
Supplemental Material: On the Role of Hidden States of Modern Hopfield Network in Transformer

A Acknowledgment

MT was partially supported by JSPS KAKENHI (22H05116), JST CREST (JPMJCR22N4) and AMED under Grant Number JP25wm0625422.

B Limitations

The limitation of this paper is that due to the constraints of computer resources, the experiments are limited to academic scale: At most, GPT-2 (Medium) trained on Wikitext103, ViT-L trained on CIFAR100, and ViT-B trained on ImageNet-1k. It will be an interesting future direction to see what results will be obtained with larger-scale pre-training. In addition, this paper mainly considered rank collapse and attention entropy as factors that contribute to the performance improvement of MHA, but it may be interesting to investigate whether there is any relationship with other factors. In this paper, the research was conducted based on the similarity of the mathematical structure between Hopfield networks and Transformer, but clarifying the essential deep relationship between associative memory and self-attention mechanisms is also a major future challenge.

C Preliminaries

Recall the definition of the operator p -norm:

$$\|A\|_p = \max_{x \neq 0} \frac{\|Ax\|_p}{\|x\|_p}. \quad (1)$$

The right-hand side is computed using the vector p -norm $\|\cdot\|_p$.

We will use some basic properties of the operator norm. First, the 1-norm and ∞ -norm are given in the following simple form:

$$\|A\|_1 = \max_j \sum_i |A_{ij}|, \quad (2)$$

$$\|A\|_\infty = \max_i \sum_j |A_{ij}|. \quad (3)$$

The submultiplicativity of these two norms is the following properties

$$\|AB\|_1 \leq \|A\|_1 \|B\|_1, \quad (4)$$

$$\|AB\|_\infty \leq \|A\|_\infty \|B\|_\infty. \quad (5)$$

We have the same property for another choice of norm $\|\cdot\|_p$.

We also have the following Hölder's inequality for 1- and ∞ -norms

$$\|AB\|_1 \leq \|A\|_\infty \|B\|_1. \quad (6)$$

Proof is straightforward as $\|\mathbf{A}\mathbf{B}\|_1 = \max_i \sum_k |A_{ik}| \sum_j |B_{kj}| \leq \max_i \sum_k |A_{ik}| \max_{k'} \sum_j |B_{k'j}| = \|\mathbf{A}\|_\infty \|\mathbf{B}\|_1$.

Following the paper [3], we also use the following composite of operator norms in this paper

$$\|\mathbf{A}\|_{1,\infty} = \sqrt{\|\mathbf{A}\|_1 \|\mathbf{A}\|_\infty}. \quad (7)$$

D Proof of Theorem: Convergence of the Residual

In this section, we present the proof of theorems on the phenomenon of rank collapse. In the following, we extend the proof and main theorems of [3] to our model, filling in minor errors and gaps in the proof in the literature.

First, recall the defining equation of the residual in [3]:

$$\mathbf{R}^{(\ell)} = \text{res}(\mathbf{X}^{(\ell)}) = \mathbf{X}^{(\ell)} - \mathbf{1}\mathbf{x}^{(\ell)\top}, \quad (8)$$

$$\mathbf{x}^{(\ell)} = \arg \min_{\mathbf{x}} \|\mathbf{X}^{(\ell)} - \mathbf{1}\mathbf{x}^\top\|_F, \quad (9)$$

where a row of rank one matrix for \mathbf{X} is given by

$$\mathbf{x}^{(\ell)} = \frac{1}{N} \mathbf{X}^\top \mathbf{1}. \quad (10)$$

Let's start with the simplest case. The single head self-attention layer is

$$\text{SA}(\mathbf{X}^{(\ell)}) = \mathbf{P}^{(\ell)} \mathbf{X}^{(\ell)} \mathbf{W}_V^{(\ell)}, \quad (11)$$

where the attention weight $\mathbf{P}^{(\ell)}$ is given by the attention score $\mathbf{A}^{(\ell)}$ as

$$\mathbf{P}^{(\ell)} = \text{softmax}_{\text{row}}(\mathbf{A}^{(\ell)}). \quad (12)$$

$\text{softmax}_{\text{row}}$ is the row-wise softmax function. For self-attention layer, the residual of the layer is $\text{Res}(\text{SA}(\mathbf{X}^{(\ell)})) = \mathbf{P}^{(\ell)} \mathbf{X}^{(\ell)} \mathbf{W}_V^{(\ell)} - \mathbf{1}\mathbf{z}^\top$ for certain \mathbf{z} . [3] gave inequality for the norm of this residual.

D.1 Single-Layer MHA

First, to investigate the rank collapse caused by single MHA-layer, we prove the following inequality to evaluate the degree of convergence of the residual of MHA layer $\text{Res}(\text{MHA}(\mathbf{X}^{(\ell)}))$:

Lemma D.1. *When $\alpha = 0$, for a single-head MHA layer, the residual is bounded as follows:*

$$\|\text{Res}(\text{MHA}(\mathbf{X}^{(\ell)}))\|_{1,\infty} \leq \frac{4(1-\alpha')C_1}{\sqrt{d_k}} \|\text{Res}(\mathbf{X}^{(\ell)})\|_{1,\infty}^3 + \frac{4\alpha'C_2}{\sqrt{d_k}} \|\text{Res}(\mathbf{X}^{(\ell)})\|_{1,\infty}. \quad (13)$$

In order to examine the contribution of MHA alone, in addition to not adding any shortcut paths, we also removed the shortcut paths within MHA by setting $\alpha = 0$. Extending such inequalities to the case where there are shortcut paths is straightforward by following the techniques in [3]. The coefficients here are $C_1 = \|\mathbf{W}_V^{(\ell)}\|_{1,\infty} \|\mathbf{W}_{QK}^{(\ell)}\|_1$ and $C_2 = \|\mathbf{W}_V^{(\ell)}\|_{1,\infty} \|\mathbf{A}^{(\ell-1)}\|_1$.

The attention weight of our MHA layer is the exponential moving average of query-key dot-product $\mathbf{Q}\mathbf{K}^\top$ across layers:

$$\mathbf{P}^{(\ell)} = \text{softmax}_{\text{row}}(\mathbf{A}^{(\ell)}), \quad (14)$$

$$\mathbf{A}^{(\ell)} = \frac{1}{\sqrt{d_k}} \beta \sum_{m=1}^{\ell} \alpha'^{\ell-m} \mathbf{Q}^{(m)} \mathbf{K}^{(m)\top} = \frac{1}{\sqrt{d_k}} \beta \mathbf{Q}^{(\ell)} \mathbf{K}^{(\ell)\top} + \frac{1}{\sqrt{d_k}} \alpha' \mathbf{A}^{(\ell-1)}, \quad (15)$$

where $\beta = 1 - \alpha'$. The query and key matrices here are $\mathbf{Q}^{(\ell)} = \mathbf{X}^{(\ell)} \mathbf{W}_Q^{(\ell)} + \mathbf{1} \mathbf{b}_Q^{(\ell)\top}$ and $\mathbf{K}^{(\ell)} = \mathbf{X}^{(\ell)} \mathbf{W}_K^{(\ell)} + \mathbf{1} \mathbf{b}_K^{(\ell)\top}$. Notice that the addition of the bias term $+\mathbf{1} \mathbf{b}_{Q,K}^{(\ell)\top}$ is the broadcast addition of $\mathbf{b}_{Q,K}^{(\ell)\top}$ along the token dimension. Substituting this expression gives

$$\begin{aligned} \mathbf{A}^{(\ell)} &= \frac{1}{\sqrt{d_k}} \beta \left(\mathbf{X}^{(\ell)} \mathbf{W}_Q^{(\ell)} + \mathbf{1} \mathbf{b}_Q^{(\ell)\top} \right) \left(\mathbf{X}^{(\ell)} \mathbf{W}_K^{(\ell)} + \mathbf{1} \mathbf{b}_K^{(\ell)\top} \right)^\top + \frac{1}{\sqrt{d_k}} \alpha' \mathbf{A}^{(\ell-1)} \\ &= \frac{1}{\sqrt{d_k}} \beta \mathbf{X}^{(\ell)} \mathbf{W}_{QK}^{(\ell)} \mathbf{X}^{(\ell)\top} + \frac{1}{\sqrt{d_k}} \beta \mathbf{1} \mathbf{b}_{QK}^{(\ell)\top} \mathbf{X}^{(\ell)\top} + \frac{1}{\sqrt{d_k}} \alpha' \mathbf{A}^{(\ell-1)} + (\dots) \mathbf{1}^\top. \end{aligned} \quad (16)$$

We use the following notation

$$\mathbf{W}_{QK}^{(\ell)} = \mathbf{W}_Q^{(\ell)} \mathbf{W}_K^{(\ell)\top}, \quad (17)$$

$$\mathbf{b}_{QK}^{(\ell)} = \mathbf{W}_K^{(\ell)} \mathbf{b}_Q^{(\ell)}. \quad (18)$$

Substituting the definition of the residual, we obtain $\mathbf{X}^{(\ell)} = \mathbf{R}^{(\ell)} + \mathbf{1} \mathbf{x}^{(\ell)}$

$$\begin{aligned} \mathbf{A}^{(\ell)} &= \frac{1}{\sqrt{d_k}} \left(\beta \mathbf{1} \mathbf{x}^{(\ell)\top} \mathbf{W}_{QK}^{(\ell)} \mathbf{R}^{(\ell)\top} + \beta \mathbf{R}^{(\ell)} \mathbf{W}_{QK}^{(\ell)} \mathbf{R}^{(\ell)\top} + \beta \mathbf{1} \mathbf{b}_{QK}^{(\ell)\top} \mathbf{R}^{(\ell)\top} + \alpha' \mathbf{A}^{(\ell-1)} \right) \\ &\quad + (\dots) \mathbf{1}^\top. \end{aligned} \quad (19)$$

The point here is that the last term is constant over row. Such a constant shift of attention score $\mathbf{A}^{(\ell)}$ does not affect the attention weight $\mathbf{P}^{(\ell)}$ since the softmax function is row-wise. Therefore, this term will be omitted below.

To simplify the equations, we introduce the following notation:

$$\mathbf{r}^{(\ell)\top} = \frac{1}{\sqrt{d_k}} \beta \mathbf{x}^{(\ell)\top} \mathbf{W}_{QK}^{(\ell)} \mathbf{R}^{(\ell)\top} + \frac{1}{\sqrt{d_k}} \beta \mathbf{b}_{QK}^{(\ell)\top} \mathbf{R}^{(\ell)\top}, \quad (20)$$

$$= \frac{1}{\sqrt{d_k}} \beta \left(\mathbf{R}^{(\ell)} \mathbf{W}_{QK}^{(\ell)\top} \mathbf{x}^{(\ell)} + \mathbf{R}^{(\ell)} \mathbf{b}_{QK}^{(\ell)} \right)^\top, \quad (21)$$

$$\mathbf{E}^{(\ell)} = \frac{1}{\sqrt{d_k}} \beta \mathbf{R}^{(\ell)} \mathbf{W}_{QK}^{(\ell)} \mathbf{R}^{(\ell)\top}. \quad (22)$$

The attention score is then

$$\mathbf{A}^{(\ell)} = \mathbf{E}^{(\ell)} + \mathbf{1} \mathbf{r}^{(\ell)\top} + \frac{1}{\sqrt{d_k}} \alpha' \mathbf{A}^{(\ell-1)} = \tilde{\mathbf{E}}^{(\ell)} + \mathbf{1} \mathbf{r}^{(\ell)\top}, \quad (23)$$

where $\tilde{\mathbf{E}}^{(\ell)} = \mathbf{E}^{(\ell)} + \frac{1}{\sqrt{d_k}} \alpha' \mathbf{A}^{(\ell-1)}$.

Since $\mathbf{P}^{(\ell)}$ is row-stochastic matrix, we have $\mathbf{P}^{(\ell)} \mathbf{1} = \mathbf{1}$. Therefore, $\mathbf{P}^{(\ell)} \mathbf{R}^{(\ell)} = \mathbf{P}^{(\ell)} (\mathbf{X}^{(\ell)} - \mathbf{1} \mathbf{x}^{(\ell)\top})$ gives

$$\mathbf{P}^{(\ell)} \mathbf{R}^{(\ell)} = \mathbf{P}^{(\ell)} \mathbf{X}^{(\ell)} - \mathbf{1} \mathbf{x}^{(\ell)\top}. \quad (24)$$

Using these relations, we can obtain an inequality for the norm of the Res(MHA($\mathbf{X}^{(\ell)}$)).

First, let us use the inequality used to prove the theorem for self-attention in [3]. Lemma A.3 in [3] under assumption $|\tilde{E}_{ij} - \tilde{E}_{ik}| \leq 1.256$ lead to the following element-wise inequalities

$$(\mathbf{I} - 2\tilde{\mathbf{D}}) \mathbf{1} \text{softmax}(\mathbf{r}^{(\ell)\top}) \mathbf{R}^{(\ell)} \preceq \mathbf{P}^{(\ell)} \mathbf{R}^{(\ell)} \preceq (\mathbf{I} + 2\tilde{\mathbf{D}}) \mathbf{1} \text{softmax}(\mathbf{r}^{(\ell)\top}) \mathbf{R}^{(\ell)}, \quad (25)$$

where \preceq is element-wise inequality and \mathbf{I} is the identity matrix. The matrix $\tilde{\mathbf{D}}$ is the following diagonal matrix

$$\tilde{\mathbf{D}} = \text{diag}(\mathbf{d}), \quad d_i = \max_{j,k} |\tilde{E}_{ij} - \tilde{E}_{ik}|. \quad (26)$$

This gives us the following element-wise inequality:

$$\begin{aligned} -2\tilde{\mathbf{D}} \mathbf{1} \text{softmax}(\mathbf{r}^{(\ell)\top}) \mathbf{R}^{(\ell)} &\preceq \mathbf{P}^{(\ell)} \mathbf{R}^{(\ell)} - \mathbf{1} \left(\mathbf{x}^{(\ell)\top} + \text{softmax}(\mathbf{r}^{(\ell)\top}) \mathbf{R}^{(\ell)} \right) \\ &\preceq 2\tilde{\mathbf{D}} \mathbf{1} \text{softmax}(\mathbf{r}^{(\ell)\top}) \mathbf{R}^{(\ell)}. \end{aligned} \quad (27)$$

We therefore have the element-wise inequality

$$|\mathbf{P}^{(\ell)} \mathbf{R}^{(\ell)} - \mathbf{1} (\mathbf{x}^{(\ell)\top} + \text{softmax}(\mathbf{r}^{(\ell)\top}) \mathbf{R}^{(\ell)})| \preceq 2|\tilde{\mathbf{D}} \mathbf{1} \text{softmax}(\mathbf{r}^{(\ell)\top}) \mathbf{R}^{(\ell)}|. \quad (28)$$

Here, $|\cdot|$ is the element-wise absolute value function. This elementwise inequality implies the following inequalities for single-head MHA layer $\text{MHA}(\mathbf{X}^{(\ell)}) = \mathbf{P}^{(\ell)} \mathbf{X}^{(\ell)} \mathbf{W}_V^{(\ell)}$:

$$\|\text{MHA}(\mathbf{X}^{(\ell)}) - \mathbf{res}^{(\ell)}\|_1 \leq 2\|\tilde{\mathbf{D}} \mathbf{1} \text{softmax}(\mathbf{r}^{(\ell)\top}) \mathbf{R}^{(\ell)} \mathbf{W}_V^{(\ell)}\|_1, \quad (29)$$

$$\|\text{MHA}(\mathbf{X}^{(\ell)}) - \mathbf{res}^{(\ell)}\|_\infty \leq 2\|\tilde{\mathbf{D}} \mathbf{1} \text{softmax}(\mathbf{r}^{(\ell)\top}) \mathbf{R}^{(\ell)} \mathbf{W}_V^{(\ell)}\|_\infty. \quad (30)$$

Using the norm properties of (2) and (3), this inequality can be immediately proven from (28). The shorthand notation $\mathbf{res}^{(\ell)} = \mathbf{x}^{(\ell)\top} + \text{softmax}(\mathbf{r}^{(\ell)\top}) \mathbf{R}^{(\ell)} \mathbf{W}_V$ is used in the following.

Using Hölder's inequality (6) and the submultiplicativity of the 1-norm, the right-hand side of (29) becomes

$$\begin{aligned} \|\text{MHA}(\mathbf{X}^{(\ell)}) - \mathbf{res}^{(\ell)}\|_1 &\leq 2\|\tilde{\mathbf{D}} \mathbf{1}\|_\infty \|\text{softmax}(\mathbf{r}^{(\ell)\top}) \mathbf{R}^{(\ell)} \mathbf{W}_V^{(\ell)}\|_1 \\ &\leq 2\|\tilde{\mathbf{D}} \mathbf{1}\|_\infty \|\mathbf{R}^{(\ell)}\|_1 \|\mathbf{W}_V^{(\ell)}\|_1. \end{aligned} \quad (31)$$

The inequality $\|\text{softmax}(\mathbf{r}^{(\ell)\top})\|_1 \leq 1$ used to show the second inequality is due to the fact that the magnitude of each element of the row vector on the right side is less than 1.

Similarly, by using submultiplicativity and $\|\text{softmax}(\mathbf{r}^{(\ell)\top})\|_\infty = 1$, we obtain the following from inequality (30)

$$\begin{aligned} \|\text{MHA}(\mathbf{X}^{(\ell)}) - \mathbf{res}^{(\ell)}\|_\infty &\leq 2\|\tilde{\mathbf{D}} \mathbf{1}\|_\infty \|\text{softmax}(\mathbf{r}^{(\ell)\top})\|_\infty \|\mathbf{R}^{(\ell)}\|_\infty \|\mathbf{W}_V^{(\ell)}\|_\infty \\ &= 2\|\tilde{\mathbf{D}} \mathbf{1}\|_\infty \|\mathbf{R}^{(\ell)}\|_\infty \|\mathbf{W}_V^{(\ell)}\|_\infty. \end{aligned} \quad (32)$$

Multiplying these two inequalities, we obtain the inequality for the composite of norms $\|\cdot\|_{1,\infty}$ as follows

$$\|\text{MHA}(\mathbf{X}^{(\ell)}) - \mathbf{res}^{(\ell)}\|_{1,\infty} \leq 2\|\tilde{\mathbf{D}} \mathbf{1}\|_\infty \|\mathbf{W}_V^{(\ell)}\|_{1,\infty} \|\mathbf{R}^{(\ell)}\|_{1,\infty}. \quad (33)$$

Since $\tilde{\mathbf{D}}$ depends on the input $\mathbf{X}^{(\ell)}$, let us evaluate the value of the norm $\|\tilde{\mathbf{D}} \mathbf{1}\|_\infty$ in terms of specific inequalities. From the definitions of $\tilde{\mathbf{D}}$, $\tilde{\mathbf{E}}$ and the ∞ -norm, we obtain the following inequality

$$\begin{aligned} \|\tilde{\mathbf{D}} \mathbf{1}\|_\infty &= \max_i \max_{jk} |E_{ij}^{(\ell)} - E_{ik}^{(\ell)} + \alpha' A_{ij}^{(\ell-1)} - \alpha' A_{ik}^{(\ell-1)}| \\ &\leq 2 \max_{ij} |E_{ij}^{(\ell)}| + 2\alpha' \max_{ij} |A_{ij}^{(\ell-1)}| \\ &\leq 2\|\mathbf{E}^{(\ell)}\|_1 + 2\frac{\alpha'}{\sqrt{d_k}} \|\mathbf{A}^{(\ell-1)}\|_1. \end{aligned} \quad (34)$$

Using (22), the norm $\|\mathbf{E}^{(\ell)}\|_1$ becomes

$$\begin{aligned} \|\mathbf{E}^{(\ell)}\|_1 &= \frac{\beta}{\sqrt{d_k}} \|\mathbf{R}^{(\ell)} \mathbf{W}_{QK}^{(\ell)} \mathbf{R}^{(\ell)\top}\|_1 \\ &\leq \frac{\beta}{\sqrt{d_k}} \|\mathbf{R}^{(\ell)}\|_1 \|\mathbf{W}_{QK}^{(\ell)}\|_1 \|\mathbf{R}^{(\ell)\top}\|_1 \\ &= \frac{\beta}{\sqrt{d_k}} \|\mathbf{W}_{QK}^{(\ell)}\|_1 \|\mathbf{R}^{(\ell)}\|_{1,\infty}^2, \end{aligned} \quad (35)$$

where we use $\|\mathbf{R}^{(\ell)\top}\|_1 = \|\mathbf{R}^{(\ell)}\|_\infty$ for the last equality.

Then, by combining (33), (34), and (35), we obtain the following inequality

$$\begin{aligned} \|\text{MHA}(\mathbf{X}^{(\ell)}) - \mathbf{res}^{(\ell)}\|_{1,\infty} &\leq \frac{4\beta}{\sqrt{d_k}} \|\mathbf{W}_V^{(\ell)}\|_{1,\infty} \|\mathbf{W}_{QK}^{(\ell)}\|_1 \|\mathbf{R}^{(\ell)}\|_{1,\infty}^3 \\ &\quad + \frac{4\alpha'}{\sqrt{d_k}} \|\mathbf{W}_V^{(\ell)}\|_{1,\infty} \|\mathbf{A}^{(\ell-1)}\|_1 \|\mathbf{R}^{(\ell)}\|_{1,\infty}. \end{aligned} \quad (36)$$

Since $\mathbf{R}^{(\ell)} = \text{Res}(\mathbf{X}^{(\ell)})$, we obtain Lemma D.1:

$$\begin{aligned} \|\text{Res}(\text{SA}(\mathbf{X}^{(\ell)}))\|_{1,\infty} &\leq \frac{4(1-\alpha')}{\sqrt{d_k}} \|\mathbf{W}_V^{(\ell)}\|_{1,\infty} \|\mathbf{W}_{QK}^{(\ell)}\|_1 \|\text{Res}(\mathbf{X}^{(\ell)})\|_{1,\infty}^3 \\ &\quad + \frac{4\alpha'}{\sqrt{d_k}} \|\mathbf{W}_V^{(\ell)}\|_{1,\infty} \|\mathbf{A}^{(\ell-1)}\|_1 \|\text{Res}(\mathbf{X}^{(\ell)})\|_{1,\infty}. \end{aligned} \quad (37)$$

This inequality is precisely (13).

D.2 Single-Layer Multi-Head MHA

Next, we extend the results of the previous section to the case of multi-headed MHA. The lemma we prove here is as follows:

Lemma D.2. *When $\alpha = 0$, for a multi-head MHA layer, the residual is bounded as follows:*

$$\|\text{Res}(\text{MHMHA}(\mathbf{X}^{(\ell)}))\|_{1,\infty} \leq \frac{4H(1-\alpha')C_1}{\sqrt{d_k}} \|\text{Res}(\mathbf{X}^{(\ell)})\|_{1,\infty}^3 + \frac{4H\alpha'C_2}{\sqrt{d_k}} \|\text{Res}(\mathbf{X}^{(\ell)})\|_{1,\infty}. \quad (38)$$

The coefficients used in the above inequality for the multi-head MHA are given by $C_1 = \max_h \|\mathbf{W}_{VO,h}^{(\ell)}\|_{1,\infty} \|\mathbf{W}_{QK,h}^{(\ell)}\|_1$ and $C_2 = \max_h \|\mathbf{W}_{VO,h}^{(\ell)}\|_{1,\infty} \|\mathbf{A}_h^{(\ell-1)}\|_1$.

Let us generalize (37) to the case of multi-head MHA. Multi-head version of MHA layer is

$$\text{MHMHA}(\mathbf{X}^{(\ell)}) = \sum_{h=1}^H \mathbf{P}_h^{(\ell)} \mathbf{X}^{(\ell)} \mathbf{W}_{VO,h} + \mathbf{1} \mathbf{b}_O^\top, \quad (39)$$

where $\mathbf{W}_{VO,h} = \mathbf{W}_{V,h} \mathbf{W}_{O,h}^\top$. Since this is essentially the sum of single-head contributions, (37) gives the following inequality

$$\begin{aligned} \|\text{Res}(\text{MHMHA}(\mathbf{X}^{(\ell)}))\|_{1,\infty} &\leq \frac{4(1-\alpha')}{\sqrt{d_k}} \|\text{Res}(\mathbf{X}^{(\ell)})\|_{1,\infty}^3 \sum_h \|\mathbf{W}_V^{(\ell)}\|_{1,\infty} \|\mathbf{W}_{QK}^{(\ell)}\|_1 \\ &\quad + \frac{4\alpha'}{\sqrt{d_k}} \|\text{Res}(\mathbf{X}^{(\ell)})\|_{1,\infty} \sum_h \|\mathbf{W}_V^{(\ell)}\|_{1,\infty} \|\mathbf{A}^{(\ell-1)}\|_1. \end{aligned} \quad (40)$$

This result immediately gives (38).

D.3 Multi-Layer Multi-Head MHA

Finally, consider an attention-only network consisting of only multi-head MHA layers

$$\text{AttnNet}(\mathbf{X}) = \text{MHMHA} \circ \text{MHMHA} \circ \dots \circ \text{MHMHA}(\mathbf{X}). \quad (41)$$

The residuals of the network obey the following theorem:

Theorem D.3. *When $\alpha = 0$, for a multi-head MHA-only network, the residual is bounded as follows:*

$$\begin{aligned} &\|\text{Res}(\text{AttnNet}(\mathbf{X}))\|_{1,\infty} \\ &\leq \max_{m=0}^L \left(\frac{4H(1-\alpha')C_1}{\sqrt{d_k}} \right)^{\frac{3^m-1}{2}} \left(\frac{4H\alpha'C_2}{\sqrt{d_k}} \right)^{3^m(L-m)} \|\text{Res}(\mathbf{X})\|_{1,\infty}^{3^m}. \end{aligned} \quad (42)$$

To prove this theorem, we use the following inequality for single-layer MHA, which follows immediately from (38):

$$\|\text{Res}(\text{MHMHA}(\mathbf{X}^{(\ell)}))\|_{1,\infty} \leq \max \left(\frac{8H(1-\alpha')C_1}{\sqrt{d_k}} \|\text{Res}(\mathbf{X}^{(\ell)})\|_{1,\infty}^3, \frac{8H\alpha'C_2}{\sqrt{d_k}} \|\text{Res}(\mathbf{X}^{(\ell)})\|_{1,\infty} \right). \quad (43)$$

This inequality for MHA has exactly the same structure as the inequality derived in [3] when skip connections are introduced to the self-attention layer, so the method of proving the inequality in our AttnNet is the same as in [3]. The point here is that even though our MHA completely eliminates skip connections by setting $\alpha = 0$, we can obtain the same improvement in the upper bound of the inequality as when skip connections are added to the normal attention layer.

The proof is as follows: For each layer, expand $\text{Res}(\text{AttnNet}(\mathbf{X}))$ using (1). Since the choice of upper bound at each layer is either $\frac{8H(1-\alpha')C_1}{\sqrt{d_k}} \|\text{Res}(\mathbf{X}^{(\ell)})\|_{1,\infty}^3$ or $\frac{8H\alpha'C_2}{\sqrt{d_k}} \|\text{Res}(\mathbf{X}^{(\ell)})\|_{1,\infty}$, the calculation of the upper bound of $\text{Res}(\text{AttnNet}(\mathbf{X}))$ is a repetition of these two choices. Therefore, in the tree obtained by repeated binary expansion over L layers, we need to consider the maximum value of the leaf term, which is $\max \left(\frac{4H(1-\alpha')C_1}{\sqrt{d_k}} \right)^{\frac{3^m-1}{2}} \left(\frac{4H\alpha'C_2}{\sqrt{d_k}} \right)^{3^m(L-m)} \|\text{Res}(\mathbf{X}^{(0)})\|_{1,\infty}^{3^m}$.

E Empirical Check of Theorem

We experimentally demonstrate that the upper bound inequality derived in this paper accurately explains the decaying behavior observed in actual networks. The effect of the doubly exponential decay in norms becomes particularly prominent in deep models. To provide a clear and direct setting for comparison with the theoretical analysis, we therefore conduct evaluations in multi-layer configurations. Following [5], we evaluate the following metric at initialization:

$$\frac{\|\text{Res}(\text{AttnNet}(\mathbf{X}))\|_{1,\infty}}{\|\text{AttnNet}(\mathbf{X})\|_{1,\infty}}. \quad (44)$$

Table 1 shows the layer-wise variation of this normalized norm (metric). We evaluated this norm on

depth of network	self-attention	MHA($\alpha = 0.5$)
1	0.48887348	0.87977747
2	0.07732825	0.8309065
3	0.00081001734	0.80018705
4	$1.7725031 * 10^{-6}$	0.7630522
5	$1.809994 * 10^{-6}$	0.71199465
6	$1.7835139 * 10^{-6}$	0.67957425
7	$1.7920005 * 10^{-6}$	0.6251911
8	$1.7568021 * 10^{-6}$	0.5860811
9	$1.75397 * 10^{-6}$	0.52244353
10	$1.7860738 * 10^{-6}$	0.46130562
11	$1.82849 * 10^{-6}$	0.432442
12	$1.7818496 * 10^{-6}$	0.39708787

Table 1: Normalized norm across layers in Attention-Only (ViT-T) and MHA networks evaluated on CIFAR-10 at initialization.

the CIFAR-10 dataset for Attention-Only Networks with 1 to 12 layers (ViT-T configuration) and their MHA counterparts. The left column shows the results with standard self-attention. As observed in previous studies, we confirm a sharp norm decay across layers, corresponding to rank collapse. In contrast, the right column shows that introducing MHA significantly mitigates this norm decay and suppresses the rank collapse through layers.

While the inequality we derive in this paper provides a theoretical upper bound on the norm decay, it closely aligns with the empirical behavior, and captures the rank-collapse suppression effect of MHA.

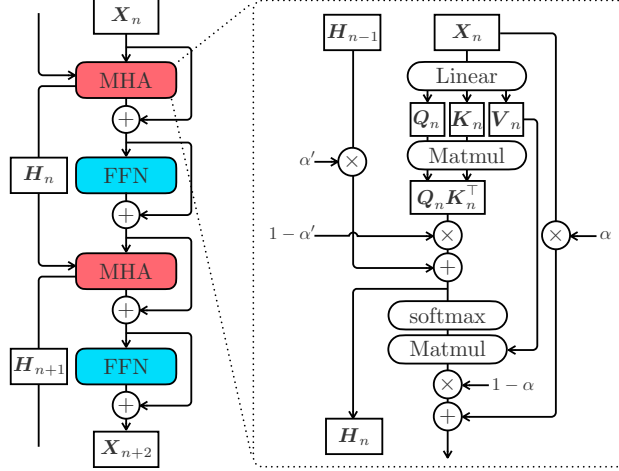


Figure 1: The architecture of the MHA examined in detail in this paper. This model corresponds to the case where the forward derivative is used for the visible state and the backward derivative for the hidden state. Simply setting $\alpha = \alpha' = 0$ reproduces normal self-attention.

F Choices of Discretization

In the paper, we employ the MHA architecture illustrated in Figure 1. However, different types of architectures can be introduced depending on the choice of discretization. In this section, we explain what results are obtained when the time evolution differential equations for visible and hidden states are discretized using forward or backward differentiation. We then comment on the reasons why we specifically address the Figure 1 case in this paper.

First, recall the state evolution equation of MCHN for Model B:

$$\tau_v \frac{d\mathbf{x}}{dt} = \text{softmax}(\mathbf{h}) \mathbf{V} - \mathbf{x}, \quad (45)$$

$$\tau_h \frac{d\mathbf{h}}{dt} = \mathbf{q} \mathbf{K}^\top - \mathbf{h}, \quad (46)$$

Here, we have rewritten the expression with key \mathbf{K} , query \mathbf{q} , and value \mathbf{V} using the correspondence with Transformer [5]. We will use forward or backward differentiation for the discretization of the left hand side of these equations.

F.1 The Case of (Forward, Forward)

First, let us consider the case where both visible and hidden states are discretized by forward differentiation. In this case, the differential equations (45,46) becomes the following difference equations

$$\frac{1}{\rho} (\mathbf{x}_{n+1} - \mathbf{x}_n) = \text{softmax}(\mathbf{h}_n) \mathbf{V}_n - \mathbf{x}_n, \quad (47)$$

$$\frac{1}{\rho'} (\mathbf{h}_{n+1} - \mathbf{h}_n) = \mathbf{q}_n \mathbf{K}_n^\top - \mathbf{h}_n, \quad (48)$$

where we use the following notation

$$\frac{\tau_v}{\Delta t} = \frac{1}{\rho}, \quad \frac{\tau_h}{\Delta t} = \frac{1}{\rho'}. \quad (49)$$

These equations are organized in a form that is easy to compare with self-attention as follows:

$$\mathbf{x}_{n+1} = (1 - \rho) \mathbf{x}_n + \rho \text{softmax}(\mathbf{h}_n) \mathbf{V}_n, \quad (50)$$

$$\mathbf{h}_n = (1 - \rho') \mathbf{h}_{n-1} + \rho' \mathbf{q}_{n-1} \mathbf{K}_{n-1}^\top. \quad (51)$$

This unfamiliar form of this attention-like model calculates the attention weights $\text{softmax}(\mathbf{h}_n)$ using information from the attention scores $\mathbf{q}_{n-1} \mathbf{K}_{n-1}^\top$ up to one layer prior through the hidden state, as shown in Figure 2. Thus, the attention weights applied to values \mathbf{V}_n do not include information about queries and keys in the same layer. Therefore, even if we consider the case $\rho' = 1$ in (51), we do not return to the usual attention, but only to an attention score of 0.

Mathematically, the usual self-attention is obtained using the equation $0 = \mathbf{h}_{n-1} + \mathbf{q}_{n-1} \mathbf{K}_{n-1}^\top$, which is obtained by taking the limit $\rho' \rightarrow \infty$ in (51). This is precisely the adiabatic limit of [3]. However, here we consider these state update equations as forward propagation equations of neural network. In actual network implementations, it is difficult to naturally perform operations equivalent to this adiabatic limit for the model. Therefore, since this model does not naturally give Transformer, we do not examine it in detail in this paper.

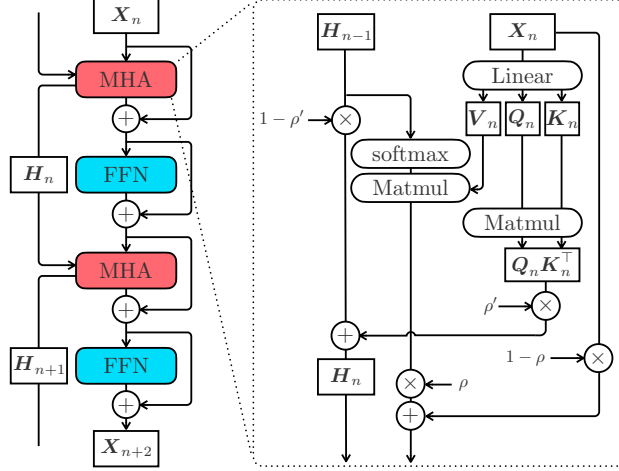


Figure 2: The architecture corresponds to the case where the forward derivative is used for the visible state and the hidden state.

F.2 The Case of (Forward, Backward)

Next, let us consider the case in which the forward derivative is used for the visible state while the backward derivative is used for the hidden state. This is the case examined in detail in this paper. In this case, the evolution equation for the hidden state is as follows

$$\frac{1}{\rho'} (\mathbf{h}_{n+1} - \mathbf{h}_n) = \mathbf{q}_{n+1} \mathbf{K}_{n+1}^\top - \mathbf{h}_{n+1}. \quad (52)$$

These equations can be organized in a form that is easy to compare with self-attention as follows:

$$\mathbf{x}_{n+1} = (1 - \rho) \mathbf{x}_n + \rho \text{softmax}(\mathbf{h}_n) \mathbf{V}_n, \quad (53)$$

$$\mathbf{h}_n = \frac{1}{1 + \rho'} \mathbf{h}_{n-1} + \frac{\rho'}{1 + \rho'} \mathbf{q}_n \mathbf{K}_n^\top. \quad (54)$$

By setting $\alpha = 1 - \rho$ and $\alpha' = \frac{1}{1 + \rho'}$, we see that these equations give Figure 1. The important point in this case is that this architecture includes usual self-attention in a natural way. It is easy to see that $\alpha = \alpha' = 0$ reproduces the original self-attention, so this can be seen as a natural extension of the Transformer. Therefore, this paper, whose purpose is to examine the implications of MCHN for Transformer, has studied this case in detail.

F.3 The Case of (Backward, Forward)

Next, let us consider the case in which the backward derivative is used for the visible state while the forward derivative is used for the hidden state. In this case, the evolution equation for the visible state

is as follows

$$\frac{1}{\rho} (\mathbf{x}_{n+1} - \mathbf{x}_n) = \text{softmax}(\mathbf{h}_{n+1}) \mathbf{V}_{n+1} - \mathbf{x}_{n+1}. \quad (55)$$

We can organize these equations in the following form, which is easy to compare with self-attention:

$$\mathbf{x}_{n+1} = \frac{1}{1+\rho} \mathbf{x}_n + \frac{\rho}{1+\rho} \text{softmax}(\mathbf{h}_{n+1}) \mathbf{V}_{n+1}, \quad (56)$$

$$\mathbf{h}_{n+1} = (1-\rho') \mathbf{h}_n + \rho' \mathbf{q}_n \mathbf{K}_n^\top. \quad (57)$$

However, a problem arises when we try to consider the state update equation for the visible state as a forward propagation equation. This is because \mathbf{V}_{n+1} is needed to calculate \mathbf{x}_{n+1} , and \mathbf{V}_{n+1} is given by the linear projection of \mathbf{x}_{n+1} . Therefore, this equation cannot be naturally interpreted as a feedforward neural network as it is. We mention the possibility of interpreting it as a recurrent network in the next example, but we will not consider this case in depth in this paper.

F.4 The Case of (Backward, Backward)

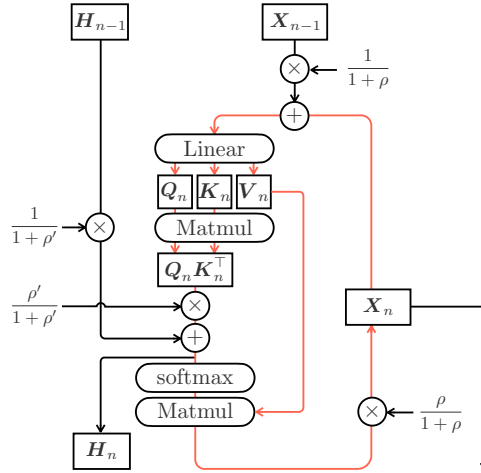


Figure 3: Possible interpretations as a recurrent neural net when both visible and hidden states use backward differentiation.

Finally, consider the case where both visible and hidden states are discretized by backward differentiation. In this case, the time evolution equations for these two states are as follows

$$\mathbf{x}_{n+1} = \frac{1}{1+\rho} \mathbf{x}_n + \frac{\rho}{1+\rho} \text{softmax}(\mathbf{h}_{n+1}) \mathbf{V}_{n+1}, \quad (58)$$

$$\mathbf{h}_{n+1} = \frac{1}{1+\rho'} \mathbf{h}_n + \frac{\rho'}{1+\rho'} \mathbf{q}_{n+1} \mathbf{K}_{n+1}^\top. \quad (59)$$

This equation (58) cannot be naturally interpreted as a feedforward neural network as in the previous case. If we try to understand it as a neural network, it may be possible to interpret it as a recurrent neural network, as shown in Figure 3. In any case, such complicated cases are beyond the scope of this paper. Therefore, we will not examine it in depth in this paper.

F.5 The Cases of Central Differentiation

In addition to the previous examples, there is also the option of using central derivatives. Using the central derivative for the visible state yields the following update rule

$$\mathbf{x}_{n+1} = 2\rho \text{softmax}(\mathbf{h}_n) \mathbf{V}_n + 2\rho \mathbf{x}_n + \mathbf{x}_{n-1}. \quad (60)$$

parameter	Small/Medium
num heads	12/16
depth	12/24
optimizer	AdamW
base learning rate	6e-4
batch size	12/4
num tokens	1024
embedding dimension	768/1024
learning rate schedule	cosine decay
lower learning rate bound	1e-6
warmup epochs	20
warmup schedule	linear
warmup learning rate	1e-6

Table 2: Detailed information on the GPT2 models and the settings used to train them.

In addition to $2\rho \mathbf{x}_n$, there is also a shortcut path that originates from \mathbf{x}_{n-1} in this forward propagation formula. This implies DenseNet-like generalization of shortcut path of Transformer and MHA.

On the other hand, using the central derivative for the hidden state yields the following equation

$$\mathbf{h}_{n+1} = 2\rho' \mathbf{q}_n \mathbf{K}_n + 2\rho' \mathbf{h}_n + \mathbf{h}_{n-1}. \quad (61)$$

This forward propagation formula complicates the computation of the attention weights because even information from two previous layers contributes to the calculation of the hidden state. Such overly complex architectures are outside the scope of this paper’s investigation.

G Experimental Setups

In this section, we describe the detailed training settings used in the experiments in this paper.

We used two sizes of GPT-2 models for text generation task in this paper, and their detailed information and training settings are summarized in Table 2.

We used four sizes of ViT models for image recognition task in this paper, and their detailed information and training settings are summarized in Table 3.

H Rank Collapse

In this section, we provide detailed information about the token similarity at each layer of GPT and ViT, and also present experimental results that could not be fully presented in the main text.

The rank collapse in [3] refers to the phenomenon in which the tokens corresponding to each row become perfectly proportional vectors in the attention network features, and the features degenerate into a matrix of rank 1. This means perfect token uniformity, where the cosine similarity of all token pairs is 1. On the other hand, the phenomenon observed in the actual Transformer is that although not all tokens are perfectly aligned, many tokens are perfectly aligned, resulting in the formation of a token population with a mutual cosine similarity of 1. In terms of rank, this is partial rank collapse, where the features degenerate into a matrix with a lower but non-1 rank. Here, we observe this phenomenon through detailed experimental results.

Figure 4 are plots that compare the token cosine similarity of GPT-2 (Medium) in each layer. In original GPT-2, there exists a population of tokens with a cosine similarity of 1. On the other hand, in the model using MHA, such token uniformity disappears. This result shows that MHA has a strong function of suppressing rank collapse.

Figures 5-12 are detailed plots comparing the token cosine similarity of ViT in each layer. In each case, in the normal ViT, there exists a population of tokens with a cosine similarity of 1 even after training. On the other hand, in the model using MHA, such token uniformity disappears. This result shows that MHA has a strong function of suppressing rank collapse.

parameter	Tiny/Small/Base/Large
num heads	3/6/12/16
depth	12/12/12/24
droppath rate	0.1/0.2/0.2/0.3
optimizer	AdamW
optimizer ϵ	1e-8
training epochs	300
base learning rate	6.25e-5
batch size	256/256/128/32
patch size	16
embedding dimension	192/384/768/1024
resized image size	(3,224,224)
learning rate schedule	cosine decay
lower learning rate bound	1e-6
warmup epochs	20
warmup schedule	linear
warmup learning rate	1e-6
cooldown epochs	10
random erasing	0.25
mixup α	0.8
cutmix α	1.0
label smoothing	0.1
RandAugment	(9,0.5)

Table 3: Detailed information on the ViT models and the settings used to train them.

I Entropy Collapse

Let’s also look at other problems that deep Transformers have besides rank collapse. Attention entropy collapse [4] is a measure of how much attention is focused on a small or large number of tokens, i.e., the degree of concentration of the attention distribution. A high attention entropy means that attention is directed to a large number of tokens, and the Transformer is distributing attention over a wide context. This is expected to allow each layer to construct a well contextualized embedding vector. On the other hand, when entropy is low, attention is directed to a small number of tokens.

When entropy is particularly low, it causes the problem of attention entropy collapse [7], which is a source of instability in Transformer training. This problem tends to occur when hyperparameters are not carefully set. We also know that concentration of attentions can lead to overfitting to some undesirable vocabulary, which can greatly impair the generalization and fairness of the language model [1, 6].

Figure 13-20 compares the attention entropy of the vanilla Transformers and our model. As can be seen from these figures, attention entropy tends to be small in some layers, but this property does not change even if attention weights are improved by MHA. In other words, MHA does not improve the performance of Transformer by improving entropy collapse.

Rank collapse means that diverse token vectors cannot be achieved, while entropy collapse means that uniform token mixing cannot be achieved. Therefore, this result is not surprising, as these two phenomena seem to be mutually exclusive. On the other hand, recent theoretical analysis [2] shows that rank collapse and entropy collapse are compatible phenomena, and further research in these contexts is an interesting future direction.

References

- [1] Giuseppe Attanasio, Debora Nozza, Dirk Hovy, and Elena Baralis. Entropy-based attention regularization frees unintended bias mitigation from lists. *arXiv preprint arXiv:2203.09192*, 2022.
- [2] Han Bao, Ryuichiro Hataya, and Ryo Karakida. Self-attention networks localize when qk-eigenspectrum concentrates. In *Proceedings of the 41st International Conference on Machine*

Learning, pages 2903–2922, 2024.

- [3] Yihe Dong, Jean-Baptiste Cordonnier, and Andreas Loukas. Attention is not all you need: Pure attention loses rank doubly exponentially with depth. In *International Conference on Machine Learning*, pages 2793–2803. PMLR, 2021.
- [4] Hamidreza Ghader and Christof Monz. What does attention in neural machine translation pay attention to? *Proceedings of the Eighth International Joint Conference on Natural Language Processing*, pages 30—39, 2017.
- [5] Hubert Ramsauer, Bernhard Schäfl, Johannes Lehner, Philipp Seidl, Michael Widrich, Lukas Gruber, Markus Holzleitner, Thomas Adler, David Kreil, Michael K Kopp, et al. Hopfield networks is all you need. In *International Conference on Learning Representations*, 2021.
- [6] Abdelrahman Zayed, Goncalo Mordido, Samira Shabanian, and Sarath Chandar. Should we attend more or less? modulating attention for fairness. *arXiv preprint arXiv:2305.13088*, 2023.
- [7] Shuangfei Zhai, Tatiana Likhomanenko, Etai Littwin, Dan Busbridge, Jason Ramapuram, Yizhe Zhang, Jiatao Gu, and Joshua M Susskind. Stabilizing transformer training by preventing attention entropy collapse. In *International Conference on Machine Learning*, pages 40770–40803. PMLR, 2023.

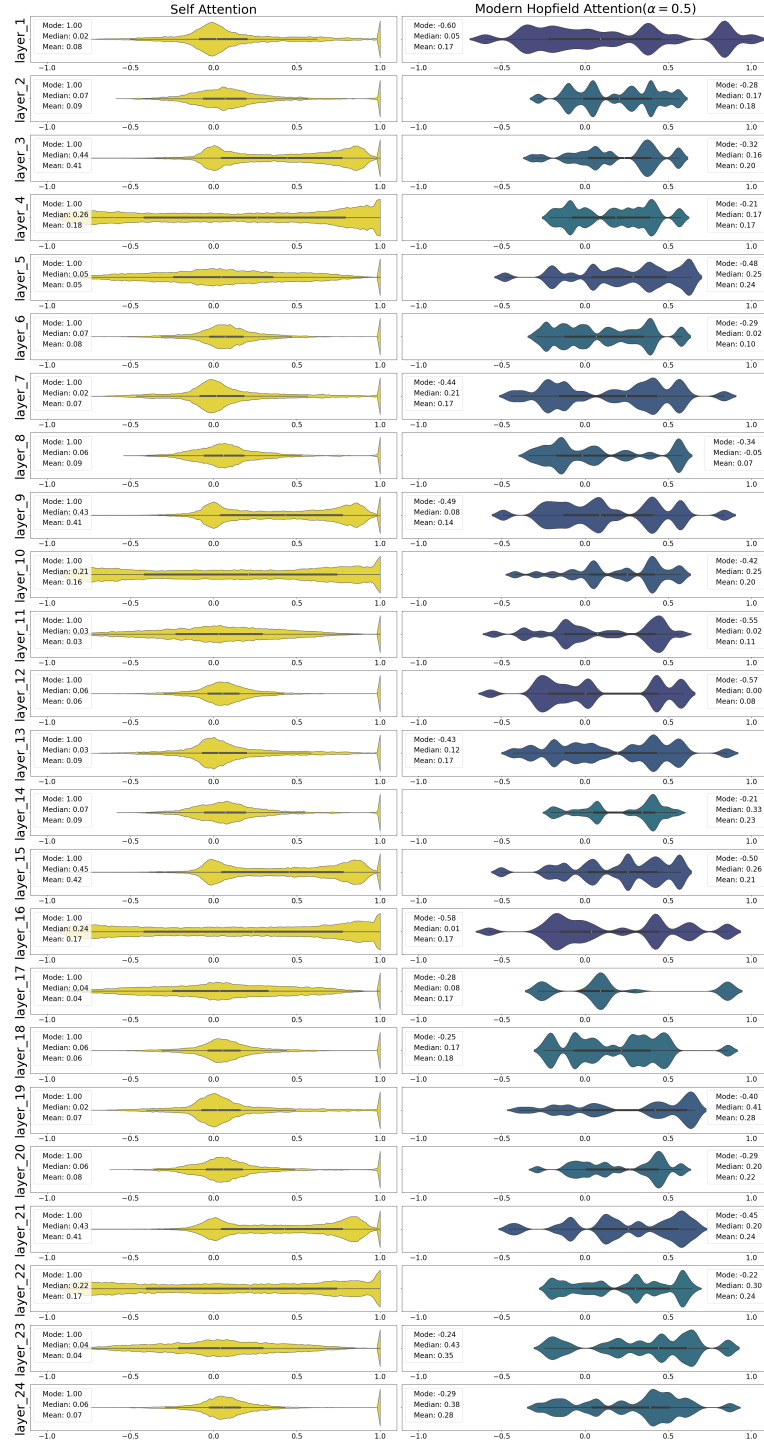


Figure 4: The violin plots of the cosine similarity of GPT-2 (Medium) with usual self-attention and MHA for $\alpha = 0.5$. The model is trained with Wikitext103. We can see that the group of perfectly aligned tokens that exists at a peak around a similarity of 1 in self-attention disappears in the MHA cases.

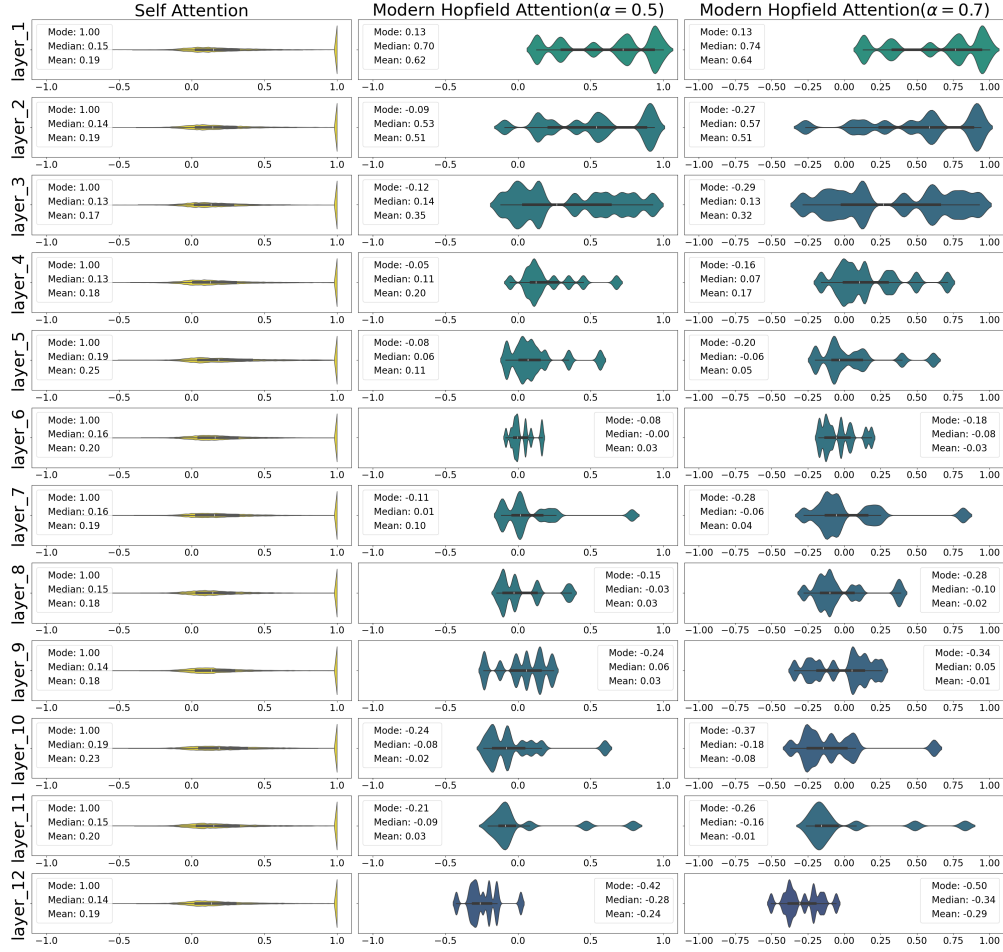


Figure 5: The violin plots of the cosine similarity of ViT-T with usual self-attention, MHA for $\alpha = 0.5$ and MHA for $\alpha = 0.7$. The model is trained with CIFAR10. We can see that the group of perfectly aligned tokens that exists at a peak around a similarity of 1 in self-attention disappears in the MHA cases.

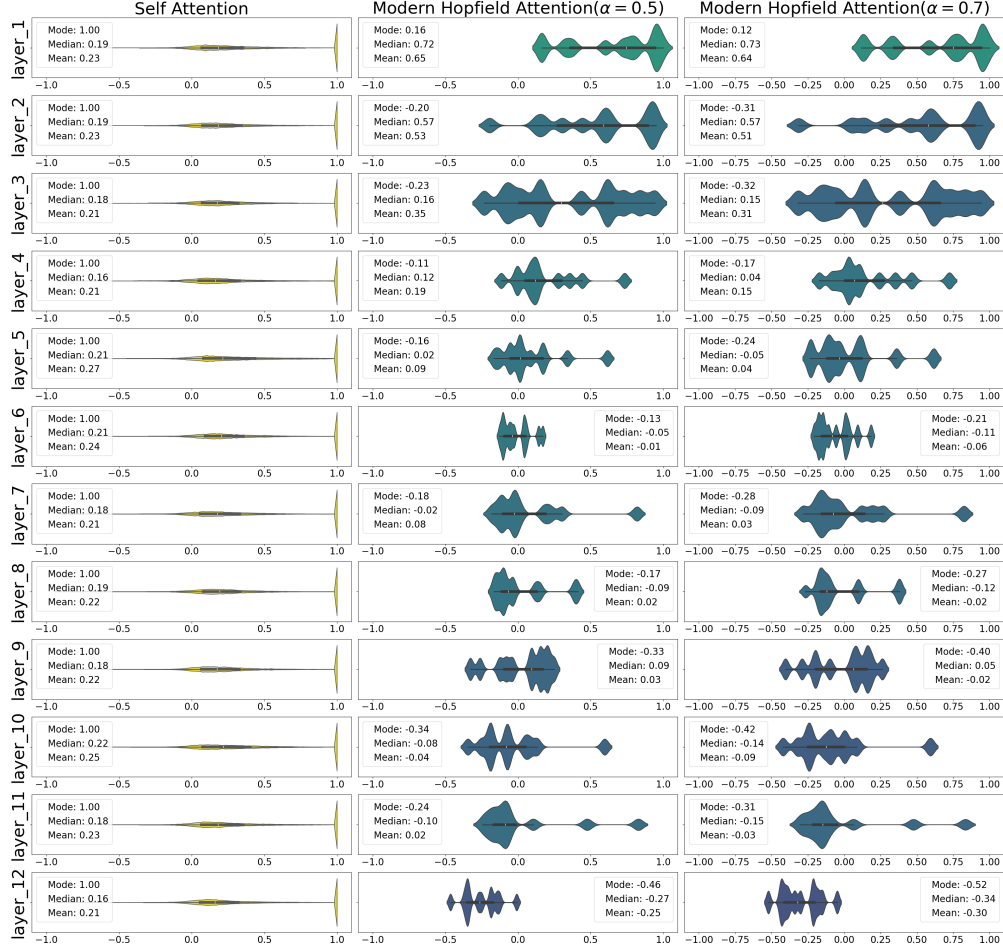


Figure 6: The violin plots of the cosine similarity of ViT-S with usual self-attention, MHA for $\alpha = 0.5$ and MHA for $\alpha = 0.7$. The model is trained with CIFAR10.

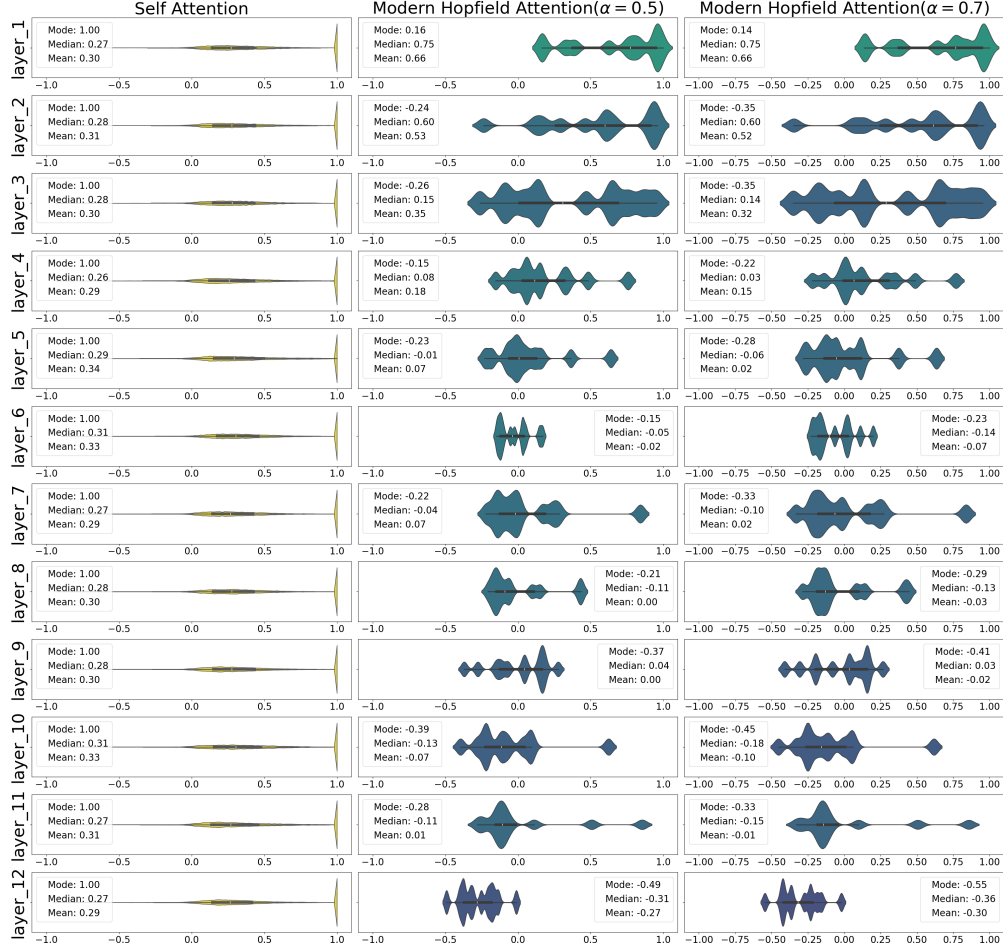


Figure 7: The violin plots of the cosine similarity of ViT-B with usual self-attention, MHA for $\alpha = 0.5$ and MHA for $\alpha = 0.7$. The model is trained with CIFAR10.

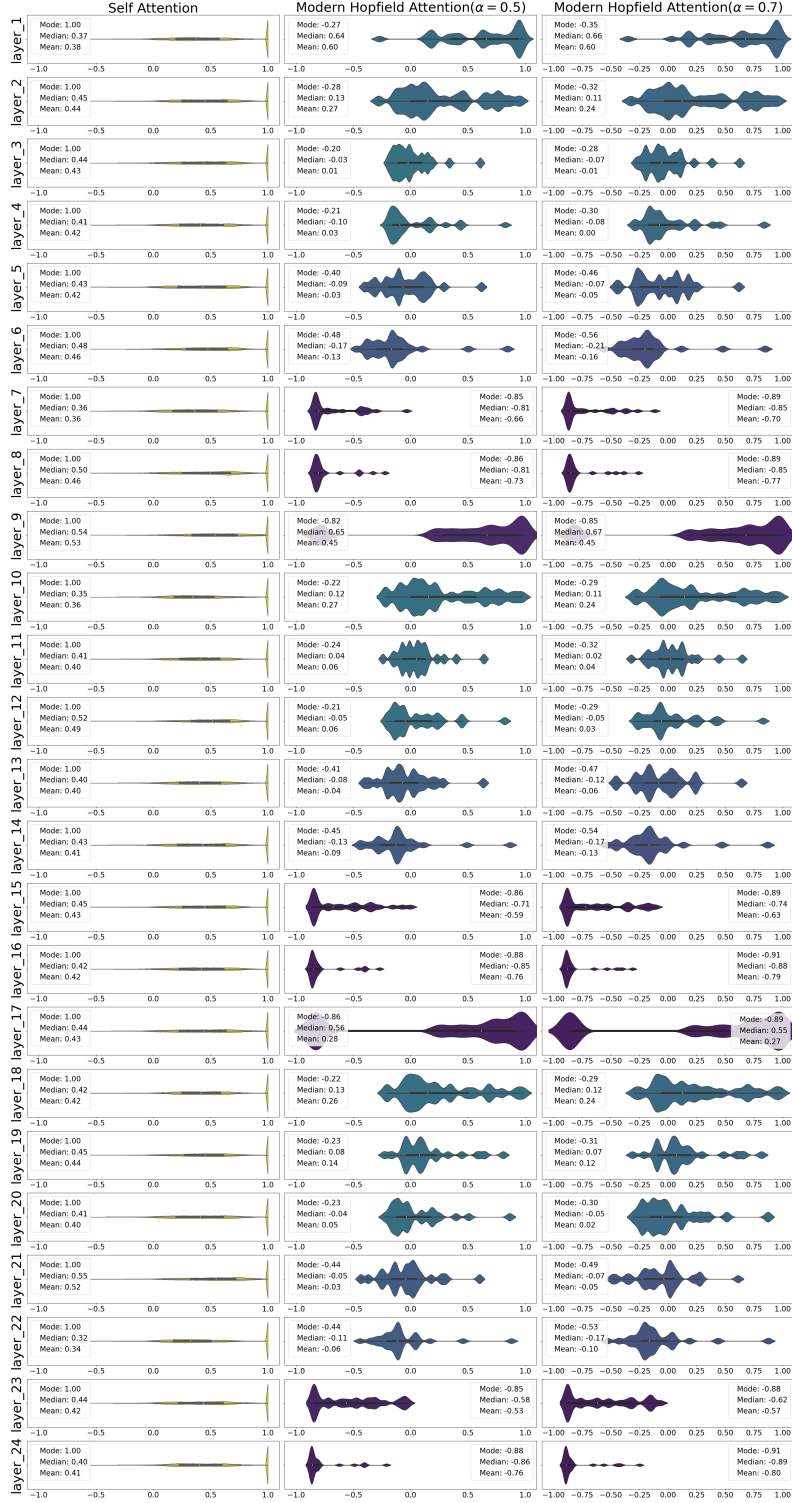


Figure 8: The violin plots of the cosine similarity of ViT-L with usual self-attention, MHA for $\alpha = 0.5$ and MHA for $\alpha = 0.7$. The model is trained with CIFAR10.



Figure 9: The violin plots of the cosine similarity of ViT-T with usual self-attention, MHA for $\alpha = 0.5$ and MHA for $\alpha = 0.7$. The model is trained with CIFAR100. We can see that the group of perfectly aligned tokens that exists at a peak around a similarity of 1 in self-attention disappears in the MHA cases.



Figure 10: The violin plots of the cosine similarity of ViT-S with usual self-attention, MHA for $\alpha = 0.5$ and MHA for $\alpha = 0.7$. The model is trained with CIFAR100.



Figure 11: The violin plots of the cosine similarity of ViT-B with usual self-attention, MHA for $\alpha = 0.5$ and MHA for $\alpha = 0.7$. The model is trained with CIFAR100.

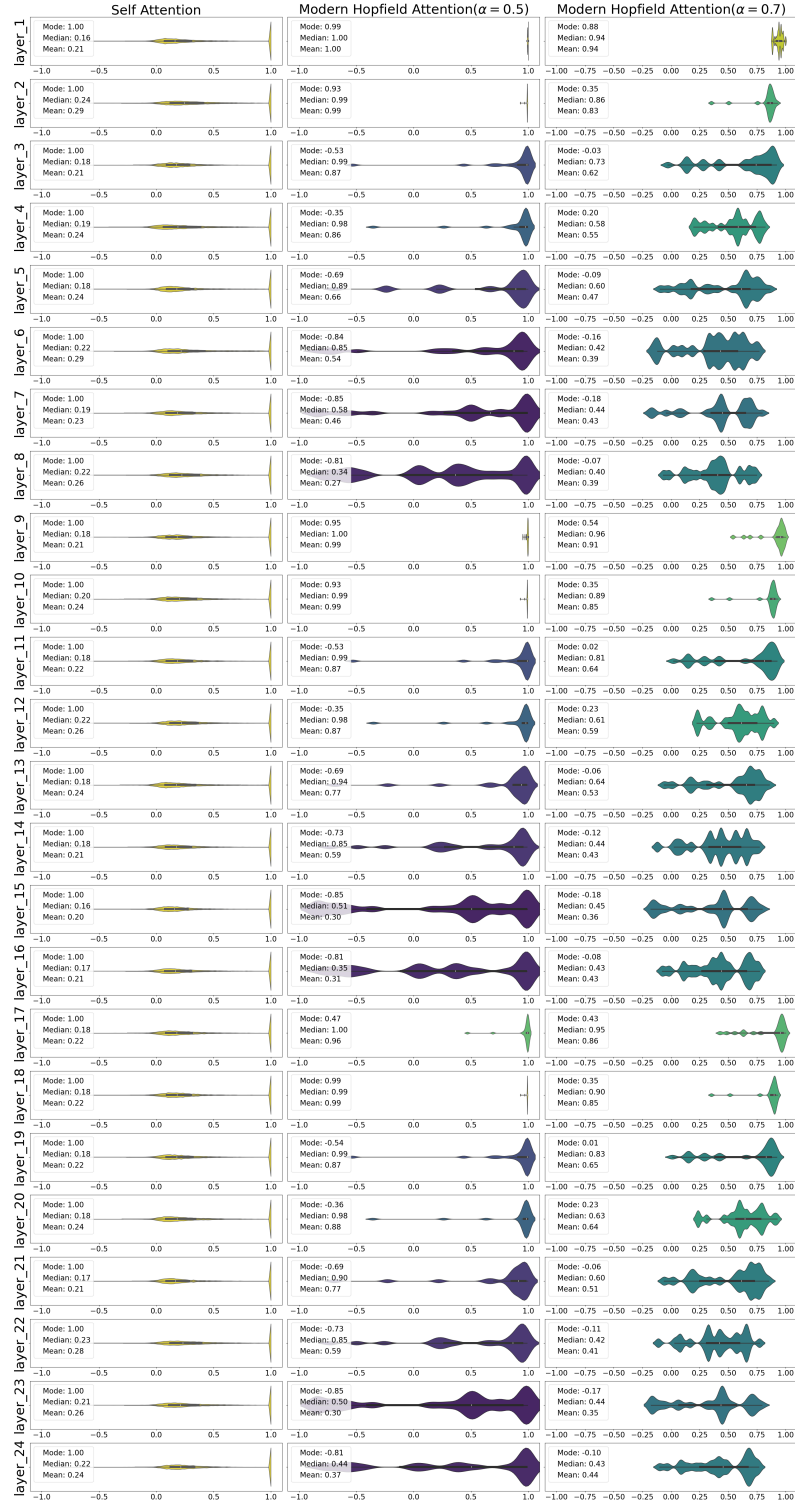


Figure 12: The violin plots of the cosine similarity of ViT-L with usual self-attention, MHA for $\alpha = 0.5$ and MHA for $\alpha = 0.7$. The model is trained with CIFAR100.

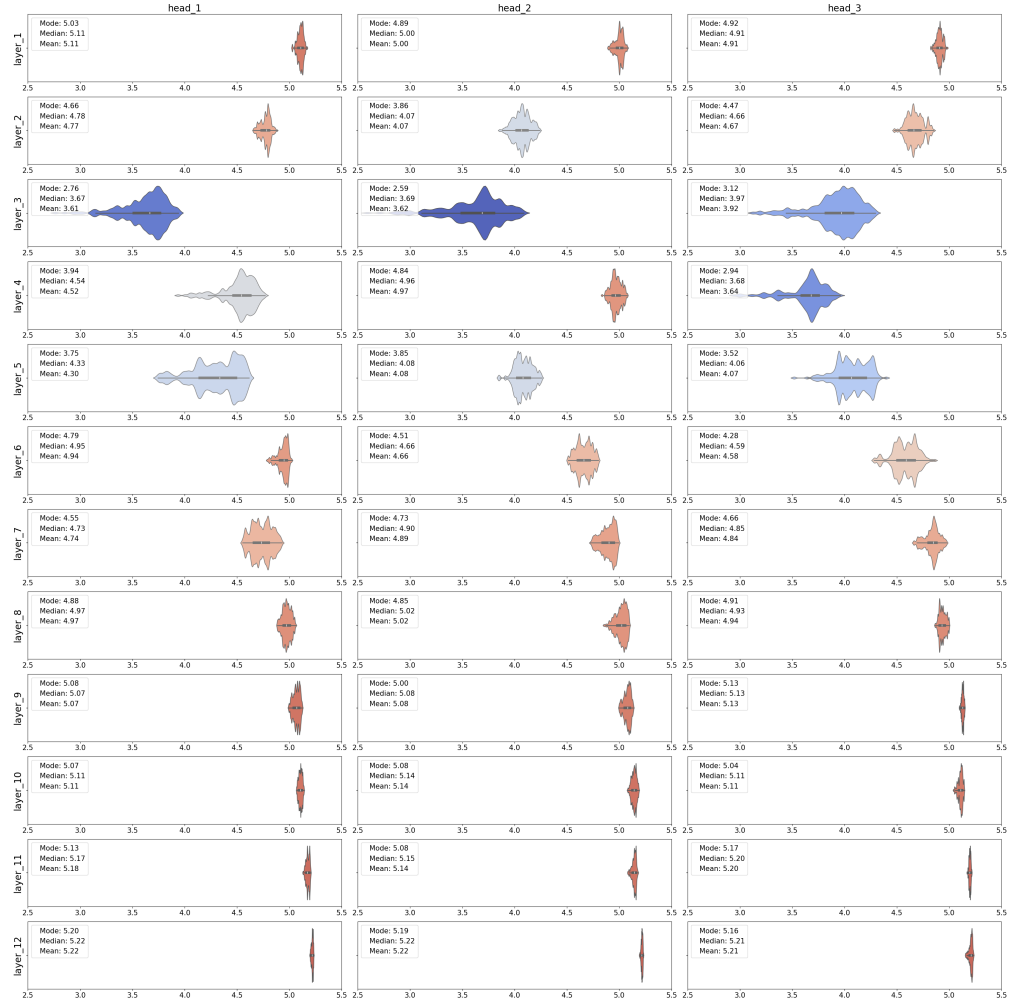


Figure 13: The violin plot of the attentional entropy for each layer and each head of ViT-T trained with CIFAR10 is shown.

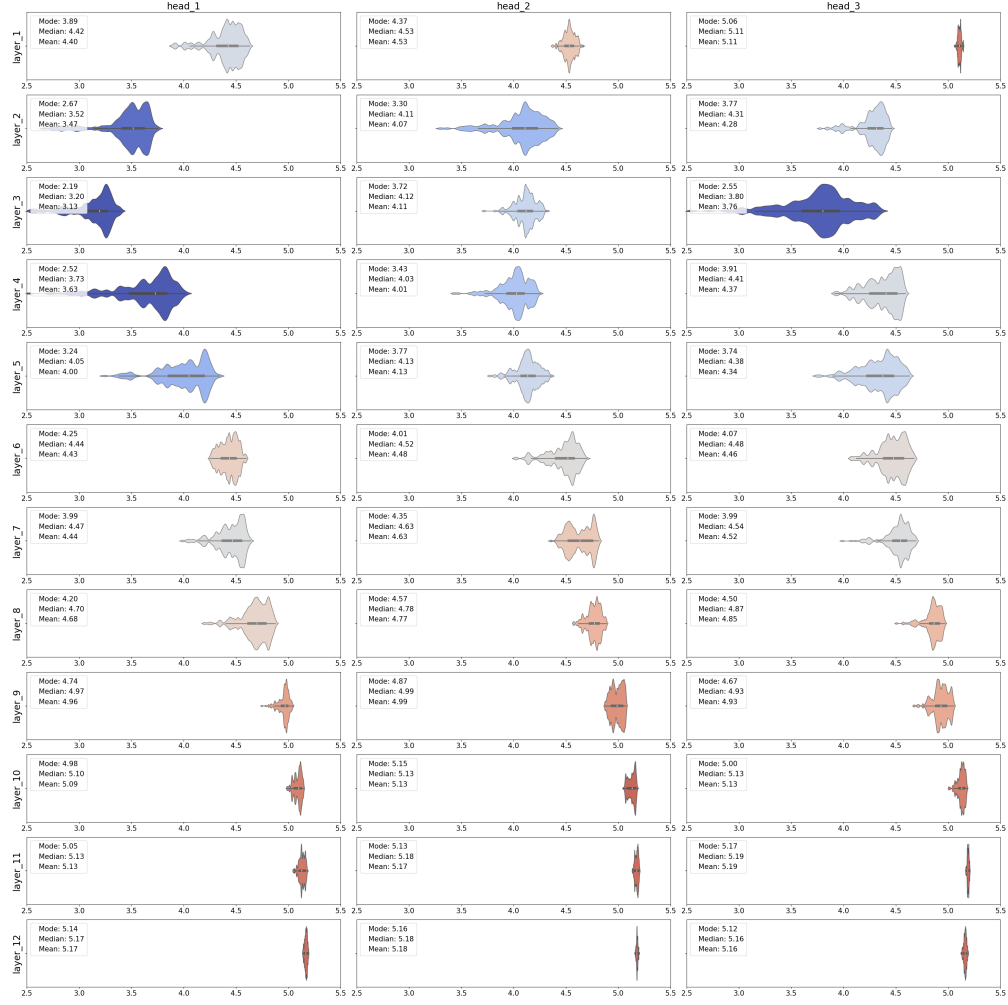


Figure 14: The violin plot of the attentional entropy for each layer and each head of MHA version of ViT-T ($\alpha = 0.5$) trained with CIFAR10 is shown.

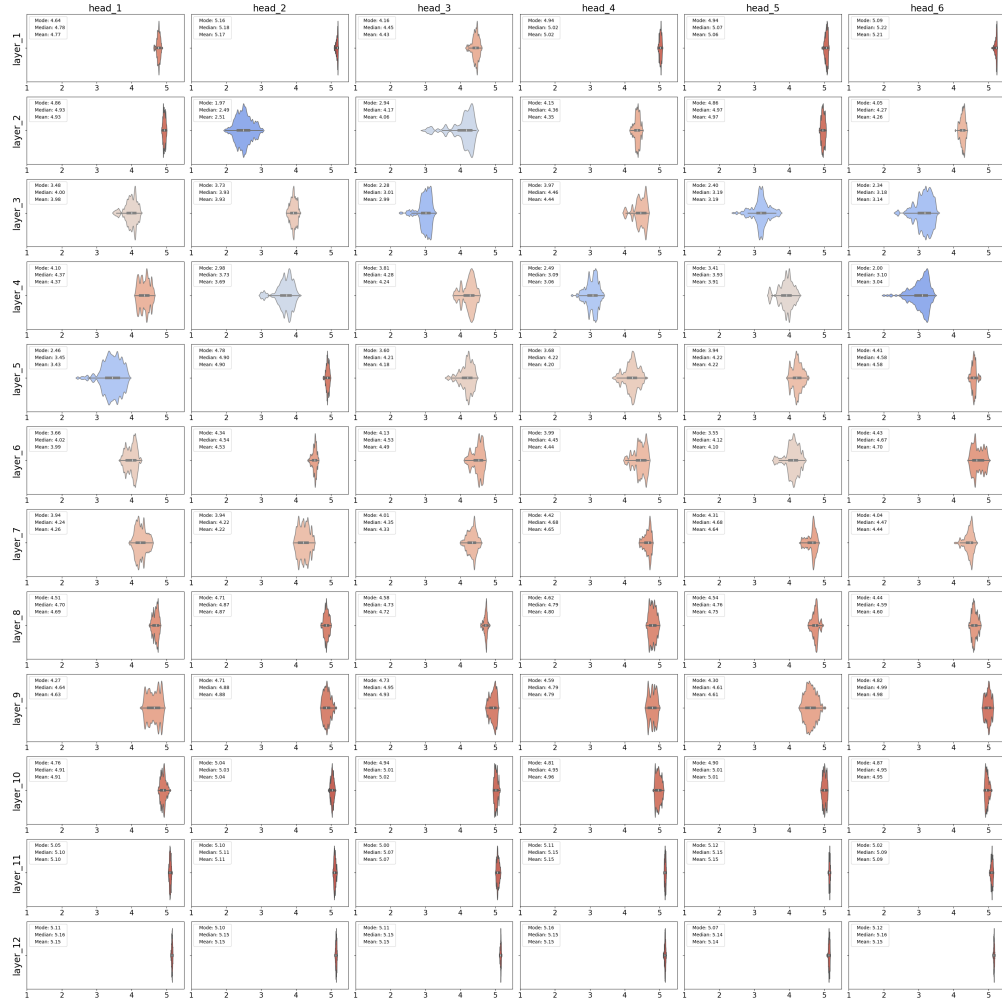


Figure 15: The violin plot of the attentional entropy for each layer and each head of ViT-S trained with CIFAR10 is shown.

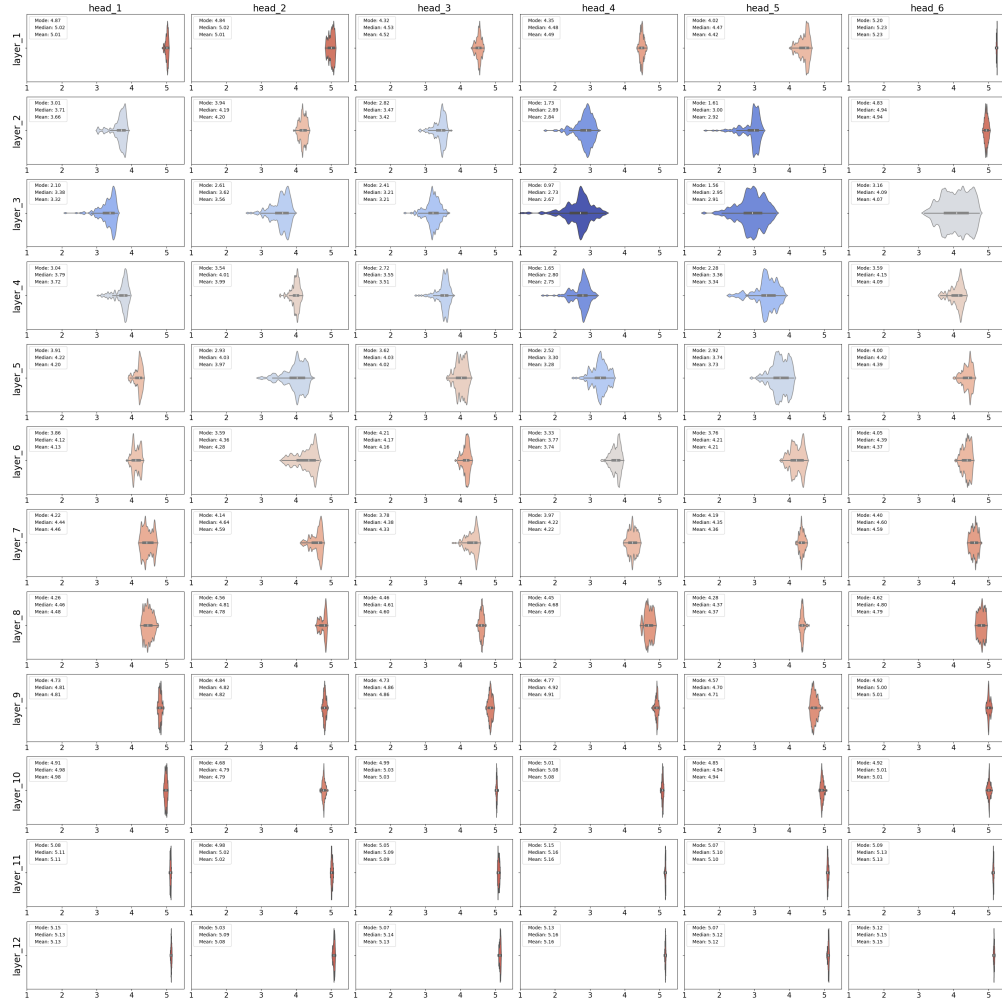


Figure 16: The violin plot of the attentional entropy for each layer and each head of MHA version of ViT-S ($\alpha = 0.5$) trained with CIFAR10 is shown.



Figure 17: The violin plot of the attentional entropy for each layer and each head of ViT-B trained with CIFAR10 is shown.

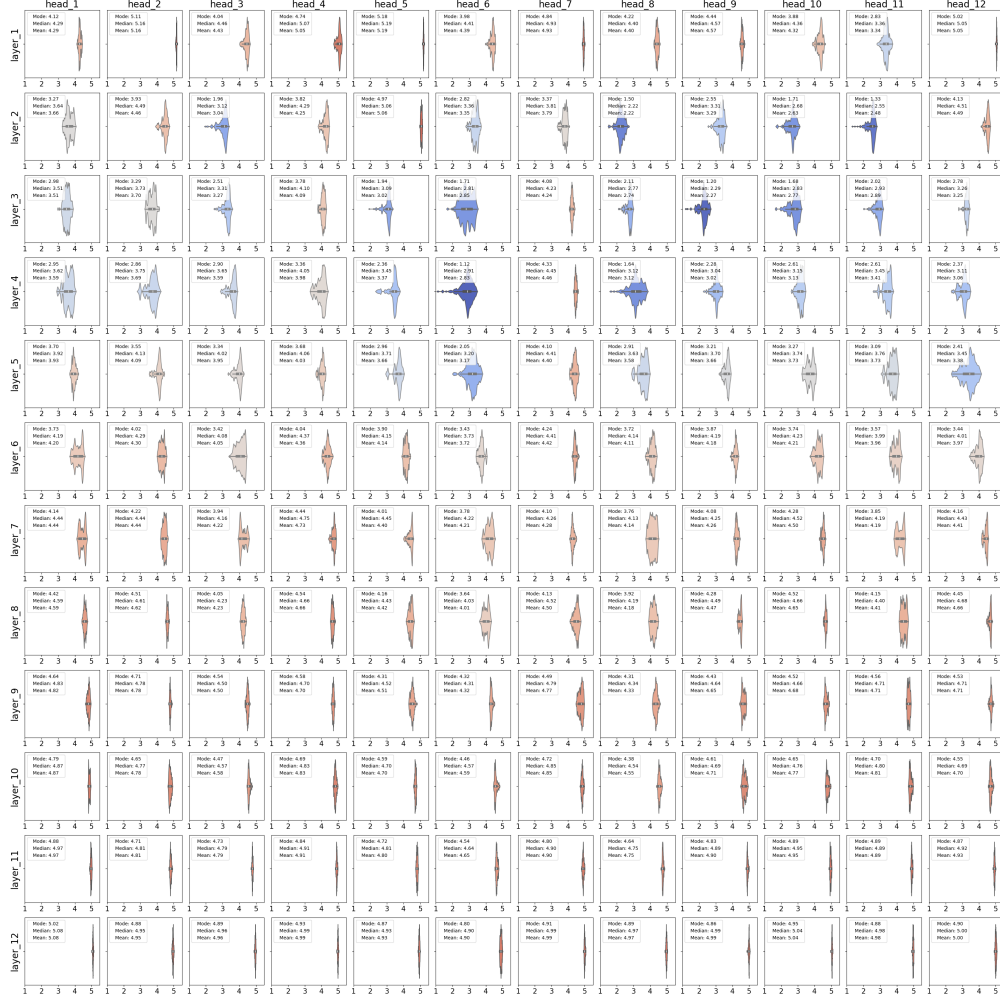


Figure 18: The violin plot of the attentional entropy for each layer and each head of MHA version of ViT-B ($\alpha = 0.5$) trained with CIFAR10 is shown.

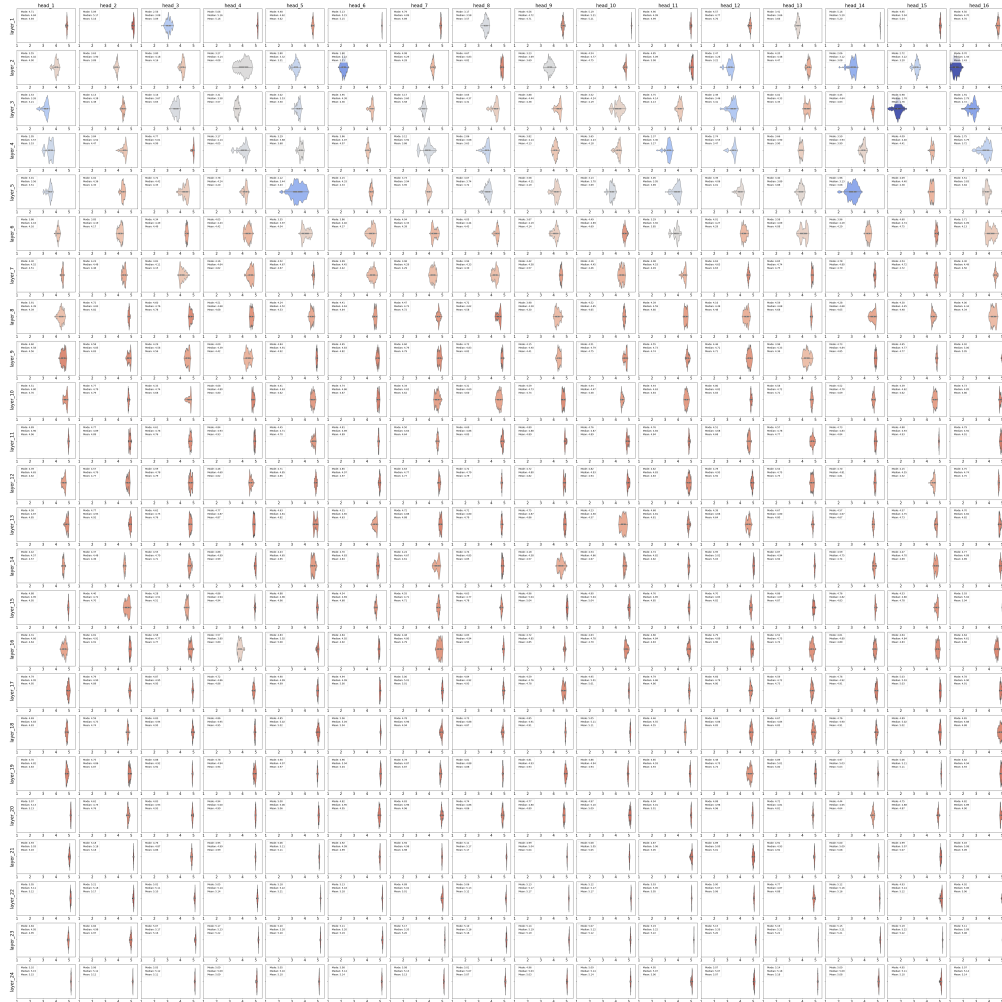


Figure 19: The violin plot of the attentional entropy for each layer and each head of ViT-L trained with CIFAR10 is shown.



Figure 20: The violin plot of the attentional entropy for each layer and each head of MHA version of ViT-L ($\alpha = 0.5$) trained with CIFAR10 is shown.