

Adaptive Multi-pass Decoder for Neural Machine Translation

Xinwei Geng Xiao Cheng Feng Bing Qin* Ting Liu

Harbin Institute of Technology, China

{xwgeng, xcfeng, qinb, tliu}@ir.hit.edu.cn

Abstract

Although end-to-end neural machine translation (NMT) has achieved remarkable progress in the recent years, the idea of adopting multi-pass decoding mechanism into conventional NMT is not well explored. In this paper, we propose a novel architecture called *adaptive multi-pass decoder*, which introduces a flexible multi-pass polishing mechanism to extend the capacity of NMT via reinforcement learning. More specifically, we adopt an extra policy network to automatically choose a suitable and effective number of decoding passes, according to the complexity of source sentences and the quality of the generated translations. Extensive experiments on Chinese-English translation demonstrate the effectiveness of our proposed adaptive multi-pass decoder upon the conventional NMT with a significant improvement about 1.55 BLEU.

1 Introduction

In the past several years, end-to-end neural machine translation (NMT) (Kalchbrenner and Blunsom, 2013; Sutskever et al., 2014; Cho et al., 2014; Bahdanau et al., 2014) has attracted increasing attention from both academic and industry communities. Compared with conventional statistical machine translation (SMT) (Brown et al., 1993; Koehn et al., 2003), which needs to explicitly model latent structures, NMT adopts a unified *encoder-decoder* framework to directly transform a source sentence into a target sentence. Furthermore, the introduction of attention mechanism (Bahdanau et al., 2014) enhances the capability of NMT in capturing long-distance dependencies.

Recently, a number of authors have endeavored to adopt the polishing mechanism into NMT. Similar to human cognitive process for writing a good paper, their models first create a complete

Source	<i>héshìzhù xiānshēng dē wěirèngqī wéi yī nián , yǐ pèihé qí wéi fāngwěihuì wěiyuán dē rèngqī jièmán rìqī , qūā xīn wěirèngwěiyuān dē rèngqī zé wéi liǎngnián .</i>
Reference	<i>all appointments are for two years , except that of mr ho sai - chu 's which is for one year in order to tie in with the expiry date of his appointment as an ha member .</i>
1st-pass	<i>mr ho sai - chu 's UNK is a year - long term of two years with a term of two years as the term of his term of office of the ha .</i>
2nd-pass	<i>mr ho sai - chu 's UNK is a year - long term of two years with a term of two years to serve as the term of office of the ha .</i>
3rd-pass	<i>mr ho sai - chu 's UNK is a year - long term of two years with a term of two years to tie in with the expiry date of his term of office .</i>
4th-pass	<i>mr ho sai - chu has been serving as a member of authority for a term of two years with a term of two years .</i>

Table 1: Translation examples of more decoding passes with the proposed multi-pass decoder.

draft and then polish it based on global understanding of the whole draft (Niehues et al., 2016; Chatterjee et al., 2016; Zhou et al., 2017; Xia et al., 2017; Junczys Dowmunt and Grundkiewicz, 2017) . Moreover, Zhang et al. (2018) introduces a backward decoder to better exploit the right-to-left target-side contexts. Generally these methods employ two separate decoders to accomplish the polishing task.

Although these polishing mechanism-based approaches demonstrate their effectiveness with two-pass decoding, the idea of multi-pass decoding is not well explored for NMT. Motivated by it, we first propose a novel multi-pass decoder to perform the translation procedure with a fixed number of decoding passes, referred to as decoding depth. According to the preliminary results, just as expected, multi-pass decoding really benefit to most translations. However, in some cases, the more decoding passes perhaps lead to the poor translation. For example in Table 1, the 3rd-pass de-

* Corresponding author.

coding achieves a better result compared to 1st- and 2nd-pass decoding. Nevertheless, a drastic decrease arises, when we perform the 4th-pass decoding. Therefore, it's necessary to introduce a flexible multi-pass decoding, which has the ability to adaptively choose the suitable decoding passes.

Towards above goal, we further propose a novel framework called *adaptive multi-pass decoder* to automatically choose a proper decoding depth using reinforcement learning. Our model considers multi-pass decoding as a sequential decision making process, where continuing decoding or halt is chosen at each step. An extra policy network is employed to learn to automatically choose to continue next pass decoding or halt via reinforcement learning. For the purpose of making accurate and effective choices, the policy network employs recurrent neural network to capture the complexity of source sentence as well as the difference between the consecutive generated translations. Extensive experiments on Chinese-English translation show the proposed adaptive multi-pass decoder is capable of choosing a suitable decoding depth and significantly improves translation performance over conventional NMT model.

2 Background

Given a source sentence $\mathbf{x} = x_1, \dots, x_m, \dots, x_M$ and a target sentence $\mathbf{y} = y_1, \dots, y_n, \dots, y_N$, end-to-end neural machine translation directly models translation probability word by word as a single, large neural network:

$$P(\mathbf{y}|\mathbf{x}; \theta) = \prod_{n=1}^N P(y_n|\mathbf{x}, \mathbf{y}_{<n}; \theta) \quad (1)$$

where θ is a set of model parameters and $\mathbf{y}_{<n}$ denotes a partial translation. Prediction of n -th word is generally made in an *encoder-decoder* framework:

$$P(y_n|\mathbf{x}, \mathbf{y}_{<n}; \theta) = g(\mathbf{y}_{n-1}, \mathbf{s}_n, \mathbf{c}_n) \quad (2)$$

where $g(\cdot)$ is a non-linear function, \mathbf{y}_{n-1} denotes the previously generated word, \mathbf{s}_n is n -th decoding hidden state, and \mathbf{c}_n is a context vector for generating n -th target word. The decoder state \mathbf{s}_n is computed by RNNs as follows:

$$\mathbf{s}_n = f(\mathbf{s}_{n-1}, \mathbf{y}_{n-1}, \mathbf{c}_n) \quad (3)$$

where $f(\cdot)$ is an activation function. Actually it's found gated RNN alternatives such as LSTM

(Hochreiter and Schmidhuber, 1997) or GRU (Cho et al., 2014) often achieve better performance than vanilla ones. \mathbf{c}_n is a dynamic vector that selectively summarizes certain parts of source sentence at each decoding step:

$$\mathbf{c}_n = \sum_{m=1}^M \alpha_{n,m} \mathbf{h}_m \quad (4)$$

where $\alpha_{m,n}$ measures how well \mathbf{x}_m and \mathbf{y}_n are aligned, calculated by attention model (Bahdanau et al., 2014; Luong et al., 2015), and \mathbf{h}_m is the encoder hidden state of the m -th source word. For the purpose of capturing both forward and backward contexts, bidirectional RNN (Schuster and Paliwal, 1997) is often employed as the encoder which converts the source sentence into an annotation sequence $\mathbf{h} = \{\mathbf{h}_1, \dots, \mathbf{h}_m, \dots, \mathbf{h}_M\}$, where $\mathbf{h}_m = [\vec{\mathbf{h}}_m, \overleftarrow{\mathbf{h}}_m]$ captures information about m -th word with respect to the preceding and following words in the source sentence respectively.

Although the introduction of RNNs as a decoder has resulted in substantial improvements in terms of translation quality, simultaneously it imposes a serious restriction on the capability of *encoder-decoder* framework caused by the structure of RNNs. That is, when the RNN decoder generates the t -th word \mathbf{y}_t in decoding phase, only $\mathbf{y}_{<t}$ can be utilized, while the possible words $\mathbf{y}_{>t}$ are directly neglected. Thus, it's difficult to capture global information especially the un-generated words for the current dominant RNN decoder without new significant innovation. Under the premise of preserving the original structure, a promising alternative to address the aforementioned issue is to incorporate with auxiliary neural networks to extend the RNN decoder.

Towards above goal, polishing mechanism-based methods first capture the global information through a complete draft created by SMT or NMT, and then take it as input to finally generate a translation. Compared with conventional NMT, polishing mechanism-based methods make a more accurate prediction at each time-step due to the extra global understanding, resulting in more fluent and grammatically correct translation. While these approaches have demonstrated the effectiveness, previous approaches follow pre-defined routes to perform the decoding procedure, not considering choosing a suitable decoding depth for the complexity of source sentences completely.

Therefore, it's important to develop a novel

framework for making an accurate and effective choice about which decoding depth is appropriate for the source sentence.

3 Adaptive Multi-pass Decoder

In this section, we present an adaptive multi-pass decoder for neural machine translation, as illustrated in Figure 1. It could choose a proper decoding depth, depending on the complexity of the source sentence. As shown in Figure 1, our model includes three major components: an encoder to summarize source sentences with parameter set θ_e , a multi-pass decoder for multi-pass decoding with parameter set θ_d , and a policy network to choose a suitable depth with parameter set θ_p . The encoder of our model is identical to that of the dominant NMT which is modeled using a bidirectional RNN. Please refer to Bahdanau et al. (2014) for more details. We will elaborate the *multi-pass decoder* and *policy network* for adaptive multi-pass decoding in the following subsections.

3.1 Multi-pass Decoder

The multi-pass decoder is extended from the one of the dominant NMT model to leverage the target-side context. Similar to the dominant NMT model, our multi-pass decoder also performs the decoding under the semantic guide of source-side context captured by the encoder, whereas more importantly and differently, the global understanding through the target-side context provided by last pass decoding, is able to strongly assist our model to produce a better translation.

Given the source-side and target-side contexts separately captured by the encoder and last pass decoding, the multi-pass decoder learns to generate next target word, based on previous generated words. Using the multi-pass decoder with parameter set θ_d , we calculate the conditional probability of the translation \hat{y}^l at the l -th decoding pass as follows:

$$\begin{aligned} & P(\hat{y}^l | \mathbf{x}, \hat{y}^{l-1}; \theta_e, \theta_d) \\ &= \prod_{n=1}^{N_l} P(\hat{y}_n^l | \mathbf{x}, \hat{y}_{<n}^l, \hat{y}^{l-1}; \theta_e, \theta_d) \\ &= \prod_{n=1}^{N_l} g^{dec}(\hat{y}_{n-1}^l, \mathbf{s}_n^{l,dec}, \mathbf{c}_n^{l,enc}, \mathbf{c}_n^{l,dec}) \end{aligned} \quad (5)$$

where $g^{dec}(\cdot)$ is a non-linear function, and $\mathbf{s}_n^{l,dec}$ denotes the n -th decoding state within the l -th decoding pass. N_l indicates the length of generated

translation at the l -th decoding pass. The decoding state $\mathbf{s}_n^{l,dec}$ is obtained by RNNs as follows:

$$\mathbf{s}_n^{l,dec} = f^{dec}(\mathbf{s}_{n-1}^{l,dec}, \hat{\mathbf{y}}_{n-1}^l, \mathbf{c}_n^{l,enc}, \mathbf{c}_n^{l,dec}) \quad (6)$$

where $f^{dec}(\cdot)$ is the GRU activation function. $\mathbf{c}_n^{l,enc}$ and $\mathbf{c}_n^{l,dec}$ denote source-side and target-side contexts at the n -th time step within the l -th decoding pass, respectively. It should be noted that when the multi-pass decoder performs the first decoding, there doesn't exist any generated translation. To address this case, the first-pass target-side context $\mathbf{c}^{1,dec}$ is set to zero.

Among the aforementioned contexts, $\mathbf{c}_n^{l,enc}$ is obtained as the weighted sum of the source-side hidden states $\{\mathbf{h}_m\}$, while we take the target-side hidden states $\{\mathbf{s}_n^{l-1,dec}\}$ produced by last pass decoding as input to compute $\mathbf{c}_n^{l,dec}$. Similar to the dominant NMT model, we adopt the attention model (Bahdanau et al., 2014; Luong et al., 2015) to calculate the weights, which indicate the alignment probability. We assume that $attn^{enc}$ denotes the encoder-decoder attention model, which takes the source annotations $\{\mathbf{h}_m\}$ as input, while $attn^{dec}$ are introduced to calculate the weight which measures how well the decoding state $\mathbf{s}_n^{l,dec}$ attends the last-pass hidden states $\{\mathbf{s}_n^{l-1,dec}\}$.

Assuming \mathbf{s}^a indicates the decoding state, which attends the annotations $\{\mathbf{s}_k^b\}$ with a length K , our attention model calculates the context vector \mathbf{s}^c as follows:

$$\mathbf{s}^c = \sum_{k=1}^K \alpha_k \mathbf{s}_k^b \quad (7)$$

$$\alpha_k = \frac{\exp(e_k)}{\sum_{k'=1}^K \exp(e_{k'})} \quad (8)$$

$$e_k = (v_a)^T \tanh(W_a \mathbf{s}^a + U_a \mathbf{s}_k^b) \quad (9)$$

where v_a , W_a and U_a are the parameters of attention model.

Given a training (\mathbf{x}, \mathbf{y}) , the translation route can be demonstrated as: $\mathbf{x} \rightarrow \hat{y}^1 \rightarrow \dots \rightarrow \hat{y}^l \rightarrow \dots \rightarrow \hat{y}^{L(\mathbf{x},\mathbf{y})-1} \rightarrow \mathbf{y}$. The intermediate translations $\{\hat{y}^l\}$ are generated by decoding. Given a training corpus $D = \{\mathbf{x}, \mathbf{y}\}$, we define the object function using cross-entropy at last pass decoding as follows:

$$\begin{aligned} J^{dec}(\theta_e, \theta_d) &= -\frac{1}{|D|} \arg \min_{\theta_e, \theta_d} \sum_{(\mathbf{x}, \mathbf{y}) \in D} \\ & \{\log P(\mathbf{y} | \mathbf{x}, \hat{\mathbf{y}}^{L(\mathbf{x},\mathbf{y})-1}; \theta_e, \theta_d)\} \end{aligned} \quad (10)$$

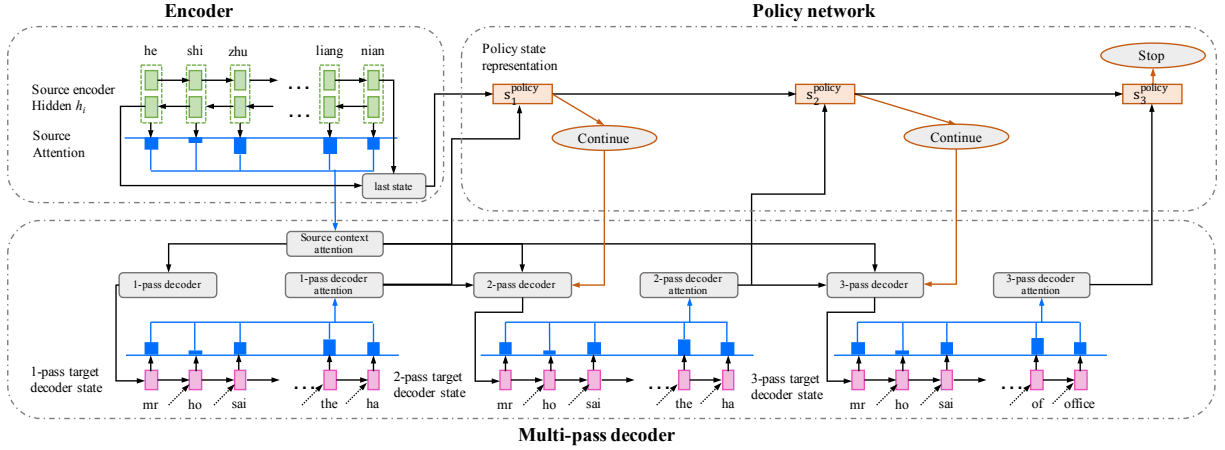


Figure 1: The architecture of our *adaptive multi-pass decoder*. Given the annotation sequence produced by the encoder, a policy network is adopted to choose a suitable action from the set $\{Continue, Stop\}$, which indicates continuing next pass decoding, or halt respectively. Different from the conventional decoder which only obtains the source-side context with the source attention model $attn^{enc}$, our multi-pass decoder also captures the target-side context of last-pass decoding with decoder attention model $attn^{dec}$. The policy network also use $attn^{policy}$ to collect useful information from the multi-pass decoding to choose an accurate and effective action to generate a good translation. Note that in this work the same parameters set of decoder and the corresponding attention is shared among different decoding passes. For this figure, we demonstrate a translation procedure with 3-pass decoding controlled by adaptive multi-pass decoder.

where $P(\mathbf{y}|\mathbf{x}, \hat{\mathbf{y}}^{L(\mathbf{x}, \mathbf{y})-1}; \theta_e, \theta_d)$ is conditional probability computed by multi-pass decoder. $L(\mathbf{x}, \mathbf{y})$ indicates the decoding depth for the instance (\mathbf{x}, \mathbf{y}) . For effectiveness, note that all the intermediate translations $\{\hat{\mathbf{y}}^l\}$ are generated by greedy search in training and testing phase.

3.2 Policy Network

The multi-pass decoding can be converted into sequential decision making process, in which a policy is adopted to choose next pass decoding or halt. It's expected to automatically choose an accurate and effective decoding depth to generate a good translation. For example, if the source sentence is exhausted to obtain the corresponding translation such as the long sentences, we assume more decoding passes are needed to improve the translation, while only one pass decoding is enough to tackle the simple case.

Our main idea is to use reinforcement learning to control the decoding depth. We parameterize the available action $\mathbf{a}_l \in \{Continue, Stop\}$, where *Continue* and *Stop* indicate continuing next decoding pass and halt respectively, by a policy network $\pi(\mathbf{a}_l | \mathbf{s}_l^{policy}; \theta_p)$, where \mathbf{s}_l^{policy} represents the policy state at the l -th decoding pass. For the purpose of making a better choice about the decoding depth and direction, it's necessary to consider whether or not the source sentence is easy to

obtain a good translation and compared with the last pass decoding, whether the quality of translation can be improved. Thus, supervised by this guideline, the policy state \mathbf{s}_l^{policy} is calculated by GRU to model the difference between the consecutive two decoding passes as follows:

$$\mathbf{s}_l^{policy} = f^{policy}(\mathbf{s}_{l-1}^{policy}, m_l) \quad (11)$$

where f^{policy} is the activation function, and m_l captures the useful information with respect to the policy network at the l -th decoding pass. In this work, we use the attention models $attn^{policy}$ to collect the decoding progress, denoted as m_l of the l -th decoding pass. In order to take account of the complexity of source sentence itself, the initial policy state \mathbf{s}_0^{policy} is computed by $\mathbf{s}_0^{policy} = \tanh(W_{init}\mathbf{h}_M)$, where \mathbf{h}_M is last state source annotations, and W_{init} is the parameters of initializing the policy state. Finally, we take the policy state \mathbf{s}_l^{policy} as input to calculate the policy as follows:

$$\pi(\mathbf{a}_l | \mathbf{s}_l^{policy}; \theta_p) = softmax(W_p \mathbf{s}_l^{policy} + b_p) \quad (12)$$

where W_p and b_p are the parameters of the policy network. In this work we use REINFORCE algorithm (Williams, 1992), which is an instance of a broader class of algorithms called policy gradient methods (Sutton and Barto, 1998), to learn the parameter set θ_p such that the sequence of actions

$\mathbf{a} = \{\mathbf{a}_1, \dots, \mathbf{a}_l, \dots, \mathbf{a}_{L(x,y)}\}$ maximizes the total expected reward. The expected reward for an instance is defined as:

$$J^{policy}(\theta_p) = \mathbb{E}_{\pi(\mathbf{a} | s^{policy}; \theta_p)} r(\hat{\mathbf{y}}^{L(x,y)}) \quad (13)$$

where $r(\hat{\mathbf{y}}^{L(x,y)})$ is the reward at the $L(x,y)$ -th decoding pass. In this work, we use BLEU (Papineni et al., 2002) of the final translation $\hat{\mathbf{y}}^{L(x,y)}$ generated by greedy search as input to compute our reward as follows:

$$r(\hat{\mathbf{y}}^{L(x,y)}) = \text{BLEU}(\hat{\mathbf{y}}^{L(x,y)}, \mathbf{y}) \quad (14)$$

4 Experiments

In this section, we describe experimental settings and report empirical results.

4.1 Setup

We evaluated the proposed *adaptive multi-pass decoder* on Chinese-English translation task. The evaluation metric was case-insensitive BLEU (Papineni et al., 2002) calculated by the `multi-bleu.perl`¹ script. The training corpus² consisted of 1.25M bilingual sentences with 27.9M Chinese words and 34.5M English words. We used the NIST 2002 (MT02) as the validation set for hyper-parameter optimization and model selection, and NIST 2003 (MT03), 2004 (MT04), 2005 (MT05) and 2006 (MT06) as test sets.

To effectively train the NMT model, we trained each model with sentences of length up to 50 words. Besides, we limited vocabulary size to 30K for both languages and map all the out-of-vocabulary words in the Chinese-English corpus to a special token UNK. We applied *Rmsprop* (Graves, 2013) to train models and selected the best model parameters according to the model performance on the development set. During this procedure, we set the following hyper-parameters: word embedding dimension as 620, hidden layer size as 1000, learning rate as 5×10^{-4} , batch size as 80, gradient norm as 1.0, and dropout rate as 0.3.

In the experiments, we compared our approach against the following state-of-the-art SMT and NMT systems:

¹<https://github.com/moses-smt/mosesdecoder/blob/master/scripts/generic/multi-bleu.perl>

²The training corpus includes LDC2002E18, LDC2003E07, LDC2003E14, part of LDC2004T07, LDC2004T08 and LDC2005T06

1. *Moses*³: an open source phrase-based translation system with default configuration and a 4-gram language model trained on the target portion of training data. Note that we used all data to train MOSES (Koehn et al., 2007).
2. *RNNSearch*: a variant of the attention-based NMT system (Bahdanau et al., 2014) with slight changes from dl4mt tutorial⁴.
3. *Deliberation Network*⁵: a re-implementation of attention-based NMT system with two independent left-to-right decoders (Xia et al., 2017). The first-pass decoder is identical to one of RNNSearch to generate a draft translation, while the second-pass decoder polishes it with an extra attention over the first pass decoder. The second-pass decoder is integrated with the first-pass decoder via reinforcement learning.
4. *ABDNMT*: As a comparison with the Deliberation Network, ABDNMT utilizes first-pass backward decoder to generate a translation with greedy search, and the second-pass forward decoder refines it with attention model (Zhang et al., 2018). For fairness, we replace the first-pass backward decoder with a forward decoder.

We set the beam size of all above-mentioned models as 10 in our work. Deliberation Network and ABDNMT were initialized with the pre-trained RNNSearch as Xia et al. (2017) and Zhang et al. (2018) described. Our multi-pass decoder was also initialized with RNNSearch and other parameters were randomly initialized from a uniform distribution on $[-0.1, 0.1]$. Besides, for effectiveness, we set the maximum decoding depth of our adaptive multi-pass decoder as 5.

4.2 Results on Chinese-English Translation

The experimental results of our model and baseline models on Chinese-English machine translation datasets are depicted in Table 2.

³<http://www.statmt.org/moses>

⁴<https://github.com/nyu-dl/dl4mt-tutorial>

⁵We reproduce the deliberation network based on REINFORCE and gumbel-softmax (Jang et al., 2016), separately, but there still exists a gap with its best performance. We attribute this to that our reimplementation may be different from the original model in some unknown details.

System	#Params	Speed	MT02	MT03	MT04	MT05	MT06	Ave.
Moses	–	–	33.79	30.86	32.71	30.02	30.49	31.02
RNNSearch	83.99M	87	39.68	36.51	40.20	36.87	36.43	37.50
Deliberation Network	125.16M	162	40.98	37.82	40.56	37.67	37.20	38.31
ABDNMT	122.86M	132	41.12	38.01	41.20	38.07	37.59	38.71
2-pass decoder	87.81M	160	41.18	37.76	41.06	38.02	37.41	38.56
3-pass decoder	87.81M	245	41.28	37.99	40.72	37.86	37.63	38.55
4-pass decoder	87.81M	293	41.05	37.86	40.87	38.18	37.57	38.62
5-pass decoder	87.81M	322	40.88	37.70	40.84	38.06	37.97	38.64
Adaptive multi-pass decoder	96.01M	180	41.42	38.39	41.43	38.54	37.86	39.05

Table 2: Evaluation of the NIST Chinese-English translation task. The BLEU scores are case-insensitive. “Params” denotes the number the parameters in each model. The “Speed” denotes the generation speed in seconds on the development set. RNNSearch is an attention-based neural machine translation model (Bahdanau et al., 2014) with one-pass left-to-right decoding. RNNSearch(R2L) is a variant of RNNSearch with one-pass right-to-left decoding. As a comparison, Deliberation Network (Xia et al., 2017) and ABDNMT (Zhang et al., 2018) involve two independent decoders to adopt polishing mechanism to extend the ability of conventional NMT. Deliberation Network utilizes two left-to-right decoders coupled with reinforcement learning. However, ABDNMT exploits a backward decoder to perform first-pass right-to-left decoding. {2,3,4,5}-pass decoder utilizes our multi-pass decoder with a fixed number of decoding passes. Furthermore, adaptive multi-pass decoder involves a policy network to enhance our multi-pass decoder to choose a proper decoding depth.

Parameters RNNSearch, Deliberation Network and ABDNMT have 83.99M, 125.16M and 122.86M parameters, respectively. And the parameter size of our {2,3,4,5}-pass decoder and adaptive multi-pass decoder are about 87.81M and 96.01M, respectively.

Fixed Decoding Depth {2,3,4,5}-pass decoders perform the left-to-right decoding by the multi-pass decoder with a fixed number of decoding passes. In contrast to the related machine translation systems, our fixed number-pass decoder significantly outperforms Moses and RNNSearch by 7.53 and 1.05 BLEU points at least, as Table 2 presents. More importantly, our proposed multi-pass decoder obtains much better performance with an increase of only 3.82M parameters over RNNSearch. As a comparison with Deliberation Network involves two-pass decoding, the multi-pass decoder has a minimum increase of 0.24 BLEU score. Nevertheless, our multi-pass decoder proves its effectiveness due to the less parameters consumption of 37.35M in contrast to Deliberation Network. These results verify our hypothesis that the more decoding passes can polish the generated output to improve the translation quality. The underlying reason is that the attention component $attn^{dec}$ within our multi-pass decoder can capture the extra target-side contexts to obtain a global understanding to assist the translation procedure.

Towards the effect of the decoding depth set {2,3,4,5}, our multi-pass decoders obtain the approximate results, but the whole curve of BLEU

is on an upward trend. Specifically, the multi-pass decoder with decoding depth 5 achieves the best performance with 38.64 BLEU, while the one with decoding depth 3 performs the worst among the decoding depth set with 38.55 BLEU. Although the average results of {2,3,4,5}-pass decoder are approximate, the distinction of {2,3,4,5}-pass decoder on NIST03, NIST04, NIST05, NIST06 and NIST08 is not negligible. These results indirectly prove the necessity of flexibility mechanism.

Adaptive Decoding Depth our proposed adaptive multi-pass decoder involves an extra policy network which controls the decoding depth according to the complexity of the source sentence and the differences between the consecutive generated translations. As shown in Table 2, the proposed adaptive multi-pass decoder obtains an improvement about 0.41 to 0.5 BLEU on average over the {2,3,4,5}-pass decoder, which demonstrates the effectiveness of the policy network. Specifically, the adaptive multi-pass decoder outperforms the multi-pass decoder with a fixed decoding depth by 0.69, 0.71, 0.68 and 0.45 BLEU scores on NIST03, NIST04, NIST05 and NIST06 datasets at most. In contrast to the Moses, RNNSearch, Deliberation Network and ABDNMT, the adaptive multi-pass decoder has the corresponding improvement about 8.03, 1.55, 0.74 and 0.34 BLEU points, respectively. More importantly, our adaptive multi-pass decoder outperforms ABDNMT, Deliberation Network model with a decrease of 26.85M, 29.15M parameters.

In order to further demonstrate the effective-

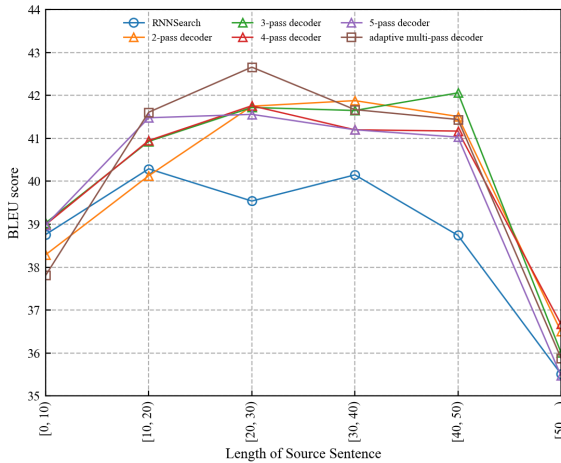


Figure 2: Performance of the generated translations with respect to the lengths of the source sentences on the development dataset.

ness of adaptively choosing the decoding depth, we investigate the ratio of decoding passes consumed by our multi-pass decoder on the development dataset, as shown in Table 3. Our adaptive multi-pass decoder chooses one-pass decoding in a high ratio of 46.57%, while in most about 53.43% cases our model leverages more than one pass decoding to produce a translation. The average decoding depth of our model is calculated as: $(1 \times 46.36\% + 2 \times 20.84\% + 3 \times 13.10\% + 4 \times 13.55\% + 5 \times 6.15\%) = 2.12$. Moreover, our ratio of the samples tends to decrease as the decoding depth rises on a whole. Since time consumption correlates with decoding depth, our adaptive multi-pass decoder proves its superior performance due to fewer parameters and less decoding passes.

Depth	1	2	3	4	5
Ratio(%)	46.57	20.45	13.00	13.60	6.38

Table 3: The ratio of decoding depth chosen by adaptive multi-pass decoder on the development dataset.

Time Consumption Due to the multi-pass decoding mechanism, the major limitation of our proposed multi-pass decoder is time cost. In training phrase, we spend more time training the multi-pass decoder than RNNSearch, Deliberation Network and ABDNMT. However, in testing phrase, as illustrated in Table 2, our adaptive multi-pass decoder spends about 180s completing the entire testing procedure, in comparison with the corresponding 87s, 162s, 132s of RNNSearch, Deliberation Network and ABDNMT, due to the auxiliary policy network. These results are consistent with above conclusion drew according to the decoding

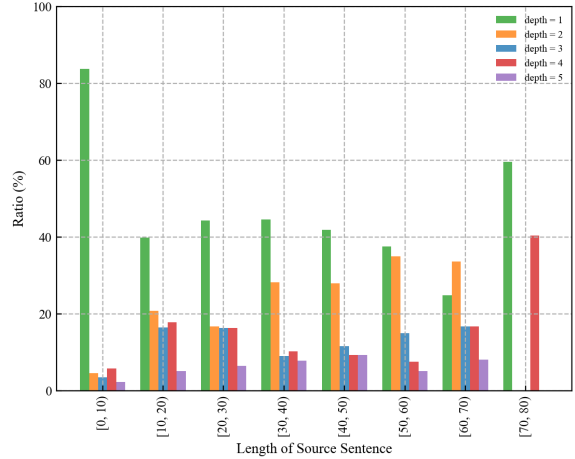


Figure 3: Ratio of decoding depth set $\{1, 2, 3, 4, 5\}$ controlled by our adaptive multi-pass decoder with respect to each length segment of the source sentences on the development dataset.

depth. Therefore, it’s proven the necessity of our proposed auxiliary policy network to choose the decoding depth.

4.3 Effect of Source Sentence Length

Following Bahdanau et al. (2014), we group sentences of similar lengths together and compute the BLEU score for each group, as shown in Figure 2. Obviously, our proposed adaptive multi-pass decoder outperforms RNNSearch in all length segments. Compared with $\{2, 3, 4, 5\}$ -pass decoders, our adaptive multi-pass decoder outperforms most even all the multi-pass decoders with fixed decoding depth in the length segments.

For the purpose of investigating the flexibility of policy network, we calculate the ratios of decoding depth set $\{1, 2, 3, 4, 5\}$ on each sentence group with similar length, as illustrated in Figure 3. The ratio of one-pass decoding remains high level on each length segment, but explicitly is dominant on the length segment $[0, 10)$. In contrast, the ratios of remaining decoding depths show upwards trends on a whole. These results indirectly proves that our policy network has the capability of choosing the proper decoding depth. That is, when the source sentence is difficult to be translated such as some long sentences, more decoding passes are consumed to improve the translation quality, while in simple cases such as short source sentences, one-pass decoding is adequate.

4.4 Case Study

To better understand the effectiveness of flexible polishing mechanism adopted by policy network, Table 4 provides a Chinese-English transla-

Source	<i>xīnhuáshè běijīng 4 yuè 5 rì diàn dà míng dǐng dǐng dē wēiruǎn gōngsī zǒngcái biē · gàicǐ rìqián bèi jiānádà yì jiā guǎngbō diàn tái " shuàn le " yì bǎ , zài yúrénjié nàitiān bèi kāi le yīgè dàwángxiào .</i>
Reference	<i>xinhua news agency, beijing, april 5, bill gates, the all-famous microsoft chairman, was duped by a canadian radio station the the other day and fell a victim to a big prank on april fools' day.</i>
1st-pass	<i>xihua news agency report of april 5 th from beijing (by staff reporter UNK UNK) - the president of microsoft's microsoft corporation, gates, was recently " UNK " by a radio station in canada and was hit by a UNK day on the day of the day.</i>
2nd-pass	<i>xinhua news agency, beijing, april 5, a few days ago, microsoft's president, microsoft corporation, was " UNK " by a radio station in canada.</i>
3rd-pass	<i>xinhua news agency, beijing, april 5, microsoft's president bill gates, the president of microsoft, was " UNK " by a radio station in canada in few days ago.</i>
4th-pass	<i>xinhua news agency, beijing, april 5, microsoft's president bill gates, the president of microsoft, was " UNK " by a radio station in canada in few days ago.</i>
5th-pass	<i>xinhua news agency, beijing, april 5, microsoft's president bill gates, the president of microsoft, was " UNK " by a radio station in canada in few days ago.</i>

Table 4: Translation examples at each decoding depth of adaptive multi-pass decoder.

tion example. Our proposed adaptive multi-pass decoder has the ability to polish the generated hypothesis again and again. As shown in Table 4, we force our adaptive multi-pass decoder to perform the multi-pass decoding with fixed depth sets $\{1,2,3,4,5\}$. The translation quality has an upwards trend with decoding depth 1 to 3, and the decoding with depth set $\{4,5\}$ generates the identical translation as the decoding depth 3. Moreover, given the same source sentence, we use the proposed adaptive multi-pass decoder to choose the decoding depth. As expected, our adaptive multi-pass chooses 3-pass decoding which generates best translation and consumes least time, rather than $\{4,5\}$ -pass decoding. Therefore, these results proves the effectiveness of our adaptive multi-pass decoder.

5 Related Work

In this work, we mainly focus on how to adopt adaptive polishing mechanism into NMT model, which has attracted intensive attention in recent years. We will elaborate polishing mechanism-based methods in the following pages.

The polishing mechanism-based approaches first generate a complete draft, and then improve the quality of it based on the global understanding of the whole draft. A related work is post-editing (Niehues et al., 2016; Chatterjee et al., 2016; Zhou et al., 2017; Junczys Dornant and Grundkiewicz, 2017): a source sentence e is first translated to f , and then f is refined by another model. Niehues et al. (2016) used phrase-based statistical machine translation (PBMT) to pre-translate the source sentence into target language, which was taken as input of NMT to generate the final translation. Zhou et al. (2017) combined phrase-based statistical machine translation (PBMT), hi-

erarchical phrase-based statistical machine translation (HPMT) and NMT with a unified architecture, similar to the dominant NMT model. Compared with the dominant NMT model, two attention models were involved to compute the context vectors. Specifically, an attention model is utilized to calculate the context vector for each machine system, while the other attention model obtains the context vector over the all context vectors of machine systems.

In above works, the generating and refining are two separate processes. As a comparison, Xia et al. (2017) proposed deliberation network, which consists of two decoders: a first-pass decoder generates a draft, which is taken as input of second-pass decoder to obtain a better translation. All the components of deliberation network are coupled together and jointly optimized in an end-to-end way via reinforcement learning. Instead of first-pass forward decoder, Zhang et al. (2018) adopted a backward decoder to capture the right-to-left target-side contexts, which is taken as input to assist the second-pass forward decoder to obtain a better translation. Besides, the another difference with deliberation network is the second-pass decoder is integrated with the first-pass decoder without reinforcement learning.

For the purpose of exploring polishing mechanism, our model adopts adaptive multi-pass decoding strategy. Compared with the previous works which consumes no more than two decoding passes, our multi-pass decoder makes an attempt to perform the multi-pass decoding. More importantly, we adopt adaptive decoding depth controlled by policy network to extend the capacity of our multi-pass decoder.

6 Conclusion

In this paper, we propose a novel architecture called *adaptive multi-pass decoder* to adopt polishing mechanism into the NMT model via reinforcement learning. Towards this goal, a novel multi-pass decoder is introduced to generate the translation, conditioned on the source- and target-side contexts. Simultaneously, the multi-pass decoding is supervised by a policy network which learns to choose a suitable action from continuing next pass decoding or halt at each time step to maximize the BLEU of the final translation. As a result, our model has the capability of controlling the decoding depth to generate a better translation. Extensive experiments on Chinese-English translation demonstrate the effectiveness of the proposed adaptive multi-pass decoder.

In this paper, we focus on utilizing multi-pass decoder to polish the translation. Our proposed multi-pass decoder performs the multi-pass decoding mechanism with only forward decoding. One promising direction is to incorporate the backward decoding into our architecture. More specifically, we can extend the policy network to choose the backward decoding except for forward decoding and halting.

Acknowledgments

We would like to thank the anonymous reviewers for their insightful comments. We also thank Heng Gong and Shuang Chen for helpful discussion. This work was supported by the National Natural Science Foundation of China (NSFC) via grant 61632011, 61772156 and 61502120.

References

- Dzmitry Bahdanau, Kyunghyun Cho, and Yoshua Bengio. 2014. Neural machine translation by jointly learning to align and translate. *arXiv e-prints*, abs/1409.0473.
- Peter E. Brown, Stephen A. Della Pietra, Vincent J. Della Pietra, and Robert L. Mercer. 1993. The mathematics of statistical machine translation: Parameter estimation. *Computational Linguistics*, 19(2).
- Rajen Chatterjee, José G. C. de Souza, Matteo Negri, and Marco Turchi. 2016. The fbk participation in the wmt 2016 automatic post-editing shared task. In *Proceedings of the First Conference on Machine Translation: Volume 2, Shared Task Papers*, pages 745–750. Association for Computational Linguistics.
- Kyunghyun Cho, Bart van Merriënboer, Caglar Gulcehre, Dzmitry Bahdanau, Fethi Bougares, Holger Schwenk, and Yoshua Bengio. 2014. Learning phrase representations using rnn encoder–decoder for statistical machine translation. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 1724–1734. Association for Computational Linguistics.
- Xiaocheng Feng, Jiang Guo, Bing Qin, Ting Liu, and Yongjie Liu. 2017. Effective deep memory networks for distant supervised relation extraction. In *Proceedings of the Twenty-Sixth International Joint Conference on Artificial Intelligence, IJCAI*, pages 19–25.
- Xiaocheng Feng, Bing Qin, and Ting Liu. 2018. A language-independent neural network for event detection. *Science China Information Sciences*, 61(9):092106.
- Alex Graves. 2013. Generating sequences with recurrent neural networks. *arXiv preprint arXiv:1308.0850*.
- Sepp Hochreiter and Jürgen Schmidhuber. 1997. Long short-term memory. *Neural computation*, 9(8):1735–1780.
- Eric Jang, Shixiang Gu, and Ben Poole. 2016. Categorical reparameterization with gumbel-softmax. *arXiv preprint arXiv:1611.01144*.
- Marcin Junczys Dowmunt and Roman Grundkiewicz. 2017. An exploration of neural sequence-to-sequence architectures for automatic post-editing. In *Proceedings of the Eighth International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 120–129. Asian Federation of Natural Language Processing.
- Nal Kalchbrenner and Phil Blunsom. 2013. Recurrent continuous translation models. In *Proceedings of the 2013 Conference on Empirical Methods in Natural Language Processing*, pages 1700–1709. Association for Computational Linguistics.
- Philipp Koehn, Hieu Hoang, Alexandra Birch, Chris Callison-Burch, Marcello Federico, Nicola Bertoldi, Brooke Cowan, Wade Shen, Christine Moran, Richard Zens, Chris Dyer, Ondrej Bojar, Alexandra Constantin, and Evan Herbst. 2007. Moses: Open source toolkit for statistical machine translation. In *Proceedings of the 45th Annual Meeting of the Association for Computational Linguistics Companion Volume Proceedings of the Demo and Poster Sessions*, pages 177–180. Association for Computational Linguistics.
- Philipp Koehn, Franz J. Och, and Daniel Marcu. 2003. Statistical phrase-based translation. In *Proceedings of the 2003 Human Language Technology Conference of the North American Chapter of the Association for Computational Linguistics*.

- 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.
- 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*.
- Mike Schuster and Kuldip K Paliwal. 1997. Bidirectional recurrent neural networks. *IEEE Transactions on Signal Processing*, 45(11):2673–2681.
- 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.
- Richard S Sutton and Andrew G Barto. 1998. *Introduction to reinforcement learning*, volume 135. MIT press Cambridge.
- Ronald J Williams. 1992. Simple statistical gradient-following algorithms for connectionist reinforcement learning. In *Reinforcement Learning*, pages 5–32. Springer.
- 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.
- Xiangwen Zhang, Jinsong Su, Yue Qin, Yang Liu, Ronrong Ji, and Hongji Wang. 2018. Asynchronous bidirectional decoding for neural machine translation. *arXiv preprint arXiv:1801.05122*.
- Long Zhou, Wenpeng Hu, Jiajun Zhang, and Chengqing Zong. 2017. 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.