A GRU-Gated Attention Model for Neural Machine Translation[*]

Biao Zhang[1], **Deyi Xiong**[2] **and Jinsong Su**[1]

Xiamen University, Xiamen, China 361005[1]

Soochow University, Suzhou, China 215006[2]

`zb@stu.xmu.edu.cn, jssu@xmu.edu.cn`

`dyxiong@suda.edu.cn`

Abstract

Neural machine translation (NMT) heavily relies on an attention network to produce a context vector for each target word prediction. In practice, we find that context vectors for different target words are quite similar to one another and therefore are insufficient in discriminatively predicting target words. The reason for this might be that context vectors produced by the vanilla attention network are just a weighted sum of source representations that are invariant to decoder states. In this paper, we propose a novel GRU-gated attention model (GAtt) for NMT which enhances the degree of discrimination of context vectors by enabling source representations to be sensitive to the partial translation generated by the decoder. GAtt uses a gated recurrent unit (GRU) to combine two types of information: treating a source annotation vector originally produced by the bidirectional encoder as the history state while the corresponding previous decoder state as the input to the GRU. The GRU-combined information forms a new source annotation vector. In this way, we can obtain translation-sensitive source representations which are then feed into the attention network to generate discriminative context vectors. We further propose a variant that regards a source annotation vector as the current input while the previous decoder state as the history. Experiments on NIST Chinese-English translation tasks show that both GAtt-based models achieve significant improvements over the vanilla attention-based NMT. Further analyses on attention weights and context vectors demonstrate the effectiveness of GAtt in improving the discrimination power of representations and handling the challenging issue of over-translation.
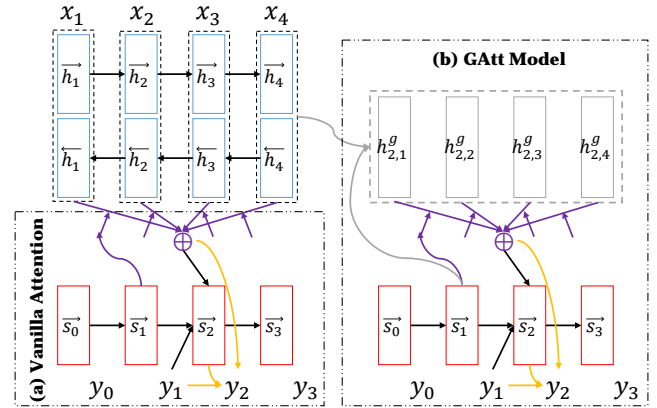
Figure 1: Illustration of the vanilla attention and the proposed GAtt. We use blue and red color to indicate the source and target side respectively. The yellow and purple color denotes the information flow for target word prediction and attention respectively, and the gray color denotes GRU-gated layer. The doted boxes show the vanilla-attention-specific and GAtt-specific components.

1 Introduction

Neural machine translation (NMT), as a large, single and end-to-end trainable neural network, has attracted wide attention in recent years [Sutskever *et al.*, 2014; Bahdanau *et al.*, 2014; Shen *et al.*, 2015; Jean *et al.*, 2015; Luong *et al.*, 2015b; Wang *et al.*, 2016]. Currently, most NMT systems use an encoder to read a source sentence into a vector and a decoder to map the vector into the corresponding target sentence. What makes NMT outperform conventional statistical machine translation (SMT) is the attention mechanism [Bahdanau *et al.*, 2014], an information bridge between the encoder and the decoder that produces context vectors by dynamically detecting relevant source words for predicting the next target word.

Intuitively, different target words would be aligned to different source words so that the generated context vectors differ significantly from one another across different decoding steps. In other words, these context vectors should be discriminative enough for target word prediction otherwise the same target words might be generated repeatedly (a well-known issue of NMT: over-translation, see Section 5.6). However, this is often not true in practice, even when "attended" source words are rather relevant. We observe that (see Section 5.5) the context vectors are very similar to each other, and that the variance in each dimension of these vec-

tors across different decoding steps is very small. These indicate that the vanilla attention mechanism suffers from its inadequacy in distinguishing different translation predictions. The reason behind, we conjecture, lies in the architecture of the attention mechanism which simply calculates a linearly weighted sum of source representations that are invariant across decoding steps. Such invariance in source representations may lead to the undesirable small variance of context vectors.

In order to handle this issue, in this paper, we propose a novel GRU-gated attention model (GAtt) for NMT. The key is that we can increase the degree of variance in context vectors by refining source representations according to the partial translation generated by the decoder. The refined source representations are composed of the original source representations and the previous decoder state at each decoding step. We show the overall framework of our model and highlight the difference between GAtt and the vanilla attention in Figure 1. GAtt significantly extends the vanilla attention by inserting a gating layer between the encoder and the vanilla attention network. Specifically, we model this gating layer with a GRU unit [Chung *et al.*, 2014], which takes the original source representations as its history and the corresponding previous decoder state as its current input. In this way, GAtt can produce translation-sensitive source representations so as to improve the variance in context vectors and therefore its discrimination ability in target word prediction.

As GRU is able to control the information flow between the history and current input through its reset and update gate, we further propose a variant of GAtt that, instead, regards the previous decoder state as the history while the original source representations as the current inputs. Both models are simple yet efficient in training and decoding.

We testify GAtt on Chinese-English translation tasks. Experimental results show that both GAtt-based models significantly outperform the vanilla attention-based NMT. We further analyze the generated attention weights and context vectors, showing that the attention weights are more accurate and the context vectors are more discriminative for target word prediction.

2 Related Work

Our work contributes to the development of attention mechanism in NMT. Originally, NMT does not have the attention mechanism and mainly relies on the encoder to summarize all source-side semantic details into a fixed-length vector [Sutskever *et al.*, 2014; Cho *et al.*, 2014]. Bahdanau et al. [2014] find that using a fixed-length vector, however, is not adequate to represent a source sentence and propose the popular attention mechanism, enabling the model to automatically search for parts of a source sentence that are relevant to the next target word. From then on, the attention mechanism has gained extensive concern. Luong et al. [2015a] explore several effective approaches to the attention network, introducing the local and global attention model. Tu et al. [2016] introduce a coverage vector to keep track of the attention history such that the attention network can pay more attention to untranslated source words. Mi et al. [2016] leverage well-trained word alignments to directly supervise the attention weights in NMT. Yang et al. [2016] bring a recurrence along the context vector to help adjust the future attention. Cohn et al. [2016] incorporate several structural bias, such as position bias, markov condition and fertilities, into the attention-based neural translation model. However, all these models mainly focus on how to make the attention weights more accurate. As we mentioned, even with well-designed attention models, context vectors may be lack of discrimination ability for target word prediction.

Another closely related work is the interactive attention model [Meng *et al.*, 2016] which treats source representations as a memory and models the interaction between the decoder and this memory during translation via reading and writing operations. To some extent, our model can also be regarded as a memory network, which only includes the reading operation. However, our reading operation differs significantly from that in the interactive attention, where we employ the GRU unit for composition while they merely use the content-based addressing. Compared with the interactive attention, our GAtt, without the writing operation, is more efficient in both training and decoding.

The gate mechanism in our GAtt is built on the GRU unit. GRU usually acts as a recurrent unit that leverages a reset gate and an update gate to control how much information flow from the history state and the current input respectively [Chung *et al.*, 2014]. It is an extension of the vanilla recurrent neural network (RNN) unit with the advantage of alleviating the vanishing and exploding gradient problems during training [Bengio *et al.*, 1994], and also a simplification of the LSTM model [Hochreiter and Schmidhuber, 1997] with the advantage of efficient computation. The idea of using GRU as a gate mechanism, to the best of our knowledge, has never been investigated before.

Additionally, our model is also related with the tree-structured LSTM [Tai *et al.*, 2015], where LSTM is adapted to compose vary-sized children nodes and current input node in a dependency tree into the current hidden state. GAtt differs significantly from the tree-structured LSTM in that the latter employs the sum operation to deal with the vary-sized representations, while our model leverages the attention mechanism.

3 Background

In this section, we briefly review the vanilla attention-based NMT [Bahdanau *et al.*, 2014]. Unlike conventional SMT, NMT directly maps a source sentence $\mathbf{x} = \{x_1, \ldots, x_n\}$ to its target translation $\mathbf{y} = \{y_1, \ldots, y_m\}$ using an encoder-decoder framework. The encoder reads the source sentence $\mathbf{x}$, and encodes the representation of each word $\mathbf{h}_i$ by summarizing the information of neighboring words. As shown by the blue color in Figure 1, this is achieved by a bidirectional RNN, specifically the bidirectional GRU model.

The decoder is a conditional language model which generates the target sentence word by word using the following conditional probability (see the yellow lines in Figure 1 (a)):

$$p(y_j|\mathbf{x}, \mathbf{y}_{<j}) = softmax\left(g(E_{y_{j-1}}, \mathbf{s}_j, \mathbf{c}_j)\right) \qquad (1)$$

where $\mathbf{y}_{<j} = \{y_1, \cdots, y_{j-1}\}$ is a partial translation, $E_{y_{j-1}} \in \mathbb{R}^{d_w}$ is the embedding of previously generated target word y_{j-1}, $\mathbf{s}_j \in \mathbb{R}^{d_h}$ is the j-th target-side decoder state and $g(\cdot)$ is a highly non-linear function. Please refer to [Bahdanau *et al.*, 2014] for more details. What we concern in this paper is $\mathbf{c}_j \in \mathbb{R}^{2d_h}$, which is the translation-sensitive context vector produced by the attention mechanism.

Attention Mechanism acts as a bridge between the encoder and the decoder, which makes them tightly coupled. The attention network aims at recognizing which source words are relevant to the next target word and giving high attention weights to these words in computing the context vector $\mathbf{c}_j$. This is based on the encoded source representations $\mathbf{H} = \{\mathbf{h}_1, \cdots, \mathbf{h}_n\}$ and the previous decoder state $\mathbf{s}_{j-1}$ (see the purple color in Figure 1 (a)). Formally,

$$\mathbf{c}_j = \text{Att}(\mathbf{H}, \mathbf{s}_{j-1}) \tag{2}$$

Att$(\cdot)$ denotes the whole process. It first computes an attention weight α_{ji} to measure the degree of relevance of a source word x_i for predicting the target word y_j via a feed-forward neural network:

$$\alpha_{ji} = \frac{\exp(e_{ji})}{\sum_k \exp(e_{jk}))} \tag{3}$$

The relevance score e_{ji} is estimated via an *alignment model* as in [Bahdanau *et al.*, 2014]: $e_{ji} = v_a^T \tanh(W_a \mathbf{s}_{j-1} + U_a \mathbf{h}_i)$. Intuitively, the higher attention weight α_{ji} is, the more important word x_i is for the next word prediction. Therefore, Att$(\cdot)$ generates $\mathbf{c}_j$ by directly weighting the source representations $\mathbf{H}$ with their corresponding attention weights $\{\alpha_{ji}\}_{i=1}^n$:

$$\mathbf{c}_j = \sum_i \alpha_{ji} \mathbf{h}_i \tag{4}$$

Although this vanilla attention model is very successful, we find that, in practice, the resulted context vectors $\{\mathbf{c}_j\}_{j=1}^m$ are very similar to one another. In other words, these context vectors are not discriminative enough. This is undesirable because it makes the decoder (Eq. (1)) hesitate in deciding which target word should be predicted. We attempt to solve this problem in the next section.

4 GRU-Gated Attention for NMT

The problem mentioned above reveals some shortcomings of the vanilla attention mechanism. Let's revisit the generation of $\mathbf{c}_j$ in Eq. (4). As different target words might be aligned to different source words, the attention weights of source words vary across different decoding steps. However, no matter how the attention weights of source words vary, the source representations $\mathbf{H}$ remain the same, i.e. they are decoding-invariant. And this invariance would limit the discrimination power of the generated context vectors.

Accordingly, we attempt to break up this invariance by refining the source representations before they are input to the vanilla attention network at each decoding step. To this end, we propose the GRU-gated attention (GAtt), which, similar to the vanilla attention, can be formulated into the following form:

$$\mathbf{c}_j = \text{GAtt}(\mathbf{H}, \mathbf{s}_{j-1}) \tag{5}$$

The gray color in Figure 1 (b) highlights the major difference between GAtt and the vanilla attention. Specifically, GAtt consists of two layers: a *gating layer* and an *attention layer*.

Gating Layer. This layer aims at refining the source representations according to the previous decoder state $\mathbf{s}_{j-1}$ so as to compute translation-relevant source representations. Formally,

$$\mathbf{H}_j^g = \text{Gate}(\mathbf{H}, \mathbf{s}_{j-1}) \tag{6}$$

The Gate$(\cdot)$ should be capable of dealing with the complex interactions between the source sentence and the partial translation, and freely controlling the semantic match and information flow between them. Instead of using conventional gating mechanism [Chung *et al.*, 2014], we directly choose the whole GRU unit to perform this task. For a source representation $\mathbf{h}_i$, GRU treats it as the history representation and refines it using the current input, i.e. the previous decoder state $\mathbf{s}_{j-1}$:

$$\begin{aligned}
\mathbf{z}_{ji} &= \sigma(W_z \mathbf{s}_{j-1} + U_z \mathbf{h}_i + b_z) \\
\mathbf{r}_{ji} &= \sigma(W_r \mathbf{s}_{j-1} + U_r \mathbf{h}_i + b_r) \\
\overline{h}_{ji} &= \tanh(W \mathbf{s}_{j-1} + U[\mathbf{r}_{ji} \odot \mathbf{h}_i] + b) \\
\mathbf{h}_{ji}^g &= (1 - \mathbf{z}_{ji}) \odot \mathbf{h}_i + \mathbf{z}_{ji} \odot \overline{h}_{ji}
\end{aligned} \tag{7}$$

where $\sigma(\cdot)$ is the sigmoid function, and $\odot$ denotes the element-wise multiplication. Intuitively, the reset gate $\mathbf{r}_{ji}$ and update gate $\mathbf{z}_{ji}$ measure the degree of the semantic match between the source sentence and partial translation. The former determines how much the original source information could be used to combine the partial translation, while the latter defines how much the original source information can be kept around. As a result, $\mathbf{h}_{ji}^g$ becomes translation-sensitive, rather than decoding-invariant, which is desired to strengthen the discrimination power of $\mathbf{c}_j$.

Attention Layer. This layer is the same as the vanilla attention mechanism:

$$\mathbf{c}_j = \text{Att}(\mathbf{H}_j^g, \mathbf{s}_{j-1}) \tag{8}$$

The Att$(\cdot)$ in Eq. (8) denotes the same procedure as that in Eq. (2). However, instead of paying attention to the original source representations $\mathbf{H}$, this layer relies on the gate-refined source representations $\mathbf{H}_j^g$. Notice that $\mathbf{H}_j^g$ is adaptive during decoding, indicated with the subscript j. Ideally, we expect $\mathbf{H}_j^g$ is decoding-specific enough such that $\mathbf{c}_j$ can vary significantly across different target words.

Notice that Gate$(\cdot)$ is not a multi-stepped RNN. It is simply a composition function, or only one-stepped RNN. Therefore, it is computationally efficient. To train our model, we employ the standard training objective, i.e. maximizing the log-likelihood of the training data, and optimize the model parameters using the standard stochastic gradient algorithm.

Model Variant We refer to the above model as *GAtt*, which regards the source representations as the history and the previous decoder state as the current input. Which information should be treated as input or history does not matter, especially for the GRU unit since GRU is able to control the information flow freely. We can also use the previous decoder state as the history and the source representations as the current input. We refer to this model as *GAtt-Inv*. Formally,

$$\mathbf{c}_j = \text{GAtt-Inv}(\mathbf{H}, \mathbf{s}_{j-1}) \tag{9}$$

Metric	System	MT05	MT02	MT03	MT04	MT06	MT08	AVG
BLEU	*Moses*	31.70	33.61	32.63	34.36	31.00	23.96	31.11
	RNNSearch	34.72	37.95	35.23	37.32	33.56	26.12	34.04
	GAtt	36.20	**39.30**$^{\Uparrow+}$	**37.22**$^{\Uparrow++}$	39.61$^{\Uparrow++}$	34.81$^{\Uparrow+}$	**27.56**$^{\Uparrow+}$	35.70
	GAtt-Inv	36.50	38.76$^{\Uparrow+}$	36.89$^{\Uparrow++}$	**40.09**$^{\Uparrow++}$	**35.46**$^{\Uparrow++}$	27.32$^{\Uparrow+}$	**35.70**
TER	*Moses*	58.24	57.81	57.22	56.11	56.90	60.33	57.67
	RNNSearch	59.58	56.48	58.47	56.55	57.72	61.68	58.18
	GAtt	57.25	**54.32**	56.13	54.15	55.90	**59.82**	56.06
	GAtt-Inv	56.67	54.66	**56.05**	**53.69**	**55.64**	59.93	**55.99**

Table 1: BLEU and TER scores on the Chinese-English translation tasks. **AVG** = average BLEU/TER score on all test sets. We highlight the best results in bold for each test set. "↑/⇑": significantly better than *Moses* ($p < 0.05/p < 0.01$); "+/++": significantly better than *GroundHog* ($p < 0.05/p < 0.01$). Higher BLEU scores and lower TER scores indicate better translation quality.

Metric	System	MT05	MT02	MT03	MT04	MT06	MT08	AVG
BLEU	*RNNSearch+GAtt*	38.13	41.13	38.95	41.17	36.75	29.28	37.46
	RNNSearch+GAtt-Inv	37.54	40.67	38.84	41.12	37.10	28.94	37.33
	GAtt+GAtt-Inv	38.88	40.78	38.72	42.08	37.54	29.59	37.94
	RNNSearch+GAtt+GAtt-Inv	**39.24**	**41.97**	**40.01**	**42.67**	**38.19**	**30.34**	**38.64**
TER	*RNNSearch+GAtt*	56.05	53.68	55.03	53.30	54.71	58.98	55.14
	RNNSearch+GAtt-Inv	55.95	53.30	55.08	53.27	54.86	59.15	55.13
	GAtt+GAtt-Inv	55.04	52.82	**54.20**	**52.24**	53.99	58.27	54.30
	RNNSearch+GAtt+GAtt-Inv	**54.74**	**52.59**	54.22	52.32	**53.76**	**57.88**	**54.15**

Table 2: BLEU and TER scores of the ensemble of different NMT systems.

with,

$$\mathbf{c}_j = \text{Att}(\mathbf{H}_j^{g\prime}, \mathbf{s}_{j-1}) \quad \mathbf{H}_j^{g\prime} = \text{Gate}(\mathbf{s}_{j-1}, \mathbf{H})$$

The major difference lies at the order of the inputs in $\text{Gate}(\cdot)$, since the inputs to GRU are directional. We verify both model variants through the following experiments.

5 Experiments

5.1 Setup

We evaluated the effectiveness of our model on Chinese-English translation tasks. Our training data consists of 1.25M sentence pairs, with 27.9M Chinese words and 34.5M English words respectively[1]. We chose the NIST 2005 dataset as the development set to perform model selection, and the NIST 2002, 2003, 2004, 2006 and 2008 datasets as our test sets. There are 878, 919, 1788, 1082 and 1664 sentences in NIST 2002, 2003, 2004, 2005, 2006, 2008 dataset respectively. We evaluated the translation quality using the case-insensitive BLEU-4 metric [Papineni *et al.*, 2002][2] and TER metric [Snover *et al.*, 2006][3]. We performed paired bootstrap sampling [Koehn, 2004] for statistical significance test using the script in *Moses*[4].

5.2 Baselines

We compared our proposed model against the following two state-of-the-art SMT and NMT systems:

---

[1]This data is a combination of LDC2002E18, LDC2003E07, LDC2003E14, Hansards portion of LDC2004T07, LDC2004T08 and LDC2005T06.

[2]https://github.com/moses-smt/mosesdecoder/blob/master/scripts/generic/multi-bleu.perl

[3]http://www.cs.umd.edu/ snover/tercom/

[4]https://github.com/moses-smt/mosesdecoder/blob/master/scripts/analysis/bootstrap-hypothesis-difference-significance.pl

- *Moses* [Koehn *et al.*, 2007]: an open source state-of-the-art phrase-based SMT system.
- *RNNSearch* [Bahdanau *et al.*, 2014]: a state-of-the-art attention-based NMT system using the vanilla attention mechanism. We further feed the information of y_{j-1} to the attention, and implemented the decoder with two GRU layers, following the suggestions in *dl4mt*[5].

For *Moses*, we trained a 4-gram language model on the target portion of training data using the SRILM[6] toolkit with modified Kneser-Ney smoothing. The word alignments were obtained with GIZA++ [Och and Ney, 2003] on the training corpora in both directions, using the "grow-diag-final-and" strategy [Koehn *et al.*, 2003]. All other parameters were kept as the default settings.

For *RNNSearch*, we limit the vocabulary of both source and target languages to be the most frequent 30K words, covering approximately 97.7% and 99.3% of the two corpora respectively. The words that do not appear in the vocabulary were mapped to a special token "*UNK*". We trained our model with the sentences of length up to 50 words in the training data. Following the settings in [Bahdanau *et al.*, 2014], we set d_w = 620, d_h = 1000. We initialized all parameters randomly according to a normal distribution ($\mu = 0, \sigma = 0.01$) except the square matrices which are initialized with random orthogonal matrices. We used the Adadelta algorithm [Zeiler, 2012] for optimization, with a batch size of 80 and gradient norm as 5. The model parameters were selected according to the maximum BLEU points on the development set. Additionally, during decoding, we used the beam-search algorithm, and set the beam size to 10.

For *GAtt*, we randomly initialized its parameters as what we do in *RNNSearch*. All the other settings are the same as *RNNSearch*. All NMT systems were trained on a GeForce

---

[5]https://github.com/nyu-dl/dl4mt-tutorial/tree/master/session3

[6]http://www.speech.sri.com/projects/srilm/download.html

Source	他 说 ， 难民 重新 融入 社会 是 临时 政府 的 工作 重点 之一 。
Reference	*he said the refugees ' re-integration into society is one of the top priorities on the interim government 's agenda .*
RNNSearch	*he said that refugee is one of the key government tasks .*
GAtt	*he said that the re - integration of the refugees was one of the key tasks of the interim government .*
GAtt-Inv	*he said that the refugees ' integration into society is one of the key tasks of the interim government .*

Table 3: Example translations generated by different systems. We use red color to highlight interesting parts.
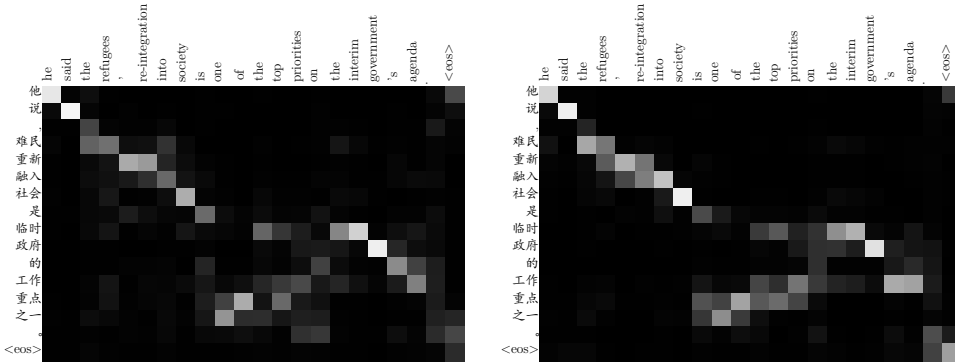

Figure 2: Visualization of the attention weights for RNNSearch (left) and GAtt (right).

GTX 1080 using the computational framework *Theano*. In one hour, the *RNNSearch* system processes about 2769 batches while *GAtt* processes 1549 batches.

5.3 Translation Results

The results are summarized in Table 1. Both GAtt and GAtt-Inv outperform both Moses and RNNSearch. Specially, GAtt yields 35.70 BLEU and 56.06 TER scores on average, with improvements of 4.59 BLEU and 1.61 TER points over Moses, and 1.66 BLEU and 2.12 TER points over RNNSearch; GAtt-Inv achieves 35.70 BLEU and 55.99 TER scores on average, with gains of 4.59 BLEU and 1.68 TER points over Moses, and 1.66 BLEU and 2.19 TER points over RNNSearch. All improvements are statistically significant.

It seems that GAtt-Inv obtains very slightly better performance than GAtt in terms of TER on average. However, these improvements are neither significant nor consistent. In other words, GAtt is as efficient as GAtt-Inv. This is reasonable, since the difference of GAtt and GAtt-Inv lies at the order of inputs to GRU, and GRU is able to control its information flow from each input through its reset and update gate.

5.4 Effects of Model Ensemble

We further testify whether the ensemble of our models and RNNSearch can generate better performance against any single system. We ensemble different systems by simply averaging their predicted target word probabilities at each decoding step, as suggested in [Luong *et al.*, 2015b]. We show the results in Table 2. Not surprisingly, all the ensemble systems achieves significant improvements over the best single system. And the ensemble of "*RNNSearch+GAtt+GAtt-Inv*" produces the best results, 38.64 BLEU and 54.15 TER scores on average. This demonstrates that these neural models are complementary and beneficial to each other.

System	SAER	AER
Tu et al. [2016]	64.25	50.50
RNNSearch	64.10	50.83
GAtt	**56.19**	**43.53**
GAtt-Inv	58.29	43.60

Table 4: SAER and AER scores of word alignments deduced by different neural systems. The lower the score, the better the alignment quality.

5.5 Translation Analysis

In order to have a deep understanding of how the proposed models work, we dug into the translated sentences of different neural systems. Table 3 shows an example. All the neural models generate very fluent translations. However, RNNSearch only translates the rough meaning of the source sentence, ignoring important sub-phrases " 重新 融入 社会" and " 临时". These missing translations resonate with the finding of Tu et al. [2016]. sIn contrast, GAtt and GAtt-Inv are able to capture these two sub-phrases, generating the key translations "*integration*" and "*interim*".

To find the underlying reason, we investigated their generated attention weights. Rather than using the generated target sentences, we feed the same reference translations into RNNSearch and GAtt for making a fair comparison[7]. Figure 2 visualizes the attention weights. Both RNNSearch and GAtt have very intuitive attentions, e.g. "*refugees*" is aligned to " 难民", "*government*" is aligned to " 政府". However, compared against those of RNNSearch, the attentions learned by GAtt are more focused and accurate. In other words, the refined source representations in GAtt help the attention mechanism concentrate its weights on translation-related words.

To verify this point, we evaluated the quality of word alignments induced from different neural systems in terms of

---

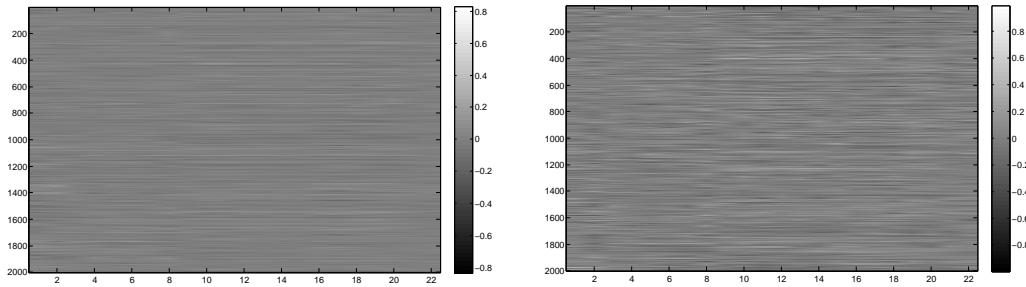[7]We do not analyze GAtt-Inv because it is very similar to GAtt.

Figure 3: Visualization of context vectors for RNNSearch (left) and GAtt (right) generated as in Figure 2. The horizontal and vertical axis denotes the translation and context dimension respectively.

alignment error rate (AER) [Och and Ney, 2003] and the soft version (SAER) of AER, following Tu et al. [Tu *et al.*, 2016].[8] Table 4 display the evaluation results of word alignments. We find that both GAtt and GAtt-Inv significantly outperform RNNSearch in terms of both AER and SAER. Specifically, GAtt obtains a gain of 7.91 SAER and 7.3 AER points over RNNSearch. As we obtain word alignments by connecting target words to source words with the highest alignment probabilities computed according to their attention weights, the consistent improvements of our model over RNNSearch on AER score indicate that our model indeed learns more accurate attentions.

Another very important question is whether GAtt enhances the discrimination of the context vectors. We answer this question by visualizing these vectors, as shown in Figure 3. We can observe that the heatmap of RNNSearch is very smooth, which varies very slightly across different decoding steps (the horizontal axis). This means that these context vectors are very similar to one another, thus lacking of discrimination. In contrast, there are obvious variations in GAtt. Statistically, the mean variance of the context vectors across different dimensions in RNNSearch is 0.0057, while it is 0.0365 in GAtt, 6 times larger than that of RNNSearch. Additionally, across different decoding steps, the mean variance is 0.0088 in RNNSearch, while it is 0.0465 in GAtt. All these strongly suggest that our model makes the context vectors more discriminative across different target words.

5.6 Over-Translation Evaluation

Over-translation or repeatedly predicting the same target words [Tu *et al.*, 2016] is a challenging problem for NMT. We conjecture that the reason behind the over-translation issue is partially due to the small differences in context vectors learned by the vanilla attention mechanism. As the proposed GAtt can improve the discrimination power of context vectors, we hypothesize that our model can deal better with the over-translation issue than the vanilla attention network. To testify this hypothesis, we introduce a metric called *N-Gram Repetition Rate* (N-GRR) that calculates the portion of re-

System	1-gram	2-gram	3-gram	4-gram
Reference	12.94	1.80	0.93	1.29
RNNSearch	19.12	5.26	3.27	2.97
GAtt	18.09	4.11	2.50	2.46
GAtt-Inv	16.79	3.39	1.99	1.94

Table 5: N-GRR scores of different systems on all test sets with N ranges from 1 to 4. The lower the score, the better the system deals with the over-translation problem.

peated n-grams in a sentence:

$$\text{N-GRR} = \frac{1}{CR} \sum_{c=1}^{C} \sum_{r=1}^{R} \frac{|\text{N-grams}_{c,r}| - |u(\text{N-grams}_{c,r})|}{|\text{N-grams}_{c,r}|} \tag{10}$$

where $|\text{N-grams}_{c,r}|$ denotes the number of total n-grams in the r-th translation of the c-th sentence in the testing corpus and $u(\text{N-grams}_{c,r})$ the number of n-grams after duplicate n-grams are removed. In our test sets, there are $C = 6606$ sentences with $r = 4$ and $r = 1$ translations for the *Reference* and *NMT systems* respectively. If we compare N-GRR scores of machine-generated translation against those of reference translations, we can roughly know how serious the over-translation problem is.

We show N-GRR results in Table 5. Compared with reference translations (Reference), RNNSearch yields significant high scores, indicating that RNNSearch generates redundant repeated n-grams in translations, and therefore the over-translation problem in RNNSearch is serious. In contrast, both GAtt and GAtt-Inv achieve considerable improvements over RNNSearch in terms of N-GRR. Especially, we find that GAtt-Inv performs better than GAtt on all n-grams, which is in accordance with the translation results in Table 1. These N-GRR results strongly suggest that the proposed models are able to handle the over-translation issue and that generating more discriminative context vectors makes NMT suffer less from the over-translation issue.

6 Conclusion

In this paper, we have presented a novel GRU-gated attention model (GAtt) for NMT. Instead of using decoding-invariant source representations, GAtt produces new source representations that vary across different decoding steps according to the partial translation so as to improve the discrimination of context vectors for translation. This is achieved by a gating layer that regards the source representations and previous decoder state as the history and input to a gated recurrent unit. Experiments on Chinese-English translation tasks

---

[8]Notice that we used the same dataset and evaluation script as Tu et al. [Tu *et al.*, 2016]. We refer the readers to [Tu *et al.*, 2016] for more details.

demonstrate the effectiveness of our model. In-depth analysis further reveals that our model is able to significantly reduce repeated redundant translations (over-translations).

In the future, we would like to apply our model to other sequence learning tasks as our model is easily to be adapted to any other sequence-to-sequence tasks (e.g. document summarization, neural conversion model, speech recognition, .etc). Additionally, except for the GRU unit, we will explore more different end-to-end neural architectures, such as convolutional neural network, LSTM unit as the gate mechanism plays a very important role in our model. Finally, we are interested in adapting our GAtt model as a tree-structured unit to compose different nodes in a dependency tree.

References

[Bahdanau *et al.*, 2014] Dzmitry Bahdanau, Kyunghyun Cho, and Yoshua Bengio. Neural machine translation by jointly learning to align and translate. In *Proc. of ICLR*, 2014.

[Bengio *et al.*, 1994] Yoshua Bengio, Patrice Simard, and Paolo Frasconi. Learning long-term dependencies with gradient descent is difficult. *IEEE Transactions on Neural Networks*, 5:157–166, 1994.

[Cho *et al.*, 2014] Kyunghyun Cho, Bart van Merrienboer, Caglar Gulcehre, Dzmitry Bahdanau, Fethi Bougares, Holger Schwenk, and Yoshua Bengio. Learning phrase representations using rnn encoder–decoder for statistical machine translation. In *Proc. of EMNLP*, pages 1724–1734, Doha, Qatar, October 2014.

[Chung *et al.*, 2014] Junyoung Chung, Çaglar Gülçehre, KyungHyun Cho, and Yoshua Bengio. Empirical evaluation of gated recurrent neural networks on sequence modeling. *CoRR*, 2014.

[Cohn *et al.*, 2016] Trevor Cohn, Cong Duy Vu Hoang, Ekaterina Vymolova, Kaisheng Yao, Chris Dyer, and Gholamreza Haffari. Incorporating structural alignment biases into an attentional neural translation model. *CoRR*, abs/1601.01085, 2016.

[Hochreiter and Schmidhuber, 1997] Sepp Hochreiter and Jürgen Schmidhuber. Long short-term memory. *Neural Comput.*, 9:1735–1780, 1997.

[Jean *et al.*, 2015] Sébastien Jean, Kyunghyun Cho, Roland Memisevic, and Yoshua Bengio. On using very large target vocabulary for neural machine translation. In *Proc. of ACL-IJCNLP*, pages 1–10, July 2015.

[Koehn *et al.*, 2003] Philipp Koehn, Franz Josef Och, and Daniel Marcu. Statistical phrase-based translation. In *Proc. of NAACL*, NAACL '03, pages 48–54, 2003.

[Koehn *et al.*, 2007] Philipp Koehn, Hieu Hoang, Alexandra Birch, Chris Callison-Burch, Marcello Federico, Nicola Bertoldi, Brooke Cowan, Wade Shen, Christine Moran, Richard Zens, Chris Dyer, Ondřej Bojar, Alexandra Constantin, and Evan Herbst. Moses: Open source toolkit for statistical machine translation. In *Proc. of ACL*, pages 177–180, 2007.

[Koehn, 2004] Philipp Koehn. Statistical significance tests for machine translation evaluation. In *Proc. of EMNLP*, 2004.

[Luong *et al.*, 2015a] Thang Luong, Hieu Pham, and Christopher D. Manning. Effective approaches to attention-based neural machine translation. In *Proc. of EMNLP*, pages 1412–1421, September 2015.

[Luong *et al.*, 2015b] Thang Luong, Ilya Sutskever, Quoc Le, Oriol Vinyals, and Wojciech Zaremba. Addressing the rare word problem in neural machine translation. In *Proc. of ACL-IJCNLP*, pages 11–19, July 2015.

[Meng *et al.*, 2016] Fandong Meng, Zhengdong Lu, Hang Li, and Qun Liu. Interactive attention for neural machine

translation. In *Proc. of COLING*, pages 2174–2185, Osaka, Japan, December 2016.

[Mi *et al.*, 2016] Haitao Mi, Zhiguo Wang, and Abe Ittycheriah. Supervised attentions for neural machine translation. In *Proc. of EMNLP*, pages 2283–2288, Austin, Texas, November 2016.

[Och and Ney, 2003] Franz Josef Och and Hermann Ney. A systematic comparison of various statistical alignment models. *Comput. Linguist.*, 29(1):19–51, March 2003.

[Papineni *et al.*, 2002] Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu. Bleu: a method for automatic evaluation of machine translation. In *Proc. of ACL*, pages 311–318, 2002.

[Shen *et al.*, 2015] Shiqi Shen, Yong Cheng, Zhongjun He, Wei He, Hua Wu, Maosong Sun, and Yang Liu. Minimum risk training for neural machine translation. *CoRR*, abs/1512.02433, 2015.

[Snover *et al.*, 2006] Matthew Snover, Bonnie Dorr, Richard Schwartz, Linnea Micciulla, and John Makhoul. A study of translation edit rate with targeted human annotation. In *In Proceedings of Association for Machine Translation in the Americas*, pages 223–231, 2006.

[Sutskever *et al.*, 2014] Ilya Sutskever, Oriol Vinyals, and Quoc V. Le. Sequence to sequence learning with neural networks. *CoRR*, abs/1409.3215, 2014.

[Tai *et al.*, 2015] Kai Sheng Tai, Richard Socher, and Christopher D. Manning. Improved semantic representations from tree-structured long short-term memory networks. In *Proc. of ACL*, pages 1556–1566, Beijing, China, July 2015.

[Tu *et al.*, 2016] Zhaopeng Tu, Zhengdong Lu, Yang Liu, Xiaohua Liu, and Hang Li. Coverage-based neural machine translation. *CoRR*, abs/1601.04811, 2016.

[Wang *et al.*, 2016] Xing Wang, Zhengdong Lu, Zhaopeng Tu, Hang Li, Deyi Xiong, and Min Zhang. Neural machine translation advised by statistical machine translation. *CoRR*, abs/1610.05150, 2016.

[Yang *et al.*, 2016] Z. Yang, Z. Hu, Y. Deng, C. Dyer, and A. Smola. Neural Machine Translation with Recurrent Attention Modeling. *ArXiv e-prints*, July 2016.

[Zeiler, 2012] Matthew D. Zeiler. ADADELTA: an adaptive learning rate method. *CoRR*, abs/1212.5701, 2012.