You Need to Pay Better Attention

Mehran Hosseini*
Department of Informatics
King's College London
London, UK
mehran.hosseini@kcl.ac.uk

Peyman Hosseini*
School of Electronic Engineering & Computer Science
Queen Mary University of London
London, UK
s.hosseini@qmul.ac.uk

Abstract

We introduce three new attention mechanisms that outperform standard multi-head attention in terms of efficiency and learning capabilities, thereby improving the performance and broader deployability of Transformer models. Our first contribution is *Optimised Attention*, which performs similarly to standard attention, but has 3/4 as many parameters and one matrix multiplication fewer per head. Next, we introduce *Efficient Attention*, which performs on par with standard attention with only 1/2 as many parameters as many parameters and two matrix multiplications fewer per head and is up to *twice as fast* as standard attention. Lastly, we introduce *Super Attention*, which surpasses standard attention by a significant margin in both vision and natural language processing tasks while having fewer parameters and matrix multiplications. In addition to providing rigorous mathematical comparisons, we evaluate the presented attention mechanisms on MNIST, CIFAR100, IMDB Movie Reviews, and Amazon Reviews datasets.

1 Introduction

Not many ideas have had as profound an effect on the field of *Artificial Intelligence* (*AI*) as the *attention mechanism* (Bahdanau et al., 2015). Introduced as a method to improve machine translation, the attention mechanism revolutionised the way neural networks process and interpret data. By allowing models to focus on specific parts of the input while disregarding irrelevant information, it mimics a form of cognitive attention in humans. It not only enhanced the capability and efficiency of Language Models (LM) but also paved the way for the development of advanced AI architectures like the Transformer model (Vaswani et al., 2017).

These advances have had far-reaching impacts, extending beyond Natural Language Processing (NLP) to other areas such as image recognition (Dosovitskiy et al., 2021), autonomous systems (Mott et al., 2019), and even healthcare (Choi et al., 2016), where AI can now make more nuanced and context-aware decisions.

Numerous attention mechanisms have been put forward even before the seminal paper of Bahdanau et al. (2015). Nonetheless, the standardisation of the attention mechanism put forward by Vaswani et al. (2017) remains predominant even in 2024.

"The bigger the better" has been the prevailing maxim in AI in the last few years. Larger Language Models (LLM), such as Llama 2 (Touvron et al., 2023a,b), GPT-4 (Achiam et al., 2023), and Gemini (Anil et al., 2023) have demonstrated unprecedented capabilities in NLP tasks.

However, the behemothic sizes of these models have introduced numerous challenges, such as expensive and slow training and inference, leading to secondary problems such as high carbon emission, contributing to global warming (Dhar, 2020). Furthermore, such models are impossible

*Equal contribution; ordered alphabetically.

not only to run but even to store on edge devices such as smartphones, consumer laptops, and even powerful personal workstations.

In the last few years, there have been numerous attempts to address this problem using quantisation (Jacob et al., 2018), Low-Rank Adaptation (LoRA) (Hu et al., 2022), Quantised LoRA (QLoRA) (Dettmers et al., 2023), and sparsification (Ashkboos et al., 2024).

There have also been attempts to optimise the speed and GPU utilisation of attention-based models. Notable examples include Flash Attention Dao et al. (2022) and its successor, Flash Attention 2 Dao (2024). We explain all these approaches in more detail in the related work in Section 5.

All these approaches focus on techniques to improve the performance of attention-based models without altering the attention mechanism. In this paper, we look into the attention mechanism itself and put forward three attention mechanisms, *Optimised Attention*, *Efficient Attention*, and *Super Attention*. Our contributions are founded on three observed principles:

1. Two consecutive linear transformations result in a linear transformation.
2. *Multi-Head Attention* (*MHA*) provides little to no gain compared to single head attention.
3. A linear kernel between each input increases the learning capabilities.

Using Principle 1, Optimised Attention omits the W^V kernel (see Eq. (4)), while preserving the learning capabilities of standard attention. We use Principle 1 once more to introduce Optimised Attention, which not only omits W^V but also W^K (see Eq. (5)). Optimised Attention also utilises Principal 2 to reduce the number of parameters while performing on par with standard attention in terms of learning capabilities. Finally, using Principal 3 and building on top of Efficient Attention, Super Attention introduces a new learnable kernel W^A boosting the performance of the attention mechanism in both vision and NLP tasks compared to standard attention, while being more efficient and having fewer parameters.

We validate our findings on image classification tasks on MNIST and CIFAR100 datasets as well as on text sentiment analysis on IMDB and Amazon Reviews datasets.

In summary, our contributions are as follows.

- We introduce *Optimised Attention* in Section 3.1, which
  - ◇ reduces the attention layer's size by 1/4 and its computational cost by h matrix multiplication, where h is the number of heads, thereby reducing its training and inference time by 3–10% as we show in Section 4.1,
  - ◇ performs similarly to standard attention in terms of learning capabilities as we demonstrate in Section 4.1, and
  - ◇ is equivalent to the standard multi-head attention in terms of linear rank as we show in Section 3.1.
- We introduce *Efficient Attention* in Section 3.2, which is our most efficient attention mechanism,
  - ◇ reducing the attention layer's size by 1/2 and its computational cost by $2h$ matrix multiplications, thereby reducing its training and inference time by 11–50% as we show in Section 4.1, and
  - ◇ performing as well as the standard attention in terms of loss and accuracy while being up to twice as fast as we demonstrate in Section 4.1.
- We introduce *Super Attention* in Section 3.3, which is our most capable attention mechanism,
  - ◇ reducing the attention layer's size by 1/4 and its computational cost by $2h - 1$ matrix multiplications, when the context length is equal to or smaller than the model dimension, thereby reducing the training and inference time by 4–45% as we show in Section 4.1, and
  - ◇ outperforming standard attention by 2–7% in terms of accuracy in both vision and language classification tasks.

2 Preliminaries

We introduce the notations and definitions that we will use throughout the paper in this section. For natural numbers $d_m, d_k \in \mathbb{N}$, we denote the d_m-dimensional real *vectors space* by $\mathbb{R}^{d_m}$ and the set of all real $d_m \times d_k$ *matrices* by $\mathbb{R}^{d_m \times d_k}$, noting that all matrices can be regarded as 2D *tensors*

and vice versa. Given a set $\mathcal{A} \subseteq \mathbb{R}^{d_m}$, we denote the smallest real vector space containing $\mathcal{A}$ by $\mathrm{span}(\mathcal{A})$. Similarly, given matrices for a mtrix $W \in \mathbb{R}^{d_m \times d_k}$, we denote the smallest real vectors space containing the columns of W's by $\mathrm{span}(W)$. For a *subspace* $\mathcal{S} \leq \mathbb{R}^{d_m}$, the *dimension* of $\mathcal{S}$, denoted $\dim(\mathcal{S})$, is the size of the largest *linearly independent* set in $\mathcal{S}$. The *rank* of a matrix $W \in \mathbb{R}^{d_m \times d_k}$, denoted $\mathrm{rank}(W)$, is the number of linearly independent columns (or rows) in W. The rank-nullity theorem implies that $\mathrm{rank}(W) = \dim(\mathrm{span}(W))$ and $\mathrm{rank}(W) \leq \min(d_m, d_k)$. For a more in-depth introduction on these see (Meyer, 2023, Chapters 2 & 4).

We use the definition of the attention mechanism used in the implementations of MHA in machine learning frameworks, such as Torch, JAX, TensorFlow, and Keras.

Definition 1 (Standard Attention)**.** The (*multi-head*) *attention* mechanism on *input* tensors $Q, K, V \in \mathbb{R}^{\ell \times d_m}$ is defined as

$$O = (H_1\ H_2\ \cdots\ H_h)W^O, \tag{1}$$
$$H_i = S_i V_i', \tag{2}$$
$$S_i = \mathrm{softmax}(\frac{Q_i' K_i'^{\intercal}}{\sqrt{d_k}}), \tag{3}$$
$$V_i' = V W_i^V, \tag{4}$$
$$K_i' = K W_i^K, \tag{5}$$
$$Q_i' = Q W_i^Q, \tag{6}$$

where O is the *output*; Q_i', K_i', V_i', S_i, and H_i are the *query*, *key*, *value*, *attention score*, and *head value* of the i-th *head*, respectively. The natural numbers ℓ, d_m and h are the *context length*, *model dimension*, and *number of heads*, respectively. Moreover, $W_i^Q, W_i^K \in \mathbb{R}^{d_m \times d_k}$ and $W_i^V \in \mathbb{R}^{d_m \times d_v}$, where d_k and d_v are the *key* and *value dimensions*, respectively.

Parameters d_m, d_k, d_v and h are often chosen so that $d_k = d_v = d_m/h$, and in most recent models, including tranformer models, Q, K, and V are set to X, a single input tensor; whereby, the attention mechanism is called *self-attention*.

We use the notation used in Definition 1 throughout the paper; in particular in Definitions 2–4.

3 Revising the Attention Mechanism

We delve into the mathematical underpinnings of the attention mechanism and present enhanced attention mechanisms that are *more efficient* (in terms of *number of parameters* and *computation cost*) and *more potent* (in terms of attaining *higher accuracies* and *lower losses*).

Specifically, we introduce *Optimised Attention* in Section 3.1, *Efficient Attention* in Section 3.2, and *Super Attention* in Section 3.3. We provide a detailed mathematical analysis of each of them in their corresponding sections. We evaluate all mechanisms in Section 4.1.

3.1 Optimised Attention: Absorbing W_i^V's into W^0

We start by optimising operations (1) and (4) of the attention mechanism. We do this by absorbing $W_1^V, W_2^V, \ldots, W_h^V$ into W^O, thereby reducing the computational cost of the attention layer by h matrix multiplications without significantly affecting the performance as we prove in Section 4.

In standard attention, the output O can be written as

$$\begin{aligned} O = (H_1\ H_2\ \cdots\ H_h)W^O &= (S_1 V W_1^V\ S_2 V W_2^V\ \cdots\ S_h V W_h^V) \begin{pmatrix} W_1^O \\ W_2^O \\ \vdots \\ W_h^O \end{pmatrix} \\ &= S_1 V W_1^V W_1^O + S_2 V W_2^V W_2^O + \cdots + S_h V W_h^V W_h^O, \end{aligned} \tag{7}$$

where W_i^O is the matrix that contains rows $(i-1)d_v+1,\ldots,id_v$ of W^O for $i=1,2,\ldots,h$. By the rank-nullity theorem, for each head, we have that

$$\begin{aligned}\dim(\text{span}(VW_i^VW_i^O)) &= \text{rank}(VW_i^VW_i^O) \leq \text{rank}(W_i^VW_i^O),\\ &\leq \min(\text{rank}(W_i^V),\text{rank}(W_i^O)) = \min(d_m,d_v) = d_v.\end{aligned}$$

In other words, $VW_i^VW_i^O$ has at most d_v independent columns, and the linear function $V \mapsto VW_i^VW_i^O$ maps the columns of V into a d_v-dimensional subspace of $\mathbb{R}^{d_m}$.

Thus, standard attention uses two matrix consecutive multiplication to embed the columns of V into a d_v-dimensional subspace of $\mathbb{R}^{d_m}$, which is inefficient according to Principal 1, which we validate in Section 4. In Optimised attention, we achieve the same effect by one slicing and one matrix multiplication, thereby reducing the computational cost of attention during training and inference.

In more details, we propose that instead of multiplying V from the right by W_i^V, to slice V into $V_1,\ldots,V_h$, where V_i consists of columns $(i-1)d_v+1,\ldots,id_v$ of V. Then, in the attention mechanism, instead of computing $S_iVW_i^VW_i^O$, we compute $S_iV_iW_i^O$, which has fewer parameters and matrix multiplications (see Remark 1). We refer to this optimised attention mechanism as Optimised Attention. As we show in Section 4, Optimised Attention considerably improves the efficiency of the attention layer without affecting the model's performance.

Definition 2 (Optimised Attention)**.** Using the notation of Definition 1, *Optimised Attention* is the attention mechanism defined by the following set of equations:

$$O = (H_1,H_2,\ldots,H_h)W^O, \tag{8}$$

$$H_i = S_iV_i, \tag{9}$$

$$S_i = \text{softmax}(\frac{Q_i'K_i'^\intercal}{\sqrt{d_k}}), \tag{10}$$

$$K_i' = KW_i^K, \tag{11}$$

$$Q_i' = QW_i^Q. \tag{12}$$

Remark 1. Optimised Attention is more efficient than standard attention in the sense that it has h matrix multiplication and d_m^2 parameters fewer than standard attention.

Proof. Compared to Optimised Attention, standard attention has extra $W_1^V,W_2^V,\ldots,W_h^V$, which are multiplied from the right to V, amounting to a total of $d_md_vh = d_m^2$ parameters and h matrix multiplications. □

3.2 Efficient Attention: Absorbing W^K into W^Q

We now turn our focus to the attention scores S_i in Eq. (3). Let us denote the pre-softmax scores by

$$A_i = \frac{QW_i^QW^{K\intercal}{}_i^\intercal K^\intercal}{d_k}, \tag{13}$$

so that $S_i = \text{softmax}(A_i)$. Let $W_i^{QK} = W_i^QW^{K\intercal}{}_i^\intercal$. By the rank-nullity theorem, we have that

$$\text{rank}(W_i^{QK}) = \min(\text{rank}(W_i^Q),\text{rank}(W_i^K)) \leq \min(d_m,d_k) = d_k, \tag{14}$$

because $W_i^Q,W_i^K \in \mathbb{R}^{d_m\times d_k}$. In other words, $\text{rank}(W_i^{QK}) \leq d_k$ even though $W_i^{QK} \in \mathbb{R}^{d_m\times d_m}$. In turn, this implies that $\text{rank}(A_i) \leq d_k$ for $i=1,2,\ldots,h$. Thus, most rows (and columns) in W_i^{QK} and A_i are *linearly dependent* (except at most d_k of them), which is less than ideal. Therefore, the combined rank of all A_i's from different heads is at most hd_k. Since h and d_k are often chosen such that $d_k = d_m/h$, the overall combined rank from all heads is at most d_m, which is what one would ideally obtain from a single $W^{QK} \in \mathbb{R}^{d_m\times d_m}$ instead of h matrices.

To address these, we introduce *Efficient Attention*. Efficient Attention builds on top of Optimised Attention and optimises it even further as follows. First, we apply Principle 1 to Eq. (14) and replace $W_i^QW^{K\intercal}{}_i^\intercal$ with a single $\hat{W}_i^Q \in \mathbb{R}^{d_m\times d_m}$. This has two advantages: **(i)** reduces the number of matrix

multiplications required and **(ii)** allows A_i's to have full d_m rank. However, this also increases the layer size when the number of heads is greater than 2 as we are replacing $2d_m d_k$ parameters in $W_i^Q, W_i^k \in \mathbb{R}^{d_m \times d_k}$ by $d_m^2 = h d_m d_k$ parameters of $\hat{W}_i^Q$.

To prevent the increase in size, we apply Principle 2 and limit the number of heads to one. As we demonstrate in Section 4.1, the models with single-head Efficient Attention perform on par with the model using multi-head standard attention while being significantly faster and smaller.

Definition 3 (Efficient Attention)**.** Using the notation of Definition 2, *Efficient Attention* is the attention mechanism defined by the following set of equations:

$$O = HW^O, \tag{15}$$
$$H = SV, \tag{16}$$
$$S = \text{softmax}(\frac{Q'K^\intercal}{\sqrt{d_k}}), \tag{17}$$
$$Q' = QW^Q. \tag{18}$$

Remark 2. Efficient Attention is more efficient than Optimised Attention and standard attention in the sense that it has h matrix multiplication and d_m^2 parameters fewer than Optimised Attention and $2h$ multiplication and $d_m(d_v h + d_m)$ parameters fewer than standard attention.

Proof. In Efficient Attention, we replace all $W_i^Q W^{K^\intercal}_i$'s with a single $W^Q \in \mathbb{R}^{d_m \times d_m}$. Therefore, we have reduced the number of matrix multiplications by h, thereby improving the training and inference time of the model. We have also reduced the model size as W^A has d_m^2 parameters, while $W_1^K, W_2^K, \ldots, W_h^K$ and $W_1^Q, W_2^Q, \ldots, W_h^Q$ have a total of $2hd_m d_k$ parameters, which based on the common choices of h and d_k in practice, amounts to $2d_m^2$ parameters. From this and Remark 1, it follows that Efficient Attention has $h + h = 2h$ matrix multiplication and $d_m^2 + d_m^2 = 2d_m$ parameters fewer than standard attention. □

Efficient Attention reduces both the size and computational cost of the model, while preserving the overall rank of pre-softmax scores. More concretely, for given query Q and key K, if we denote the corresponding pre-softmax scores in Efficient Attention by A and in standard attention by $A_1, A_2 \ldots, A_h$, it follows from Equations (17–18) that

$$\begin{aligned} \max_A(\dim(\text{span}(A))) &= \max_{W^Q}(\min(\text{rank}(Q), \text{rank}(W^Q), \text{rank}(K))) \\ &= \min(\text{rank}(Q), d_m, \text{rank}(K)). \end{aligned} \tag{19}$$

and from Equations (3) and (5–6) that

$$\begin{aligned} \max_{A_1,\ldots,A_h}(\dim(\text{span}(\bigcup_{i=1}^h A_i))) &= \max_{W^Q}(\min(\text{rank}(Q), \dim(\text{span}(\bigcup_{i=1}^h W_i^Q W_i^{K^\intercal})), \text{rank}(K))) \\ &= \min(\text{rank}(Q), hd_k, \text{rank}(K)). \end{aligned} \tag{20}$$

From Equations (19–20) and the fact that $hd_k = d_m$, we conclude that

$$\max_A(\dim(\text{span}(A))) = \max_{A_1,\ldots,A_h}(\dim(\text{span}(\bigcup_{i=1}^h A_i))) \tag{21}$$

for all queries and keys.

In other words, Eq. (21) tells us that the amount of linearly independent information in $A_1, A_2, \ldots, A_h$ (from h-head standard attention) is equivalent to the amount of linearly independent information in A (from single head Efficient Attention). In Section 4.1, we study the effect of this in practice by showing that single-head efficient attention performs about the same, and sometime better, compared to multi-head standard attention while being significantly faster and smaller.

3.3 Super Attention: Introducing W^A

In standard attention, all of the inputs Q, K, and V undergo linear transformations via multiplication by their corresponding kernels from the right, as described in Equations (4–6). As we discussed in Section 3.1, this is redundant for V as V is consecutively multiplied from the right by W^V and W^O. Thus, following Principal 1, we omit one of them. We also discussed in Section 3.2, how we can omit W^K as after transposing $K' = KW^K$, key and query kernels end up next to each other (see Eq. (13)), and thus, we can omit one of them.

All three attention mechanisms, we discussed so far, have a learnable linear kernel between Q and $K^\intercal$ but not between $K^\intercal$ and V. To better see this, let us write the equation for one of the attention mechanisms discussed so far, e.g., Efficient Attention by combining Equations (15–18):

$$O = \text{softmax}(\frac{QW^Q K^\intercal}{d_m})VW^O. \tag{22}$$

As we see, there are no learnable parameters in between $K^\intercal$ and V, connecting the two. The intuition behind directly multiplying V by the attention scores S is that the attention scores indicate how much "attention" should be paid to each of the velues in V.

Despite the intuition, this results in loss of performance as evident in Section 4.1. We use Principal 3 to address this by introducing a new attention mechanism in Definition 4 with an additional learnable kernel W^A which comes in between S and V. The values V are then multiplied by W^A from the left (see Eq. (26)), aligning and mixing the values before the attention score are applied to them.

Definition 4 (Super Attention)**.** Using the notation of Definition 3, *Super Attention* is the attention mechanism defined by the following set of equations:

$$O = HW^O, \tag{23}$$
$$H = SV', \tag{24}$$
$$S = \text{softmax}(\frac{Q'K^\intercal}{\sqrt{d_k}}), \tag{25}$$
$$V' = W^A V, \tag{26}$$
$$Q' = QW^Q, \tag{27}$$

where $W^A \in \mathbb{R}^{\ell \times \ell}$ is the *alignment kernel*, which vertically (i.e., for values corresponding to different tokens) aligns and mixes the values before the attention scores are applied to them.

Remark 3. Super Attention is more efficient than standard attention whenever the model dimension d_m is greater than or equal to the context length ℓ. This means that Super Attention has at least $2h - 1$ matrix multiplication and d_m^2 parameters fewer than standard attention.

Proof. Looking at the Equations (15–18) and (23–27), we observe that Super Attention and Efficient Attention have the same defining equations, except that Super Attention has an the additional linear transformation in Eq. (26), where V is multiplied by $W^A \in \mathbb{R}^{\ell \times \ell}$. This amounts to a total of ℓ^2 additional parameters and one matrix multiplication.

By Remark 2, Efficient Attention has $2h$ multiplication and $2d_m^2$ parameters fewer than standard attention. Therefore, Super Attention has $2h - 1$ matrix multiplication and $2d_m^2 - \ell^2$ parameters fewer than standard attention. Since $\ell \leq d_m$, we have that $2d_m^2 - \ell^2 \geq d_m^2$. Thus Super Attention has d_m^2 fewer parameters than standard attention. □

To better understand Super Attention, let us write its complete equation. By combining Equations (23–27), we have that

$$O = \text{softmax}(\frac{QW^Q K^\intercal}{d_m})W^A VW^O. \tag{28}$$

In Eq. (28), W^A comes in between the attention scores S and values V, aligning and mixing the values (tokenwise) before the attention scores are applied to them. As we show next, this results in a far better learning performance compared to the other attention mechanisms.

4 Evaluation

We evaluate all the attention mechanisms discussed here in vision and natural language applications. We have chosen classification tasks in both domains for two reasons. First, our limited computing resource of one Nvidia RTX 4090 GPU. Second, classification tasks provide clear comparison metrics like accuracy. For the evaluation, we train Transform models using each attention mechanism, discussed here, until the learning curves flatten. To ensure the reliability, we report results averaged over five training runs. We then evaluate the performance of all the attention mechanisms, in terms of loss and accuracy, in image classification on MNIST (LeCun et al., 2010) and CIFAR100 (Krizhevsky, 2009) datasets and text sentiment analysis on IMDB Movie Reviews (Maas et al., 2011) and Amazon Reviews (Ni et al., 2019) datasets. We have chosen these datasets as they each introduce different challenges because of varying dataset sizes, input sizes, and number of classes.

Additionally, we analyse the performance of each attention mechanism on an edge device to demonstrate how our contribution can be used for wider deployability of AI models on user devices. To this end, we compare the inference speed for all Transformer models on each task in Section 4.1.4. Our results indicate that the Transformer models using Efficient and Super Attention are around 25–45% faster than their standard counterparts on a device with limited resources while being on par or better.

4.1 Performance Comparison

We compare the proposed attention mechanisms against standard attention in this section. In all experiments, all attention mechanisms except standard and Optimised Attention use a single head. There are two reasons why we use a single head for the rest of attention mechanisms. First, we have found that using multiple heads provides us with little extra gain in most cases. This is even the case for standard attention as evident in (Vaswani et al., 2017, Table 3); nonetheless, we have varied the number of heads for standard and Optimised attention in Tables 1 to 4, to further showcase this. Remember that we also provided the intuition as to why this is the case in Section 3.2. Second, except for Optimised and standard Attention, the model sizes increase by the number of heads as in the other models as $W^Q \in \mathbb{R}^{d_m \times d_m}$ is always a square matrix (see Definitions 3 and 4).

Experimental Setup. We have implemented all experiments in Keras with the JAX backend using the examples provided in `keras.io/examples` with minor dataset-specific adjustments, e.g., modifying the number of classes, layers, etc. The reported results in all experiments are obtained by averaging the results over 5 runs. Where relevant, we have included 95% Confidence Intervals (CI). While we report the results for standard and Optimised attention for varying number of heads, we consider 4 heads as the comparison benchmark against the others.

4.1.1 Ablation Study on Number of Heads

In practice, Transformer (as well as other attention-based) models are implemented using standard multi-head attention. In (Vaswani et al., 2017), the authors suggest that using multiple heads could lead to learning richer representations and ultimately better performance. Since increasing the number of heads does not increase the number of parameters in standard and Optimised attention, we conduct ablation studies on the number of heads for both these mechanism. However, for Efficient and Super attention, we always use a single head.

The results, detailed in Tables 1 to 4, indicate that increasing the number of attention heads increases the training time across all tested models. Specifically, in computer vision tasks, increasing the number of heads from 1 to 4 (6 for CIFAR-100) leads to a training time surge of 1–4% and 1–3% in standard and Optimised attention models, respectively. In natural language tasks, these number are 11–50% for standard attention and 8–59% for Optimised Attention. As showcased in Table 5, at inference time on an edge device, increasing the number of heads increases the inference time 30–55% and 29–51% for standard and Optimised attention models respectively.

For other performance metrics like train/test accuracy and loss, Tables 1 to 4 show that increasing the number of heads increases the computational cost of training the models but does not yield a significant, if any, boost in performance.

4.1.2 Vision Transformers

Vision Transformers are increasingly adopted across computer vision. As such, we evaluate the proposed mechanisms, for use in ViT, on two widely used image classification datasets, MNIST (LeCun et al., 2010) and CIFAR100 (Krizhevsky, 2009).

MNIST. We trained ViT models with different attention mechanisms, all with two attention layers and model dimension $d_m = 64$. As expected, Super Attention outperforms all other architectures, in terms of accuracy, by at least 5.7% and standard attention by 6.6%. The smallest attention layer size belongs to Efficient Attention, which performs on par with standard attention. The complete results are presented in Table 1.

Table 1: Averages of different metrics over five runs in the MNIST experiment. The numbers in parentheses indicate the ranking of each mechanism for that metric. Ablation studies on the number of heads for standard and Optimised attention models show that increasing the number of heads does not meaningfully affect performance. As expected, the Efficient Attention model has the smallest attention layer size and the Super Attention model performs the best in terms of accuracy and loss.

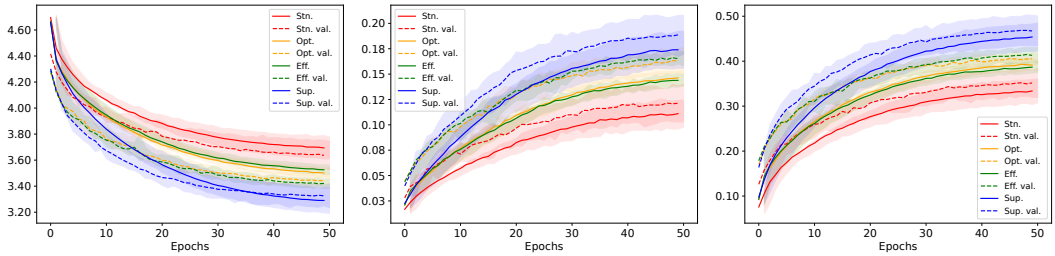
Att.	h	d_m	d_k	# Param.	Avg. Time (s)	Acc. (%)	Loss	Test Acc. (%)	Test. Loss
Stn.	1	64	64	16,640	40.33	71.7	0.83	89.6	0.41
	2	64	32	16,640	40.43	69.5	0.86	87.5	0.43
	4	64	16	16,640 (4)	40.84 (4)	73.0 (3)	0.79 (3)	88.5 (2)	0.39 (3)
Opt.	1	64	64	12,480	38.25	70.0	0.87	86.4	0.51
	2	64	32	12,480	38.28	74.3	0.78	88.7	0.39
	4	64	16	12,480 (2)	38.57 (2)	71.0 (4)	0.82 (4)	87.6 (4)	0.43 (4)
Eff.	1	64	64	**8,320 (1)**	**36.48 (1)**	73.9 (2)	0.75 (2)	88.2 (3)	0.36 (2)
Sup.	1	64	64	12,480 (2)	39.34 (3)	**79.6 (1)**	**0.59 (1)**	**90.0 (1)**	**0.31 (1)**

CIFAR100. Classifying CIFAR100 images presents considerable difficulty due to the large number of classes in the dataset. This complexity necessitates the maximal utilisation of the attention layers, thereby presenting the perfect challenge for comparing the attention mechanisms discussed here. We trained ViT models with eight attention layers, each with $d_m = 144$. As presented in Table 2, the Super Attention model surpasses all other architectures achieving 45.4% top-5 accuracy as opposed to standard attention with 33.4% top-5 accuracy. The Efficient Attention model has the smallest attention layer size, only half of that of the standard attention model.

For further insight, we have provided the accuracy and validation accuracy curves in Fig. 1. We have also included the results for varying numbers of heads in the standard attention model in Table 2.

Table 2: Averages of different metrics over five runs in the CIFAR100 experiment. The numbers in parentheses indicate the ranking of each mechanism for that metric. Ablation studies on the number of heads for standard and Optimised attention models show that increasing the number of heads does not meaningfully affect performance. As expected, the Efficient Attention model has the smallest attention layer size and the Super Attention model performs the best in terms of accuracy and loss.

Att.	h	d_m	d_k	# Param.	Avg. Time	Acc.	Loss	Top 5	Test Acc.	Test Loss	Test Top 5
Stn.	1	144	144	83,520	113.48	12.5	3.64	35.8	15.3	3.52	40.3
	2	144	72	83,520	116.16	12.2	3.65	35.2	14.6	3.54	39.4
	4	144	36	83,520 (4)	115.94 (4)	11.1 (4)	3.69 (4)	33.4 (4)	12.5 (4)	3.64 (4)	36.0 (4)
	6	144	24	83,520	118.27	13.3	3.58	37.1	15.6	3.49	40.6
Opt.	1	144	144	62,640	107.08	14.4	3.54	38.9	17.2	3.43	43.2
	2	144	72	62,640	107.41	14.9	3.50	39.6	17.5	3.41	43.5
	4	144	36	62,640 (2)	107.94 (2)	14.6 (2)	3.50 (2)	39.1 (2)	16.3 (3)	3.45 (3)	41.7 (3)
	6	144	24	62,640	109.82	14.6	3.49	39.5	16.4	3.45	41.7
Eff.	1	144	144	**41,760 (1)**	**100.15 (1)**	14.4 (3)	3.52 (3)	38.7 (3)	16.7 (2)	3.44 (2)	42.6 (2)
Sup.	1	144	144	62,640 (2)	110.97 (3)	**17.4 (1)**	**3.29 (1)**	**45.4 (1)**	**19.4 (1)**	**3.29 (1)**	**47.6 (1)**



(a) Categorical Cross Entropy Loss (b) Accuracy (c) Top 5 Accuracy

Figure 1: Average and 95% CI of train/validation loss, accuracy, and top 5 accuracy of the models using each attention mechanism over 50 training epochs on CIFAR100 dataset.

4.1.3 Natural Language Processing

Now, we evaluate the attention mechanisms introduced here in Transformer models of different sizes for sentiment analysis on IMDB and Amazon Reviews datasets. Similarly to Section 4.1.2, the Transformer models using Efficient Attention results in the smallest models and Super Attention achieves the highest performance. The differences in performance are more pronounced in the more challenging Amazon Reviews dataset as presented in Tables 3 and 4.

IMDB. The IMDB dataset includes 50,000 reviews with binary labels, indicating negative and positive sentiments. The Transformer models, used in this experiment, all have a single attention layer with model dimension and context length 32. The complete results are presented in Table 3.

Table 3: Averages of different metrics over five runs in the IMDB experiment. The numbers in parentheses indicate the ranking of each mechanism for that metric. Ablation studies on the number of heads for standard and Optimised attention models show that increasing the number of heads does not meaningfully affect performance. As expected, the Efficient Attention model has the smallest attention layer size and the Super Attention model performs the best in terms of accuracy and loss.

Att.	h	d_m	d_k	# Param.	Avg. Time	Acc. (%)	Loss	Test Acc. (%)	Test Loss
Stn.	1	32	32	4,224	0.284	96.09	0.0821	78.09	0.461
	2	32	16	4,224	0.297	95.51	0.112	78.14	0.467
	4	32	8	4,224 (4)	0.315 (4)	95.70 (4)	0.086 (3)	77.62 (3)	0.474 (3)
Opt.	1	32	32	3,168	0.283	96.62	0.070	78.00	0.461
	2	32	16	3,168	0.299	96.77	0.073	78.00	0.460
	4	32	8	3,168 (2)	0.305 (3)	96.31 (3)	0.095 (4)	77.85 (2)	**0.472 (1)**
Eff.	1	32	32	**2,112 (1)**	**0.274 (1)**	96.66 (2)	0.080 (2)	77.58 (4)	0.478 (4)
Sup.	1	32	32	3,168 (2)	0.289 (2)	**97.68 (1)**	**0.063 (1)**	**78.21 (1)**	**0.472 (1)**

Amazon Reviews. The Amazon Reviews dataset poses a different challenge than the IMDB dataset as it is a significantly larger dataset with 3,650,000 reviews, containing a wider range of sentiments in $1, 2, \ldots, 5$; higher values indicate more positive sentiment. The Transformer models, used in this experiment, all have three attention layers with model dimension and context length 64. The complete results are presented in Table 4.

4.1.4 Edge Device Performance

Our main motivation for introducing Optimised, Efficient, and Super Attention is to allow running more capable models on edge devices. We calculated the inference times of the Transformer models, we trained before, on a MacBook Pro with an M2 Chip for each task/attention mechanism in Table 5.

Table 4: Averages of different metrics over five runs in the Amazon Reviews experiment. The numbers in parentheses indicate the ranking of each mechanism for that metric. Ablation studies on the number of heads for standard and Optimised attention models show that increasing the number of heads does not meaningfully affect performance. As expected, the Efficient Attention model has the smallest attention layer size and the Super Attention model performs the best in accuracy and loss.

Att.	h	d_m	d_k	# Param.	Avg. Time	Acc.	Loss	Test Acc.	Test Loss
Stn.	1	64	64	16,640	13.60	61.33	0.897	52.84	1.094
	2	64	32	16,640	15.80	63.61	0.851	52.71	1.091
	4	64	16	16,640 (4)	20.38 (4)	62.54 (2)	0.868 (2)	52.74 (4)	1.097 (4)
Opt.	1	64	64	12,480	12.54	60.71	0.909	52.79	1.093
	2	64	32	12,480	14.37	62.04	0.884	52.93	1.090
	4	64	16	12,480 (2)	19.89 (3)	61.64 (4)	0.876 (4)	52.88 (3)	1.090 (3)
Eff.	1	64	64	**8,320 (1)**	**10.87 (1)**	62.23 (3)	0.873 (3)	53.25 (2)	1.082 (2)
Sup.	1	64	64	12,480 (2)	11.96 (2)	**66.65 (1)**	**0.776 (1)**	**53.87 (1)**	**1.070 (1)**

5 Related Work

After the adoption of Transformers, different research directions have emerged to address different shortcomings of the attention mechanism and Transformer models.

The computational complexity of Transformers increases quadratically in the input length. Sparse attention reduces the computational complexity by focusing on key input parts (Child et al., 2019). A notable application of this is Longformer (Beltagy et al., 2020; Zhang et al., 2021a), which employs a unique attention pattern combining local and global attention.

Despite their efficiency in handling long sequences, sparse attention models like Longformer struggle in tasks that require a comprehensive analysis of the entire sequence, where understanding full the context is essential. Therefore, a new line of research has emerged that focuses on optimising multi-head attention for modern GPUs without changing its structure. Some of the most prominent examples include Flash Attention (Dao et al., 2022) and its successor, Flash Attention 2 (Dao, 2024). Flash Attention's optimisation involves reordering the attention computation and utilising efficient memory handling techniques like tiling, allowing for faster processing and reduced memory demands. Flash Attention-2 further enhances this by refining computational aspects, particularly for handling

Table 5: Total inference times (in seconds) for each attention mechanism/dataset pair on an Apple M2 chip over 5,000 samples. Ablation studies on the number of heads for standard and Optimised attention models show that increasing the number of heads lead to a significant increase in inference time on edge devices. As expected, Efficient and Super Attention models are the fastest. Also, Optimised Attention models are faster than their standard counterpart with the same number of heads while performing equally well as we discussed before.

Name	h	CIFAR100	MNIST	IMDB	Amazon
Standard	1	35.68	2.53	0.219	1.43
	2	41.34	2.72	0.247	1.54
	4	51.52 (4)	3.27 (4)	0.284 (3)	2.09 (4)
	6	55.47	-	-	-
Optimised	1	34.51	2.45	0.213	1.40
	2	40.70	2.64	0.242	1.52
	4	51.01 (3)	3.16 (3)	0.284 (3)	2.05 (3)
	6	52.18	-	-	-
Efficient	1	**32.29 (1)**	**2.32 (1)**	**0.208 (1)**	1.23 (2)
Super	1	33.45 (2)	2.50 (2)	0.221 (2)	**1.22 (1)**

longer sequences. These modifications do not change the core structure of the standard multi-head attention but make it more efficient for large-scale applications.

Since the adoption of LLMs and large Foundation Models (FMs), a significant amount of work has been done on improving the scalability and deployability of such models. Hu et al. (2022) introduced LoRA for adapting pre-trained models with minimal additional parameters by focusing on altering the rank of weight matrices. This approach allows for efficient fine-tuning of large models, enhancing their practicality across a broader range of applications. Building on this, QLoRA (Dettmers et al., 2023) incorporates quantisation, reducing the precision of numerical representations within the model. This results in a substantial reduction in both memory and computational demands, thereby making large models more accessible and efficient for deployment in various settings.

Quantisation, striving to make neural networks more efficient in memory and computation, has revolutionized the adoption of FMs, particularly those based on Transformers. Recent advances include mixed-precision post-training quantisation for vision transformers, which maintains attention mechanism integrity (Liu et al., 2021). This involves novel quantisation strategies, like similarity-aware and ranking-aware techniques. Moreover, Ding et al. (2022) unveild a cutting-edge framework enhancing quantised model accuracy without significant performance degradation. Beyond post-training quantisation, research explores methods like quantisation-aware training (Jacob et al., 2018; Nagel et al., 2022), mixed-precision training (Micikevicius et al., 2018), dynamic quantisation (Zhang et al., 2021b), and layer-wise quantisation (Chen et al., 2019), aiming to balance model performance with computational and memory efficiency. Despite their benefits in reducing neural networks' memory and computational demands, these quantisation techniques face challenges, including potential performance drops in complex tasks and increased vulnerability to adversarial attacks, highlighted by (Hong et al., 2021; Gupta and Ajanthan, 2022).

Finally, there are other lines of work like sparsification that make a neural network sparse, meaning reducing the number of non-zero elements in the network's weights. This can involve pruning weights that have little effect on the output, leading to a network with fewer connections and parameters. Recently, Ashkboos et al. (2024) introduced a new post-training sparsification technique for large language models that reduces model size by compressing weight matrices with a 1-10% performance degradation. In addition to a degradation of performance, increasing sparsity could lead to reduced robustness as shown by Timpl et al. (2022).

Conclusions

This paper presents a significant leap forward in the evolution of attention mechanisms, particularly addressing the challenges posed by large foundation and language models. Our introduction of *Optimised Attention*, *Efficient Attention*, and *Super Attention* marks a transformative change in the efficiency and efficacy of AI systems. These mechanisms, introduced here, not only reduce computational costs and model sizes—thereby making AI more accessible and sustainable—but also maintain, and in case of Super Attention enhance the performance of attention-based models.

Optimised Attention is the ideal replacement for standard attention where using multiple heads is an essential part of the model design. It is the most similar attention mechanism to standard attention among the ones introduced here. It reduces the attention layer size by 25% as well as its computational cost while performing similarly in vision and natural language tasks.

Efficient Attention is the most efficient full attention mechanism that we are aware of, performing on par with the standard attention on both vision and natural language tasks while having *half* as many parameters. We showed that models using Efficient Attention are up to *twice as fast* compared to their counterparts that use the standard attention. We believe Efficient Attention can replace standard attention in the models that use attention mechanism, allowing them to be smaller, faster, and deployable on a wider range of devices.

Super Attention outperforms standard attention as well as Optimised and Efficient Attention in both vision and natural language tasks by a substantial margin while being smaller than standard attention by at least 25% and faster by up to 45% when the context size is equal to or smaller than the model dimensions. As such, Super Attention is an ideal replacement for standard attention in tasks where high performance is essential and the context size is proportional to the model dimension.

The impressive performance of the attention mechanisms introduced here in diverse tasks underscores their versatility and potential to redefine the landscape of AI. As AI continues to evolve, the developments presented in this paper will likely play a pivotal role in shaping the future of efficient, powerful, accessible and environmentally conscious AI.

Acknowledgments and Disclosure of Funding

This work is partially supported by the UK EPSRC via the Centre for Doctoral Training in Intelligent Games and Game Intelligence (IGGI; EP/S022325/1).

References

D. Bahdanau, K. Cho, and Y. Bengio, "Neural machine translation by jointly learning to align and translate," in *3rd International Conference on Learning Representations, ICLR*. OpenReview.net, 2015.

A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, L. Kaiser, and I. Polosukhin, "Attention is all you need," in *Advances in neural information processing systems, NeurIPS*. Curran Associates, Inc., 2017, pp. 5998–6008.

A. Dosovitskiy, L. Beyer, A. Kolesnikov, D. Weissenborn, X. Zhai, T. Unterthiner, M. Dehghani, M. Minderer, G. Heigold, S. Gelly, J. Uszkoreit, and N. Houlsby, "An image is worth 16x16 words: Transformers for image recognition at scale," in *9th International Conference on Learning Representations, ICLR*. OpenReview.net, 2021.

A. Mott, D. Zoran, M. Chrzanowski, D. Wierstra, and D. J. Rezende, "Towards interpretable reinforcement learning using attention augmented agents," in *Advances in Neural Information Processing Systems, NeurIPS*, 2019, pp. 12 329–12 338.

E. Choi, M. T. Bahadori, J. Sun, J. Kulas, A. Schuetz, and W. F. Stewart, "RETAIN: an interpretable predictive model for healthcare using reverse time attention mechanism," in *Advances in Neural Information Processing Systems, NeurIPS*, 2016, pp. 3504–3512.

H. Touvron, T. Lavril, G. Izacard *et al.*, "Llama: Open and efficient foundation language models," 2023, arXiv preprint arXiv:2302.13971.

H. Touvron, L. Martin, K. Stone *et al.*, "Llama 2: Open foundation and fine-tuned chat models," 2023, arXiv preprint arXiv:2307.09288.

J. Achiam, S. Adler, S. Agarwal *et al.*, "GPT-4 technical report," 2023.

R. Anil, S. Borgeaud, Y. Wu *et al.*, "Gemini: a family of highly capable multimodal models," 2023, arXiv preprint arXiv:2312.11805.

P. Dhar, "The carbon impact of artificial intelligence." *Nature Machine Intelligence*, vol. 2, no. 8, pp. 423–425, 2020.

B. Jacob, S. Kligys, B. Chen, M. Zhu, M. Tang, A. G. Howard, H. Adam, and D. Kalenichenko, "Quantization and training of neural networks for efficient integer-arithmetic-only inference," in *IEEE/CVF Conference on Computer Vision and Pattern Recognition, CVPR*. Computer Vision Foundation / IEEE, 2018, pp. 2704–2713.

E. J. Hu, Y. Shen, P. Wallis, Z. Allen-Zhu, Y. Li, S. Wang, L. Wang, and W. Chen, "Lora: Low-rank adaptation of large language models," in *10th International Conference on Learning Representations, ICLR*. OpenReview.net, 2022.

T. Dettmers, A. Pagnoni, A. Holtzman, and L. Zettlemoyer, "Qlora: Efficient finetuning of quantized llms," 2023, arXiv preprint arXiv:2305.14314.

S. Ashkboos, M. L. Croci, M. G. do Nascimento, T. Hoefler, and J. Hensman, "Slicegpt: Compress large language models by deleting rows and columns," in *12th International Conference on Learning Representations, ICLR*. OpenReview.net, 2024.

T. Dao, D. Y. Fu, S. Ermon, A. Rudra, and C. Ré, "Flashattention: Fast and memory-efficient exact attention with io-awareness," in *Advances in Neural Information Processing Systems, NeurIPS*, vol. 35. Curran Associates, Inc., 2022, pp. 16 344–16 359.

T. Dao, "Flashattention-2: Faster attention with better parallelism and work partitioning," 2024.

C. D. Meyer, *Matrix Analysis and Applied Linear Algebra*, 2nd ed., ser. Other Titles in Applied Mathematics. SIAM, 2023.

Y. LeCun, C. Cortes, C. Burges *et al.*, "Mnist handwritten digit database," http://yann.lecun.com/exdb/mnist, 2010, accessed: 2020-06-13.

A. Krizhevsky, "Learning multiple layers of features from tiny images," University of Toronto, Tech. Rep., 2009.

A. L. Maas, R. E. Daly, P. T. Pham, D. Huang, A. Y. Ng, and C. Potts, "Learning word vectors for sentiment analysis," in *The 49th Annual Meeting of the Association for Computational Linguistics, ACL*. The Association for Computer Linguistics, 2011, pp. 142–150.

J. Ni, J. Li, and J. McAuley, "Justifying recommendations using distantly-labeled reviews and fine-grained aspects," in *Empirical Methods in Natural Language Processing EMNLP*. Association for Computational Linguistics, 2019, pp. 188–197.

R. Child, S. Gray, A. Radford, and I. Sutskever, "Generating long sequences with sparse transformers," 2019, arXiv preprint arXiv:1904.10509.

I. Beltagy, M. E. Peters, and A. Cohan, "Longformer: The long-document transformer," 2020, arXiv preprint arXiv:2004.05150.

P. Zhang, X. Dai, J. Yang, B. Xiao, L. Yuan, L. Zhang, and J. Gao, "Multi-scale vision longformer: A new vision tansformer for high-resolution image encoding," in *IEEE/CVF International Conference on Computer Vision, ICCV*. IEEE, 2021, pp. 2978–2988.

Z. Liu, Y. Wang, K. Han, W. Zhang, S. Ma, and W. Gao, "Post-training quantization for vision transformer," in *Advances in Neural Information Processing Systems, NeurIPS*, vol. 34. Curran Associates, Inc., 2021, pp. 28 092–28 103.

Y. Ding, H. Qin, Q. Yan, Z. Chai, J. Liu, X. Wei, and X. Liu, "Towards accurate post-training quantization for vision transformer," in *30th ACM International Conference on Multimedia, MM*. ACM, 2022, pp. 5380–5388.

M. Nagel, M. Fournarakis, Y. Bondarenko, and T. Blankevoort, "Overcoming oscillations in quantization-aware training," in *39th International Conference on Machine Learning, ICML*, ser. Proceedings of Machine Learning Research, vol. 162. PMLR, 2022, pp. 16 318–16 330.

P. Micikevicius, S. Narang, J. Alben, G. F. Diamos, E. Elsen, D. García, B. Ginsburg, M. Houston, O. Kuchaiev, G. Venkatesh, and H. Wu, "Mixed precision training," in *6th International Conference on Learning Representations, ICLR*. OpenReview.net, 2018.

Z. Zhang, W. Shao, J. Gu, X. Wang, and P. Luo, "Differentiable dynamic quantization with mixed precision and adaptive resolution," in *38th International Conference on Machine Learning, ICML*, vol. 139. Curran Associates, Inc., 2021, pp. 12 546–12 556.

S. Chen, W. Wang, and S. J. Pan, "Deep neural network quantization via layer-wise optimization using limited training data," in *AAAI Conference on Artificial Intelligence*. AAAI Press, 2019, pp. 3329–3336.

S. Hong, M. Panaitescu-Liess, Y. Kaya, and T. Dumitras, "Qu-anti-zation: Exploiting quantization artifacts for achieving adversarial outcomes," in *Advances in Neural Information Processing Systems, NeurIPS*. Curran Associates, Inc., 2021, pp. 9303–9316.

K. Gupta and T. Ajanthan, "Improved gradient-based adversarial attacks for quantized networks," in *AAAI Conference on Artificial Intelligence*. AAAI Press, 2022, pp. 6810–6818.

L. Timpl, R. Entezari, H. Sedghi, B. Neyshabur, and O. Saukh, "Understanding the effect of sparsity on neural networks robustness," 2022, arXiv preprint arXiv:2206.10915.