Multilevel Attention

Tingfu Zhou[1], Yujie Li[1*]

[1] University of Michigan, Michigan, Ann Arbor, USA
Email: tingfu@umich.edu, yujievli@umich.edu

***Abstract*—Attention mechanism has been widely used in the large language model (LLM). However, whether existing LLMs really have reasoning ability is questionable. Since even the state-of-art LLM still will make simple mistakes. We believe the key to getting real reasoning ability is to circularly use key and value to form a real reasoning link. Based on this, we introduce the idea of reasoning link and propose multilevel attention. Several experiments show that our multilevel attention can effectively improve the divergent thinking ability of the transformer based model.**

I. Introduction

Attention mechanism is largely inspired by human attention cognitive function [1], [2]. When a person observes an object, he or she will focus on a particular part of the picture that he or she is interested in. For example, when people watch the painting Mona Lisa, they will usually focus on Mona Lisa herself instead of the background environment. This allows people to extract high value information from massive information with limited processing resource. Due to this advantage, attention mechanism is imported in deep learning [2].

In deep learning, attention mechanism can match query and key, and output the best matching value [3]. This mechanism has been widely used in the mainstream large language model (LLM). But there is always a double of whether the exiting attention based models really have reasoning ability [4], or they just mechanically follow reasoning enhance strategies, such as prompt engineering. Moreover, Wei et al.[5] shows that with the increase of scale large language model, LLM will emerge some ability that is not present in small scale of LLM, which leads to the booming computing resource requirement. And there is still a long way to achieve a clear interpretability of this emergence [4]. Even when using the state-of-art large language model such as GPT-4, its ability to compute the answers in areas like complex reasoning, physics, and mathematics is still limited [6].

Existing attention mechanisms can be divided into four criteria [7]: the softness of attention, input representations, forms of input feature, and output representations. However, the core idea of attention mechanism is unchanged. That is, using query and key to calculate the attention distribution. Combine the attention distribution with value to get the weighted value. Then compute the context vector using the weighted value. Here are several improvement directions of attention mechanism [8]: sparse attention, low-rank self-attention, linearized attention, prototype, and memory compression, attention with prior. These improvement directions focus on improving existing attention mechanism that does not change the original core idea mentioned above.

Inspired by the idea of attention mechanism and multilayer perceptron, we assume the key to achieving the real reasoning ability is to recurrently use key and value to form a real reasoning link. Based on this new attention mechanism, we introduce multilevel attention. It is worth noting that multilevel attention is different from recurrent neural network (RNN). In RNN, the hidden state is used in recurrence. In multilevel attention, we recurrently connect key and value to form the reasoning link.

The main contributions to this work are as follows:

- We introduce the idea of reasoning link and propose the multilevel attention mechanism.
- Several experiments are done to analyze the effect of multilevel attention.

The remainder of this article is organized as follows: Section II briefly introduces the related work. Section III introduces the proposed Multilevel attention mechanism and its structure. Section V shows experiments and analysis of Multilevel attention. Section VI is the conclusion and future work.

II. Related Work

Attention mechanism was first applied in machine translation by Bahdanau et al. [3]. Since then, there are various kinds of attention mechanisms have been proposed [9], [7], [10]. Although the principle of attention mechanism is the same [7], there are some attention models try to improve the capturing relation ability (between feature vector) of attention mechanism. In the aspect of input representations, here are co-attention, hierarchical attention, and self-attention. Co-attention introduced by[11] can use two attention modules to process multiple input, which allow co-attention to process different kinds of information and exploit variously. Hierarchical attention [12] is composed of word-level and sentence-level. These two attention mechanism will allow model aggregate important information into different level. Self-attention focus on the input data itself, that is, query, key and value are different representations of the same input data. Another input representations attention mechanism is multi-hop attention, also known as multi-step attention, [13], [14]

[*] Corresponding author.

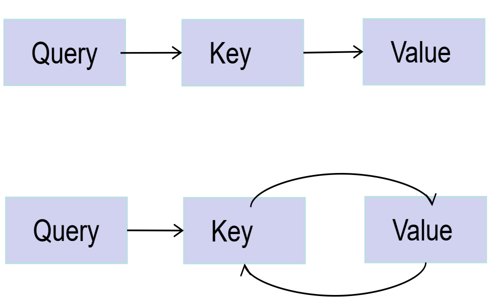

Fig. 1: Traditional attention and Multilevel attention

use multiple round of attention, which iteratively transform to extract different information in each step.

In terms of using output representation to improve capturing relation ability, there are multi-head, multi-dimension, multi-query and grouped-query attention. Multi-Head attention [15] use multiple attention modules in parallel to utilizing different version of the same query. Multi-dimensional attention [16] replaces weight scores vectors with a matrix to calculates a feature-wise score vector for keys. Multi-query attention [17] allocates a unified set of key and value heads to all query heads. While grouped-query [18] is a mixture of multi-head and multi-query. It assigns a singular set of key and value heads to specific groups of query heads.

There are also some popular modifications directions to improve capturing relation ability. For example, bidirectional attention mechanism [19] allows a model to consider both preceding and following context information when processing sequence data. However, our multilevel attention focus on principle of attention mechanism. Thus, we will not explain them in detail.

III. Multilevel Attention

A. Principle of Multilevel Attention

In classic attention based algorithm, like transformer [15], the attention distribution calculated by attention function will be put into a position-wise feed-forward network. During this process, the serial information has been aggregated, and feed-forward network use these aggregated information to map a semantic space. But we assume the serial information extracted by existing attention mechanism does not fully represent the true relationship or process of object in the reasoning process. Because the serial information that existing attention mechanism extracted is the map of query and a set of key-value pair. But we believe reasoning ability need an inference chain. That is, an inference chain from query to key-value pair, and then from former key-value pair to other key-value pair. And the final outcome is computed from the inference chain. Using this core idea, we propose the multilevel attention mechanism. Multilevel attention is different with prompt engineering, for example CoT prompting [20], process optimization [21], [22], or external knowledge enhance [23], [24]. These methods do no change the attention mechanism they used, while our multilevel attention is a different mechanism.

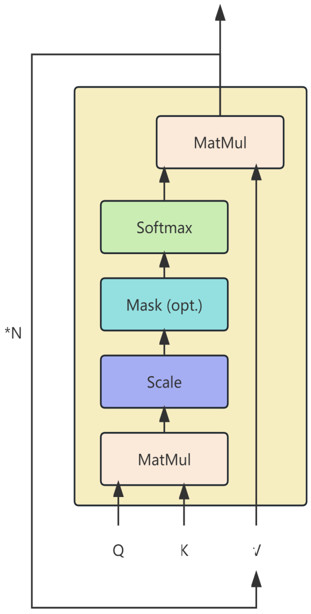

Fig. 2: Multilevel attention

Multilevel attention is forming a reasoning link from query to key to value and then to key to value and repeat this process and then get the outcome. Firstly, we apply dot-product between query Q and key K. Then, we divide the outcome by the dimension of key d_k, and mask the sequence after the predicted position. We use $softmax$ function as the activation. And we dot-product the outcome with value V. Finally, the outcome is assigned to the new value V' of the next iteration. This process is shown in figure 2.

We choose scaled dot-product attention as a basic method due to its simplicity and convenience for calculation compared to additive attention.

$$Attention(Q, K, V) = softmax(\frac{QK^T}{\sqrt{d_k}})V \tag{1}$$

In Multilevel Attention, we take $Attention(Q, K, V)$ as the new value of next iteration V'_{i+1}, where i is the iteration number of Multilevel Attention. Then start a new round of attention calculation. This circulation will repeat N times to form a reasoning link.

$$V'_2 = softmax(\frac{QK^T}{\sqrt{d_k}})V \tag{2}$$

$MultiAttention(Q, K, V'_{N-1})$ is equal to

$$softmax(\frac{QK^T}{\sqrt{d_k}})V'_{N-1} \tag{3}$$

It's worth noting that initially, we take the output of $Attention(Q, K, V)$ as the new Q'_{i+1}, K'_{i+1}, and V'_{i+1}. However, we found that the train loss will be constantly NAN and the accuracy will be constantly zero. The reason is that when the query is iterated, the original meaning of the query is lost. Thus, the finial outcome can not match the original query. After modification, we use the multilevel attention mechanism shown above. Then the outcome becomes normal.

IV. Problem Set-up

Model Architecture The model we use is classic transformer model. We replace the scaled dot-product attention with our multilevel attention.

Experiment Problem We used machine translation to We used machine translation to test our model.

Dataset The machine translation datasets is Chinese-English translation mission from [1]. The size of it is 2 MB. The training part and testing part are separated.

Evaluation We use Training loss, training accuracy, and BLEU [25] to evaluate the predicted sequence.

Hardware A laptop with Intel I7-13700HX, NVIDIA RTX 4070 (8GB), 16 GB of memory.

V. Experiment and Result

A. *Comparison with transformer*

We use transformer as compared algorithm. And the parameter of both algorithm are the same as parameters set in the transformer article [15]. $MultiAttention_2$ means that the multilevel attention repeat 2 times. $MultiAttention_3$ means that the multilevel attention repeat 3 times, and $MultiAttention_4$ means that the multilevel attention repeat 4 times. The finial training loss and training accuracy of Multilevel Attention models and Transformer is shown in table I. And the curve of training loss and training accuracy of Multilevel Attention models and Transformer is shown in figure 3. Table II shows the time consumption of Multilevel Attention models and Transformer.

From table I and figure 3 we can see that the performant gap between Multilevel Attention model and Transformer is small. The finial training loss is distributed around 0.40, while the finial training accuracy is stable at 91 %. Also, table II shows that the time consumption of transformer and Multilevel Attention model is similar. Thus, to further exam our Multilevel Attention model, we use a case study and calculate its BLEU score.

The expect predicted outcome is:"we should protect environment". Following is the outcome of transformer.

Transformer:
$< /s >$ To maintain the environment environment environment protection environment protection environment protection environment protection environment protection environment protection environment protection environment protection environment protection environment $< /s >$ environment $< /s >$ environment $< /s >$ environment $< /s > < /s > < /s > < /s > < /s > < /s > < /s >$

And the BLEU score of this translation is zero. Apparently, the model is overfit, which means that this dataset is too small for transformer. The outcome of other Multilevel Attention models show similar result. We choose $MultiAttention_2$ as an example. Following is the outcome:

MultiAttention2:
$< /s >$ We must vigorously protect the environment protect the environment protect the environment protect the environment and work with a fast environment a fast environment a fast environment a fast environment a fast environment a fast environment a fast environment a fast environment $< /s >$ Environment $< /s >$ Environment $< /s >$ Environment $< /s >$ Environment $< /s >$ Environment $< /s >$ Environment $< /s > < /s > < /s > < /s >$

For all the predicted outcome of Multilevel Attention model, their BLEU score is zero. Hence, the dataset is too small for transformer based model. This explains the high training accuracy and high training loss. (However, it is hard to queue up a GPU for 24 hours in Great Lake. And the limit of my Laptop GPU make it difficult to train a large dataset. Thus, I temporarily use the small dataset to finish the experiment.)

Despite the overfit problem, we can still find some excellent signals of Multilevel Attention. Compared to transformer, the translation of Multilevel Attention models further expands the relevant meaning on the basis of the original meaning. This is a signal of divergent thinking ability. For example, for the word "protect", Transformer just output the "protection" itself. While $MultiAttention_2$ output "vigorously protect", and $MultiAttention_{100}$ output "effectively protect". Multilevel Attention add some adverb to describe the word "protect", though there are no such requirements.

Another example is "We should". Transformer outputs the word "To maintain" instead of "we". While $MultiAttention_4$ outputs "We want", and $MultiAttention_5$ outputs "we need to", $MultiAttention_{50}$ and $MultiAttention_{100}$ output "We must". It shows that Multilevel Attention output some related word of "should". Hence, compared to Transformer, Multilevel Attention mechanism can find the associate relation between different key and value.

However, compared to classic scaled dot-product attention, Multilevel Attention is expected to reasonably expand the searching area in the word vector space. The current experimental phenomena do not show this very well. Because these phenomena actually occur frequently in machine translation tasks. The reason for this is that the query does not match the appropriate key in the word vector space very well, thus multiple similar keys will be matched.

B. *Effect of multilevel attention*

In the ablation study, we change the number of repeat time of Multilevel Attention. We repeat the Multilevel Attention 2, 3, 4, 5, 10, 20, 50 and 100 times separately. As shown in II, the time consumption gap between different $MultiAttention$

[1] https://github.com/P3n9W31/transformer-pytorch

TABLE I: Finial training loss and training accuracy

Algorithm	Training Loss	Training Accuracy
Transformer	0.400	91.221%
$MultiAttention_2$	0.406	91.495%
$MultiAttention_3$	0.408	91.426%
$MultiAttention_4$	0.402	91.676%
$MultiAttention_5$	0.411	91.394%
$MultiAttention_{10}$	0.406	91.635%
$MultiAttention_{20}$	0.406	91.537%
$MultiAttention_{50}$	0.401	91.473%
$MultiAttention_{100}$	0.404	91.160%

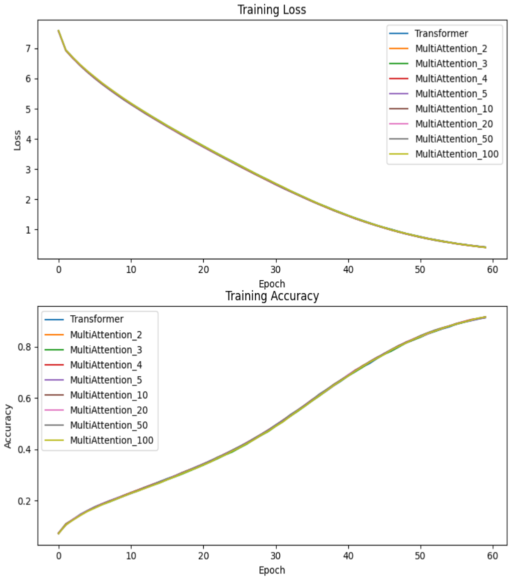

Fig. 3: Training loss and training accuracy

TABLE II: Time consumption

Algorithm	Training Time (min)
Transformer	30:55
$MultiAttention_2$	27:37
$MultiAttention_3$	31:43
$MultiAttention_4$	29:34
$MultiAttention_5$	30:43
$MultiAttention_{10}$	31:00
$MultiAttention_{20}$	31:39
$MultiAttention_{50}$	39:47
$MultiAttention_{100}$	30:32

model is very small. Figure 3 and table I show that the performance of different multilevel model is similar in training loss and training accuracy.

It seems that the repeat times of Multilevel Attention does not cause negative effect to the time efficiency. Since the tensor product can be easily calculated by modern GPU. However, due to the overfit problem, it is hard to analysis the effect of Multilevel Attention repetition. As the case study shown, there are some signals of divergent thinking ability. But the BLEU score of all $MultiAttention$ is zero, which make it is difficult to further exam the effectiveness of Multilevel Attention repetition.

VI. Conclusion and Future work

In this paper, we modify the principle of attention mechanism to improve its ability of founding the relation between key and value. We introduce the idea of reasoning link, and propose multilevel attention. Several experiments show that multilevel attention can improve divergent thinking ability of transformer based model. However, due to the limit of my laptop computational ability, the dataset is not large enough to fully train the model. Thus, further experiment with large dataset is needed for this model. I will look for computing resources to do further experiment in the following day.

References

[1] L. Itti, C. Koch, and E. Niebur, “A model of saliency-based visual attention for rapid scene analysis,” *IEEE Transactions on pattern analysis and machine intelligence*, vol. 20, no. 11, pp. 1254–1259, 1998.

[2] V. Mnih, N. Heess, A. Graves *et al.*, “Recurrent models of visual attention,” *Advances in neural information processing systems*, vol. 27, 2014.

[3] D. Bahdanau, K. Cho, and Y. Bengio, “Neural machine translation by jointly learning to align and translate,” *arXiv preprint arXiv:1409.0473*, 2014.

[4] S. Qiao, Y. Ou, N. Zhang, X. Chen, Y. Yao, S. Deng, C. Tan, F. Huang, and H. Chen, “Reasoning with language model prompting: A survey,” *arXiv preprint arXiv:2212.09597*, 2022.

[5] J. Wei, Y. Tay, R. Bommasani, C. Raffel, B. Zoph, S. Borgeaud, D. Yogatama, M. Bosma, D. Zhou, D. Metzler *et al.*, “Emergent abilities of large language models,” *arXiv preprint arXiv:2206.07682*, 2022.

[6] Y. Liu, T. Han, S. Ma, J. Zhang, Y. Yang, J. Tian, H. He, A. Li, M. He, Z. Liu *et al.*, “Summary of chatgpt/gpt-4 research and perspective towards the future of large language models,” *arXiv preprint arXiv:2304.01852*, 2023.

[7] Z. Niu, G. Zhong, and H. Yu, “A review on the attention mechanism of deep learning,” *Neurocomputing*, vol. 452, pp. 48–62, 2021.

[8] T. Lin, Y. Wang, X. Liu, and X. Qiu, “A survey of transformers,” *AI Open*, 2022.

[9] G. Brauwers and F. Frasincar, “A general survey on attention mechanisms in deep learning,” *IEEE Transactions on Knowledge and Data Engineering*, 2021.

[10] N. Zhang and J. Kim, “A survey on attention mechanism in nlp,” in *2023 International Conference on Electronics, Information, and Communication (ICEIC)*. IEEE, 2023, pp. 1–4.

[11] J. Lu, J. Yang, D. Batra, and D. Parikh, “Hierarchical question-image co-attention for visual question answering,” *Advances in neural information processing systems*, vol. 29, 2016.

[12] Z. Yang, D. Yang, C. Dyer, X. He, A. Smola, and E. Hovy, “Hierarchical attention networks for document classification,” in *Proceedings of the 2016 conference of the North American chapter of the association for computational linguistics: human language technologies*, 2016, pp. 1480–1489.

[13] N. K. Tran and C. Niedereée, “Multihop attention networks for question answer matching,” in *The 41st international ACM SIGIR conference on research & development in information retrieval*, 2018, pp. 325–334.

[14] Z. Yang, X. He, J. Gao, L. Deng, and A. Smola, “Stacked attention networks for image question answering,” in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2016, pp. 21–29.

[15] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, Ł. Kaiser, and I. Polosukhin, “Attention is all you need,” *Advances in neural information processing systems*, vol. 30, 2017.

[16] T. Shen, T. Zhou, G. Long, J. Jiang, S. Pan, and C. Zhang, “Disan: Directional self-attention network for rnn/cnn-free language understanding,” in *Proceedings of the AAAI conference on artificial intelligence*, vol. 32, no. 1, 2018.

[17] N. Shazeer, “Fast transformer decoding: One write-head is all you need,” *arXiv preprint arXiv:1911.02150*, 2019.

[18] J. Ainslie, J. Lee-Thorp, M. de Jong, Y. Zemlyanskiy, F. Lebrón, and S. Sanghai, "Gqa: Training generalized multi-query transformer models from multi-head checkpoints," *arXiv preprint arXiv:2305.13245*, 2023.

[19] J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova, "Bert: Pre-training of deep bidirectional transformers for language understanding," *arXiv preprint arXiv:1810.04805*, 2018.

[20] J. Wei, X. Wang, D. Schuurmans, M. Bosma, F. Xia, E. Chi, Q. V. Le, D. Zhou *et al.*, "Chain-of-thought prompting elicits reasoning in large language models," *Advances in Neural Information Processing Systems*, vol. 35, pp. 24 824–24 837, 2022.

[21] X. Wang, J. Wei, D. Schuurmans, Q. Le, E. Chi, S. Narang, A. Chowdhery, and D. Zhou, "Self-consistency improves chain of thought reasoning in language models," *arXiv preprint arXiv:2203.11171*, 2022.

[22] Y. Li, Z. Lin, S. Zhang, Q. Fu, B. Chen, J.-G. Lou, and W. Chen, "Making language models better reasoners with step-aware verifier," in *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, 2023, pp. 5315–5333.

[23] F. Petroni, T. Rocktäschel, P. Lewis, A. Bakhtin, Y. Wu, A. H. Miller, and S. Riedel, "Language models as knowledge bases?" *arXiv preprint arXiv:1909.01066*, 2019.

[24] J. Davison, J. Feldman, and A. M. Rush, "Commonsense knowledge mining from pretrained models," in *Proceedings of the 2019 conference on empirical methods in natural language processing and the 9th international joint conference on natural language processing (EMNLP-IJCNLP)*, 2019, pp. 1173–1178.

[25] K. Papineni, S. Roukos, T. Ward, and W.-J. Zhu, "Bleu: a method for automatic evaluation of machine translation," in *Proceedings of the 40th annual meeting of the Association for Computational Linguistics*, 2002, pp. 311–318.