Bidirectional Modeling for Simultaneous Neural Machine Translation

Anonymous ACL submission

Abstract

Simultaneous Neural Machine Translation (SimulNMT) generates the output before the entire input sentence is available and only uses the unidirectional attention from left-to-right so that its decoding highly relies on future forecast according to word ordering rules. However, it is utopian that the word order strictly obeys the grammar rules in a language, especially in oral. To address the mismatch between SimulNMT expecting strict word order and free word order in real scenario, we propose a bidirectional modeling. In detail, we train another backward model where the input sentence is from right-to-left and keep the target sentence from left-to-right. Then we join this backward model into the standard forward SimulNMT model during decoding. This strategy enhances the robustness of SimulNMT and empowers the model to be more adaptable for the inconstant word ordering phenomenon. Experiments show that our method brings improvement over the strong baselines.

1 Introduction

Neural Machine Translation (NMT), built on the encoder-decoder framework has achieved advanced translation performance in recent years other than the traditional statistical machine translation (Kalchbrenner and Blunsom, 2013; Sutskever et al., 2014; Cho et al., 2014; Bahdanau et al., 2015; Vaswani et al., 2017). Inside an NMT model, the encoder encodes a source input sentence, and the decoder generates the target language sentence by iteratively predicting the output token according to the entire input with the partially decoded output so far. However, these *offline* models mentioned above are not well adaptable for real-time speech-to-speech interpretation, such as international conferences, symposiums, and business. Thus, *online* (or simultaneous) NMT is quite desirable for such scenarios, which starts the decoding process right after reading the first few words of the source sentence instead of waiting for the end.

SimulNMT has caused widespread concern in the NMT community recently (Cho and Esipova, 2016; Jaitly et al., 2016; Dalvi et al., 2018; Ma et al., 2019; Zheng et al., 2019a; Zhang et al., 2019; Arivazhagan et al., 2019; Ma et al., 2020; Ren et al., 2020; Elbayad et al., 2020). Ma et al. (2019) propose a popular wait-k decoding algorithm where the decoding process is always k words after the source input instead of single read-writes. This simple approach guarantees the translation quality and controls the translation delay at the same time. For dynamic online decoding, reinforcement learning (RL) and imitation learning (IL) are also used to optimize the read/write policy (Grissom II et al., 2014; Luo et al., 2017; Gu et al., 2017; Press and Smith, 2018; Zheng et al., 2019b).

However, all of the methods are decoded from left-to-right without the future information, ignoring that word order is flexible so that the resulting translations cannot always obey the grammar rules in practical use, especially in oral. The SimulNMT performance may be greatly hindered by the mismatch about the word order forecast between the requirement of SimulNMT and the actual scenario in linguistics..

To address such a mismatch issue in the current SimulNMT, we propose a bidirectional modeling strategy in this work. In detail, we train another *backward* model, in contrast to the *forward* model shown in the Figure 1, which inputs the source sentence from right-to-left and keeps the target sentence left-to-right order. Then we joint this *backward* model into the *forward* model during decoding. This decoding policy enhances the robustness of SimulNMT and allows the model to be more adaptable for the inconstant word order phenomenon. Experiments show that our method significantly improves the translation performance

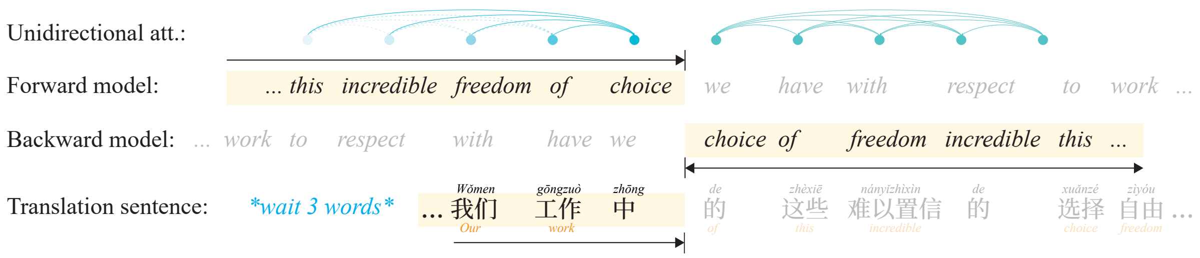


Figure 1: The example for *forward* model and *backward* model

over the strong baselines.

2 Backward Modeling as Future Forecasting

2.1 Problem Formalization

Given a source sentence $\mathbf{x} = \{x_1, ..., x_i, ..., x_{L_x}\}$ in the document to be translated and a target sentence $\mathbf{y} = \{y_1, ..., y_i, ..., y_{L_y}\}$, we denote $\mathbf{x}_{\leq t}$ as the a substring of $\mathbf{x}$ containing words $\{x_1, ..., x_t\}$, and similarly for $\mathbf{y}_{\leq t}$ and $\mathbf{y}_{<t}$. The NMT model computes the probability of translation from the source sentence to the target sentence word by word:

$$P(\mathbf{y}|\mathbf{x}) = \prod_{t=1}^{L_y} P(y_t|\mathbf{y}_{<t}, \mathbf{x}), \tag{1}$$

In this paper, we focus on the SimulNMT model based on the Transformer (Vaswani et al., 2017) which contains an encoder and a decoder and respectively processes the source and target sentences. Both are composed of a stack of N (usually equal to 6) identical layers. The critical component is multi-headed attention, which concatenates the outputs from multiple attention heads.

2.2 Simultaneous Neural Machine Translation (SimulNMT)

SimulNMT starts decoding the translation before the entire input sentence is available. Formally, we use z_t to represent the number of source tokens read when decoding y_t. In the SimulNMT model, the decoder predicts y_t by considering the first z_t source states, and each source state only encodes the information from the z_{t-1} source tokens read so far.

Unidirectional Transformer Encoder In the most encoder-decoder model, encoding the source tokens at a given position includes information from the past and future time-steps. However, the encoder has to be recomputed when the new source token is inputted. To reduce the cost of re-encoding the source sequence, Elbayad et al. (2020) propose unidirectional encoders for SimulNMT by masking the self-attention and only consider the previous time-steps. In this way, source sentences are encoded once without updating the encoder states at each time step.

Wait-k Strategy Human simultaneous interpretation usually starts translating a few seconds after the speakers' speech and finishes a few seconds accordingly after the speaker finishes. Inspired by this, Ma et al. (2019) present a wait-k policy, which first waits for the k source tokens and then translates simultaneously with the rest of the source sentence. When $k = \infty$, the full source sentence is read before decoding. For a wait-k decoding path, $z_t = \min\{k + t - 1, L_x\}$. The SimulNMT model computes the probability with regard to the single wait-k decoding path $\mathbf{z}^k$:

$$P(\mathbf{y}|\mathbf{x}, \mathbf{z}^k) = \prod_{t=1}^{L_y} P(y_t|\mathbf{y}_{<t}, \mathbf{x}_{\leq \mathbf{z}_t^k}, \mathbf{z}_{<t}^k) \tag{2}$$

The wait-k strategy is most effective when trained for the specific k (Zheng et al., 2019b). However, it requires training models individually for each potential value of k for translation.

2.3 Backward Modeling

Current SimulNMT methods translate the output tokens word by word from left to right, which is denoted as *forward* model. Considering the flexible word order phenomenon, we train another *backward* model by the contrast, which takes the source sentence as input from right to left, and keeps the target sentence in standard left-to-right order illustrated in Figure 2. It is worth noting that we adopt unidirectional self-attention in the *forward* modeling as in the standard SimulNMT, but a bidirectional self-attention in the *backward* modeling.

Task	wait-k	Zheng et al.	Elbayad et al.	This work
IWSLT14 En→De	26.74	–	26.40	27.39 (↑0.65)
IWSLT14 De→En	30.15	30.17	30.48	32.13 (↑1.65)
IWSLT15 En→Vi	28.31	–	29.19	29.98 (↑0.79)
IWSLT15 Vi→En	21.89	–	22.32	23.20 (↑0.88)

Table 1: The results of our proposed models for $k_{train} = k_{eval} = 7$.

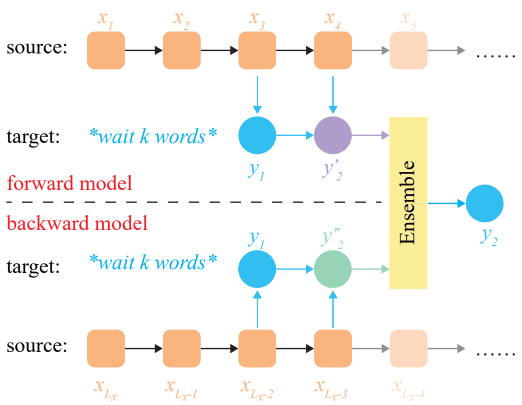


Figure 2: The framework of our proposed model using wait-k strategy and $k = 3$

On the one hand, *backward* modeling allows the whole to be exposed to a new order, and on the other hand, bidirectional modeling can bring more comprehensive features as an aid to normal *forward* SimulNMT. The probability of the translation in the *backward* model is calculated as follows:

$$P'(\mathbf{y}|\mathbf{x},\mathbf{z}^k) = \prod_{t=1}^{L_y} P(y_t|\mathbf{y}_{<t}, \mathbf{x}_{>(L_x-\mathbf{z}_t^k)}, \mathbf{z}_{<t}^k) \quad (3)$$

Then we train an ensemble model to joint this *backward* model output into the *forward* model during decoding.

$$P_{ensemble} = w_1 \otimes P_{backward} + w_2 \otimes P_{forward} \quad (4)$$

where w_1 and w_2 respectively mean the weights of the *backward* and the *forward* models. In our work, we set $w_1 = 0.1$ and $w_2 = 1.0$ based on the preliminary experiments.

3 Experiments

We briefly denote English, German, Vietnamese as En, De, and Vi respectively and conduct our simulNMT experiments on two small-scale datasets: IWSLT14 En↔De (Cettolo et al.) and IWSLT15 En↔Vi (Luong et al., 2015)[1], and a large-scale dataset: WMT15 En→De translation. We train a *forward* and *backward* models individually for each language pair.

[1]The tokenized data is downloaded from `https://nlp.stanford.edu/projects/nmt/`

3.1 Setup

Datasets For IWSLT14 En↔De, following (Edunov et al., 2018), we train on 160K pairs and randomly selected 7K sentences for validation and held-out from the training corpus, and the test set is the concatenation of *dev2010*, *dev2012*, *tst2010*, *tst2011* and *tst2012* of 6,750 pairs similar to the validation set. For IWSLT15 En↔Vi, like (Ma et al., 2020), we train on 133K pairs and use *tst2012* (1,553 pairs) as the validation set and *tst2013* (1,268 pairs) as the test set. All data is tokenized, lower-cased, and segmented with a byte-pair encoding (BPE) of 10K types (Sennrich et al., 2016).

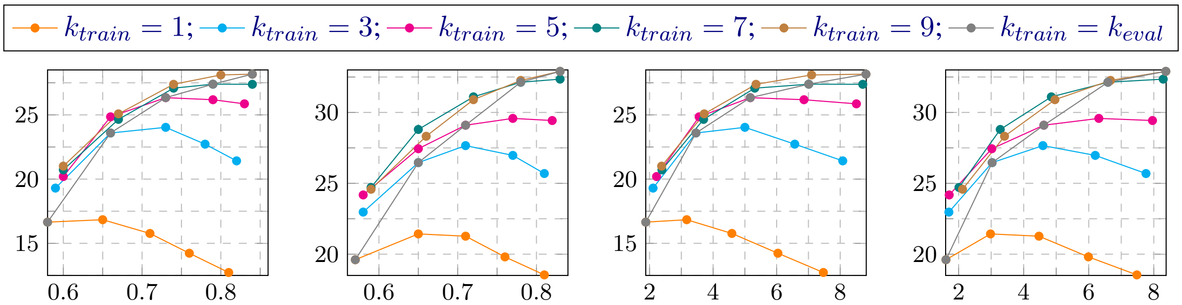
Models Both *forward* model and *backward* model are based on Transformer. For IWSLT14 En↔De and IWSLT15 En↔Vi, in the Transformer, we set the embedding dimension, feed-forward layer dimension, number of layers as 512, 1024, and 6, respectively.

Evaluation We evaluate the translation quality of all models by the tokenized word-level BLEU score (Papineni et al., 2002).

We also use Average Proportion (AP) (Ma et al., 2019) and Average Lagging (AL) (Cho and Esipova, 2016) to evaluate the translation delay. AP means the average proportion of source tokens required for translation, and AL means the average number of the delayed words.

3.2 Main Results and Analysis

We first evaluate models trained with different wait-k decoding paths on the IWSLT14 En↔De datasets. Figure 3 presents the performance of the models trained with $k_{train} \in \{1, 3, 5, 7, 9\}$ on these two datasets. Each curve with specified color represents each trained model, which is evaluated across different wait-k decoding paths with



(a) BLEU-AP, En→De (b) BLEU-AP, De→En (c) BLEU-AL, En→De (d) BLEU-AL, De→En

Figure 3: BLEU-AP and BLEU-AL curve on IWSLT14 De↔En.

	$k=3$			$k=5$			$k=7$		
	BLEU	AP	AL	BLEU	AP	AL	BLEU	AP	AL
Our model	23.59	0.66	3.46	26.34	0.73	5.17	27.39	0.79	7.01
- only with *forward* model	23.45	0.65	3.43	26.26	0.71	5.09	26.68	0.78	6.99
- only with *backward* model	8.54	0.66	2.91	7.11	0.73	4.42	8.57	0.78	6.02
- with two different *forward* model	22.68	0.65	3.40	25.78	0.71	5.13	27.3	0.79	6.96

Table 2: Ablation Study in IWSLT14 En→De

$k_{eval} \in \{1, 3, 5, 7, 9\}$. The results show that models trained on wait-7 (i.e. $k_{train} = 7$) generalize well on other evaluation paths. Like most simulNMT models using wait-k policy, the performance drops when far from the training path, for example, when $k_{train} = 1$ and $k_{eval} = 9$.

We also compare our method with the related work on IWSLT datasets when $k_{train} = k_{eval} = 7$. As shown in Table 1, our proposed method achieves the highest BLEU scores than the baselines and related works. Especially on IWSLT14 En→De, we perform better by a great margin. The evaluation results on multiple benchmarks show that our approach can obtain better scores than the baseline, and the latency is not affected much. This shows that the performance of SimulNMT models can be effectively improved through additional model design. We empirically verified that using our proposed bidirectional modeling is simple and effective.

4 Ablation Study

To investigate the importance of the *forward* model and *backward* model, we provide three groups of ablation study: (1) only with the *forward* model, (2) only with the *backward* model, and (3) the ensemble model with two *forward* models with a different seed. We work on the IWSLT14 En→De task and study the effect to wait-$\{3, 5, 7\}$. The results are shown in Table 2, and the latency metrics (AP and AL) are not significantly influenced. The BLEU score drops when removing any feature, which indicates that they all benefit the model. Specifically, the *forward* model plays the most critical role in our model. This reveals that due to the use of unidirectional attention in the *forward* modeling of SimulNMT, although real-time efficiency is satisfied, translation quality suffers from a negative impact. And with an additional feature from *backward* modeling added for enhancing the *forward* modeling, it indeed enhances the model's ability to adapt to the flexible order without affecting the latency.

5 Conclusion

In this work, we proposed a bidirectional modeling strategy for simultaneous neural machine translation. Motivated by the observation that the word order is free in practical use, while SimulNMT expects strict word order, we train a *backward* decoding model to let wait-k forecast the future information. Then we fuse the backward model into the *forward* model for ensemble decoding. Experiments on four translation tasks indicate the effectiveness of our model design. Experimental results demonstrate that this method is simple and effective. For future work, we will enhance this model performance by jointing more considerable auxiliary models besides the *backward* model.

References

Naveen Arivazhagan, Colin Cherry, Wolfgang Macherey, Chung-Cheng Chiu, Semih Yavuz, Ruoming Pang, Wei Li, and Colin Raffel. 2019. Monotonic infinite lookback attention for simultaneous machine translation. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 1313–1323, Florence, Italy. Association for Computational Linguistics.

Dzmitry Bahdanau, Kyunghyun Cho, and Yoshua Bengio. 2015. Neural machine translation by jointly learning to align and translate. In *Proceedings of the 3rd International Conference on Learning Representations*, pages 1–15, San Diego, USA.

Mauro Cettolo, Jan Niehues, Sebastian Stüker, Luisa Bentivogli, and Marcello Federico. Report on the 11th iwslt evaluation campaign, iwslt 2014.

Kyunghyun Cho and Masha Esipova. 2016. Can neural machine translation do simultaneous translation? *arXiv preprint arXiv:1606.02012*.

Kyunghyun Cho, Bart van Merriënboer, Caglar Gulcehre, Dzmitry Bahdanau, Fethi Bougares, Holger Schwenk, and Yoshua Bengio. 2014. Learning phrase representations using RNN encoder–decoder for statistical machine translation. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 1724–1734, Doha, Qatar. Association for Computational Linguistics.

Fahim Dalvi, Nadir Durrani, Hassan Sajjad, and Stephan Vogel. 2018. Incremental decoding and training methods for simultaneous translation in neural machine translation. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 2 (Short Papers)*, pages 493–499, New Orleans, Louisiana. Association for Computational Linguistics.

Sergey Edunov, Myle Ott, Michael Auli, David Grangier, and Marc'Aurelio Ranzato. 2018. Classical structured prediction losses for sequence to sequence learning. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)*, pages 355–364, New Orleans, Louisiana. Association for Computational Linguistics.

Maha Elbayad, Laurent Besacier, and Jakob Verbeek. 2020. Efficient wait-k models for simultaneous machine translation. *arXiv preprint arXiv:2005.08595*.

Alvin Grissom II, He He, Jordan Boyd-Graber, John Morgan, and Hal Daumé III. 2014. Don't until the final verb wait: Reinforcement learning for simultaneous machine translation. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 1342–1352, Doha, Qatar. Association for Computational Linguistics.

Jiatao Gu, Graham Neubig, Kyunghyun Cho, and Victor O.K. Li. 2017. Learning to translate in real-time with neural machine translation. In *Proceedings of the 15th Conference of the European Chapter of the Association for Computational Linguistics: Volume 1, Long Papers*, pages 1053–1062, Valencia, Spain. Association for Computational Linguistics.

Navdeep Jaitly, Quoc V. Le, Oriol Vinyals, Ilya Sutskever, David Sussillo, and Samy Bengio. 2016. An online sequence-to-sequence model using partial conditioning. In *NIPS*, pages 5067–5075.

Nal Kalchbrenner and Phil Blunsom. 2013. Recurrent continuous translation models. In *Proceedings of the 2013 Conference on Empirical Methods in Natural Language Processing*, pages 1700–1709, Seattle, Washington, USA. Association for Computational Linguistics.

Yuping Luo, Chung-Cheng Chiu, Navdeep Jaitly, and Ilya Sutskever. 2017. Learning online alignments with continuous rewards policy gradient. In *2017 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 2801–2805.

Minh-Thang Luong, Christopher D Manning, et al. 2015. Stanford neural machine translation systems for spoken language domains. In *Proceedings of the international workshop on spoken language translation*, pages 76–79.

Mingbo Ma, Liang Huang, Hao Xiong, Renjie Zheng, Kaibo Liu, Baigong Zheng, Chuanqiang Zhang, Zhongjun He, Hairong Liu, Xing Li, Hua Wu, and Haifeng Wang. 2019. STACL: Simultaneous translation with implicit anticipation and controllable latency using prefix-to-prefix framework. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 3025–3036, Florence, Italy. Association for Computational Linguistics.

Xutai Ma, Juan Pino, James Cross, Liezl Puzon, and Jiatao Gu. 2020. Monotonic multihead attention. *a8th International Conference on Learning Representations*.

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 of the Association for Computational Linguistics*, pages 311–318, Philadelphia, Pennsylvania, USA. Association for Computational Linguistics.

Ofir Press and Noah A Smith. 2018. You may not need attention. *arXiv preprint arXiv:1810.13409*.

Yi Ren, Jinglin Liu, Xu Tan, Chen Zhang, Tao Qin, Zhou Zhao, and Tie-Yan Liu. 2020. SimulSpeech: End-to-end simultaneous speech to text translation. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages

3787–3796, Online. 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, Berlin, Germany. 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.

Chen Zhang, Xu Tan, Jinglin Liu, Yi Ren, Tao Qin, and Tie-Yan Liu. 2019. Simuls2s: End-to-end simultaneous speech to speech translation. *Openreview*.

Baigong Zheng, Kaibo Liu, Renjie Zheng, Mingbo Ma, Hairong Liu, and Liang Huang. 2020. Simultaneous translation policies: From fixed to adaptive. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 2847–2853, Online. Association for Computational Linguistics.

Baigong Zheng, Renjie Zheng, Mingbo Ma, and Liang Huang. 2019a. Simpler and faster learning of adaptive policies for simultaneous 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 1349–1354, Hong Kong, China. Association for Computational Linguistics.

Baigong Zheng, Renjie Zheng, Mingbo Ma, and Liang Huang. 2019b. Simultaneous translation with flexible policy via restricted imitation learning. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 5816–5822, Florence, Italy. Association for Computational Linguistics.