Neural Machine Translation With Explicit Phrase Alignment

Jiacheng Zhang, Huanbo Luan, Maosong Sun, Feifei Zhai, Jingfang Xu, and Yang Liu

***Abstract*—While neural machine translation has achieved state-of-the-art translation performance, it is unable to capture the alignment between the input and output during the translation process. The lack of alignment in neural machine translation models leads to three problems: it is hard to (1) interpret the translation process, (2) impose lexical constraints, and (3) impose structural constraints. These problems not only increase the difficulty of designing new architectures for neural machine translation, but also limit its applications in practice. To alleviate these problems, we propose to introduce explicit phrase alignment into the translation process of arbitrary neural machine translation models. The key idea is to build a search space similar to that of phrase-based statistical machine translation for neural machine translation where phrase alignment is readily available. We design a new decoding algorithm that can easily impose lexical and structural constraints. Experiments show that our approach makes the translation process of neural machine translation more interpretable without sacrificing translation quality. In addition, our approach achieves significant improvements in lexically and structurally constrained translation tasks.**

***Index Terms*—Alignment, machine translation, natural language processing, neural-networks.**

I. Introduction

NEURAL machine translation (NMT), which leverages neural networks to map between natural languages, has made remarkable progress in the past several years [1]–[3]. Capable of learning representations from data, NMT has achieved significant improvements over conventional statistical machine translation (SMT) [4] and become the new *de facto* paradigm in the machine translation community.

Despite its success, NMT suffers from a major drawback: there is no alignment to explicitly indicate the correspondence between the input and the output. As all internal information of an NMT model is represented as real-valued vectors or matrices, it is hard to associate a source word with its translational equivalents on the target side. Although the attention weights between the input and the output are available in the RNNsearch model [2],[1] these weights only reflect relevance rather than translational equivalence [5]. To aggravate the situation, attention weights between the input and the output are even unavailable in modern NMT models such as Transformer [3].

The lack of alignment in NMT leads to at least three problems. First, it is difficult to interpret the translation process of NMT models without alignment. In conventional SMT [4], the translation process can be seen as a sequence of interpretable decisions, in which alignment plays a central role. It is hard to include such interpretable decisions in NMT models without the access to alignment. Although visualization tools such as layer-wise relevance propagation [6] can be used to measure the relevance between two arbitrary neurons in NMT models, the hidden states in neural networks still do not have clear connections to interpretable language structures.

Second, it is difficult to impose **lexical constraints** on NMT systems [7] without alignment. For example, given an English sentence

`American peot Edgar Allan Poe,`

one requires that the English phrase "Edgar Allan Poe" must be translated by NMT systems as a Chinese word "ailunpo". Such lexical constraints are important for both automatic machine translation and interactive machine translation. In automatic machine translation, it is desirable to incorporate the translations of infrequent numbers, named entities, and technical terms into NMT systems [8]. In interactive machine translation, human experts expect that the NMT system can be controlled and include specified translations in the system output [9]. Although Hokamp and Liu [7] and Post and Vilar [10] provide solutions to impose lexical constraints, their methods can only ensure that the specified target words or phrases will appear in the system output. As a result, the ignorance of the alignment to the source side might deteriorate the adequacy of system output (see Table III).

Manuscript received December 30, 2019; revised June 25, 2020 and January 5, 2021; accepted January 22, 2021. Date of publication February 9, 2021; date of current version March 3, 2021. This work was supported in part by the National Key R&D Program of China under Grant 2017YFB0202204, in part by the National Natural Science Foundation of China under Grants 61925601, and 61772302, and in part by the Beijing Academy of Artificial Intelligence. *(Corresponding author: Yang Liu.)*

Jiacheng Zhang is with ByteDance AI Lab, Shanghai 200233, China and also with the Natural Language Processing Group, Department of Computer Science and Technology, Tsinghua University, Beijing 100084, China (e-mail: grit31@126.com).

Huanbo Luan and Maosong Sun are with the Department of Computer Science and Technology, Tsinghua University, Beijing 100084, China (e-mail: luanhuanbo@gmail.com; sms@tsinghua.edu.cn).

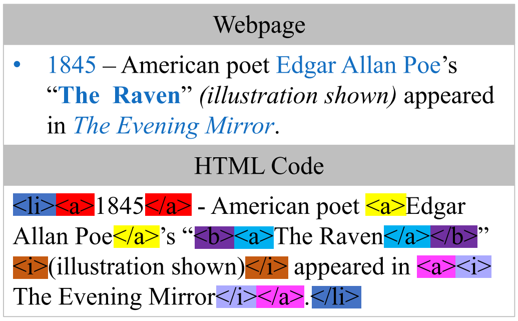
Feifei Zhai is with Fanyu Technology Company, Beijing 100190, China, and also with Sogou Machine Translation Team at Sogou Inc, Beijing 100084, China (e-mail: zhaifeifei@sogou-inc.com).

Jingfang Xu is with Sogou Inc., Beijing 100084, China (e-mail: xujingfang@sogou-inc.com).

Yang Liu is with the Institute for Artificial Intelligence, Tsinghua University, Beijing 100084, China, with the Beijing National Research Center for Information Science and Technology, Beijing 100084, China, with the Beijing Academy of Artificial Intelligence, Beijing 100084, and also with Jiangsu Collaborative Innovation Center for Language Competence, Jiangsu 330022, China (e-mail: liuyang2011@tsinghua.edu.cn).

Digital Object Identifier 10.1109/TASLP.2021.3057831

[1] RNNsearch is a recurrence-based NMT model with attention mechanism.



Webpage

- 1845 – American poet Edgar Allan Poe's "**The Raven**" *(illustration shown)* appeared in *The Evening Mirror*.

HTML Code

<li><a>1845</a> - American poet <a>Edgar Allan Poe</a>'s "<b><a>The Raven</a></b>" <i>(illustration shown)</i> appeared in <a><i>The Evening Mirror</i></a>.</li>

Fig. 1. Example structural constraints. The translation of a source string enclosed in a pair of HTML tags must be confined by the same tag pair on the target side.

Third, it is difficult to impose **structural constraints** on NMT systems without alignment. Fig. 1 shows an example of webpage and its HTML code. Unlike lexical constraints, structural constraints require that source strings enclosed in paired HTML tags must be translated as single units and the translations must be enclosed in the same paired HTML tags. For example, the Chinese translation of "⟨a ⟩ The Raven ⟨/a⟩" should be "⟨ a⟩wuya⟨/a⟩". It is challenging for NMT models trained on plain text to translate such structured text. While removing these HTML tags before translation and inserting tags back after translation will maintain translation quality but often violate structural constraints [11], [12], only translating the plain text within tags and concatenating the translations and tags in a monotonic way can strictly conform to structural constraints but impair translation quality [13].

In this work, we propose to introduce phrase alignment into the translation process of arbitrary NMT models. The basic idea is to develop an NMT model that treats phrase alignment as a latent variable. During decoding, the NMT model is used to score a search space similar with conventional phrase-based SMT [4], in which phrase alignment is readily available. While the use of the trained NMT model keeps the capabilities of NMT in learning representations from data and capturing non-local dependencies, the availability of phrase alignment makes it possible to include interpretable decisions in the translation process. We take advantage of the availability of phrase alignment to design a new decoding algorithm that applies to all the unconstrained, lexically constrained, and structurally constrained translation tasks. Experiments show that the use of phrase-based search space does not hurt the translation performance of NMT models on the unconstrained translation task. Moreover, our approach significantly improves over state-of-the-art methods on the lexically and structurally constrained translation tasks.

II. Related Work

Our work is related to three lines of research: (1) interpreting NMT, (2) constrained decoding for NMT, and (3) combining SMT and NMT.

A. Interpreting NMT

Our work is related to attempts on interpreting NMT [6], [14]. Modern NMT models such as Transformer [3] have multiple layers. There is no direct attention between the input layer and the output layer. Li *et al.* [14] pointed out that word alignment by attention is inconsistent for different layers of Transformer and the best layer only achieves an alignment error rate (AER) of 45.22, so it is not possible to interpret NMT with attention weights.

To interpret the internal working of NMT, Ding *et al.* [6] calculated the relevance between source and target words with layer-wise relevance propagation. Such relevance measures the contribution of each source word to target word instead of translational equivalence between source and target words. Li *et al.* [14] predicted alignment with an external alignment model trained on the output of a statistical word aligner and use prediction differences to quantify the relevance between source and target words. However, their external alignment model is not identical to the alignment in the translation process. Our approach differs from prior studies by introducing explicit phrase alignment into the translation process of NMT models, which makes each step in generating a target sentence interpretable to human experts.

B. Constrained Decoding for NMT

Our work is also closely related to imposing lexical constraints on the decoding process of NMT [7], [10], [15]. Hokamp and Liu [7] proposed a lexically constrained decoding algorithm for NMT. Their approach can ensure that pre-specified target strings will appear in the system output. Post and Vilar [10] improved the efficiency of lexically constrained decoding by introducing dynamic beam allocation. One drawback of the two methods is that they cannot impose lexical constraints on the source side due to the lack of alignment. Chatterjee *et al.* [15] and Hasler *et al.* [16] relied on the attention weights in the RNNsearch model [2] to impose source-aware lexical constraints with guided beam search. However, their methods can not be applied to Transformer [3]. With translation options, it is also easy to impose source-aware lexical constraints using our approach for arbitrary NMT models.

The direction of imposing structural constraints remains much unexplored, especially for NMT. Most prior studies have focused on SMT. Although the ideal solution is to directly train NMT models on parallel corpora for structured text [17]–[19], such labeled datasets are hard to construct and remain limited in quantity. Therefore, a more practical solution is to use off-the-shelf machine translation systems tailored for unstructured text to translate structured text [11]–[13]. But these approaches face the risk of performance degradation or failure to impose structural constraints correctly. Our work proposes a structurally constrained decoding algorithm for NMT to preserve structural constraints without sacrificing translation quality.

C. Combining SMT and NMT

Several authors have endeavored to combine the merits of SMT and NMT [20]–[22]. Wang *et al.* [23] and Wang *et al.* [24] treated SMT features as extra features in NMT decoding. Huang *et al.* [25] introduced the concept of reordering into their model, but their solution was neural-based. Zhang *et al.* [26] reranked NMT candidates with phrased-based decoding scores. Stahlberg

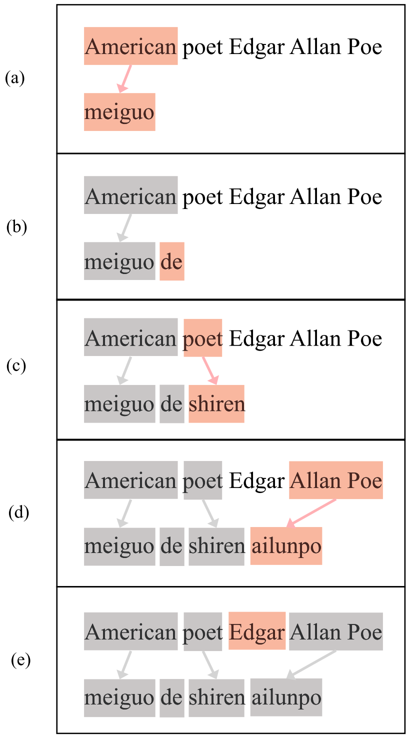

(a)
American poet Edgar Allan Poe
meiguo
(b)
American poet Edgar Allan Poe
meiguo de
(c)
American poet Edgar Allan Poe
meiguo de shiren
(d)
American poet Edgar Allan Poe
meiguo de shiren ailunpo
(e)
American poet Edgar Allan Poe
meiguo de shiren ailunpo

Fig. 2. Neural machine translation with explicit phrase alignment.

et al. [20] proposed to use the lattice output by SMT as the search space of NMT. The major difference is that our work allows for both source word omission and target word insertion, which proves to be helpful in reducing the gap between phrase-based and neural spaces. In this work, we only use NMT models to score the translations in a phrase-based space. It is possible to exploit SMT features as suggested by Dahlmann *et al.* [22].

III. Approach

Our work aims to introduce phrase alignment into the translation process of arbitrary NMT models. Fig. 2 illustrate the central idea of our approach. During decoding, the target sentence and phrase alignment are generated simultaneously. As the target sentence grows from left to right, it is easy to apply arbitrary NMT models to calculate translation probabilities in an incremental way. A key difference of our approach from conventional phrase-based SMT [4] is that unaligned source and target phrases are allowed to reduce the discrepancy between the search spaces of SMT and NMT models. For example, Fig. 2(b) uses an unaligned target phrase (i.e., “de”) and Fig. 2(e) uses an unaligned source phrase (e.g., “Edgar”). With access to phrase alignment, we develop a decoding algorithm that is capable of preserving lexical and structural constraints without sacrificing translation quality.

A. Modeling

Let $\mathbf{x} = x_1, \ldots, x_I$ be a source sentence and $\mathbf{y} = y_1, \ldots, y_J$ be a target sentence. We use x_0 to denote an empty source word that connects to all unaligned target phrases and y_0 to denote an empty target word that connects to all unaligned source phrases. We use $\mathbf{z} = z_1, \ldots, z_K$ to denote the phrase alignment between the source and target sentences. Each link $z_k = (i_b, i_e, j_b, j_e)$ is a 4-tuple, where i_b is the beginning position of the source phrase, i_e is the ending position of the source phrase, j_b is the beginning position of the target phrase, and j_e is the ending position of the target phrase. For example, the phrase alignment in Fig. 2 comprises five links: $z_1 = (1,1,1,1)$, $z_2 = (0,0,2,2)$, $z_3 = (2,2,3,3)$, $z_4 = (4,5,4,4)$, and $z_5 = (3,3,0,0)$. For convenience, we use $\mathbf{x}_{z_k}$ to denote the source phrase spanning from i_b to i_e and $\mathbf{y}_{z_k}$ to denote the target phrase spanning from j_b to j_e. For example, $\mathbf{x}_{z_4}$ is “Allan Poe” and $\mathbf{y}_{z_4}$ is “alunpo”.

More formally, our approach is based on a latent variable model given by

$$P(\mathbf{y}|\mathbf{x};\boldsymbol{\theta}) = \sum_{\mathbf{z}} P(\mathbf{y}, \mathbf{z}|\mathbf{x};\boldsymbol{\theta}), \tag{1}$$

where $\boldsymbol{\theta}$ is a set of model parameters.

The probability of generating the target sentence $\mathbf{y}$ and phrase alignment $\mathbf{z}$ given the source sentence $\mathbf{x}$ can be further factored as

$$P(\mathbf{y}, \mathbf{z}|\mathbf{x};\boldsymbol{\theta}) = \prod_{k=1}^{K} P(z_k|\mathbf{x}, \mathbf{y}_{z_1}, \ldots, \mathbf{y}_{z_{k-1}}, \mathbf{z}_{<k};\boldsymbol{\theta}) \\ P(\mathbf{y}_{z_k}|\mathbf{x}, \mathbf{y}_{z_1}, \ldots, \mathbf{y}_{z_{k-1}}, \mathbf{z}_k;\boldsymbol{\theta}), \tag{2}$$

where $P(z_k|\mathbf{x}, \mathbf{y}_{z_1}, \ldots, \mathbf{y}_{z_{k-1}}, \mathbf{z}_{<k};\boldsymbol{\theta})$ is a *phrase alignment* model and $P(\mathbf{y}_{z_k}|\mathbf{x}, \mathbf{y}_{z_1}, \ldots, \mathbf{y}_{z_{k-1}};\boldsymbol{\theta})$ is a *phrase translation* model. Note that $\mathbf{z}_{<k} = z_1, \ldots z_{k-1}$ is a partial phrase alignment. As it is challenging to estimate the phrase alignment model from data due to the exponential search space of phrase alignments, we assume that the alignment model has a uniform distribution for simplicity and leave the learning of the alignment model for future work.

We distinguish between two kinds of phrase translation models: *non-empty* and *empty*. For non-empty target phrases, the phrase translation probability can be decomposed as a product of word-level translation probabilities:

$$P(\mathbf{y}_{z_k}|\mathbf{x}, \mathbf{y}_{z_1}, \ldots, \mathbf{y}_{z_{k-1}};\boldsymbol{\theta}) \\ = \prod_{l=1}^{|\mathbf{y}_{z_k}|} P(\mathbf{y}_{z_k}^{(l)}|\mathbf{x}, \mathbf{y}_{z_1}, \ldots, \mathbf{y}_{z_{k-1}}, \mathbf{y}_{z_k}^{(1)}, \ldots, \mathbf{y}_{z_k}^{(l-1)};\boldsymbol{\theta}_n), \tag{3}$$

where $\mathbf{y}_{z_k}^{(l)}$ is the l-th word in the target phrase $\mathbf{y}_{z_k}$ and $\boldsymbol{\theta}_n$ denotes the set of model parameters related to non-empty phrases. Note that the word-level translation probabilities in Eq. (3) can be easily calculated by arbitrary NMT models.

For the empty target phrase such as $\mathbf{y}_{z_5} = y_0$, we define the phrase translation probability as

$$P(\mathbf{y}_{z_k}|\mathbf{x}, \mathbf{y}_{z_1}, \ldots, \mathbf{y}_{z_{k-1}};\boldsymbol{\theta}) \\ = P(y_0|\mathbf{x}_{z_k}, \mathbf{x}/\mathbf{x}_{z_k};\boldsymbol{\theta}_e), \tag{4}$$

where $\mathbf{x}_{z_k}$ is the source phrase aligned to y_0, $\mathbf{x}/\mathbf{x}_{z_k}$ is the surrounding context on the source side, and $\boldsymbol{\theta}_e$ is the set of model parameters related to empty phrases. For simplicity, we restrict that unaligned source phrase $\mathbf{x}_{z_k}$ to be a single source word. Note that $\boldsymbol{\theta} = \boldsymbol{\theta}_n \cup \boldsymbol{\theta}_e$.

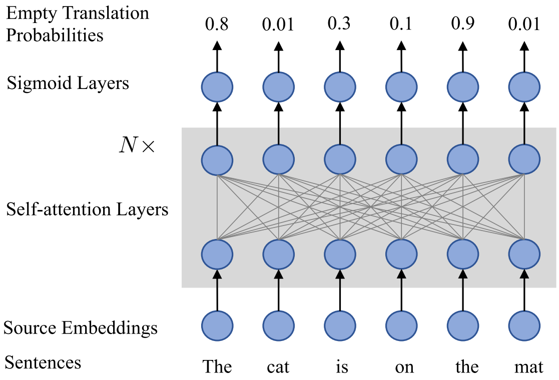


Fig. 3. The empty translation model. Function words like "the" normally have higher empty translation probabilities than non-function words like "cat" and "mat".

We use the self-attention based encoder [3] followed by a sigmoid layer (Fig. 3) to model the translation probability of empty target phrases. The encoder takes $\mathbf{x}_{z_k}$ and $\mathbf{x}/\mathbf{x}_{z_k}$ as input and output the probability of omitting $\mathbf{x}_{z_k}$.

B. Training

Given a parallel corpus $D = \{\langle \mathbf{x}^{(s)}, \mathbf{y}^{(s)}\rangle\}_{s=1}^{S}$, the standard training objective is to maximize the log-likelihood of the training data:

$$\hat{\boldsymbol{\theta}} = \underset{\boldsymbol{\theta}}{\text{argmax}} \left\{ \sum_{s=1}^{S} \log P(\mathbf{y}^{(s)}|\mathbf{x}^{(s)}; \boldsymbol{\theta}) \right\}, \qquad (5)$$

where S is the size of the parallel corpus.

As training the latent-variable model requires to enumerate all possible phrase alignments, it is impractical to directly estimate $\boldsymbol{\theta}_n$ and $\boldsymbol{\theta}_e$ jointly. Instead, we propose to train the two models separately. For the non-empty translation model in Eq. (3), the training objective is given by

$$\hat{\boldsymbol{\theta}}_n = \underset{\boldsymbol{\theta}_n}{\text{argmax}} \left\{ \sum_{s=1}^{S} \sum_{j=1}^{|\mathbf{y}^{(s)}|} \log P(y_j^{(s)}|\mathbf{x}^{(s)}, \mathbf{y}_{<j}^{(s)}; \boldsymbol{\theta}_n) \right\}. \qquad (6)$$

For the empty translation model in Eq. (4), we can use an external word alignment tool [27] to generate word alignments for the parallel corpus D. It is easy to decide whether a source word is unaligned or not based on the word alignments. As a result, the training objective for the empty translation model is given by

$$\hat{\boldsymbol{\theta}}_e = \underset{\boldsymbol{\theta}_e}{\text{argmin}} \left\{ \sum_{s=1}^{S} \sum_{i=1}^{|\mathbf{x}^{(s)}|} \text{CE}(\mathbf{x}^{(s)}, \mathbf{u}^{(s)}, \boldsymbol{\theta}_e, i) \right\}, \qquad (7)$$

where $\mathbf{u}^{(s)} = u_1^{(s)}, \ldots, u_I^{(s)}$ is an indicator vector corresponding to the s-th source sentence $\mathbf{x}^{(s)}$ that indicates whether $x_i^{(s)}$ is unaligned and $\text{CE}(\cdot)$ is the cross entropy loss defined as

$$\begin{aligned} &\text{CE}(\mathbf{x}^{(s)}, \mathbf{u}^{(s)}, \boldsymbol{\theta}_e, i) = \\ &- u_i^{(s)} \log P(y_0|x_i^{(s)}, \mathbf{x}^{(s)}/x_i^{(s)}; \boldsymbol{\theta}_e) \\ &+ (1-u_i^{(s)}) \log \left(1 - P(y_0|x_i^{(s)}, \mathbf{x}^{(s)}/x_i^{(s)}; \boldsymbol{\theta}_e)\right). \end{aligned} \qquad (8)$$

C. Decoding

Given the learned model parameters $\hat{\boldsymbol{\theta}} = \hat{\boldsymbol{\theta}}_n \cup \hat{\boldsymbol{\theta}}_e$ and an unseen source sentence $\mathbf{x}$, our goal is to find the target sentence $\hat{\mathbf{y}}$ and phrase alignment $\hat{\mathbf{z}}$ with the highest probability without violating pre-specified constraints:

$$\hat{\mathbf{y}}, \hat{\mathbf{z}} = \underset{\mathbf{y},\mathbf{z}\ s.t.\ C(\mathbf{x},\mathbf{y},\mathbf{z},\mathcal{C})=1}{\text{argmax}} \left\{ P(\mathbf{y}, \mathbf{z}|\mathbf{x}; \hat{\boldsymbol{\theta}}) \right\}, \qquad (9)$$

where $C(\mathbf{x}, \mathbf{y}, \mathbf{z}, \mathcal{C})$ is a function that checks whether the resulting translation and alignment conform to a set of pre-specified constraints $\mathcal{C}$. The function returns 1 if all constraints are satisfied and 0 otherwise.

As it is computationally expensive to enumerate all possible phrases and alignments during decoding, we resort to an external bilingual phrase table [4] to restrict the search space. Before decoding, the candidate translations of each source phrase, which are usually referred to as *translation options*, can be collected by matching the phrase table against the input sentence. Note that unlike Koehn *et al.* [4], our approach allows a source phrase or a target phrase to be unaligned.

It is easy for our approach to impose lexical constraints during the option collection process simply by replacing the translation of the pre-specified source phrase with the pre-specified target phrase. To achieve this, we restrict that (1) the pre-specified source phrase must be translated into a continuous segment and (2) its translation options do not overlap with other words. To impose structural constraints, we restrict that the translation options within a paired HTML tags do not intersect with those outside.

As unconstrained decoding is a special case of structurally constrained decoding and lexically constrained decoding can be achieved by restricting translation options, we focus on describing the structurally constrained decoding algorithm. We use a deductive system to formally describe the decoding process. An item in the deductive system is a 4-tuple $[\mathbf{x}, \mathbf{c}, S, \mathbf{y}]$ defined as follows: [2]

1) *Source sentence* $\mathbf{x}$: To capture structural constraints, we add *open constraint tags* (e.g., "⟨c1⟩" and "⟨c2⟩") and *close constraint tags* (e.g., "⟨/c1⟩" and "⟨/c2⟩") to the input, as shown in Fig. 4. Note that sentence boundaries can also be seen as constraints.
2) *Coverage vector* $\mathbf{c}$: A vector that consists of 0's and 1's to indicate which source words have been covered. The coverage vector is initialized as $\{0\}^I$.
3) *Stack* S: A stack that stores constraint tags. The decoding algorithm uses the stack to preserve structural constraints.
4) *Translation* $\mathbf{y}$: Partial translation generated during the decoding process.

Each item is associated with a log probability p yielded by our model. Note that a translation option can also be represented as an item $[\mathbf{x}, \mathbf{c}, \emptyset, \mathbf{y}]$. Except for the position of the source phrase, all other positions in $\mathbf{c}$ are set to 0. $\mathbf{y}$ is simply the target phrase. The log probability of a translation option is set to 0.

As shown in Fig. 5, the deductive system comprises three inference rules:

[2] As it is easy to obtain phrase alignment during the decoding process, we omit it in the item for simplicity.

1	2	3	4	5	6	7	8	9
<c1>	American	poet	<c2>	Edgar	Allan	Poe	</c2>	</c1>

Step	Rule	Coverage	Stack	Translation
0		000000000		
1	Push	100000000	<c1>	
2	Translate	110000000	<c1>	meiguo
3	Translate	110000000	<c1>	meiguo de
4	Translate	111000000	<c1>	meiguo de shiren
5	Push	111100000	<c1> <c2>	meiguo de shiren
6	Translate	111111100	<c1> <c2>	meiguo de shiren ailunpo
7	Push	111111110	<c1> <c2> </c2>	meiguo de shiren ailunpo
8	Pop	111111110	<c1>	meiguo de shiren ailunpo
9	Push	111111111	<c1> </c1>	meiguo de shiren ailunpo
10	Pop	111111111		meiguo de shiren ailunpo

Fig. 4. An example derivation of structurally constrained decoding. After decoding, the HTML tags can be easily recovered with explicit phrase alignment.

Item: $[\mathbf{x}, \mathbf{c}, S, \mathbf{y}]$

Axiom: $[\mathbf{x}, \{0\}^I, \emptyset, \epsilon]$

Inference rules:

Translate $$\frac{p : [\mathbf{x}, \mathbf{c}_1, S, \mathbf{y}_1], 0 : [\mathbf{x}, \mathbf{c}_2, \emptyset, \mathbf{y}_2]}{p + \log P(\mathbf{y}_2|\mathbf{x}, \mathbf{y}_1) : [\mathbf{x}, \mathbf{c}_1 + \mathbf{c}_2, S, \mathbf{y}_1\mathbf{y}_2]}$$

Push $$\frac{p : [\mathbf{x}, \mathbf{c}_1, S, \mathbf{y}], 0 : [\mathbf{x}, \mathbf{c}_2, \emptyset, s]}{p : [\mathbf{x}, \mathbf{c}_1 + \mathbf{c}_2, S|s, \mathbf{y}]}$$

Pop $$\frac{p : [\mathbf{x}, \mathbf{c}, S|s_1 s_2, \mathbf{y}]}{p : [\mathbf{x}, \mathbf{c}, S, \mathbf{y}]}$$

Goal: $[\mathbf{x}, \{1\}^I, \emptyset, \hat{\mathbf{y}}]$

Fig. 5. The deductive system of structurally constrained decoding.

1) *Translate*: Translate a source phrase using a translation option. In Fig. 5, $[\mathbf{x}, \mathbf{c}_1, S, \mathbf{y}]$ is a current item and $[\mathbf{x}, \mathbf{c}_2, \emptyset, \mathbf{y}_2]$ is a translation option. This rule is activated in two cases: the translation option covers an uncovered source phrase within the constraint ,[3] at the top of the stack, or the source phrase is empty (i.e., $\mathbf{c}_2 = \{0\}^I$).
2) *Push*: Push a constraint tag to the stack. The algorithm constructs a special translation option $[\mathbf{x}, \mathbf{c}_2, \emptyset, s]$ for a constraint tag s. For the open constraint tag "⟨c⟩," this rule is activated when all source words within the constraint are uncovered and the algorithm starts to translate any source phrase within the constraint. For the close constraint tag "⟨/c⟩," this rule is activated when all source words within the constraint are covered.
3) *Pop*: Pop the top two constraint tags from the stack. This rule is activated if the top two elements in the stack are paired open and close tags (e.g., "⟨c1⟩" and "⟨/c1⟩").

Similar to lexically constrained decoding [7], [10], we use an $I \times J$ matrix $\mathbf{M}$ to store all items generated during decoding, where I is the length of input and J is the maximum length of the output. Each element $M_{i,j}$ is a stack of items with i source words covered and j target words generated. While the time complexity of the decoding algorithm in standard NMT is $\mathcal{O}(bJ)$, the time complexity of our algorithm is $\mathcal{O}(bIJ)$, where b is the beam size (i.e., the maximum number of items stored in each stack). To speed up the decoding, our approach only keeps top-b items for all stacks with the same number of generated target words (i.e., $M_{*,j}$). As a result, the time complexity of our algorithm is reduced to $\mathcal{O}(kJ)$, which is identical to that of Post and Vilar [10].

IV. Experiments

A. Setup

We evaluate our approach on the Chinese-English and English-German translation task. We apply byte pair encoding [28] to split words into subwords. For Chinese-English translation, the training set contains 1.25 M sentence pairs from LDC[4] with 29.8 M Chinese tokens and 35.8 M English tokens after byte pair encoding [28] with 32 K merges. The NIST 2006 dataset is used as the development set and the NIST 2008 datasets is used as the test set. The evaluation metric for Chinese-English translation task is case-insensitive BLEU4 [29] as calculated by the *multi-bleu.perl* script. For English-German translation task, we use the standard WMT 2014 dataset containing 4.47 M

[3] By "within the constraint," we mean that the constraint is the innerest one that encloses a token. For example, in Fig. 4 "Edgar" is within the inner constraint c2 rather than the outer constraint c1.

[4] The training set is composed of LDC2002E18, LDC2003E07, LDC2003E14, part of LDC2004T07, LDC2004T08, and LDC-2005T06.

TABLE I
EFFECT OF EMPTY PHRASES ON THE SOURCE AND TARGET SIDES ON THE DEVELOPMENT SET

empty		BLEU
source	target	
×	×	41.69
×	√	47.43
√	×	39.28
√	√	**47.77**

sentence pairs with 128.9 M English tokens and 132.5 M German tokens after byte pair encoding with 32 K merges. We use the newstest2013 dataset as the development set and the newstest2014 dataset as the test set. The evaluation metric for English-German translation task is case-sensitive BLEU4 [29] score.

The NMT model used in our experiments is Transformer [3]. The number of layers is set to 6 for both encoder and decoder. The hidden size is set to 512 and the filter size is set to 2048. There are 8 separate heads in the multi-head attention. We use Adam [30] to optimize model parameters. During training, each batch contains approximately 25 000 tokens. We adopt the learning rate decay policy as described by Vaswani *et al.* [3]. The length penalty [31] is used and the hyper-parameter α is set to 0.6.

For our approach, we use the training set to train the non-empty translation model in Eq. (3). The same training set is also used to obtain an aligned parallel corpus using GIZA++ [27], which is used to extract a bilingual phrase table [4] to collect translation options and train the empty translation model in Eq. (4). We filter phrase pairs which co-occur less than 5 times in the training set. For each source phrase, we reserve 30 translation options with the highest phrase translation probability. The phrase table after filtering contains 1.00 M phrase pairs for Chinese-English and 3.46 M phrase pairs for English-German. The translation options of the empty source phrase are restricted to most frequent words of which the probabilities of aligning to the empty source phrase are higher than 0.2 on the training set.

We train the NMT model with 8 GTX 1080Ti GPUs, the training speed is 34 000 tokens per second on both translation directions. We train the Chinese-English NMT model for 10 hours and the English-German NMT model for 40 hours. We train the empty model with 2 GTX 1080Ti GPUs, the training speed is 75 000 tokens per second on both translation directions. We train the empty model for approximately 3 hours.

B. Results on Unconstrained Decoding

In this experiment, we compare our method with the standard Transformer model [3].

Effect of Empty Phrases: Table I shows the effect of empty source and target phrases on the Chinese-English development set. The empty source phrase allows for target word insertion and the empty target phrase permits source word omission. It is clear that introducing empty phrases on both sides is beneficial for improving translation quality, suggesting that it is important to use empty phrases to reduce the discrepancy between the phrase-based search space and neural models. An interesting finding is that allowing for target word insertion but disabling source word omission dramatically hurts the translation performance (i.e., 39.28). We find that the decoder tends to insert many meaningless target words.

TABLE II
COMPARISON BETWEEN THE STANDARD TRANSFORMER MODEL AND OUR LATENT VARIABLE MODEL

model	ZH-EN		EN-DE	
	Dev	Test	Dev	Test
Transformer	47.44	37.49	26.17	27.03
this work	**47.77**	**38.14**	26.07	26.69

Comparison With Transformer: Table II shows the comparison between the standard Transformer model and our latent variable model. Our model is different from the standard model in two aspects. First, our model uses a phrase lattice to represent the search space. Second, empty phrases are introduced to make the search space more flexible than that of conventional SMT. We find that our model slightly improves over the standard model, suggesting that we can use the phrase-based search space to replace the standard search space for lexically and structurally constrained decoding.

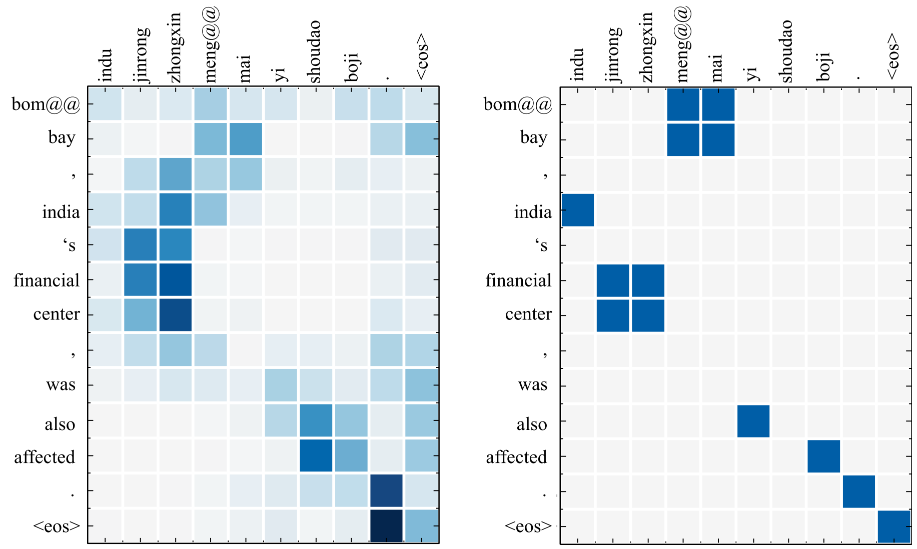
Visualization: Fig. 6 shows the comparison between the attention and alignment. As there is no attention between the input and output in the Transformer model, the heatmap in Fig. 6 is taken from the encoder-decoder attention in the third layer. In the heatmap, the attention weight is averaged over 8 different heads. While the attention matrix only reveals the relevance between source and target words, the phrase alignment generated by our model is more useful for achieving lexically and structurally constrained decoding.

C. Results on Lexically Constrained Decoding

In this experiment, we compare our method with dynamic beam allocation (DBA) proposed by Post and Vilar [10].[5] We ask human experts to pre-specify 1005 lexical constraints for the NIST 2008 Chinese-English translation dataset and 1581 lexical constraints for the newstest2014 English-German translation dataset. They are mostly translations of named entities.

On Chinese-English translation task, we find that imposing lexical constraints using DBA achieves a BLEU score of 38.54 and our approach achieves a BLEU score of 39.43. Table III shows some example translations. Given a lexical constraint (“taose,” “color blossoms”), unconstrained decoding fails to generate “color blossoms” on the target side. DBA is capable of enforcing the target phrase of the lexical constraint to appear in the translation. However, there is an extra target word “peach” (highlighted in bold) that is also connected to “taose”. In other words, “taose” is translated twice in a wrong way. To make things worse, DBA omits the source phrase “hai ting hao de” (highlighted in italic). Similar findings are also observed on the second example, in which the Chinese word “wendiya” is translated twice by DBA: “avandia” and “man dim” (highlighted

[5] We do not compare to GBS because it is difficult to compare with GBS exactly due to its variable beam size [10] Moreover, DBA improves considerably over GBS when beam size is constant.



(a) attention

(b) phrase alignment

Fig. 6. Comparison between attention and alignment. the alignment shows the translational equivalence between source and target phrases while the attention only reflects the relevance.

TABLE III

Example Translations of Two Lexically Constrained Decoding Algorithms. We Use DBA to Denote the Dynamic Beam Allocation Method Proposed by [10]. Lexical Constraints Are Highlighted in Different Colors. We Find That Although DBA Is Able to Include All Specified Target Phrases in the Translations, It Tends to Either Translate the Specified Source Phrases Repeatedly (Highlighted in Bold) or Omitting Source Phrases (Highlighted in Italic)

lexical constraints	(“taose”, “color blossoms”)
source	“ taose ” qian ban duan *hai ting hao de* , dajia dou shi xinren .
reference	the first half of “ color blossoms ” is quite good . they are all first-timers .
no constraint	the first half of the “ peach color ” is still quite good . people are new people .
DBA	in the first half of the “ **peach** color blossoms , ” people are new people .
this work	the first half of the “ color blossoms ” is still quite good . people are new people .
lexical constraints	(“yaoguanju”, “fda”), (“7 yue 30 ri”, “july 30”), (“wendiya”, “avandia”)
source	yaoguanju jiang yu 7 yue 30 ri juxing youguan wendiya anquanxing de tingzhenghui
reference	the fda will hold a hearing into the safety of avandia on july 30 .
no constraint	the drug administration will hold hearings on the safety of wendiya on july 30 .
DBA	fda avandia will hold a hearing on the safety of **man dim** on july 30 .
this work	the fda will hold hearings on the safety of avandia on july 30 .

in bold). On English-German translation task, imposing lexical constraints using DBA achieves a BLEU score of 27.51 and our approach achieves a BLEU score of 27.53.

We observe that 6.9% of the source phrases of lexical constraints on the test set are repeatedly translated by DBA while the proportion drops to 0.3% for our approach. One possible reason is that DBA ignores the source side of a lexical constraint and thus inevitably impairs the adequacy of the resulting translation.

D. Results on Structurally Constrained Decoding

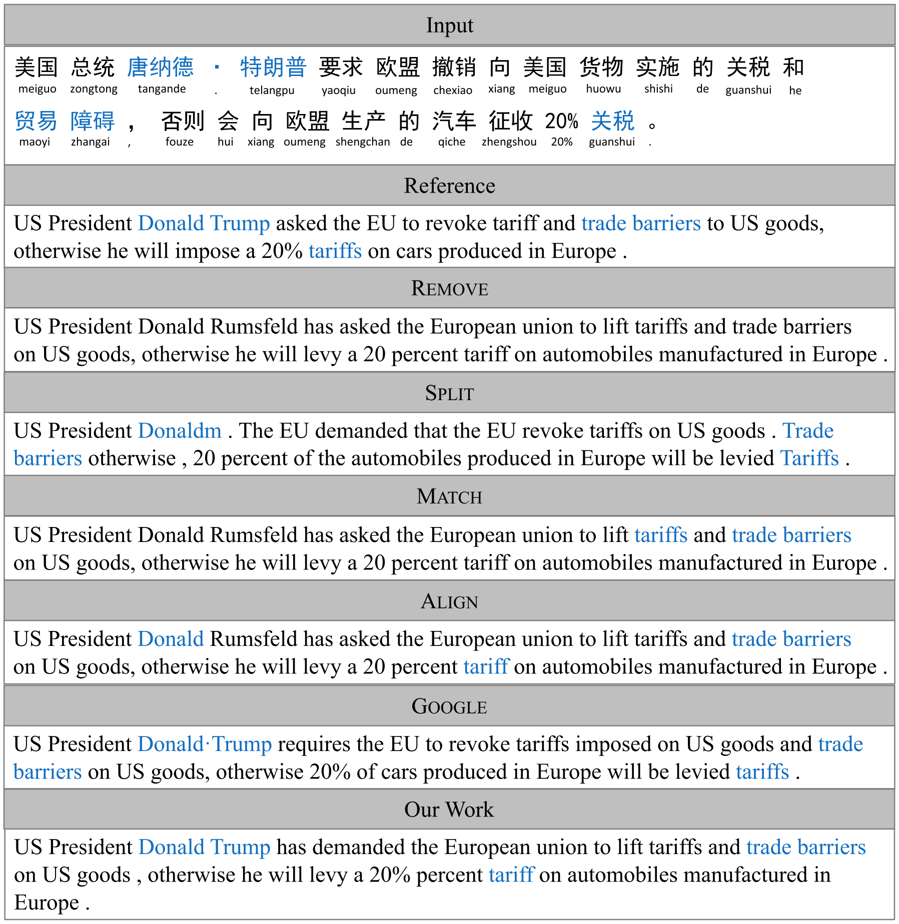
We evaluate our structurally constrained decoding algorithm on a webpage translation task.

Dataset: As labeled data is limited in quantity for webpage translation, we still use the unstructured Chinese-English and English-German dataset that contains 1.25 M and 4.47 M sentence pairs as the training set respectively. We build a test set for Chinese-English structured text translation based on the webpages of Wikipedia. The Chinese-English test set contains 500 sentence pairs with HTML tags retained. On average, each sentence pair in the test set has 15.1 Chinese words, 17.6 English words and 2.3 pairs of HTML tags. The English-German test set contains 501 sentence pairs where each sentence pair has 10.3 English words, 10.8 German words and 1.9 pairs of HTML tags averagely.

Baselines: We compare our approach with the following five baselines: [6]

1) Remove: Remove all HTML tags before decoding and do not insert tags back to translations after decoding.
2) Split [13]: Split the input by tags before decoding, translate textual parts independently, and concatenate translations monotonically after decoding.
3) Match [11]: Remove all HTML tags before decoding and insert tags back to translations by matching.

[6] We did not compare with the methods that train SMT models on parallel corpora for webpage translation because these datasets are not publicly available.



Input

美国 总统 唐纳德 · 特朗普 要求 欧盟 撤销 向 美国 货物 实施 的 关税 和
meiguo zongtong tangande . telangpu yaoqiu oumeng chexiao xiang meiguo huowu shishi de guanshui he

贸易 障碍 ， 否则 会 向 欧盟 生产 的 汽车 征收 20% 关税 。
maoyi zhangai , fouze hui xiang oumeng shengchan de qiche zhengshou 20% guanshui .

Reference

US President Donald Trump asked the EU to revoke tariff and trade barriers to US goods, otherwise he will impose a 20% tariffs on cars produced in Europe .

Remove

US President Donald Rumsfeld has asked the European union to lift tariffs and trade barriers on US goods, otherwise he will levy a 20 percent tariff on automobiles manufactured in Europe .

Split

US President Donaldm . The EU demanded that the EU revoke tariffs on US goods . Trade barriers otherwise , 20 percent of the automobiles produced in Europe will be levied Tariffs .

Match

US President Donald Rumsfeld has asked the European union to lift tariffs and trade barriers on US goods, otherwise he will levy a 20 percent tariff on automobiles manufactured in Europe .

Align

US President Donald Rumsfeld has asked the European union to lift tariffs and trade barriers on US goods, otherwise he will levy a 20 percent tariff on automobiles manufactured in Europe .

Google

US President Donald·Trump requires the EU to revoke tariffs imposed on US goods and trade barriers on US goods, otherwise 20% of cars produced in Europe will be levied tariffs .

Our Work

US President Donald Trump has demanded the European union to lift tariffs and trade barriers on US goods , otherwise he will levy a 20% percent tariff on automobiles manufactured in Europe .

Fig. 7. Example translations of all approaches to structured text translation. The input is a sentence with three pairs of HTML tags. Strings enclosed in HTML tags are highlighted in blue.

4) Align [12]: Remove all HTML tags before decoding and insert tags back to translations using word alignments generated by GIZA++.
5) Google: The Google Translate online system.[7] HTML tags are not removed before decoding.

All the baselines except Google share the same Transformer model with our approach and use the neural search space. The statistical bilingual word alignment system is trained on the same training set with our approach.

Results on Webpage Translation: Table IV shows the comparison of imposing structural constraints with existing methods on the test set. As Remove ignores all HTML tags, it is not capable of imposing structural constraints. Split ensures that the structural constraints can be imposed correctly because the sentence segments between HTML tags are translated independently, but the translation quality drops dramatically. Match and Align take the full advantage of standard NMT to translate the textual parts but often fail to recover HTML tags correctly after decoding. Our approach achieves the best performance in terms of both evaluation metrics by fully preserving the structural constraints without losing translation quality. We also report the result of Google online translation system. According to the translations, it seems that Google uses a strategy similar to Split but achieves much higher BLEU scores because it used much larger training data than all other methods.

TABLE IV
RESULTS ON THE WEBPAGE TRANSLATION TASK. "W/O TAG" DENOTES THE BLEU SCORE WITHOUT CONSIDERING HTML TAGS AND "W/ TAG" DENOTES THE BLEU SCORE CONSIDERING HTML TAGS. IN WHICH "W/ TAG" IS THE STANDARD EVALUATION METRIC FOR WEBPAGE TRANSLATION [19]. WE FOLLOW TEZCAN AND VANDEGHINSTE [19] TO PREPROCESS EACH HTML TAG TO ONE TOKEN

Training Data	Method	ZH-EN		EN-DE	
		w/o tag	w/ tag	w/o tag	w/ tag
Limited	Remove	28.66	17.14	23.20	9.87
	Split	17.40	23.20	16.00	29.07
	Match	28.66	30.40	23.20	33.57
	Align	28.66	31.92	23.20	32.97
	Ours	**28.67**	**34.05**	**23.23**	**34.59**
Unlimited	Google	**30.10**	**35.79**	**25.27**	**35.76**

Fig. 7 shows example translations of all approaches to structured text translation. Remove treats it as an unstructured text translation task. Therefore, there are no HTML tags in its translation. Split divides the input into seven textual parts. Each part is translated separately without access to other parts. After decoding, the translations of seven parts and HTML tags are

[7] [Online]. Available: https://translate.google.com/

concatenated in a monotonic way. The splitting severely impairs the translation quality of NMT. Match and Align are extensions of Remove. Note that they have the same textual translation. The difference is that Match requires a second pass to translate the strings enclosed in HTML tags separately. Then, inserting HTML tags is done by string matching. Although this approach avoid the quality loss problem, it faces the risk of matching failure (e.g., the first tag pair) or erroneous matching (e.g., the second tag pair). Align inserts HTML tags according to word alignment, which is inevitably erroneous.

V. Conclusion

We have presented a latent variable model for neural machine translation that treats phrase alignment as an unobserved latent variable. The introduction of phrase alignment makes it possible to decompose the translation process of arbitrary NMT models into interpretable steps. Our approach achieves comparable performance with the state-of-the-art model on unconstrained translation. Allowing for target word insertion and source word omission benefits translation quality by reducing the discrepancy between the phrase-based search space and neural models. In addition, it is also convenient to use our approach to impose lexical and structural constraints thanks to the availability of phrase alignment. Experiments show that the proposed method achieves significantly better performance on both lexically and structurally constrained translation tasks.

The main limitation of this study is that our approach depends on an external phrase table, which is obtained using an external statistical alignment model. In the future, we would like to investigate how to remove the dependence of our approach on an external phrase table and try to learn the latent variables from data automatically.

Acknowledgment

The authors would like to thank all the reviewers for their suggestions.

References

[1] I. Sutskever, O. Vinyals, and Q. V. Le, "Sequence to sequence learning with neural networks," in *Proc. Neural Inf. Process. Syst.*, 2014, pp. 3104–3112.

[2] D. Bahdanau, K. Cho, and Y. Bengio, "Neural machine translation by jointly learning to align and translate," in *Proc. Int. Conf. Learn. Representations*, 2015.

[3] A. Vaswani *et al.*, "Attention is all you need," in *Proc. Neural Inf. Process. Syst.*, 2017, pp. 6000–6010.

[4] P. Koehn, F. J. Och, and D. Marcu, "Statistical phrase-based translation," in *Proc. North Amer. Chapter Assoc. Comput. Linguistics: Hum. Lang. Technol.*, 2003, pp. 48–54.

[5] P. Koehn and R. Knowles, "Six challenges for neural machine translation," in *Proc.1st Workshop Neural Mach. Trans.*, 2017, pp. 28–39.

[6] Y. Ding, Y. Liu, H. Luan, and M. Sun, "Visualizing and understanding neural machine translation," in *Proc. Assoc. Comput. Linguistics*, 2017, pp. 1150–1159.

[7] C. Hokamp and Q. Liu, "Lexically constrained decoding for sequence generation using grid beam search," in *Proc. Assoc. Comput. Linguistics*, 2017, pp. 1535–1546.

[8] M.-T. Luong, I. Sutskever, Q. V. Le, O. Vinyals, and W. Zaremba, "Addressing the rare word problem in neural machine translation," in *Proc. Assoc. Comput. Linguistics*, 2015, pp. 11–19.

[9] S. Cheng, S. Huang, H. Chen, X.-Y. Dai, and J. Chen, "Primt: A. pick-revise framework for interactive machine translation," in *Proc. North Amer. Chapter Assoc. Comput. Linguistics: Hum. Lang. Technol.*, 2016, pp. 1240–1249.

[10] M. Post and D. Vilar, "Fast lexically constrained decoding with dynamic beam allocation for neural machine translation," in *Proc. North Amer. Chapter Assoc. Comput. Linguistics: Hum. Lang. Technol.*, 2018, pp. 1314–1324.

[11] R. N. Zantout and A. Guessoum, "An automatic english-arabic html page translation system," *J. Netw. Comput. Appl.*, vol. 24, no. 4, pp. 333–357, 2001.

[12] E. Joanis, D. Stewart, S. Larkin, and R. Kuhn, "Transferring markup tags in statistical machine translation: A. two-stream approach," in *Mach. Trans. Summit XIV*, 2013, pp. 73–81.

[13] F. Al-Anzi, K. Al-Zame, M. Husain, and H. Al-Mutairi, "Automatic english/arabic html home page translation tool," in *Proc. 1st Workshop Technol. Arabizing Internet*, 1997.

[14] X. Li, G. Li, L. Liu, M. Meng, and S. Shi, "On the word alignment from neural machine translation," in *Proc. Assoc. Comput. Linguistics*, 2019, pp. 1293–1303.

[15] R. Chatterjee, M. Negri, M. Turchi, M. Federico, L. Specia, and F. Blain, "Guiding neural machine translation decoding with external knowledge," in *Proc. 2nd Conf. Mach. Trans.*, 2017, pp. 157–168.

[16] E. Hasler, A. De Gispert, G. Iglesias, and B. Byrne, "Neural machine translation decoding with terminology constraints," in *Proc. NAACL-HLT*, 2018, pp. 506–512.

[17] J. Du, J. Roturier, and A. Way, "Tmx markup: A challenge when adapting smt to the localisation environment," *Eur. Assoc. Mach. Trans.*, 2010.

[18] T. Hudík and A. Ruopp, "The integration of moses into localization industry," in *Proc. Eur. Assoc. Mach. Trans., 2011*, 2011, pp. 47–54.

[19] A. Tezcan and V. Vandeghinste, "Smt-cat integration in a technical domain: Handling xml markup using pre & post-processing methods," in *Proc. Eur. Assoc. Mach. Trans.*, 2011, pp. 55–62.

[20] F. Stahlberg, E. Hasler, A. Waite, and B. Byrne, "Syntactically guided neural machine translation," in *Proc. Assoc. Comput. Linguistics*, 2016, pp. 299–305.

[21] H. Khayrallah, G. Kumar, K. Duh, M. Post, and P. Koehn, "Neural lattice search for domain adaptation in machine translation," in *Proc. Int. Joint Conf. Nat. Lang. Process.*, 2017, pp. 20–25.

[22] L. Dahlmann, E. Mastusov, P. Petrushkov, and S. Khadivi, "Neural machine translation leveraging phrase-based models in a hybrid search," in *Proc. EMNLP*, 2017, pp. 1411–1420.

[23] X. Wang, Z. Lu, Z. Tu, H. Li, D. Xiong, and M. Zhang, "Neural machine translation advised by statistical machine translation," in *Proc. AAAI*, 2017, pp. 3330–3336.

[24] X. Wang, Z. Tu, D. Xiong, and M. Zhang, "Translating phrases in neural machine translation," in *Proc. Empirical Methods Nat. Lang. Process.*, 2017, pp. 1421–1431.

[25] P.-S. Huang, C. Wang, S. Huang, D. Zhou, and L. Deng, "Towards neural phrase-based machine translation," in *Proc. Int. Conf. Learn. Representations*, 2018.

[26] J. Zhang *et al.*, "Thumt: An open source toolkit for neural machine translation," 2017, *arXiv:1706.06415*.

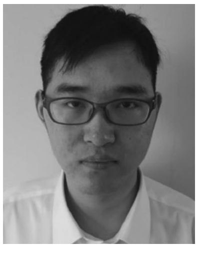
[27] F. J. Och and H. Ney, "A systematic comparison of various statistical alignment models," *Comput. Linguistics*, vol. 29, no. 1, pp. 19–51, 2003.

[28] R. Sennrich, B. Haddow, and A. Birch, "Neural machine translation of rare words with subword units," in *Proc. Assoc. Comput. Linguistics*, 2016, pp. 1715–1725.

[29] K. Papineni, S. Roukos, T. Ward, and W.-J. Zhu, "Bleu: A. method for automatic evaluation of machine translation," in *Proc. Assoc. Comput. Linguistics*, 2002, pp. 311–318.

[30] D. Kingma and J. Ba, "Adam: A. method for stochastic optimization," in *Proc. Int. Conf. Learn. Representations, 2015*, 2015.

[31] Y. Wu *et al.*, "Google's neural machine translation system: bridging the gap between human and machine translation," 2016, *arXiv:1609.08144v2*.



Jiacheng Zhang received the Ph.D. degree from the Department of Computer Science and Technology, Tsinghua University, Beijing, China, in 2020. He is currently a Researcher with ByteDance AI Lab, Shanghai, China. His research interests include machine translation and natural language processing.

Huanbo Luan received the Ph.D. degree from the Institute of Computing Technology, Chinese Academy of Sciences, Beijing, China, in 2008. He is currently an Assistant Research Fellow with the Department of Computer Science and Technology, Tsinghua University, Beijing, China. His research interests include machine translation and natural language processing.

Maosong Sun received the Ph.D. degree in computational linguistics from the City University of Hong Kong, Hong Kong, in 2004. He is currently a Professor with the Department of Computer Science and Technology, Tsinghua University, Beijing, China. His research interests include natural language processing, web intelligence, and machine learning.

Feifei Zhai received the Ph.D. degree from the Institute of Automation, Chinese Academy of Sciences, Beijing, China, in 2014. He is currently the Director with Fanyu AI Research, Beijing Fanyu Technology Company, Ltd. His research interests include natural language processing and machine translation.

Jingfang Xu received the Ph.D. degree in electronics and information engineering from Tsinghua University, Beijing, China, in 2007. She is currently the Vice President with Sogou Inc., Beijing, China. Her research interests include natural language processing and information retrieval.

Yang Liu received the Ph.D. degree from the Institute of Computing Technology, Chinese Academy of Sciences, Beijing, China, in 2007. He is currently a tenured Full Professor with the Department of Computer Science and Technology, Tsinghua University, Beijing, China. His research interests include natural language processing and machine translation.