Improving Autoregressive NMT with Non-Autoregressive Model

Long Zhou[1,2], **Jiajun Zhang**[1,2], **Chengqing Zong**[1,2,3]
[1]National Laboratory of Pattern Recognition, CASIA, Beijing, China
[2]School of Artificial Intelligence, University of Chinese Academy of Sciences, Beijing, China
[3]CAS Center for Excellence in Brain Science and Intelligence Technology, Shanghai, China
{long.zhou, jjzhang, cqzong}@nlpr.ia.ac.cn

Abstract

Autoregressive neural machine translation (NMT) models are often used to teach non-autoregressive models via knowledge distillation. However, there are few studies on improving the quality of autoregressive translation (AT) using non-autoregressive translation (NAT). In this work, we propose a novel ***Encoder-NAD-AD*** framework for NMT, aiming at boosting AT with global information produced by NAT model. Specifically, under the semantic guidance of source-side context captured by the encoder, the non-autoregressive decoder (NAD) first learns to generate target-side hidden state sequence in parallel. Then the autoregressive decoder (AD) performs translation from left to right, conditioned on source-side and target-side hidden states. Since AD has global information generated by low-latency NAD, it is more likely to produce a better translation with less time delay. Experiments on WMT14 En⇒De, WMT16 En⇒Ro, and IWSLT14 De⇒En translation tasks demonstrate that our framework achieves significant improvements with only 8% speed degeneration over the autoregressive NMT.

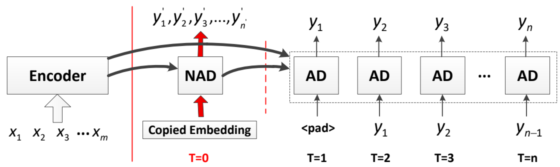

Figure 1: Decoding illustration of our proposed *Encoder-NAD-AD* framework including an encoder, non-autoregressive decoder (NAD) and autoregressive decoder (AD).

1 Introduction

Neural machine translation (NMT) based on *encoder-decoder* framework has gained rapid progress over recent years (Sutskever et al., 2014; Bahdanau et al., 2015; Wu et al., 2016; Gehring et al., 2017; Vaswani et al., 2017; Zhang and Zong, 2020). All these high-performance NMT models generate target languages from left to right in an autoregressive manner. An obvious limitation of autoregressive translation (AT) is that the inference process can hardly be parallelized, and the inference time is linear with respect to the length of the target sequence.

To speed up the inference of machine translation, non-autoregressive translation (NAT) models have been proposed, which generate all target tokens independently and simultaneously (Gu et al., 2017; Lee et al., 2018; Kaiser et al., 2018; Libovický and Helcl, 2018). Although NAT is successfully trained with the help from an AT model as its teacher via knowledge distillation (Kim and Rush, 2016), there is no work focusing on improving the quality of AT using NAT. Therefore, a natural question arises, *can we boost AT with NAT?*

In this paper, we propose a novel and effective *Encoder-NAD-AD* framework for NMT, in which the newly added non-autoregressive decoder (NAD) can provide target-side global information when autoregressive decoder (AD) translates, as illustrated in Figure 1. Briefly speaking, the encoder is first used to encode the source sequence into a sequence of vector representations. NAD then reads the encoder representations and generates a coarse target sequence in parallel. Given the source-side and target-side contexts separately captured by the encoder and NAD, AD learns to generate final translation token by token.

Our proposed model can fully combine two major advantages compared to previous work (Vaswani et al., 2017; Xia et al., 2017). On the one hand, due to the lower latency during inference of NAT, the decoding efficiency of our proposed framework is only slightly lower than

Proceedings of the 1st Workshop on Automatic Simultaneous Translation, pages 24–29
July 10, 2020.

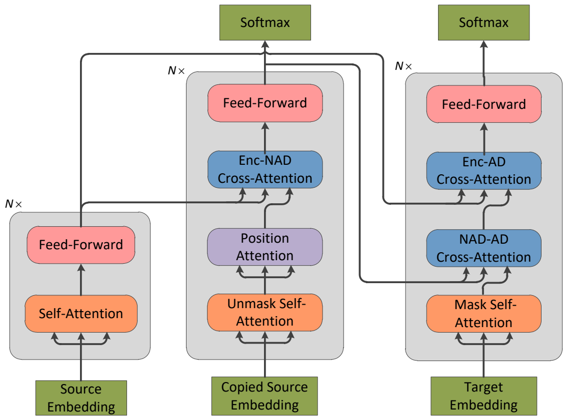

Figure 2: The extended Transformer translation model that exploits global information produced by NAT. We omit the residual connection and layer normalization in each sub-layer for simplicity.

the standard NMT models, as shown in Figure 1. On the other hand, since AD can asses the global target-side context provided by NAD, it has the potential to generate a better translation by fully exploiting source-side and target-side contexts. We conduct massive experiments on WMT14 En⇒De, WMT16 En⇒Ro and IWSLT14 De⇒En translations tasks. Experimental results demonstrate that our proposed model achieves substantial improvements with only 8% degradation in decoding efficiency compared to the standard NMT.

2 The Framework

Our goal in this work is to improve autoregressive NMT using the non-autoregressive model with lower latency during inference. Figure 2 shows the model architecture of the proposed framework. Next, we will detail individual components and introduce an algorithm for training and inference.

2.1 The Neural Encoder

The neural encoder of our model is identical to that of the dominant Transformer model, which is modeled using the self-attention network. The encoder is composed of a stack of N identical layers, each of which has two sub-layers:

$$\begin{aligned} \widetilde{h}^l &= \mathrm{LN}(h^{l-1} + \mathrm{MHAtt}(h^{l-1}, h^{l-1}, h^{l-1})) \\ h^l &= \mathrm{LN}(\widetilde{h}^l + \mathrm{FFN}(\widetilde{h}^l)) \end{aligned} \tag{1}$$

where the superscript l indicates layer depth, h^l denotes the source hidden state of l-th layer, LN is layer normalization, FFN means feed-forward networks, and MHAtt denotes the multi-head attention mechanism (Vaswani et al., 2017).

2.2 Non-Autoregressive Decoder

We initialize the non-autoregressive decoder inputs using copied source inputs from the encoder side by the fertility mechanism (Gu et al., 2017). For each layer in non-autoregressive decoder, the lowest sublayer is the unmasked multi-head self-attention network, and it also uses residual connections around each of the sublayers, followed by layer normalization.

$$z_1^l = \mathrm{LN}(z^{l-1} + \mathrm{MHAtt}(z^{l-1}, z^{l-1}, z^{l-1})) \tag{2}$$

The second sub-layer is a positional attention. We follow (Gu et al., 2017) and use the positional encoding p as both query and key and the decoder states as the value:

$$z_2^l = \mathrm{LN}(z_1^l + \mathrm{MHAtt}(z_1^l, p^l, p^l)) \tag{3}$$

The third sub-layer is Enc-NAD cross-attention that integrates the representation of corresponding source sentence, and the fourth sub-layer is a FFN:

$$\begin{aligned} z_3^l &= \mathrm{LN}(z_2^l + \mathrm{MHAtt}(z_2^l, h^N, h^N)) \\ z^l &= \mathrm{LN}(z_3^l + \mathrm{FFN}(z_3^l)) \end{aligned} \tag{4}$$

where h^N is the source hidden state of top layer.

2.3 Autoregressive Decoder

For each layer in autoregressive decoder, the lowest sub-layer is the masked multi-head self-attention network:

$$s_1^l = \mathrm{LN}(s^{l-1} + \mathrm{MHAtt}(s^{l-1}, s^{l-1}, s^{l-1})) \tag{5}$$

The second sub-layer is NAD-AD cross-attention that integrates non-autoregressive sequence context into autoregressive decoder:

$$s_2^l = \mathrm{LN}(s_1^l + \mathrm{MHAtt}(s_1^l, z^N, z^N)) \tag{6}$$

In addition, the decoder both stacks Enc-AD cross-attention and FFN sub-layers to seek task-relevant input semantics to bridge the gap between the input and output languages:

$$\begin{aligned} s_3^l &= \mathrm{LN}(s_2^l + \mathrm{MHAtt}(s_2^l, h^N, h^N)) \\ s^l &= \mathrm{LN}(s_3^l + \mathrm{FFN}(s_3^l)) \end{aligned} \tag{7}$$

2.4 Training and Inference

Given a set of training examples $\{x^{(z)}, y^{(z)}\}_{z=1}^{Z}$, the training algorithm aims to find the model parameters that maximize the likelihood of the training

#	System	Architecture	En⇒De	En⇒Ro	De⇒En
		Existing NAT Systems			
1	(Gu et al., 2017)	NAT	17.35	26.22	-
2	(Lee et al., 2018)	NAT-IR (adaptive)	18.91	-	-
3	(Wang et al., 2019)	NAT-AR	20.61	-	23.89
		Existing AT Systems			
4	(Wu et al., 2016)	Google-NMT	24.60	-	-
5	(Gehring et al., 2017)	ConvS2S	26.36	-	-
6	(Vaswani et al., 2017)	Transformer	27.30	-	-
7	(Xia et al., 2017)	Deliberate Network	27.56	33.18	33.95
		Our NMT Systems			
8		Transformer	27.06	32.28	32.87
9	*this work*	NAT	21.25	26.60	27.06
10		Our Model	**27.65**$^{\uparrow}$	**33.17**$^{\Uparrow}$	**34.01**$^{\Uparrow}$

Table 1: Comparing with existing NMT systems on WMT14 En⇒De, WMT16 En⇒Ro, and IWSLT14 De⇒En test sets. "↑/⇑" indicates statistically significant (p<0.05/0.01) from the Transformer baseline.

data:

$$J(\theta) = \frac{1}{Z}\sum_{z=1}^{Z}\{log\ P(y_{ad}^{(z)}|x^{(z)}, \theta_{enc}, \theta_{nad}, \theta_{ad}) + \lambda * \ log\ P(\widetilde{y}_{nad}^{(z)}|x^{(z)}, \theta_{enc}, \theta_{nad})\} \tag{8}$$

where $\widetilde{y}_{nad}$ is the reference of NAT, which can be obtained from standard NMT model via sequence-level knowledge distillation (Gu et al., 2017; Lee et al., 2018; Wang et al., 2019), and λ is a hyperparameter used to balance the preference between the two terms. Once our model is trained, we use the decoding algorithm shown in Figure 1 to translate source language with little time wasted over the autoregressive NMT.

3 Experiments

We use 4-gram NIST BLEU (Papineni et al., 2002) as the evaluation metric, and *sign-test* (Collins et al., 2005) to test for statistical significance.

3.1 Datasets

We conduct experiments on three widely used public machine translation corpora: WMT14 English-German[2] (En⇒De), WMT16 English-Romanian[3] (En⇒Ro), and IWSLT14 German-English[4] (De⇒En), whose training sets consist of 4.5M, 600K, 153K sentence pairs, respectively. We employ 37K, 40K, and 10K shared BPE (Sennrich et al., 2016) tokens for En⇒De, En⇒Ro, and De⇒en respectively. For En⇒De, we use `newstest2013` as the validation set and `newstest2014` as the test set. For En⇒Ro, we use `newsdev-2016` and `newstest-2016` as development and test sets. For De⇒En, we use 7K data split from the training set as the validation set and use the concatenation of `dev2010`, `tst2010`, `tst2011`, and `tst2012` as the test set, which is widely used in prior works (Bahdanau et al., 2017; Wang et al., 2019).

3.2 Model Settings

We build the described models modified from the open-sourced tensor2tensor[5] toolkit. For our proposed model, we employ the Adam optimizer with β_1=0.9, β_2=0.998, and ϵ=10^{-9}. For En⇒De and En⇒Ro, we use the hyperparameter settings of `base` Transformer model as Vaswani et al. (2017), whose encoder and decoder both have 6 layers, 8 attention-heads, and 512 hidden sizes. We follow Gu et al. (2017) to use the same `small` Transformer setting for IWSLT14 because of its smaller dataset. For evaluation, we use argmax decoding for NAD, and beam search with a beam size of k=4 and length penalty α=0.6 for AD. We also re-implement and compare with deliberate network (Xia et al., 2017) based on strong Transformer, which adopts the two-pass decoding method and uses the autoregressive decoding manner for the first decoder.

[2]http://www.statmt.org/wmt14/translation-task.html
[3]http://www.statmt.org/wmt16/translation-task.html.
[4]https://wit3.fbk.eu/
[5]https://github.com/tensorflow/tensor2tensor

Models	Latency	Degeneration
Transformer	251ms	0%
Deliberate Network	422ms	68%
NAT	16ms	(16× speedup)
Our Model	271ms	8%

Table 2: Decoding efficiency of different models. Latency is computed as average of per sentence decoding time on the test set of De⇒En.

3.3 Results and Analysis

In this section, we evaluate and analyze the proposed approach on En⇒De, En⇒Ro, and De⇒En translation tasks.

Model Complexity We first compare the model parameters and training speed in De⇒En for Transformer baseline, deliberate network, and our proposed model, which have 10.3M, 16.3M, and 18.0M parameters, respectively. Although our model uses more parameter than deliberate network due to additional position attention network, its training speed is significantly faster than deliberate network (1.8 steps/s vs. 0.7 steps/s)

Translation Quality We report the translation performance in Table 1, from which we can make the following conclusions: (1) Our proposed model (row 10) significantly outperforms Transformer baseline (row 8) by 0.59, 0.89, and 1.14 BLEU points in three translation tasks, respectively. (2) Compared to the existing deliberate network which uses greedy search for the one-pass decoding, our model can obtain a comparable performance. (3) Our NAT model (row 9) can achieve a competitive or even better model accuracy than previous NAT models (rows 1-3).

Decoding Speed Table 2 shows the decoding efficiency of different models. The deliberate network achieves the translation improvement at the cost of the substantial drop in decoding speed (68% degeneration). However, due to the high efficiency during inference of non-autoregressive models (16× speedup than Transformer), the decoding efficiency of our proposed framework is only slightly lower (8% degeneration) than the standard autoregressive Transformer models.

Case Study To better understand how our model works, we present a translation example sampled form De⇒En task in Table 3. The standard AT model incorrectly translates the phrase "geschrieben sein könnte" into "may be", and omits word "geschrieben". This problem is well addressed by the *Encoder-NAD-AD* framework, since AD can access the global information contained in the draft sequence generated by NAD, and therefore outputs a better sentence.

Source	ich sage dann mit meinen eigenen worten, was zwischen diesem gerüst geschrieben sein könnte .
Reference	then i will say , in my own words , what could be written within this framework .
AT	i then say to my own words , which *may be* between that framework .
NAT	i i say with my own words , which could be written between this scaffold .
Our Model	i then say , in my own words , what could be written between this framework ?

Table 3: Translation examples from De⇒En task. The *italic fonts* indicate the incomplete translation problem.

4 Related Work

There are many design choices in the *encoder-decoder* framework based on different types of layers, such as RNN-based (Sutskever et al., 2014), CNN-based (Gehring et al., 2017), and self-attention based (Vaswani et al., 2017) approaches. Particularly, relying entirely on the attention mechanism, the Transformer introduced by Vaswani et al. (2017) can improve the training speed as well as model performance.

In term of speeding up the decoding of the neural Transformer, Gu et al. (2017) modified the autoregressive architecture to directly generate target words in parallel. In past two years, non-autoregressive and semi-autoregressive models have been extensively studied (Oord et al., 2017; Kaiser et al., 2018; Lee et al., 2018; Libovický and Helcl, 2018; Wang et al., 2019; Guo et al., 2018; Zhou et al., 2019a). Previous work shows that NAT can be improved via knowledge distillation from AT models. In contrast, the idea of improving AT with NAT is not well explored.

The most relevant to our proposed framework is deliberation network (Xia et al., 2017), which leverages the global information by observing both back and forward information in sequence decoding through a deliberation process. Recently, Zhang et al. (2018) proposed asynchronous bidirectional decoding for NMT (ABD-NMT), which extended the conventional encoder-decoder framework by introducing a backward decoder. Different from ABD-NMT, synchronous bidirectional sequence generation model perform left-to-right decoding and right-to-left decoding simultaneously and interactively (Zhou et al., 2019b; Zhang et al., 2020). Besides, Geng et al. (2018) introduced a adaptive multi-pass decoder to standard NMT models. However, the above models improve translation quality while greatly reducing inference efficiency.

5 Conclusion

In this work, we propose a novel *Encoder-NAD-AD* framework for NMT, aiming at improving the quality of autoregressive decoder with global information produced by the newly added non-autoregressive decoder. We extensively evaluate the proposed model on three machine translation tasks (En⇒De, En⇒Ro, and De⇒En). Compared to existing deliberation network (Xia et al., 2017) which suffers from serious decoding speed degradation, our proposed model achieves a significant improvement in translation quality with little degradation of decoding efficiency compared to the state-of-the-art autoregressive NMT.

References

Dzmitry Bahdanau, Philemon Brakel, Kelvin Xu, Anirudh Goyal, Ryan Lowe, Joelle Pineau, Aaron Courville, and Yoshua Bengio. 2017. An actor-critic algorithm for sequence prediction. *In Proceedings of ICLR 2017*.

Dzmitry Bahdanau, Kyunghyun Cho, and Yoshua Bengio. 2015. Neural machine translation by jointly learning to align and translate. *In Proceedings of ICLR 2015*.

Michael Collins, Philipp Koehn, and Ivona Kučerová. 2005. Clause restructuring for statistical machine translation. In *Proceedings of the 43rd annual meeting on association for computational linguistics*, pages 531–540. Association for Computational Linguistics.

Jonas Gehring, Michael Auli, David Grangier, Denis Yarats, and Yann N. Dauphin. 2017. Convolutional sequence to sequence learning. In *Proceedings of the 34th International Conference on Machine Learning*, volume 70 of *Proceedings of Machine Learning Research*, pages 1243–1252, International Convention Centre, Sydney, Australia. PMLR.

Xinwei Geng, Xiaocheng Feng, Bing Qin, and Ting Liu. 2018. Adaptive multi-pass decoder for neural machine translation. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 523–532, Brussels, Belgium. Association for Computational Linguistics.

Jiatao Gu, James Bradbury, Caiming Xiong, Victor OK Li, and Richard Socher. 2017. Non-autoregressive neural machine translation. In *Proceedings of ICLR 2017*.

Junliang Guo, Xu Tan, Di He, Tao Qin, Linli Xu, and Tie-Yan Liu. 2018. Non-autoregressive neural machine translation with enhanced decoder input. In *Proceedings of AAAI 2019*.

Łukasz Kaiser, Aurko Roy, Ashish Vaswani, Niki Pamar, Samy Bengio, Jakob Uszkoreit, and Noam Shazeer. 2018. Fast decoding in sequence models using discrete latent variables. In *Proceedings of ICML 2018*.

Yoon Kim and Alexander M. Rush. 2016. Sequence-level knowledge distillation. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, pages 1317–1327. Association for Computational Linguistics.

Jason Lee, Elman Mansimov, and Kyunghyun Cho. 2018. Deterministic non-autoregressive neural sequence modeling by iterative refinement. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 1173–1182, Brussels, Belgium. Association for Computational Linguistics.

Jindřich Libovický and Jindřich Helcl. 2018. End-to-end non-autoregressive neural machine translation with connectionist temporal classification. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 3016–3021, Brussels, Belgium. Association for Computational Linguistics.

Aaron van den Oord, Yazhe Li, Igor Babuschkin, Karen Simonyan, Oriol Vinyals, Koray Kavukcuoglu, George van den Driessche, Edward Lockhart, Luis C Cobo, Florian Stimberg, et al. 2017. Parallel wavenet: Fast high-fidelity speech synthesis. *arXiv preprint arXiv:1711.10433*.

Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu. 2002. Bleu: a method for automatic evaluation of machine translation. In *Proceedings of the 40th annual meeting on association for computational linguistics*, pages 311–318. Association for Computational Linguistics.

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. Association for Computational Linguistics.

Ilya Sutskever, Oriol Vinyals, and Quoc V Le. 2014. Sequence to sequence learning with neural networks. In Z. Ghahramani, M. Welling, C. Cortes, N. D. Lawrence, and K. Q. Weinberger, editors, *Advances in Neural Information Processing Systems 27*, pages 3104–3112. Curran Associates, Inc.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Ł ukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. In I. Guyon, U. V. Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, and R. Garnett, editors, *Advances in Neural Information Processing Systems 30*, pages 5998–6008. Curran Associates, Inc.

Yiren Wang, Fei Tian, Di He, Tao Qin, ChengXiang Zhai, and Tie-Yan Liu. 2019. Non-autoregressive machine translation with auxiliary regularization. In *Proceedings of AAAI 2019*.

Yonghui Wu, Mike Schuster, Zhifeng Chen, Quoc V Le, Mohammad Norouzi, Wolfgang Macherey, Maxim Krikun, Yuan Cao, Qin Gao, Klaus Macherey, Jeff Klingner, Apurva Shah, Melvin Johnson, Xiaobing Liu, Lukasz Kaiser, Stephan Gouws, Yoshikiyo Kato, Taku Kudo, Hideto Kazawa, Keith Stevens, George Kurian, Nishant Patil, Wei Wang, Cliff Young, Jason Smith, Jason Riesa, Alex Rudnick, Oriol Vinyals, Greg Corrado, Macduff Hughes, and Jeffrey Dean. 2016. Google's neural machine translation system: Bridging the gap between human and machine translation. *arXiv preprint arXiv:1609.08144*.

Yingce Xia, Fei Tian, Lijun Wu, Jianxin Lin, Tao Qin, Nenghai Yu, and Tie-Yan Liu. 2017. Deliberation networks: Sequence generation beyond one-pass decoding. In I. Guyon, U. V. Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, and R. Garnett, editors, *Advances in Neural Information Processing Systems 30*, pages 1784–1794. Curran Associates, Inc.

Jiajun Zhang, Long Zhou, Yang Zhao, and Chengqing Zong. 2020. Synchronous bidirectional inference for neural sequence generation. *Artif. Intell.*, 281:103234.

Jiajun Zhang and Chengqing Zong. 2020. Neural machine translation: Challenges, progress and future. volume abs/2004.05809.

Xiangwen Zhang, Jinsong Su, Yue Qin, Yang Liu, Rongrong Ji, and Hongji Wang. 2018. Asynchronous bidirectional decoding for neural machine translation. *In Proceedings of AAAI 2018*.

Long Zhou, Jiajun Zhang, Heng Yu, and Chengqing Zong. 2019a. Sequence generation: From both sides to the middle. In *Proceedings of IJCAI 2019*.

Long Zhou, Jiajun Zhang, and Chengqing Zong. 2019b. Synchronous bidirectional neural machine translation. *Transactions of the Association for Computational Linguistics*, 7:91–105.