Synchronous Bidirectional Neural Machine Translation

Long Zhou[1,2], **Jiajun Zhang**[1,2]*, **Chengqing Zong**[1,2,3]
[1]National Laboratory of Pattern Recognition, CASIA, Beijing, China
[2]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

Existing approaches to neural machine translation (NMT) generate the target language sequence token by token from left to right. However, this kind of unidirectional decoding framework cannot make full use of the target-side future contexts which can be produced in a right-to-left decoding direction, and thus suffers from the issue of unbalanced outputs. In this paper, we introduce a synchronous bidirectional neural machine translation (SB-NMT) that predicts its outputs using left-to-right and right-to-left decoding simultaneously and interactively, in order to leverage both of the history and future information at the same time. Specifically, we first propose a new algorithm that enables synchronous bidirectional decoding in a single model. Then, we present an interactive decoding model in which left-to-right (right-to-left) generation does not only depend on its previously generated outputs, but also relies on future contexts predicted by right-to-left (left-to-right) decoding. We extensively evaluate the proposed SB-NMT model on large-scale NIST Chinese-English, WMT14 English-German, and WMT18 Russian-English translation tasks. Experimental results demonstrate that our model achieves significant improvements over the strong Transformer model by 3.92, 1.49 and 1.04 BLEU points respectively, and obtains the state-of-the-art performance on Chinese-English and English-German translation tasks.[1]

1 Introduction

Neural machine translation has significantly improved the quality of machine translation in recent years (Sutskever et al., 2014; Bahdanau et al., 2015; Zhang and Zong, 2015; Wu et al., 2016; Gehring et al., 2017; Vaswani et al., 2017). Recent approaches to sequence to sequence learning typically leverage recurrence (Sutskever et al., 2014), convolution (Gehring et al., 2017), or attention (Vaswani et al., 2017) as basic building blocks.

Model	The first 4 tokens	The last 4 tokens
L2R	**40.21%**	35.10%
R2L	35.67%	**39.47%**

Table 1: Translation accuracy of the first 4 tokens and last 4 tokens in NIST Chinese-English translation tasks. L2R denotes left-to-right decoding and R2L means right-to-left decoding for conventional NMT.

Typically, NMT adopts the encoder-decoder architecture and generates the target translation from left to right. Despite their remarkable success, NMT models suffer from several weaknesses (Koehn and Knowles, 2017). One of the most prominent issues is the problem of unbalanced outputs in which the translation prefixes are better predicted than the suffixes (Liu et al., 2016). We analyze translation accuracy of the first and last 4 tokens for left-to-right (L2R) and right-to-left (R2L) directions respectively. As shown in Table 1, the statistical results show that L2R performs better in the first 4 tokens, whereas R2L translates better in term of the last 4 tokens. This problem is mainly caused by the left-to-right unidirectional decoding, which conditions each output word on previously generated outputs only, but leaving the future information from target-side contexts unexploited during translation. The future context is commonly used in reading and writing in human cognitive process (Xia et al., 2017), and it is crucial to avoid under-translation (Tu et al., 2016; Mi et al., 2016).

To alleviate the problems, existing studies usually used independent bidirectional decoders for

* Corresponding author.

[1]The source code is available at https://github.com/wszlong/sb-nmt.

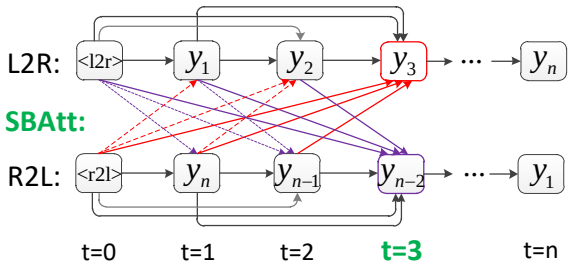

Figure 1: Illustration of the decoder in the synchronous bidirectional NMT model. L2R denotes left-to-right decoding guided by the start token $\langle l2r \rangle$ and R2L means right-to-left decoding indicated by the start token $\langle r2l \rangle$. SBAtt is our proposed synchronous bidirectional attention (see § 3.2). For instance, the generation of y_3 does not only rely on y_1 and y_2, but also depends on y_n and y_{n-1} of R2L.

NMT (Liu et al., 2016; Sennrich et al., 2016a). Most of them trained two NMT models with left-to-right and right-to-left directions respectively. Then, they translated and re-ranked candidate translations using two decoding scores together. More recently, Zhang et al. (2018) presented an asynchronous bidirectional decoding algorithm for NMT, which extended the conventional encoder-decoder framework by utilizing a backward decoder. However, these methods are more complicated than the conventional NMT framework beacuse they require two NMT models or decoders. Furthermore, the L2R and R2L decoders are independent from each other (Liu et al., 2016), or only the forward decoder can utilize information from the backward decoder (Zhang et al., 2018). It is therefore a promising direction to design a synchronous bidirectional decoding algorithm in which L2R and R2L generations can interact with each other.

Accordingly, we propose in this paper a novel framework (SB-NMT) that utilizes a single decoder to bidirectionally generate target sentences simultaneously and interactively. As shown in Figure 1, two special labels ($\langle l2r \rangle$ and $\langle r2l \rangle$) at the beginning of the target sentence guide translating from left to right or right to left, and the decoder in each direction can utilize the previously generated symbols of bidirectional decoding when generating the next token. Taking L2R decoding as an example, at each moment, the generation of the target word (e.g., y_3) does not only rely on previously generated outputs (y_1 and y_2) of L2R decoding, but also depends on previously predicted tokens (y_n and y_{n-1}) of R2L decoding. Compared to the previous related NMT models, our method has the following advantages: 1) We use a single model (one encoder and one decoder) to achieve the decoding with left-to-right and right-to-left generation, which can be processed in parallel. 2) Via the synchronous bidirectional attention model (SBAtt, §3.2), our proposed model is an end-to-end joint framework and can optimize bidirectional decoding simultaneously. 3) Compared to two-phase decoding scheme in previous work, our decoder is faster and more compact using one beam-search algorithm.

Specifically, we make the following contributions in this paper:

- We propose a synchronous bidirectional NMT model that adopts one decoder to generate outputs with left-to-right and right-to-left directions simultaneously and interactively. To the best of our knowledge, this is the first work to investigate the effectiveness of a single NMT model with synchronous bidirectional decoding.

- Extensive experiments on NIST Chinese-English, WMT14 English-German and WMT18 Russian-English translation tasks demonstrate that our SB-NMT model obtains significant improvements over the strong Transformer model by 3.92, 1.49 and 1.04 BLEU points respectively. In particular, our approach separately establishes the state-of-the-art BLEU score of 51.11 and 29.21 on Chinese-English and English-German translation tasks.

2 Background

In this paper, we build our model based on the powerful Transformer (Vaswani et al., 2017) with an encoder-decoder framework, where the encoder network first transforms an input sequence of symbols $x = (x_1, x_2, ..., x_n)$ to a sequence of continues representations $z = (z_1, z_2, ..., z_n)$, from which the decoder generates an output sequence $y = (y_1, y_2, ..., y_m)$ one element at a time. Particularly, relying entirely on the multi-head attention mechanism, the Transformer with beam search algorithm achieves the state-of-the-art results for machine translation.

Multi-Head Attention allows the model to jointly attend to information from different representation subspaces at different positions. It op-

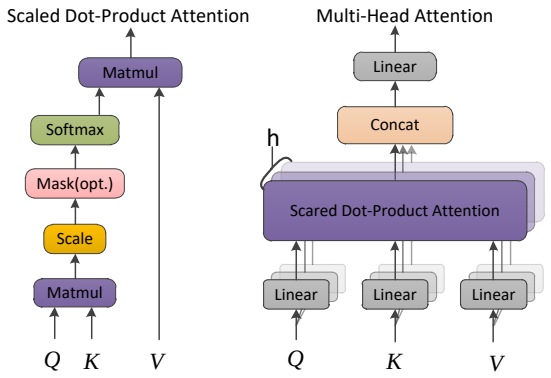

Figure 2: (left) Scaled Dot-Product Attention. (right) Multi-Head Attention.

erates on queries Q, keys K, and values V. For multi-head intra-attention of encoder or decoder, all of Q, K, V are the output hidden state matrices of the previous layer. For multi-head inter-attention of the decoder, Q are the hidden states of the previous decoder layer, and K-V pairs come from the output $(z_1, z_2, ..., z_n)$ of the encoder.

Formally, multi-head attention first obtains h different representations of (Q_i, K_i, V_i). Specifically, for each attention head i, we project the hidden state matrix into distinct query, key and value representations Q_i=QW_i^Q, K_i=KW_i^K, V_i=VW_i^V respectively. Then we perform **scaled dot-product attention** for each representation, concatenate the results, and project the concatenation with a feed-forward layer.

$$\begin{aligned} \text{MultiHead}(Q, K, V) &= \text{Concat}_i(head_i)W^O \\ head_i &= \text{Attention}(QW_i^Q, KW_i^K, VW_i^V) \end{aligned} \quad (1)$$

where W_i^Q, W_i^K, W_i^V and W^O are parameter projection matrices .

Scaled Dot-Product Attention can be described as mapping a query and a set of key-value pairs to an output. Specifically, we can then multiply query Q_i by key K_i to obtain an attention weight matrix, which is then multiplied by value V_i for each token to obtain the self-attention token representation. As shown in Figure 2, scaled dot-product attention operates on a query Q, a key K, and a value V as:

$$\text{Attention}(Q, K, V) = \text{Softmax}(\frac{QK^T}{\sqrt{d_k}})V \quad (2)$$

where d_k is the dimension of the key. For the sake of brevity, we refer the reader to Vaswani et al. (2017) for more details.

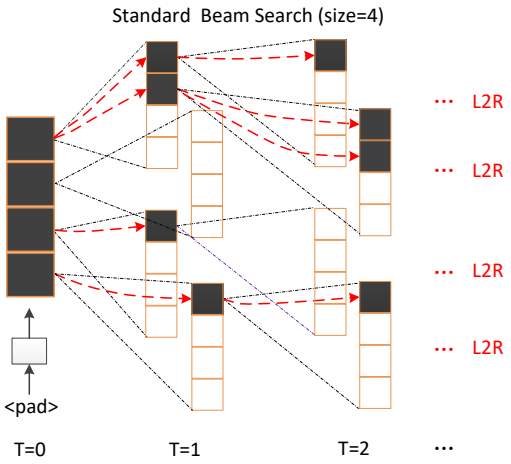

Figure 3: Illustration of the standard beam search algorithm with beam size 4. The black blocks denote the ongoing expansion of the hypotheses.

Standard Beam Search Given the trained model and input sentence x, we usually employ beam search or greedy search (beam size = 1) to find the best translation $\widehat{y} = \text{argmax}_y P(y|x)$. Beam size N is used to control the search space by extending only the top-N hypotheses in the current stack. As shown in Figure 3, the blocks represent the four best token expansions of the previous states, and these token expansions are sorted top-to-bottom from most-probable to least-probable. We define a complete hypothesis as a hypothesis which outputs EOS, where EOS is a special target token indicating the end of sentence. With the above settings, the translation y is generated token by token from left to right.

3 Our Approach

In this section, we will introduce the approach of synchronous bidirectional NMT. Our goal is to design a synchronous bidirectional beam search algorithm (§3.1) which generates tokens with both L2R and R2L decoding simultaneously and interactively using a single model. The central module is the synchronous bidirectional attention (SBAtt, see §3.2). By using SBAtt, the two decoding directions in one beam-search process can help and interact with each other, and can make full use of the target-side history and future information during translation. Then, we apply our proposed SBAtt to replace the multi-head intra-attention in the decoder part of Transformer model (§3.3), and the model is trained end-to-end by maximum likelihood using stochastic gradient descent (§3.4).

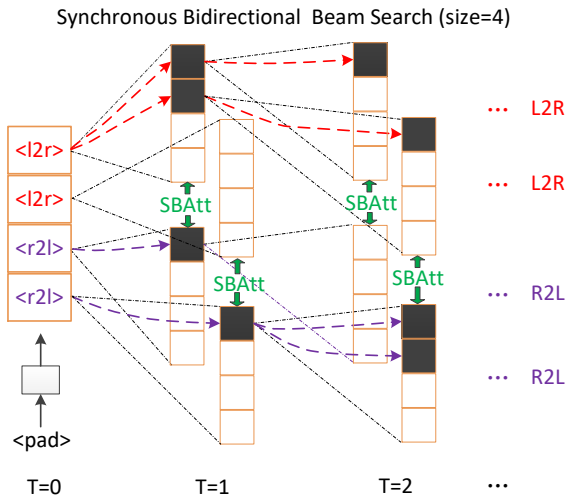

Figure 4: The synchronous bidirectional decoding of our model. ⟨l2r⟩ and ⟨r2l⟩ are two special labels, which indicate the target-side translation direction in L2R and R2L modes, respectively. Our model can decode with both L2R and R2L directions in one beam search by using SBAtt, simultaneously and interactively. SBAtt means the synchronous bidirectional attention (§3.2) performed between items of L2R and R2L decoding.

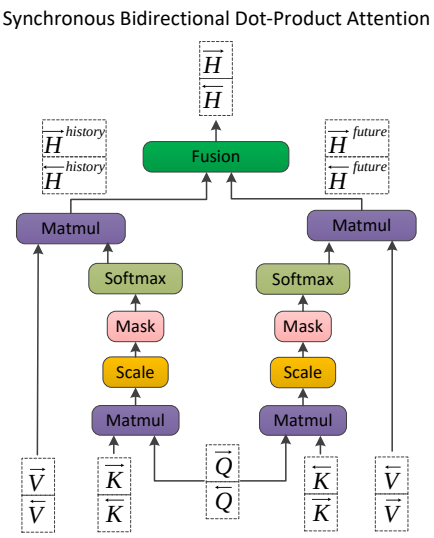

Figure 5: Synchronous bidirectional attention model based on scaled dot-product attention. It operates on forward (L2R) and backward (R2L) queries Q, keys K, values V.

3.1 Synchronous Bidirectional Beam Search

Figure 4 illustrates the synchronous bidirectional beam-search process with beam size 4. With two special start tokens which are optimized during the training process, we let half of the beam to keep decoding from left to right guided by the label ⟨l2r⟩, and allow the other half beam to decode from right to left indicated by the label ⟨r2l⟩. More importantly, via the proposed **SBAtt** (§3.2) model, L2R (R2L) generation does not only depend on its previously generated outputs, but also relies on future contexts predicted by R2L (L2R) decoding.

Note that (1) at each time step, we choose best items of the half beam from L2R decoding and best items of the half beam from R2L decoding to continue expanding simultaneously; (2) L2R and R2L beams should be thought of as parallel, with **SBAtt** computed between items of 1-best L2R and R2L, items of 2-best L2R and R2L, and so on[2]; (3) the black blocks denote the ongoing expansion of the hypotheses and decoding terminates when the end-of-sentence flag EOS is predicted; (4) in our decoding algorithm, the complete hypotheses will not participate in subsequent SBAtt, and the L2R hypothesis attended by R2L decoding may change at different time steps, while the ongoing partial hypotheses in both directions of SBAtt always share the same length; (5) finally, we output the translation result with highest probability from all complete hypotheses. Intuitively, our model is able to choose from L2R output or R2L output as final hypothesis according to their model probabilities, and if a R2L hypothesis wins, we reverse the tokens before presenting it.

[2]We also did experiments that all of L2R hypotheses attend to the 1-best R2L hypothesis, and all the R2L hypotheses attend to the 1-bset L2R hypothesis. The results of the two schemes are similar. For the sake of simplicity, we employed the previous scheme.

3.2 Synchronous Bidirectional Attention

Instead of multi-head intra-attention which prevents future information flow in the decoder to preserve the auto-regressive property, we propose a synchronous bidirectional attention (SBAtt) mechanism. With the two key modules of synchronous bidirectional dot-product attention (§3.2.1) and synchronous bidirectional multi-head attention (§3.2.2), SBAtt is capable of capturing and combining the information generated by L2R and R2L decoding.

3.2.1 Synchronous Bidirectional Dot-Product Attention

Figure 5 shows our particular attention "Synchronous Bidirectional Dot-Product Attention (SBDPA)". The input consists of queries ($[\overrightarrow{Q};\overleftarrow{Q}]$), keys ($[\overrightarrow{K};\overleftarrow{K}]$) and values ($[\overrightarrow{V};\overleftarrow{V}]$) which are all concatenated by forward (L2R) states and back-

ward (R2L) states. The new forward state $\overrightarrow{H}$ and backward state $\overleftarrow{H}$ can be obtained by synchronous bidirectional dot-product attention. For the new forward state $\overrightarrow{H}$, it can be calculated as:

$$\begin{aligned}\overrightarrow{H}^{history} &= \text{Attention}(\overrightarrow{Q}, \overrightarrow{K}, \overrightarrow{V}) \\ \overrightarrow{H}^{future} &= \text{Attention}(\overrightarrow{Q}, \overleftarrow{K}, \overleftarrow{V}) \\ \overrightarrow{H} &= \text{Fusion}(\overrightarrow{H}^{history}, \overrightarrow{H}^{future})\end{aligned} \tag{3}$$

where $\overrightarrow{H}^{history}$ is obtained by using conventional scaled dot-product attention as introduced in Equation 2, and its purpose is to take advantage of previously generated tokens, namely **history information**. We calculate $\overrightarrow{H}^{future}$ using forward query ($\overrightarrow{Q}$) and backward key-value pairs ($\overleftarrow{K}$, $\overleftarrow{V}$), which attempts at making use of **future information** from R2L decoding as effectively as possible in order to help predict the current token in L2R decoding. The role of Fusion$(\cdot)$ (green block in Figure 5) is to combine $\overrightarrow{H}^{history}$ and $\overrightarrow{H}^{future}$ by using linear interpolation, nonlinear interpolation or gate mechanism.

Linear Interpolation $\overrightarrow{H}^{history}$ and $\overrightarrow{H}^{future}$ have different importance to prediction of current word. Linear interpolation of $\overrightarrow{H}^{history}$ and $\overrightarrow{H}^{future}$ produces an overall hidden state:

$$\overrightarrow{H} = \overrightarrow{H}^{history} + \lambda * \overrightarrow{H}^{future} \tag{4}$$

where λ is a hyper-parameter decided by the performance on development set.[3]

Nonlinear Interpolation $\overrightarrow{H}$ is equal to $\overrightarrow{H}^{history}$ in the conventional attention mechanism, and $\overrightarrow{H}^{future}$ means the attention information between current hidden state and generated hidden states of the other decoding. In order to distinguish two different information sources, we present a nonlinear interpolation by adding an activation function to the backward hidden states:

$$\overrightarrow{H} = \overrightarrow{H}^{history} + \lambda * AF(\overrightarrow{H}^{future}) \tag{5}$$

where AF denotes activation function, such as $tanh$ or $relu$.

Gate Mechanism We also propose a gate mechanism to dynamically control the amount of information flow from the forward and backward contexts. Specially, we apply a feed-forward gating layer upon $\overrightarrow{H}^{history}$ as well as $\overrightarrow{H}^{future}$ to enrich the non-linear expressiveness of our model:

$$\begin{aligned}r_t, z_t &= \sigma(W^g[\overrightarrow{H}^{history}; \overrightarrow{H}^{future}]) \\ \overrightarrow{H} &= r_t \odot \overrightarrow{H}^{history} + z_t \odot \overrightarrow{H}^{future}\end{aligned} \tag{6}$$

where $\odot$ denotes element-wise multiplication. Via this gating layer, it is able to control how much past information can be preserved from previous context and how much reversed information can be captured from backward hidden states.

Similar to the calculation of forward hidden states $\overrightarrow{H}_i$, the backward hidden states $\overleftarrow{H}_i$ can be computed as follows.

$$\begin{aligned}\overleftarrow{H}^{history} &= \text{Attention}(\overleftarrow{Q}, \overleftarrow{K}, \overleftarrow{V}) \\ \overleftarrow{H}^{future} &= \text{Attention}(\overleftarrow{Q}, \overrightarrow{K}, \overrightarrow{V}) \\ \overleftarrow{H} &= \text{Fusion}(\overleftarrow{H}^{history}, \overleftarrow{H}^{future})\end{aligned} \tag{7}$$

where Fusion$(\cdot)$ is the same as introduced in Equation 4-6. Note that $\overrightarrow{H}$ and $\overleftarrow{H}$ can be calculated in parallel. We refer to the whole procedure formulated in Equation 3 and Equation 7 as SBDPA$(\cdot)$.

$$[\overrightarrow{H}; \overleftarrow{H}] = \text{SBDPA}([\overleftarrow{Q}; \overrightarrow{Q}], [\overleftarrow{K}; \overrightarrow{K}], [\overleftarrow{V}; \overrightarrow{V}]) \tag{8}$$

3.2.2 Synchronous Bidirectional Multi-Head Attention

Multi-head attention consists of h attention heads, each of which learns a distinct attention function to attend to all of the tokens in the sequence, where mask is used for preventing leftward information flow in decoder. Compared to the multi-head attention, our inputs are the concatenation of forward and backward hidden states. We extend standard multi-headed attention by letting each head attend to both forward and backward hidden states, combined via SBDPA$(\cdot)$.

$$\begin{aligned}&\text{MultiHead}([\overleftarrow{Q}; \overrightarrow{Q}], [\overleftarrow{K}; \overrightarrow{K}], [\overleftarrow{V}; \overrightarrow{V}]) \\ &= \text{Concat}([\overrightarrow{H}_1; \overleftarrow{H}_1], ..., [\overrightarrow{H}_h; \overleftarrow{H}_h])W^O\end{aligned} \tag{9}$$

and $[\overrightarrow{H}_i; \overleftarrow{H}_i]$ can be computed as follows, which is the biggest difference from conventional multi-head attention.

$$\begin{aligned}[\overrightarrow{H}_i; \overleftarrow{H}_i] = \text{SBDPA}([\overleftarrow{Q}; \overrightarrow{Q}]W_i^Q, \\ [\overleftarrow{K}; \overrightarrow{K}]W_i^K, [\overleftarrow{V}; \overrightarrow{V}]W_i^V)\end{aligned} \tag{10}$$

[3] Note that we can also set λ to be a vector and learn λ during training with standard back-propagation, and we remain it as future exploration.

encoder是普通的self-attention, q,k,v, source的token排列順序只有單向的
decoder分成training與inference兩種attention方式，decoder輸入如Fig.1，是兩個順序相反的token ids進行concate，shape=[2, batch, hidden_size]
1. training時的第一個self-attention會分成l2r和r2l各別產生H，接著用nonlinear interpolation(tanh)進行concate
l2r把concate的輸入直接當成qkv進行普通的self-attention，而r2l是將k和v依照第一個維度進行reverse，然後用q與k進行反向的attention，然後再與v做線性組合得到H

疑問: 為何前面要concate? 維持原來的維度[batch, hidden_size]也可以reverse在組合，應該是為了確保inference時的beam size永遠可以有2可以除?

2. inference時候因為有beam search，l2r一樣是原始的self-attention，H為[batch*beam, num_heads, length_tmp, hidden_size/num_heads]
r2l會依照beam維度切兩份, shape = [batch, 2, beam/2, num_heads, length_tmp, hidden_size/num_heads]
然後依照第二個維度進行reverse形成新的k, v，接著與encoder的q進行self-attention得到r2l的H, [batch*beam, num_heads, length_tmp, hidden_size/num_heads]

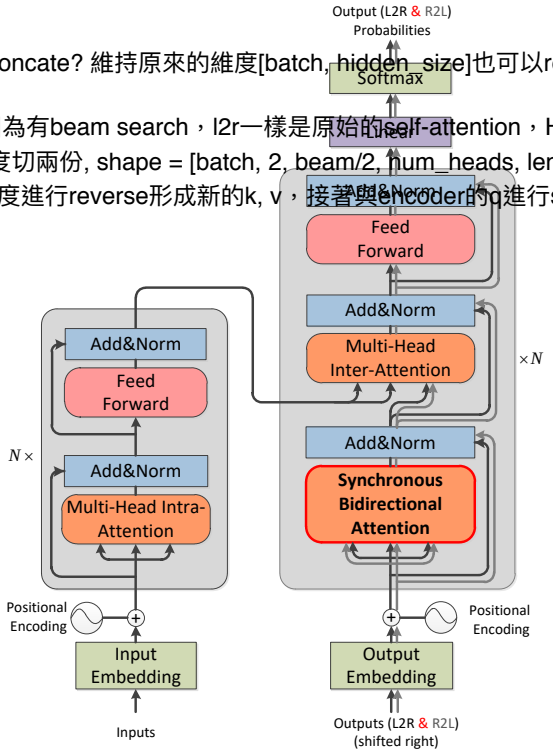

Figure 6: The new Transformer architecture with the proposed synchronous bidirectional multi-head attention network, namely SBAtt. The input of decoder is concatenation of forward (L2R) sequence and backward (R2L) sequence. Note that all bidirectional information flow in decoder runs in parallel and only interacts in synchronous bidirectional attention layer.

where W_i^Q, W_i^K, W_i^V and W^O are parameter projection matrices, which are the same as standard multi-head attention introduced in Equation 1.

3.3 Integrating Synchronous Bidirectional Attention into NMT

We apply our synchronous bidirectional attention to replace the multi-head intra-attention in the decoder, as illustrated in Figure 6. The neural encoder of our model is identical to that of the standard Transformer model. From the source tokens, learned embeddings are generated which are then modified by an additive positional encoding. The encoded word embeddings are then used as input to the encoder which consists of N blocks each containing two layers: (1) a multi-head attention layer (MHAtt), and (2) a position-wise feed-forward layer (FFN).

The bidirectional decoder of our model is extended from the standard Transformer decoder. For each layer in the bidirectional decoder, the lowest sub-layer is our proposed synchronous bidirectional attention network, and it also uses residual connections around each of the sublayers, followed by layer normalization.

$$s_d^l = \text{LayerNorm}(s_d^{l-1} + \text{SBAtt}(s^{l-1}, s^{l-1}, s^{l-1})) \quad (11)$$

where l denotes layer depth, subscript d means the decoder-informed intra-attention representation. SBAtt is our proposed synchronous bidirectional attention, and s^{l-1} is equal to $[\overrightarrow{s}^{l-1}; \overleftarrow{s}^{l-1}]$ containing forward and backward hidden states. In addition, the decoder stacks another two sublayers to seek translation-relevant source semantics to bridge the gap between the source and target language:

$$\begin{aligned} s_e^l &= \text{LayerNorm}(s_d^l + \text{MHAtt}(s_d^l, h^N, h^N)) \\ s^l &= \text{LayerNorm}(s_e^l + \text{FFN}(s_e^l)) \end{aligned} \quad (12)$$

where MHAtt denotes the multi-head attention introduced in Equation 1, and we use e to denote the encoder-informed inter-attention representation. h^N is the source top layer hidden state, and FFN means feed-forward networks.

Finally, we use a linear transformation and softmax activation to compute the probability of the next tokens based on $s^N = [\overrightarrow{s}^N; \overleftarrow{s}^N]$, namely the final hidden states of forward and backward decoding.

$$\begin{aligned} p(\overrightarrow{y}_j | \overrightarrow{y}_{<j}, \overleftarrow{y}_{<j}, x, \theta) &= \text{Softmax}(\overrightarrow{s}^N W) \\ p(\overleftarrow{y}_j | \overleftarrow{y}_{<j}, \overrightarrow{y}_{<j}, x, \theta) &= \text{Softmax}(\overleftarrow{s}^N W) \end{aligned} \quad (13)$$

where θ is shared weight for L2R and R2L decoding and W is the weight matrix.

3.4 Training

We design a simple yet effective strategy to enable synchronous bidirectional translation within a decoder. We separately add the special labels ($\langle l2r \rangle$ and $\langle r2l \rangle$) at the beginning of target sentence ($\overrightarrow{y}$ and $\overleftarrow{y}$) to guide translating from left to right or right to left. 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 data:

$$\begin{aligned} J(\theta) = \frac{1}{Z} \sum_{z=1}^{Z} \sum_{j=1}^{M} \{ & log\ p(\overrightarrow{y}_j^{(z)} | \overrightarrow{y}_{<j}^{(z)}, \overleftarrow{y}_{<j}^{(z)}, x^{(z)}, \\ & \theta) + log\ p(\overleftarrow{y}_j^{(z)} | \overleftarrow{y}_{<j}^{(z)}, \overrightarrow{y}_{<j}^{(z)}, x^{(z)}, \theta) \} \end{aligned} \quad (14)$$

Similar to asynchronous bidirectional decoding (Zhang et al., 2018) and bidirectional language models in BERT (Devlin et al., 2018), the proposed SB-NMT model also faces the same training problem that the bidirectional decoding would allow the words (the second half of the decoding sequence) to indirectly "see themselves" from the other decoding direction. To ensure consistency between model training and testing, we construct pseudo references $\overleftarrow{y}_p$ ($\overrightarrow{y}_p$) for gold $\overrightarrow{y}_g$ ($\overleftarrow{y}_g$). More specifically, we first train a L2R model using (x, $\overrightarrow{y}_g$) and a R2L model using (x, $\overleftarrow{y}_g$). Then we use the two models to translate source sentences x into pseudo target sentences $\overrightarrow{y}_p$ and $\overleftarrow{y}_p$ respectively. Finally, we get two triples (x, $\overrightarrow{y}_p$, $\overleftarrow{y}_g$) and (x, $\overrightarrow{y}_g$, $\overleftarrow{y}_p$) as our training data.

Once the proposed model is trained, we employ the bidirectional beam search algorithm to predict the target sequence, as illustrated in Figure 4. Compared to previous work that usually adopt a two-phase scheme to translate input sentences (Liu et al., 2016; Sennrich et al., 2017; Zhang et al., 2018), our decoding approach is more compact and effective.

4 Experiments

We evaluate the proposed model on three translation datasets with different size, including NIST Chinese-English, WMT14 English-German and WMT18 Russian-English translations.

4.1 Dataset

For Chinese-English, our training data includes about 2.0 million sentence pairs extracted from the LDC corpus.[4] We use NIST 2002 (MT02) Chinese-English dataset as the validation set, NIST 2003-2006 (MT03-06) as our test sets. We use BPE (Sennrich et al., 2016b) to encode Chinese and English respectively. We learn 30K merge operations and limit the source and target vocabularies to the most frequent 30K tokens.

For English-German translation, the training set consists of about 4.5 million bilingual sentence pairs from WMT 2014.[5] We use newstest2013 as the validation set and newstest2014 as the test set. Sentences are encoded using BPE, which has a shared vocabulary of about 37000 tokens. To evaluate the models, we compute the BLEU metric (Papineni et al., 2002) on tokenized, true-case output.[6]

Fusion		λ =0.1	λ =0.5	λ =1.0
Linear		51.05	50.71	46.98
Nonlinear	*tanh*	50.99	50.72	50.96
	relu	50.79	50.57	50.71
Gate		50.51		

Table 2: Experiment results on the development set using different fusion mechanism with different λs.

For Russian-English translation, we use the following resources from the WMT parallel data[7]: ParaCrawl corpus, Common Crawl corpus, News Commentary v13 and Yandex Corpus. We do not use Wiki Headlines and UN Parallel Corpus V1.0. The training corpus consists of 14M sentence pairs. We emply the Moses Tokenizer[8] for precocessing. For subword segmentation, we use 50000 joint BPE operations and choose the most frequent 52000 tokens as vocabularies. We use newstest2017 as the development set and the newtest2018 as the test set.

4.2 Setting

We build the described models by modifying the tensor2tensor[9] toolkit for training and evaluating. For our bidirectional Transformer model, we employ the Adam optimizer with β_1=0.9, β_2=0.998, and ϵ=10^{-9}. We use the same warmup and decay strategy for learning rate as Vaswani et al. (2017), with 16,000 warmup steps. During training, we employ label smoothing of value ϵ_{ls}=0.1. For evaluation, we use beam search with a beam size of k=4 (For SB-NMT, we use two L2R and R2L hypotheses respectively.) and length penalty α=0.6. Additionally, we use 6 encoder and decoder layers, hidden size d_{model}=1024, 16 attention-heads, 4096 feed forward inner-layer dimensions, and $P_{dropout}$=0.1. Our settings are close to *trans-*

[4]The corpora includes LDC2000T50, LDC2002T01, LDC2002E18, LDC2003E07, LDC2003E14, LDC2003T17 and LDC2004T07. Following previous work, we also using case-insensitive tokenized BLEU to evaluate Chinese-English which have been segmented by Stanford word segmentation and Moses Tokenizer respectively.

[5]http://www.statmt.org/wmt14/translation-task.html. All preprocessed dataset and vocab can be directly download in tensor2tensor website https://drive.google.com/open?id=0B_bZck-ksdkpM25jRUN2X2UxMm8.

[6]This procedure is used in the literature to which we compare (Wu et al., 2016; Gehring et al., 2017; Vaswani et al., 2017).

[7]http://www.statmt.org/wmt18/translation-task.html.

[8]https://github.com/moses-smt/mosesdecoder/blob/master/scripts/tokenizer/tokenizer.perl.

[9]https://github.com/tensorflow/tensor2tensor.

Model	DEV	MT03	MT04	M05	MT06	AVE	Δ
Moses	37.85	37.47	41.20	36.41	36.03	37.78	-9.41
RNMT	42.43	42.43	44.56	41.94	40.95	42.47	-4.72
Transformer	48.12	47.63	48.32	47.51	45.31	47.19	-
Transformer (R2L)	47.81	46.79	47.01	46.50	44.13	46.11	-1.08
Rerank-NMT	49.18	48.23	48.91	48.73	46.51	48.10	+0.91
ABD-NMT	48.28	49.47	48.01	48.19	47.09	48.19	+1.00
Our Model	**50.99**	**51.87**	**51.50**	**51.23**	**49.83**	**51.11**	**+3.92**

Table 3: Evaluation of translation quality for Chinese-English translation tasks using case-insensitive BLEU scores. All results of our model are significantly better than Transformer and Transformer (R2L) ($p < 0.01$).

former_big setting as defined in Vaswani et al. (2017). We employ three Titan Xp GPUs to train English-German and Russian-English translation, and one GPU for Chinese-English translation pairs. In addition, we use a single model obtained by averaging the last 20 checkpoints for English-German and Russian-English and do not perform checkpoint averaging for Chinese-English.

4.3 Baselines

We compare the proposed model against the following state-of-the-art SMT and NMT systems[10]:

- **Moses**: an open source phrase-based SMT system with default configuration and a 4-gram language model trained on the target portion of training data.
- **RNMT** (Luong et al., 2015): it is a state-of-the-art RNN-based NMT system with default setting.
- **Transformer**: it has obtained the state-of-the-art performance on machine translation, which predicts target sentence from left to right relying on self-attention (Vaswani et al., 2017).
- **Transformer (R2L)**: it is a variant of Transformer that generates translation in a right-to-left direction.
- **Rerank-NMT**: Via exploring the agreement on left-to-right and right-to-left NMT models, (Liu et al., 2016; Sennrich et al., 2016a) first run beam search for forward and reverse models independently to obtain two k-best lists, and then re-score the union of two k-best lists (k=10 in our experiments) using the joint model (adding logprobs) to find the best candidate.
- **ABD-NMT**: it is an asynchronous bidirectional decoding for NMT, which equipped the conventional attentional encoder-decoder NMT model with a backward decoder (Zhang et al., 2018). ABD-NMT adopts a two-phrase decoding scheme: (1) use backward decoder to generate reverse sequence states; (2) perform beam search on the forward decoder to find the best translation based on encoder hidden states and backward sequence states.

[10]For fair comparison, Rerank-NMT and ABD-NMT are based on strong Transformer models.

4.4 Results on Chinese-English Translation

Effect of Fusion Mechanism We first investigate the impact of different fusion mechanisms with different λs on the development set. As shown in Table 2, we find that linear interpolation is sensitive to parameters λ. Nonlinear interpolation, which is more robust than linear interpolation, achieves the best performance when we use $tanh$ with λ=0.1. Compared to gate mechanism, nonlinear interpolation is much simpler and needs less parameters. Therefore, we will use nonlinear interpolation with $tanh$ and λ=0.1 for all experiments thereafter.

Translation Quality Table 3 shows translation performance for Chinese-English. Specifically, the proposed model significantly outperforms Moses, RNMT, Transformer, Transformer (R2L), Rerank-NMT and ABD-NMT by 13.23, 8.54, 3.92, 4.90, 2.91, 2.82 BLEU points, respectively. Compared to Transformer and Transformer (R2L), our model exhibits much better performance. These results confirm our hypothesis that the two directions are mutually beneficial in bidirectional decoding. Furthermore, compared

Model	TEST
GNMT‡ (Wu et al., 2016)	24.61
Conv‡ (Gehring et al., 2017)	25.16
AttIsAll‡ (Vaswani et al., 2017)	28.40
Transformer[11]	27.72
Transformer (R2L)	27.13
Rerank-NMT	27.81
ABD-NMT	28.22
Our Model	**29.21**

Table 4: Results of WMT14 English-German translation using case-sensitive BLEU. Results with ‡ mark are taken from the corresponding papers.

Model	DEV	TEST
Transformer	35.28	31.02
Transformer (R2L)	35.22	30.57
Our Model	**36.38**	**32.06**

Table 5: Results of WMT18 Russian-English translation using case-insensitive tokenized BLEU.

to Rerank-NMT in which two decoders are relatively independent and ABD-NMT where only the forward decoder can rely on a backward decoder, our proposed model achieves substantial improvements over them on all test sets, which indicates that joint modeling and optimizing with left-to-right and right-to-left decoding behaves better in leveraging bidirectional decoding.

4.5 Results on English-German Translation

We further demonstrate the effectiveness of our model in WMT14 English-German translation tasks, and we also display the performances of some competitive models including GNMT (Wu et al., 2016), Conv (Gehring et al., 2017), and AttIsAll (Vaswani et al., 2017). As shown in Table 4, our model also significantly outperforms others and gets an improvement of 1.49 BLEU points than a strong Transformer model. Moreover, our SB-NMT model establishes a state-of-the-art BLEU score of 29.21 on the WMT14 English-German translation task.

[11]The BLEU scores for Transformer model are our reproduced results. Similar to footnote 7 in (Chen et al., 2018), our performance is slightly lower than those reported in (Vaswani et al., 2017). Additionally, we only use 3 GPUs for English-German, whereas most papers employ 8 GPUs for model training.

Model	Param	Speed	
		Train	*Test*
Transformer	207.8M	2.07	19.97
Transformer (R2L)	207.8M	2.07	19.81
Rerank-NMT	415.6M	1.03	6.51
ABD-NMT	333.8M	1.18	7.20
Our Model	207.8M	1.26	17.87

Table 6: Statistics of parameters, training and testing speeds. *Train* denotes the number of global training steps processed per second at the same batch-size sentences; *Test* indicates the amount of translated sentences in one second.

4.6 Results on Russian-English Translation

Table 5 shows the results of large-scale WMT18 Russian-English translation, and our approach still significantly outperforms the state-of-the-art Transformer model in development and test sets by 1.10 and 1.04 BLEU points respectively. Note that the BLEU score gains of English-German and Russian-English are not as significant as that on Chinese-English. The underlying reasons, which have also been mentioned in Shen et al. (2016) and Zhang et al. (2018), are that (1) the Chinese-English datasets contain four reference translations for each source sentence while the English-German and Russian-English datasets only have single reference; (2) English is more distantly related to Chinese than German and Russian, leading to the predominant improvements for Chinese-English translation when leveraging bidirectional decoding.

4.7 Analysis

We conduct analyses on Chinese-English translation, to better understand our model from different perspectives.

Parameters and Speeds In contrast to the standard Transformer, our model does not increase any parameters except for a hyper-parameter λ, as shown in Table 6. Rerank-NMT needs to train two sets of NMT models, so its parameters are doubled. The parameters of ABD-NMT are 333.8M since it has two decoders containing a backward decoder and a forward decoder. Hence, our model is more compact because it only has a single encoder-decoder NMT model.

We also show the training and testing speed of our model and baselines in Table 6. During training, our model performs approximately 1.26 train-

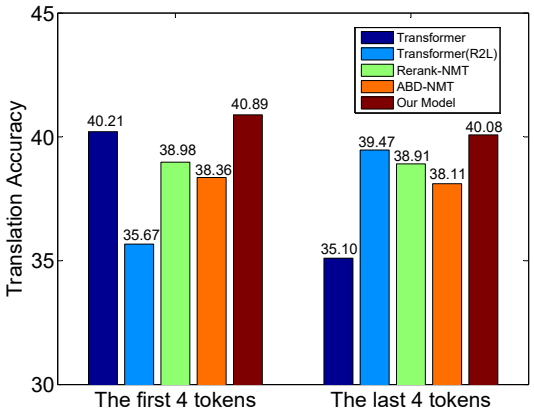

Figure 7: Translation accuracy of the first and last 4 tokens for Transformer, Transformer (R2L), Rerank-NMT, ABD-NMT and our proposed model.

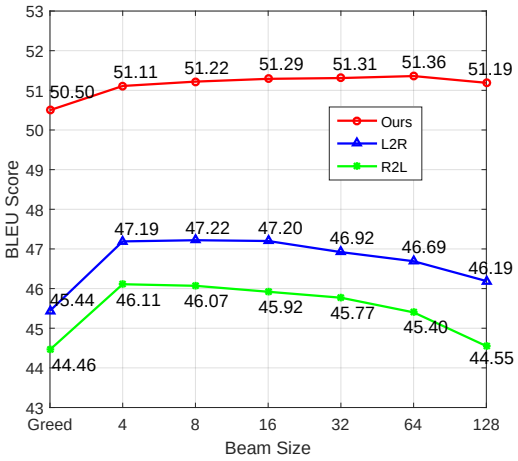

Figure 8: Translation qualities (BLEU score) of our L2R, R2L and our SB-NMT model as beam size becomes larger[12].

ing steps per second, which is faster than Rerank-NMT and ABD-NMT. When it comes to decoding procedure, the decoding speed of our model is 17.87 sentences per second with batch size 50, which is two or three times faster than Rerank-NMT and ABD-NMT.

Effect of Unbalanced Outputs According to Table 1, L2R usually does well on predicting the left-side tokens of target sequences, while R2L usually performs well on the right-side tokens. Our central idea is combine the advantage of left-to-right and right-to-left modes. To test our hypothesis, we further analyze the translation accuracy of Rerank-NMT, ABD-NMT, and our model, as shown in Figure 7. Rerank-NMT and ABD-NMT can alleviate the unbalanced output problem, but fail to improve prefix and suffix accuracies at the same time. The experimental results demonstrate that our model can balance the outputs, and gets the best translation accuracy for both the first 4 words and the last 4 words. Note that our model chooses from L2R output or R2L output as final results according to their model probabilities, and the left-to-right decoding contributes 58.6% on test set.

Effect of Varying Beam Size We observe that beam search decoding only improves translation quality for narrow beams and degrades translation quality when exposed to a larger search space for L2R and R2L decoding as illustrated in Figure 8. Additionally, the gap between greedy search and beam search is significant and can be up to about 1-2 BLEU points. Koehn and Knowles (2017) also demonstrate these phenomena in eight translation directions.

As for our SB-NMT model, we investigate the effect of different beam sizes k, as shown by the red line of Figure 8. Compared to conventional beam search, where worse translations are found beyond an optimal beam size setting (e.g., in the range of 4-32), the translation quality of our proposed model remains stable as beam size becomes larger. We attribute this to the ability of the combined objective to model both history and future translation information.

Effect of Long Sentences A well-known flaw of NMT models is the inability to properly translate long sentences. We follow Bahdanau et al. (2015) to group sentences of similar lengths together and compute a BLEU score per group (left picture). Figure 9 shows the BLEU score and the averaged length of translations for each group (right picture). Transformer and Transformer (R2L) perform very well on short source sentences, but degrade on long source sentences. Our model can alleviate this problem by taking advantage of both history and future information. In fact, incorporating synchronous bidirectional attention boosts translation performance on all source sentence groups.

Comparison to Data-Enhanced NMT In the training setup, we have obtained pseudo L2R and R2L references ($\overrightarrow{y}_p$ and $\overleftarrow{y}_p$) by using L2R and R2L models respectively. Here, we first compare our proposed model with NMT enhanced by pseudo data, and further explore the data utilization of SB-NMT by using combined data strategy

[12]For greedy search in SB-NMT, it has one item L2R decoding and one item R2L decoding. In other words, its beam size is equal to 2 compared to conventional beam search decoding.

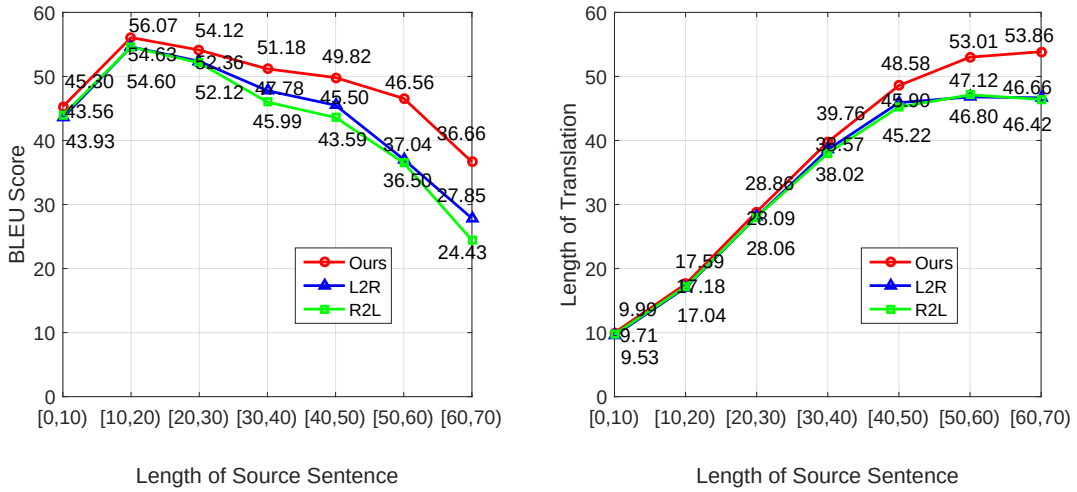

Figure 9: Performance of translations on the test set with respect to the lengths of the source sentences.

Model	TEST
Transformer (standard $\overrightarrow{y}_g$)	47.17
SB-NMT (two triples data)	51.11
Transformer ($\overrightarrow{y}_g$ + $\overrightarrow{y}_p$)	49.48
Transformer ($\overrightarrow{y}_g$ + $\overrightarrow{y}_p$ + $\overleftarrow{y}_p$)	49.99
SB-NMT (six triples data)	**52.14**

Table 7: Chinese-English BLEU scores of standard Transformer enhanced with pseudo data, and our SB-NMT model with combined data strategy.

Model	Over-Trans		Under-Trans	
	Ratio	Δ	Ratio	Δ
L2R	0.07%	-	7.85%	-
R2L	0.14%	-	7.81%	-
Ours	0.07%	-0.00%	5.42%	-30.6%

Table 8: Subjective evaluation on over-translation and under-translation for Chinese-English. Ratio denotes the percentage of source words which are over- or under-translated, Δ indicates relative improvement.

(six triples data, that is, ($\overrightarrow{y}_g$, $\overleftarrow{y}_p$), (reversed $\overleftarrow{y}_p$, $\overleftarrow{y}_g$), ($\overrightarrow{y}_p$, $\overleftarrow{y}_g$), ($\overrightarrow{y}_g$, reversed $\overrightarrow{y}_p$), ($\overrightarrow{y}_p$, $\overleftarrow{y}_p$), and (reversed $\overleftarrow{y}_p$, reversed $\overrightarrow{y}_p$)). As shown in Table 7, we find that data-enhanced Transformer outperforms the original Transformer, but still behaves worse than our proposed model. Furthermore, by making full use of training data, our model (six triple data) significantly improves the translation quality by 1.03 BLEU points than the original set (two triples data).

Subjective Evaluation We follow Tu et al. (2016) to conduct a subjective evaluation to validate the benefit of the synchronous bidirectional decoder, as shown in Table 8. Four human evaluators are asked to evaluate the translations of 100 source sentences, which are randomly sampled from the test sets without knowing which system the translation is selected from. These 100 source sentences have 2712 words. We evaluate over- or under-translation based on the number of source words which are dropped or repeated in translation[13], though we use subword (Sennrich et al., 2016b) in training and inference. Transformer and Transformer (R2L) suffer from serious under-translation problems with 7.85% and 7.81% errors. Our proposed model alleviates the under-translation problems by exploiting the combination of left-to-right and right-to-left decoding directions, reducing 30.6% of under-translation errors. It should be emphasized that the proposed model is especially effective for alleviating under-translation problem, which is a more serious translation problem for Transformer systems as seen in Table 8.

Case Study Table 9 gives three examples to show the translations of different models, in order to better understand how our model outperforms others. We find that Transformer produces translations with good prefixes (red line or dotted line), while Transformer (R2L) generates translations with better suffixes (blue line or wave line). Therefore, they are often unable to translate the whole sentence precisely. In contrast, the proposed approach can make full use of bidirectional decoding and remedy the errors in these cases.

[13]For our SB-NMT model, 2 source words are over-translated and 147 source words are under-translated. Additionally, it is interesting to combine with better scoring methods and stopping criteria (Yang et al., 2018) to strengthen the baseline and our model in the future.

Source	捷克 总统 哈维 卸任 新 总统 仍 未 确定
Reference	czech president havel steps down while new president still not chosen
L2R	czech president leaves office
R2L	the outgoing president of the czech republic is still uncertain
Ours	czech president havel leaves office , new president yet to be determined
Source	他们 正在 研制 一 种 超大型 的 叫做 炸弹 之 母 。
Reference	they are developing a kind of superhuge bomb called the mother of bombs .
L2R	they are developing a super , big , mother , called the bomb .
R2L	they are working on a much larger mother called the mother of a bomb .
Ours	they are developing a super-large scale , called the mother of the bomb .

Table 9: Chinese-English translation examples of Transformer decoding in left-to-right and right-to-left way, and our proposed models. L2R performs well in the first half sentence, whereas R2L translates well in the second half sentence.

5 Related Work

Our research is built upon a sequence-to-sequence model (Vaswani et al., 2017), but it is also related to future modeling and bidirectional decoding. We discuss these topics in the following.

Future Modeling Standard neural sequence decoders generate target sentences from left to right, and it has been proven to be important to establish the direct information flow between current predicting word and previous generated words (Zhou et al., 2017b; Vaswani et al., 2017). However, current methods still fail to estimate some desired information in the future. To address this problem, reinforcement learning methods have been applied to predict future properties (Li et al., 2017; Bahdanau et al., 2017; He et al., 2017). Li et al. (2018) presented a target foresight based attention which uses the POS tag as the partial information of a target foresight word to improve alignment and translation. Inspired by the human cognitive behaviors, Xia et al. (2017) proposed a deliberation network, which leverages the global information by observing both back and forward information in sequence decoding through a deliberation process. Zheng et al. (2018) introduced two additional recurrent layers to model translated past contents and untranslated future contents. The most relevant models in future modeling are twin networks (Serdyuk et al., 2018), which encourage the hidden state of the forward network to be close to that of the backward network used to predict the same token. However, they still used two decoders and the backward network contributes nothing during inference. Along the direction of future modeling, we introduce a single synchronous bidirectional decoder, where forward decoding can be used as future information for backward decoding, and vice versa.

Bidirectional Decoding In SMT, many approaches explored backward language models or target-bidirectional decoding to capture right-to-left target-side contexts for translation (Watanabe and Sumita, 2002; Finch and Sumita, 2009; Zhang et al., 2013). To address the issue of unbalanced outputs, Liu et al. (2016) proposed an agreement model to encourage the agreement between L2R and R2L NMT models. Similarly, some work attempted to re-rank the left-to-right decoding results by right-to-left decoding, leading to diversified translation results (Sennrich et al., 2016a; Hoang et al., 2017; Tan et al., 2017; Sennrich et al., 2017; Liu et al., 2018; Deng et al., 2018). Recently, Zhang et al. (2018) proposed asynchronous bidirectional decoding for NMT, which extended the conventional attentional encoder-decoder framework by introducing a backward decoder. Additionally, both Niehues et al. (2016) and Zhou et al. (2017a) combined the strengths of NMT and SMT, which can also be used to combine the advantages of bidirectional translation texts (Zhang et al., 2018). Compared to previous methods, our method has the following advantages: (1) We use a single model to achieve the goal of synchronous left-to-right and right-to-left decoding. (2) Our model can leverage and combine the two decoding directions in every layer of the Transformer decoder, which can run in parallel. (3) By using synchronous bidirectional attention, our model is an end-to-end joint framework and can optimize L2R and R2L decoding simultaneously. (4) Compared to two-phase decoding schemes in previous work, our decoder is more compact and faster.

6 Conclusions and Future Work

In this paper, we propose a synchronous bidirectional NMT model that performs bidirectional decoding simultaneously and interactively. The bidirectional decoder, which can take full advantage of both history and future information provided by bidirectional decoding states, predicts its outputs using left-to-right and right-to-left directions at the same time. To the best of our knowledge, this is the first attempt to integrate synchronous bidirectional attention into a single NMT model. Extensive experiments demonstrate the effectiveness of our proposed model. Particularly, our model respectively establishes state-of-the-art BLEU scores of 51.11 and 29.21 on NIST Chinese-English and WMT14 English-German translation tasks. In future work, we plan to apply this framework to other tasks, such as sequence labeling, abstractive summarization and image captioning. Additionally, it is interesting to reduce the training cost by adding noise in the target sentence and using fine-tune technology.

Acknowledgments

We would like to thank the anonymous reviewers as well as the Action Editor, George Foster, for insightful comments and suggestions. The research work has been funded by the Natural Science Foundation of China under Grant No. 61673380. This work is also supported by grants from NVIDIA NVAIL program.

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*.

Mia Xu Chen, Orhan Firat, Ankur Bapna, Melvin Johnson, Wolfgang Macherey, George Foster, Llion Jones, Mike Schuster, Noam Shazeer, Niki Parmar, Ashish Vaswani, Jakob Uszkoreit, Lukasz Kaiser, Zhifeng Chen, Yonghui Wu, and Macduff Hughes. 2018. The best of both worlds: Combining recent advances in neural machine translation. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 76–86. Association for Computational Linguistics.

Yongchao Deng, Shanbo Cheng, Jun Lu, Kai Song, Jingang Wang, Shenglan Wu, Liang Yao, Guchun Zhang, Haibo Zhang, Pei Zhang, Changfeng Zhu, and Boxing Chen. 2018. Alibaba's neural machine translation systems for wmt18. In *Proceedings of the Third Conference on Machine Translation: Shared Task Papers*, pages 368–376. Association for Computational Linguistics.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2018. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*.

Andrew Finch and Eiichiro Sumita. 2009. Bidirectional phrase-based statistical machine translation. In *Proceedings of the 2009 Conference on Empirical Methods in Natural Language Processing*, pages 1124–1132. 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.

Di He, Hanqing Lu, Yingce Xia, Tao Qin, Liwei Wang, and Tie-Yan Liu. 2017. Decoding with value networks for neural machine translation. In *Proceedings of NIPS 2017*.

Cong Duy Vu Hoang, Gholamreza Haffari, and Trevor Cohn. 2017. Towards decoding as continuous optimisation in neural machine translation. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, pages 146–156. Association for Computational Linguistics.

Philipp Koehn and Rebecca Knowles. 2017. Six challenges for neural machine translation. In

Proceedings of the First Workshop on Neural Machine Translation, pages 28–39. Association for Computational Linguistics.

Jiwei Li, Will Monroe, and Dan Jurafsky. 2017. Learning to decode for future success. *arXiv preprint arXiv:1701.06549*.

Xintong Li, Lemao Liu, Zhaopeng Tu, Shuming Shi, and Max Meng. 2018. Target foresight based attention for neural machine translation. 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 1380–1390. Association for Computational Linguistics.

Lemao Liu, Masao Utiyama, Andrew Finch, and Eiichiro Sumita. 2016. Agreement on target-bidirectional neural machine translation. In *Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 411–416. Association for Computational Linguistics.

Yuchen Liu, Long Zhou, Yining Wang, Yang Zhao, Jiajun Zhang, and Chengqing Zong. 2018. A comparable study on model averaging, ensembling and reranking in nmt. In *Natural Language Processing and Chinese Computing*, pages 299–308, Cham. Springer International Publishing.

Thang Luong, Hieu Pham, and Christopher D. Manning. 2015. Effective approaches to attention-based neural machine translation. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*, pages 1412–1421. Association for Computational Linguistics.

Haitao Mi, Baskaran Sankaran, Zhiguo Wang, and Abe Ittycheriah. 2016. Coverage embedding models for neural machine translation. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, pages 955–960. Association for Computational Linguistics.

Jan Niehues, Eunah Cho, Thanh-Le Ha, and Alex Waibel. 2016. Pre-translation for neural machine translation. In *Proceedings of COLING 2016, the 26th International Conference on Computational Linguistics: Technical Papers*, pages 1828–1836. The COLING 2016 Organizing Committee.

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*.

Rico Sennrich, Alexandra Birch, Anna Currey, Ulrich Germann, Barry Haddow, Kenneth Heafield, Antonio Valerio Miceli Barone, and Philip Williams. 2017. The university of edinburgh's neural mt systems for wmt17. In *Proceedings of the Second Conference on Machine Translation*, pages 389–399. Association for Computational Linguistics.

Rico Sennrich, Barry Haddow, and Alexandra Birch. 2016a. Edinburgh neural machine translation systems for wmt 16. In *Proceedings of the First Conference on Machine Translation: Volume 2, Shared Task Papers*, pages 371–376. Association for Computational Linguistics.

Rico Sennrich, Barry Haddow, and Alexandra Birch. 2016b. 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.

Dmitriy Serdyuk, Nan Rosemary Ke, Alessandro Sordoni, Adam Trischler, Chris Pal, and Yoshua Bengio. 2018. Twin networks: Matching the future for sequence generation. In *International Conference on Learning Representations*.

Shiqi Shen, Yong Cheng, Zhongjun He, Wei He, Hua Wu, Maosong Sun, and Yang Liu. 2016. Minimum risk training for neural machine translation. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1683–1692. 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.

Zhixing Tan, Boli Wang, Jinming Hu, Yidong Chen, and Xiaodong Shi. 2017. Xmu neural machine translation systems for wmt 17. In *Proceedings of the Second Conference on Machine Translation*, pages 400–404. Association for Computational Linguistics.

Zhaopeng Tu, Zhengdong Lu, Yang Liu, Xiaohua Liu, and Hang Li. 2016. Modeling coverage for neural machine translation. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 76–85. Association for Computational Linguistics.

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.

Taro Watanabe and Eiichiro Sumita. 2002. Bidirectional decoding for statistical machine translation. In *COLING 2002: The 19th International Conference on Computational Linguistics*.

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.

Yilin Yang, Liang Huang, and Mingbo Ma. 2018. Breaking the beam search curse: A study of (re-)scoring methods and stopping criteria for neural machine translation. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 3054–3059. Association for Computational Linguistics.

Hui Zhang, Kristina Toutanova, Chris Quirk, and Jianfeng Gao. 2013. Beyond left-to-right: Multiple decomposition structures for smt. In *Proceedings of the 2013 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 12–21. Association for Computational Linguistics.

Jiajun Zhang and Chengqing Zong. 2015. Deep neural networks in machine translation: An overview. *IEEE Intelligent Systems*, 30(5):16–25.

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*.

Zaixiang Zheng, Hao Zhou, Shujian Huang, Lili Mou, Xinyu Dai, Jiajun Chen, and Zhaopeng Tu. 2018. Modeling past and future for neural machine translation. *Transactions of the Association for Computational Linguistics*, 6:145–157.

Long Zhou, Wenpeng Hu, Jiajun Zhang, and Chengqing Zong. 2017a. Neural system combination for machine translation. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 378–384. Association for Computational Linguistics.

Long Zhou, Jiajun Zhang, and Chengqing Zong. 2017b. Look-ahead attention for generation in neural machine translation. In *Natural Language Processing and Chinese Computing*, pages 211–223, Cham. Springer International Publishing.