sepassi, Lukasz Kaiser, and Noam Shazeer. 2018. Generating wikipedia by summarizing long sequences. In *International Conference on Learning Representations*.

Niki Parmar, Ashish Vaswani, Jakob Uszkoreit, Lukasz Kaiser, Noam Shazeer, and Alexander Ku. 2018. Image transformer. *CoRR*, abs/1802.05751.

Jean Pouget-Abadie, Dzmitry Bahdanau, Bart van Merrienboer, KyungHyun Cho, and Yoshua Bengio. 2014. Overcoming the curse of sentence length for neural machine translation using automatic segmentation. *CoRR*, abs/1409.1257.

Alec Radford, Jeffrey Wu, Rewon Child, David Luan, Dario Amodei, and Ilya Sutskever. 2019. Language models are unsupervised multitask learners.

Jack W. Rae, Anna Potapenko, Siddhant M. Jayakumar, and Timothy P. Lillicrap. 2019. Compressive transformers for long-range sequence modelling. *CoRR*, abs/1911.05507.

Peter Shaw, Jakob Uszkoreit, and Ashish Vaswani. 2018. Self-attention with relative position representations. *CoRR*, abs/1803.02155.

Jinsong Su, Jiali Zeng, Deyi Xiong, Yang Liu, Mingxuan Wang, and Jun Xie. 2018. A hierarchy-to-sequence attentional neural machine translation model. *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, 26(3):623–632.

Bowen Tan, Zichao Yang, Maruan Al-Shedivat, Eric Xing, and Zhiting Hu. 2021. Progressive generation of long text with pretrained language models. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 4313–4324, Online. Association for Computational Linguistics.

Yi Tay, Dara Bahri, Liu Yang, Donald Metzler, and Da-Cheng Juan. 2020. Sparse sinkhorn attention. *CoRR*, abs/2002.11296.

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

Qiang Wang, Bei Li, Jiqiang Liu, Bojian Jiang, Zheyang Zhang, Yinqiao Li, Ye Lin, Tong Xiao, and Jingbo Zhu. 2018. The NiuTrans machine translation system for WMT18. In *Proceedings of the Third Conference on Machine Translation: Shared Task Papers*, pages

528–534, Belgium, Brussels. Association for Computational Linguistics.

Biao Zhang, Deyi Xiong, and Jinsong Su. 2020. Neural machine translation with deep attention. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 42(1):154–163.

Junnan Zhu, Yu Zhou, Jiajun Zhang, and Chengqing Zong. 2020. Attend, translate and summarize: An efficient method for neural cross-lingual summarization. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 1309–1321, Online. Association for Computational Linguistics.

Andrej Žukov-Gregorič, Yoram Bachrach, and Sam Coope. 2018. Named entity recognition with parallel recurrent neural networks. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 69–74, Melbourne, Australia. Association for Computational Linguistics.