Alleviating the Inequality of Attention Heads for Neural Machine Translation

Anonymous ACL submission

Abstract

Recent studies show that the attention heads in Transformer are not equal (Voita et al., 2019; Michel et al., 2019). We relate this phenomenon to the imbalance training of multi-head attention and the model dependence on specific heads. To tackle this problem, we propose a simple masking method: *HeadMask*, in two specific ways. Experiments show that translation improvements are achieved on multiple language pairs. Subsequent empirical analyses also support our assumption and confirm the effectiveness of the method.

1 Introduction

Recently, more and more novel network structures of neural machine translation(NMT) have been proposed (Bahdanau et al., 2015; Barone et al., 2017; Gehring et al., 2017; Vaswani et al., 2017), among which Transformer (Vaswani et al., 2017) achieves the best results. One important difference between Transformer and other translation models is its multi-head attention mechanism.

Some interesting phenomena of the attention heads are discovered recently. Voita et al. (2019) find that only a small subset of heads appear to be important for the translation task and vast majority of heads can be removed without seriously affecting performance. Michel et al. (2019) also find that several heads can be removed from trained transformer models without statistically significant degradation in test performance. It turns out that not all heads are equally important.

We speculate that this can be attributed to the imbalanced training of multi-head attention, as some heads are not trained adequately and contribute little to the model. However, this can be turned into the bottleneck for the whole model. For an analogy, if a soccer player gets used to using the right foot and spares more training opportunities for it, it will be stronger and stronger. As a result, the right foot is further relied on, while the left foot receives less training and gradually turns into the limitation.

In this paper, we firstly empirically confirm the inequality in multi-head attention. Then a new training method with two variants is proposed to avoid the bottleneck and improve the translation performance. Further analyses are also made to verify the assumption.

2 Head Inequality

Following Michel et al. (2019), we define the importance of an attention head h as

$$I_h = \mathbb{E}_{x\sim X}\left|\frac{\partial \mathcal{L}(x)}{\partial \xi_h}\right| \tag{1}$$

where $\mathcal{L}(x)$ is the loss on sample x and ξ is the head mask variable with values in $\{0, 1\}$. Intuitively, if $head_h$ is important, switching ξ_h will have a significant effect on the loss. Applying the chain rule yields the final expression for I_h:

$$I_h = \mathbb{E}_{x\sim X}\left|\mathrm{Att}_h(x)^T\frac{\partial \mathcal{L}(x)}{\partial \mathrm{Att}_h(x)}\right| \tag{2}$$

This is equivalent to the Taylor expansion method from Molchanov et al. (2017). In Transformer base (Vaswani et al., 2017), there are 3 types of attention (encoder self attention, decoder self attention, encoder-decoder attention) with 6 layers per type and 8 heads per layer. Therefore, it amounts to 144 heads. We divide them into 8 groups with 18 heads (12.5%) each group according to their importance I_h, among which, 1-18 are the most important and so on.

We then mask different groups of the heads. As is shown in Figure 1, masking a group of unimportant heads has little effect on the translation quality while masking important heads leads to a significant drop of performance. Surprisingly, almost half of the heads are not important, as it makes almost no difference whether they are masked or not.

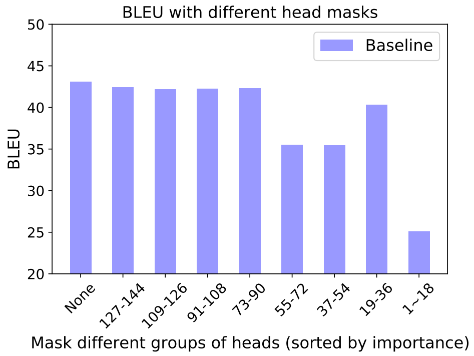


Figure 1: Mask the heads in the same group. Important ones matter much more than unimportant ones.

We also gradually masking more heads group by group in the ascending order and descending order, respectively. As is shown in Figure 2, the line starting with unimportant heads drops much slower than the one starting with important ones. It fully illustrates the inequality of different heads.

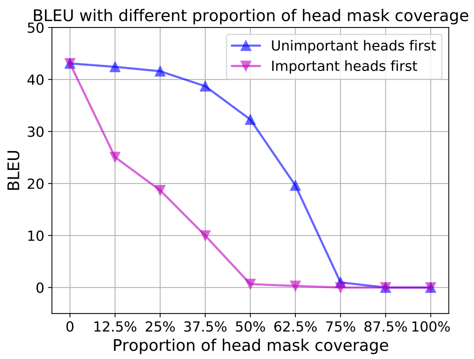


Figure 2: Mask all heads in the ascending order and descending order. The drop curves differ greatly.

Figure 1 and Figure 2 further demonstrates the inequality of the importance of attention heads. A simple assumption for explanation is that some heads coincidentally get more updating opportunities in the early stage, which makes the model learning to depend on them gradually. As a result, the model increasingly draws a strong connection with these specific heads while this local dependence prevents the rest attention heads from adequate training and restricts the overall capacity.

3 HeadMask

Since the problem refers to the unfair training of attention heads, it is natural for us to explicitly balance the training chances. We propose a simple method: *HeadMask*, which masks certain heads during training in two specific ways.

3.1 Mask Randomly

The first one is randomly picking heads and masking them in each batch. It ensures every head gets relatively equal opportunities of training and avoid partial dependence, as is shown in Algorithm 1. For the soccer analogy, it is like training the feet randomly, making both receive the same amount of practice.

Algorithm 1 HeadMask: Mask Randomly

Input: q, k, v for attention, number of masks n
Output: masked context

```
for batch in datasets do
    heads = random.sample(all_heads, n)
    for head in heads do
        ξ_head = 0
    end for
    context = attn(ξ)
end for
```

3.2 Mask Important Ones

The second one is masking the most important heads. By forcing the model neglects important heads, we hope more training chances are assigned to weaker heads. For the soccer analogy, it means training the left foot more if the right foot dominates. And once reversed, train contrarily. Its main idea is about suppressing addicted training. Specifically, the network firstly proceeds feed-forward calculation and back propagation without updating parameters to yield the importance of heads. And after picking the most important heads by sorting, mask them. During training, we only use the rest part of networks to reach the final loss and update parameters, as is shown in algorithm 2.

Algorithm 2 HeadMask: Mask Important Ones

Input: q, k, v for attention, number of masks n
Output: masked context

```
for batch in datasets do
    calculate L by feed-forward
    back propagation without updating params
    calculate importance of all heads I
    heads = argmax_n(I)
    for head in heads do
        ξ_head = 0
    end for
    context = attn(ξ)
    calculate L by feed-forward
    back propagation and update params
end for
```

4 Experiments

4.1 Datasets and Systems

We conduct experiments on four datasets, including three low-resource ones (less than 1 million). We use BPE (Sennrich et al., 2016) for Zh-En and Ro-En, adopt the preprocessed versions from Luong and Manning (2015) as well as the settings of Huang et al. (2017) for Vi-En, and follow the joint-BPE settings of Sennrich et al. (2017) for Tr-EN. More information is in Table 1.

Datasets	Scale	Dev	Test
NIST Zh-En	1.34M	MT03	MT04/05/06
WMT16 Ro-En	608K	newstest2015	newstest2016
IWSLT15 Vi-En	133K	tst2012	tst2013
WMT17 Tr-En	207K	newstest2016	newstest2017

Table 1: The information of our datasets

We follow Transformer base setting (Vaswani et al., 2017). Parameters are optimized by Adam (Kingma and Ba, 2015), with $\beta_1 = 0.9$, $\beta_2 = 0.98$, and $\epsilon = 10^{-9}$. The learning rate is scheduled according to the method proposed in Vaswani et al. (2017), with $warmup\_steps = 4000$. Label smoothing (Szegedy et al., 2016) of value=0.1 and dropout (Srivastava et al., 2014) of value=0.1 are also adopted.

Comparison We compare the baseline with masking randomly (Random-N) and masking important ones (Impt-N), where N is the mask number. In this paper, we mainly employ $N = 18(12.5\%)$.

4.2 Results

As is shown in Table 2,3,4, except for Vi-En experiments, Impt-18 yields enhancement over all language directions and reach the best result on the experiment of Ro $\rightarrow$ En. And Random-18 obtains steady improvements over all pairs and is obviously better than Impt-18. It seems the aggressive masking strategy at important heads can be too harsh and reversely restrict the model. And the random method is more expert in building a rational training pattern. In conclusion, reducing the unbalanced training among attention heads can effectively improve the translation quality.

Test sets	MT04	MT05	MT06
Baseline	46.62	43.46	43.09
Impt-18	46.94 (+0.28)	44.19 (+0.73)	43.16 (+0.07)
Random-18	**47.04 (+0.42)**	**44.33 (+0.87)**	**43.88 (+0.79)**

Table 2: Results on Experiments of Zh $\rightarrow$ En

Directions	Ro $\rightarrow$ En	Vi $\rightarrow$ En	Tr $\rightarrow$ En
Baseline	32.17	26.49	17.29
Impt-18	**32.95 (+0.78)**	26.36 (-0.13)	17.48 (+0.19)
Random-18	32.85 (+0.68)	**26.85 (+0.36)**	**17.56 (+0.27)**

Table 3: Results on Experiments of Ro/Vi/Tr $\rightarrow$ En

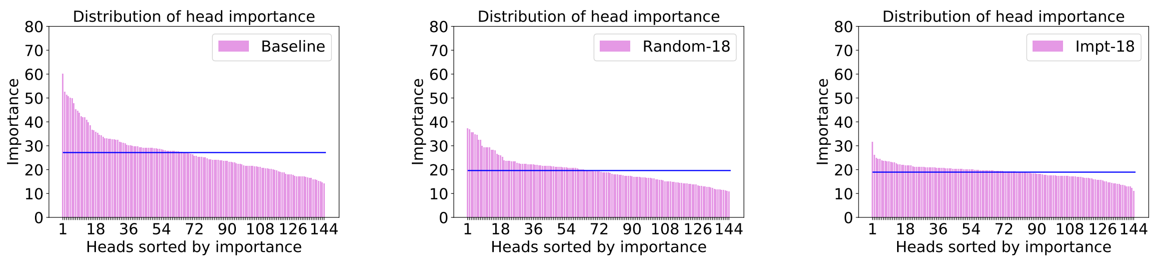
Directions	En $\rightarrow$ Ro	En $\rightarrow$ Vi	En $\rightarrow$ Tr
Baseline	31.98	28.07	15.74
Impt-18	32.47 (+0.49)	28.06 (-0.01)	16.10 (+0.36)
Random-18	**32.64 (+0.66)**	**28.46 (+0.39)**	**16.16 (+0.42)**

Table 4: Results on Experiments of En $\rightarrow$ Ro/Vi/Tr

4.3 Statistical Analysis

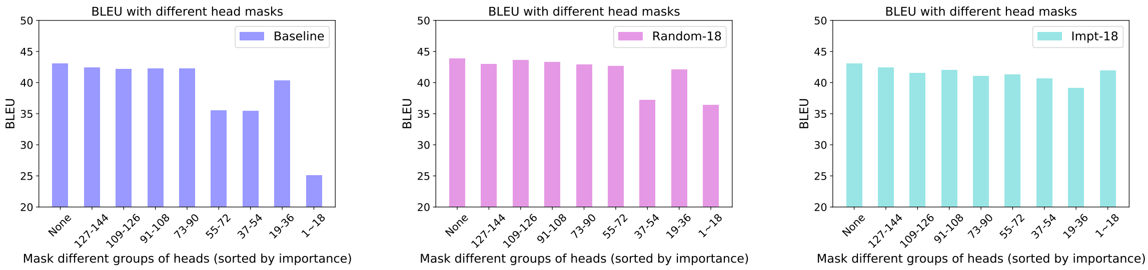
4.3.1 Flatter Distribution

To evaluate the adjusted training of heads, we check the distribution of head importance. As is shown in Figure 3, our methods make the importance distribution flatter. And the overall variance and mean are also calculated, as is shown in Table 5,6. Compared with Baseline, Impt-18 and Random-18 significantly reduce the variance of attention heads, achieving the goal of more equal training. And the mean also decreases, which proves the decline of dependence on every individual head. More specifically, Impt-18 can better resolve the imbalance, for it well prevent the emergence of "super" heads.



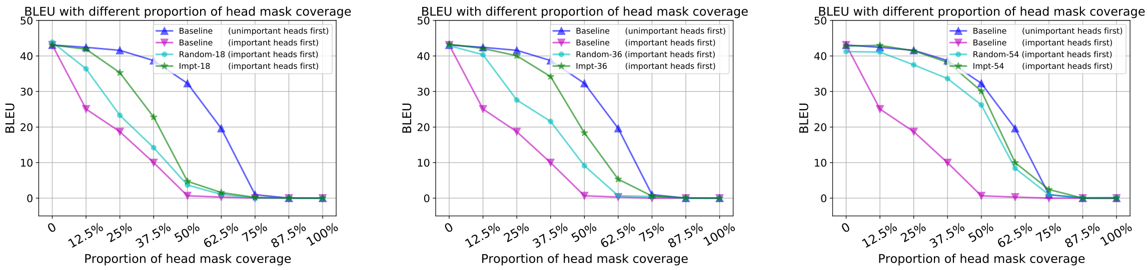
(a) Baseline (b) Random-18 (c) Impt-18

Figure 3: Distribution of importance of attention heads. Our methods make the whole distribution much flatter.



(a) Baseline (b) Random-18 (c) Impt-18

Figure 4: Our methods significantly maintain the performance even if the important heads are masked.



(a) Mask 18 heads in training (b) Mask 36 heads in training (c) Mask 54 heads in training

Figure 5: As the number of masked heads grows, the drop curves starting with important heads are moving up.

Directions	Zh2En	Ro2En	Vi2En	Tr2En
Baseline	77.28	552.93	100.73	1767.70
Random-18	33.21	255.98	48.28	900.70
Impt-18	9.13	72.73	14.13	188.87

Table 5: Our methods greatly reduce the **Variance** of the head importance, illustrating the improved equality of heads.

Directions	Zh2En	Ro2En	Vi2En	Tr2En
Baseline	27.15	47.18	17.96	83.79
Random-18	19.62	39.96	14.86	74.05
Impt-18	18.95	37.30	18.96	85.12

Table 6: Our methods reduce the **Mean** of the head importance, illustrating the lessened dependence on each head.

4.3.2 Weaker Dependence

We repeat the experiments of masking different groups of heads. As is shown in Figure 4, the translation quality is still maintained even if important heads are masked, proving the dependence on them has decreased. And Impt-18 performs more steadily since it is accustomed to such situations.

4.3.3 More Robust Models

We also repeat the experiments of masking all heads, as is shown in Figure 5. The two middle lines originally lie in the same place as the bottom one. As the number of masked heads in training (N) grows, they gradually move up and approach the top line where unimportant heads are masked first. It shows our methods make the model rely less on the important heads and become more robust.

5 Related Works

Recently, many analytical works about multi-head attention come out (Raganato and Tiedemann, 2018; Tang et al., 2018; Voita et al., 2019; Michel et al., 2019; Sun et al., 2020; Behnke and Heafield, 2020). And for the inequality of the networks, some studies focus on the model level (Frankle and Carbin, 2019), the layer level (Zhang et al., 2019), and the neuron level (Bau et al., 2019). For the mask algorithm, there are also works on the layer level (Fan et al., 2020), the word level (Provilkov et al., 2019), and the neuron level (Srivastava et al., 2014). Different from them, we mainly study the attention level and conduct a statistical analysis.

6 Conclusion

In this paper, we empirically validate the inequality of attention heads in Transformer and come up with an assumption of imbalanced training. Correspondingly, we propose a specific method in two ways to resolve the issue. Experiments show the improvements on multiple language pairs. And detailed analysis shows the alleviation of the problem and the effectiveness of our techniques.

References

Dzmitry Bahdanau, Kyunghyun Cho, and Yoshua Bengio. 2015. Neural machine translation by jointly learning to align and translate. In *ICLR*.

Antonio Valerio Miceli Barone, Jindrich Helcl, Rico Sennrich, Barry Haddow, and Alexandra Birch. 2017. Deep architectures for neural machine translation. In *WMT*.

Anthony Bau, Yonatan Belinkov, Hassan Sajjad, Nadir Durrani, Fahim Dalvi, and James Glass. 2019. Identifying and controlling important neurons in neural machine translation. In *ICLR*.

Maximiliana Behnke and Kenneth Heafield. 2020. Losing heads in the lottery: Pruning transformer attention in neural machine translation. In *EMNLP*.

Angela Fan, Edouard Grave, and Armand Joulin. 2020. Reducing transformer depth on demand with structured dropout. In *ICLR*.

Jonathan Frankle and Michael Carbin. 2019. The lottery ticket hypothesis: Finding sparse, trainable neural networks. In *ICLR*.

Jonas Gehring, Michael Auli, David Grangier, Denis Yarats, and Yann Dauphin. 2017. Convolutional sequence to sequence learning. In *ICML*.

Po-Sen Huang, Chong Wang, Dengyong Zhou, and Li Deng. 2017. Neural phrase-based machine translation. *arXiv*, abs/1706.05565.

Diederick P Kingma and Jimmy Ba. 2015. Adam: A method for stochastic optimization. In *ICLR*.

Minh-Thang Luong and Christopher D Manning. 2015. Stanford neural machine translation systems for spoken language domains. In *IWSLT*.

Paul Michel, Omer Levy, and Graham Neubig. 2019. Are sixteen heads really better than one? In *NeurIPS*.

Pavlo Molchanov, Stephen Tyree, Tero Karras, Timo Aila, and Jan Kautz. 2017. Pruning convolutional neural networks for resource efficient inference. In *ICLR*.

Ivan Provilkov, Dmitrii Emelianenko, and Elena Voita. 2019. Bpe-dropout: Simple and effective subword regularization. *arXiv*, abs/1910.13267.

Alessandro Raganato and Jörg Tiedemann. 2018. An analysis of encoder representations in transformer-based machine translation. In *BlackboxNLP@EMNLP*.

Rico Sennrich, Alexandra Birch, Anna Currey, Ulrich Germann, Barry Haddow, Kenneth Heafield, Antonio Valerio Miceli Barone, and Philip Williams. 2017. The university of edinburgh's neural mt systems for wmt17. In *WMT*.

Rico Sennrich, Barry Haddow, and Alexandra Birch. 2016. Neural machine translation of rare words with subword units. In *ACL*.

Nitish Srivastava, Geoffrey E. Hinton, Alex Krizhevsky, Ilya Sutskever, and Ruslan Salakhutdinov. 2014. Dropout: a simple way to prevent neural networks from overfitting. *JMLR*, 15(1):1929–1958.

Zewei Sun, Shujian Huang, Hao-Ran Wei, Xin-yu Dai, and Jiajun Chen. 2020. Generating diverse translation by manipulating multi-head attention. In *AAAI*.

Christian Szegedy, Vincent Vanhoucke, Sergey Ioffe, Jonathon Shlens, and Zbigniew Wojna. 2016. Rethinking the inception architecture for computer vision. In *CVPR*.

Gongbo Tang, Rico Sennrich, and Joakim Nivre. 2018. An analysis of attention mechanisms: The case of word sense disambiguation in neural machine translation. In *WMT*.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Lukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. In *NIPS*.

Elena Voita, David Talbot, F. Moiseev, Rico Sennrich, and Ivan Titov. 2019. Analyzing multi-head self-attention: Specialized heads do the heavy lifting, the rest can be pruned. In *ACL*.

Biao Zhang, Ivan Titov, and Rico Sennrich. 2019. Improving deep transformer with depth-scaled initialization and merged attention. In *EMNLP-IJCNLP*.