Attention-Only Transformers and Implementing MLPs with Attention Heads

Anonymous authors
Paper under double-blind review

Abstract

The transformer architecture is widely used in machine learning models and consists of two alternating sublayers: attention heads and MLPs. We prove that an MLP neuron can be implemented by a masked attention head with internal dimension 1 so long as the MLP's activation function comes from a restricted class including SiLU and close approximations of ReLU and GeLU. This allows one to convert an MLP-and-attention transformer into an attention-only transformer at the cost of greatly increasing the number of attention heads. We also prove that attention heads can perform the components of an MLP (linear transformations and activation functions) separately. Finally, we prove that attention heads can encode arbitrary masking patterns in their weight matrices to within arbitrarily small error.

1 Introduction

The transformer architecture was introduced in the landmark 2017 paper *Attention is All You Need* (Vaswani et al., 2023) and traditionally consists of alternating attention and multilayer-perceptron (MLP) sublayers. Although initially used for machine translation, transformers have been used across a wide range of tasks, including language modeling (Radford et al., 2018; Devlin et al., 2019; Liu et al., 2018), computer vision (Khan et al., 2022; Cornia et al., 2020), and image generation (Parmar et al., 2018). The widespread deployment of transformers has led to increasing interest in *mechanistic interpretability* (Wang et al., 2022; Conmy et al., 2023), which seeks to convert the computations of transformers into human-understandable explanations. Some interpretability efforts, such as Elhage et al. (2021), focused on attention-only transformers, finding that MLP layers were harder to interpret.

This work seeks to supplement those mechanistic interpretability methods by showing that MLP layers in transformers are equivalent to a sum of masked attention heads and therefore can be subjected to interpretability techniques that work on attention-only transformers. In Theorem 3 we show that by including a "bias token" akin to the persistent memory vectors in Sukhbaatar et al. (2019) and using a slightly unusual attention-masking pattern, an MLP layer of size ℓ can be written as the sum of ℓ attention heads with internal dimension 1. We show in Theorem 6 that one can apply this process throughout the entire transformer, converting the typical MLP-and-attention transformer into an attention-only transformer. We then show in Theorems 7 and 8 that attention heads can implement row-wise linear transformations and matrix-level activation functions separately. Finally, we show in Theorem 9 that a slightly augmented network is capable of approximating any masking pattern to within arbitrary error.

2 Background

Notation. *Throughout, we will use $M_{n,k}$ to denote the set of real-valued n-by-k matrices.*

For matrices $X \in M_{n_1,k_1}$ and $Y \in M_{n_2,k_2}$ of any size, we will write $X \oplus Y$ for the block matrix

$$X \oplus Y = \left[\begin{array}{c|c} X & \mathbf{0} \\ \hline \mathbf{0} & Y \end{array}\right] \in M_{n_1+n_2,k_1+k_2}$$

where each $\mathbf{0}$ is a correctly sized zero matrices. We will similarly write $\mathbf{1}$ for matrices with a 1 for every entry.

For matrices $X \in M_{n,k_1}$ and $Y \in M_{n,k_2}$, we will write

$$[X|Y] \in M_{n,k_1+k_2}$$

for the matrix made by appending one to the other.

For a real-valued function f and matrix X, we will write $f(X)$ for the entry-wise application of that function to the matrix.

We write

$$\begin{aligned} \mathrm{ReLU}(x) :=& \quad max(x,0) \\ \mathrm{SiLU}(x) :=& \quad x\sigma(x) \\ \mathrm{GeLU}(x) :=& \quad x\Phi(x) \end{aligned}$$

where $\sigma(x) = 1/(1+\exp(-x))$, and $\Phi(x)$ is the cumulative distribution function for the standard Gaussian distribution with mean 0 and variance 1. We will say that a generalized SiLU function *is a function of the form*

$$f(x) = a_1 \mathrm{SiLU}(a_2 x)$$

for some $a_1, a_2 \in \mathbb{R}$.

The class of generalized SiLU functions includes $\mathrm{SiLU}(x)$ and approximations of GeLU and ReLU. In particular, $\mathrm{GeLU}(x) \approx \mathrm{SiLU}(1.702x)/1.702$ (Hendrycks & Gimpel, 2023) (reaching a maximum absolute error of 0.0203 at $x = \pm 2.27$) and $\mathrm{ReLU}(x) \approx \mathrm{SiLU}(kx)/k$ for large k (reaching a maximum absolute error of $\frac{0.2785}{k}$ at $x = \pm\frac{1.278}{k}$).

Definition 1. *An* MLP with no biases and one hidden layer *is a function $f : M_{n,k} \to M_{n,k}$ of the form*

$$f(X) = \alpha(XV_1)V_2 \tag{1}$$

where $\alpha : \mathbb{R} \to \mathbb{R}$ is some real-valued function applied entry-wise to matrices, and V_1, V_2 are fixed matrices in $M_{k,\ell}$ and $M_{\ell,k}$, respectively, called parameter matrices. The number ℓ is called the size of the hidden layer*, and the function α is called the* activation function*.*

Many transformer architectures follow the convention that $\ell = 4k$ (Vaswani et al., 2023; Brown et al., 2020), but we do not require this. There are many popular choices for activation functions (Hendrycks & Gimpel, 2023), including ReLU, SiLU, and GeLU.

For describing attention heads, we largely follow the framework of Elhage et al. (2021).

Definition 2. *A* mask matrix *Λ is a matrix with entries in $\{0,1\}$ such that every row has at least one nonzero entry.*

Let $X, \Lambda \in M_{n,k}$, and suppose Λ is a mask matrix. Then define the masked softmax *function*

$$\mathrm{msoftmax}(X, \Lambda) := \mathrm{rownorm}\left(\exp(X) \odot \Lambda\right)$$

where rownorm *denotes row-wise ℓ^1 normalization, and $\odot$ denotes element-wise multiplication. That is, the masked softmax function acts like the usual row-wise softmax but applied to only the entries of X where the mask Λ is 1. At the entries where Λ is 0, the output of the masked softmax function takes the value 0.*

A masked attention head *is a function $h : M_{n,k} \to M_{n,k}$ of the form*

$$h(X) = \mathrm{msoftmax}(XW_{QK}X^T, \Lambda)XW_{OV} \tag{2}$$

for some matrices $W_{OV}, W_{QK} \in M_{k,k}$, and mask matrix $\Lambda \in M_{n,n}$. We call W_{OV} and W_{QK} the parameter matrices *for this attention head.*

For practical reasons, attention heads are rarely described (or implemented) as in Equation 2. However, one can verify that this definition encompasses the classical transformer framework in Vaswani et al. (2023), with $W_{QK} = (W_i^Q)(W_i^K)^T/\sqrt{d_k}$, and $W_{OV} = W_i^V W_i^O$, where W_i^O denotes the appropriate subblock of the W^O matrix.

For many language tasks, the masking pattern is chosen to mask later tokens from earlier tokens (Vaswani et al., 2023; Radford et al., 2018), i.e., Λ is the subdiagonal matrix with $\Lambda_{i,j} = \begin{cases} 1 & \text{if } i \leq j \\ 0 & \text{otherwise} \end{cases}$. However, in our construction in Theorem 3 and Theorem 6, we will make use of a nonstandard masking pattern in which tokens only attend to themselves and a single special token.

3 Implementing MLP Layers with Attention Heads

In this section we show that MLP layers whose activation functions are generalized SiLU functions are in fact a sum of attention heads.

The intuition for this claim is simple: both attention heads and MLPs are mostly linear, with a single nonlinearity (respectively, masked softmax and the generalized SiLU activation function). Additionally, softmax can easily play the role of the sigmoid part of SiLU since $\mathrm{softmax}([-x, 0]) = \mathrm{rownorm}([e^{-x}, 1]) = [\sigma(x), \sigma(-x)]$. Multiplying this attention pattern onto the vector $[x, 0]$, we get $x\sigma(x) + 0\sigma(-x) = \mathrm{SiLU}(x)$. The following theorem is a formalization of this intutition.

Theorem 3. *Let $f(X) = \alpha(XV_1)V_2$ be an MLP on $M_{N,D}$ with no biases and one hidden layer of size ℓ, and suppose α is a generalized SiLU function $\alpha(x) = a_1\mathrm{SiLU}(a_2x)$. Then there are ℓ masked attention heads $\{h_i\}_{i=1}^{\ell}$ on $M_{N+1,D+1}$ such that*

$$f(X) \oplus [0] = \sum_{i=1}^{\ell} h_i(X \oplus [1])$$

for all $X \in M_{N,D}$.

In particular, for the ith attention head, one uses parameter and mask matrices

$$\begin{aligned} W_{QK} &= a_2 \left[\begin{array}{c|c} \mathbf{0} & -V_1^i \\ \hline \mathbf{0} & 0 \end{array}\right] \\ W_{OV} &= a_1a_2V_1^iV_2^i \oplus [0] \\ \Lambda &= \left[\begin{array}{c|c} I_N & \mathbf{1} \\ \hline \mathbf{0} & 1 \end{array}\right] \end{aligned}$$

where the block decompositions are into size N and 1, V_1^i denotes the ith column of V_1, V_2^i denotes the ith row of V_2, and $\mathbf{1}$ denotes the column vector of all 1s.

Proof. We first prove the claim in the case of $\ell = a_1 = a_2 = 1$. In this case, since there is only one column in V_1, then $V_1 = V_1^i$, and similarly $V_2 = V_2^i$. Consider the attention matrix $\mathrm{msoftmax}((X \oplus [1])W_{QK}(X \oplus [1])^T, \Lambda)$. Multiplying matrices on the level of their blocks, we get that the first argument of the masked softmax is

$$(X \oplus [1])W_{QK}(X \oplus [1])^T = \left[\begin{array}{c|c} X & \mathbf{0} \\ \hline \mathbf{0} & 1 \end{array}\right] \left[\begin{array}{c|c} \mathbf{0} & -V_1^i \\ \hline \mathbf{0} & 0 \end{array}\right] \left[\begin{array}{c|c} X & \mathbf{0} \\ \hline \mathbf{0} & 1 \end{array}\right]^T = \left[\begin{array}{c|c} \mathbf{0} & -XV_1 \\ \hline \mathbf{0} & 0 \end{array}\right]$$

Now consider the masked softmax term in the jth row for $j \leq N$. This row has exactly two unmasked values, the diagonal entry and the rightmost entry, taking the values 0 and $-(XV_1)_j$, respectively. Applying $\exp$ and $\mathrm{rownorm}$ results in $\sigma((XV_1)_j)$ and $\sigma(-(XV_1)_j)$, respectively. Thus, the masked softmax term becomes

$$\begin{aligned} \mathrm{msoftmax}((X \oplus [1])W_{QK}(X \oplus [1])^T, \Lambda) &= \mathrm{msoftmax}(\left[\begin{array}{c|c} \mathbf{0} & -XV_1 \\ \hline \mathbf{0} & 0 \end{array}\right], \left[\begin{array}{c|c} I_{n-1} & \mathbf{1} \\ \hline \mathbf{0} & 1 \end{array}\right]) \\ &= \left[\begin{array}{c|c} \mathrm{diag}(\sigma(XV_1)) & \sigma(-XV_1) \\ \hline \mathbf{0} & 1 \end{array}\right] \end{aligned}$$

Substituting these values into the expression for $h(X)$ gives

$$
\begin{aligned}
h(X \oplus [1]) &= \text{msoftmax}((X \oplus [1])W_{QK}(X \oplus [1])^T, \Lambda)(X \oplus [1])W_{OV} \\
&= \left[\begin{array}{c|c} \text{diag}(\sigma(XV_1)) & \sigma(-XV_1) \\ \hline \mathbf{0} & 1 \end{array}\right](X \oplus [1])W_{OV} \\
&= \left[\begin{array}{c|c} \text{diag}(\sigma(XV_1)) & \sigma(-XV_1) \\ \hline \mathbf{0} & 1 \end{array}\right]\left[\begin{array}{c|c} X & \mathbf{0} \\ \hline \mathbf{0} & 1 \end{array}\right]\left[\begin{array}{c|c} V_1V_2 & \mathbf{0} \\ \hline \mathbf{0} & 0 \end{array}\right] \\
&= \left[\begin{array}{c|c} \text{diag}(\sigma(XV_1))XV_1V_2 & \mathbf{0} \\ \hline \mathbf{0} & 0 \end{array}\right] \\
&= \left[\begin{array}{c|c} \text{SiLU}(XV_1)V_2 & \mathbf{0} \\ \hline \mathbf{0} & 0 \end{array}\right] \\
&= \left[\begin{array}{c|c} f(X) & \mathbf{0} \\ \hline \mathbf{0} & 0 \end{array}\right] \\
&= f(X) \oplus [0]
\end{aligned}
$$

as desired. This completes the $\ell = a_1 = a_2 = 1$ case.

For a general a_1, a_2, apply the previous case to an MLP with weight matrices a_2V_1 and a_1V_2.

Finally, for the fully general case with $\ell > 1$, for each $1 \leq i \leq \ell$, let $f_i(X) = \alpha(XV_1^i)V_2^i$, and note that $f = \sum_{i=1}^{\ell} f_i$. Let h_i denote the attention head corresponding to f_i given by the $\ell = 1$ case. Then we have that

$$
\begin{aligned}
f(X) \oplus [0] &= \sum_{i=1}^{\ell} f_i(X) \oplus [0] \\
&= \sum_{i=1}^{\ell} h_i(X \oplus [1])
\end{aligned}
$$

as desired. □

Remark 4. *The additional term $\oplus[1]$ in Theorem 3 is similar to the* persistent vectors *of Sukhbaatar et al. (2019). In that work, the authors propose a new architecture, which they call the all-attention architecture, in which attention can also be paid to certain static vectors, learned for each attention head, called the persistent vectors. Our approach could also be implemented in that architecture with a single persistent vector $(0, 0, 0, .., 0, 1)$ shared across all attention heads.*

Note also that the W_{QK} and W_{OV} matrices used in Theorem 3 can be factored into the matrices W_Q, W_K, W_V, $W_O \in M_{D+1,1}$ from Vaswani et al. (2023) satisfying $W_{QK} = W_QW_K^T/\sqrt{D+1}$ and $W_{OV} = W_VW_O$. In particular, we can take $W_Q = W_V = a_2[V_1^i|0]^T$, $W_K = \sqrt{D+1}[\mathbf{0}|-1]^T$, and $W_O = a_1[V_2^i|0]^T$. Since W_K is shared across all attention heads, we only need to store two sets of parameters, the vectors $W_Q = W_V$ and W_O.

This provides an alternative perspective on MLP neurons: a neuron in an MLP is an attention head with internal dimension 1 and a particularly restrictive masking pattern in which each token attends only to itself and a static "bias" token.

We now have the necessary tools to show that a decoder-only transformer as in Liu et al. (2018); Radford et al. (2018) can be implemented entirely with attention heads.

Definition 5. *A transformer is a function $t : M_{N,D} \to M_{N,D}$ of the form $X_0 \mapsto X_1 \mapsto ... \mapsto X_m = t(X_0)$, where*

$$
X_{j+1} = \begin{cases} \text{LayerNorm}(X_j + \sum_i h_{j,i}(X_j)) & \textit{or} \\ \text{LayerNorm}(X_j + f_j(X_j)) & \end{cases}
$$

for some attention heads $h_{j,i}$ or MLPs with a single hidden layer f_j. Note the use of Layer Normalization (Ba et al., 2016) and skip connections*, where one performs some computation f on X_j and defines $X_{j+1} = \text{LayerNorm}(X_j + f(X_j))$, as opposed to $X_{j+1} = f(X_j)$.*

Classically, transformers alternate between attention sublayers and MLP sublayers, but we allow the existence of other architectures, including attention-only transformers and "MLP-only" transformers.

Theorem 6. *If a transformer's MLP layers are activated by a generalized SiLU function, they can be substituted with attention heads.*

Proof. We will show that we can create a new transformer t' on $M_{N+1,D+1}$ whose residual stream X'_j on every sublayer satisfies

$$X'_j = X_j \oplus [1]$$

This is sufficient to prove the main claim since the output of this new transformer will be $X'_{2m} = X_{2m} \oplus [1]$ and therefore contain the output of the original transformer.

Without loss of generality, assume that the MLP layers have no bias terms (i.e., that we've already used the "bias trick" to fold bias terms into the weight matrix).

To prove that there is a transformer t' that satisfies $X'_j = X_j \oplus [1]$ on every sublayer, we proceed by induction. For the base case of $j = 0$, we tweak the transformer's context window and embedding weights so that $X'_0 = X_0 \oplus [1]$.

We split the inductive case depending on whether the original transformer's sublayer used attention or an MLP. If the original layer was an MLP, then by Theorem 3 there are attention heads $h'_{j,i}$ such that $f_j(X) \oplus [0] = \sum h'_{j,i}(X \oplus [1])$, so in our transformer t', using these attention heads yields

$$\begin{aligned} X'_{j+1} &= \text{LayerNorm}(X'_j + \sum h'_{j,i}(X'_j)) \\ &= \text{LayerNorm}((X_j \oplus [1]) + \sum h'_{j,i}(X_j \oplus [1])) \\ &= \text{LayerNorm}((X_j \oplus [1]) + (f_j(X) \oplus [0]))) \\ &= \text{LayerNorm}(X_j + f_j(X)) \oplus [1] \\ &= X_{j+1} \oplus [1] \end{aligned}$$

as desired.

If instead, the transformer used attention heads on the jth sublayer, we must tweak our original induction heads to account for the new size. To this end, we will show that for each of the original induction heads $h = h_{j,i}$, we can create an induction head h' such that

$$h'(X \oplus [1]) = h(X) \oplus [0]$$

Let W_{QK}, W_{OV}, and Λ denote the original parameter and masking matrices for h. Then define

$$\begin{aligned} W'_{QK} &= W_{QK} \oplus [1] \\ W'_{OV} &= W_{OV} \oplus [0] \\ \Lambda' &= \Lambda \oplus [1] \end{aligned}$$

Then,

$$\begin{aligned} h'(X \oplus [1]) &= \text{msoftmax}((X \oplus [1])W'_{QK}(X \oplus [1])^T, \Lambda')(X \oplus [1])W'_{OV} \\ &= \text{msoftmax}((X \oplus [1])(W_{QK} \oplus [1])(X \oplus [1])^T, (\Lambda \oplus [1]))(X \oplus [1])(W_{OV} \oplus [0]) \\ &= \text{msoftmax}(XW_{QK}X^T \oplus [1], \Lambda \oplus [1])(XW_{OV} \oplus [0]) \\ &= (\text{msoftmax}(XW_{QK}X^T, \Lambda) \oplus [1])(XW_{OV} \oplus [0]) \\ &= \text{msoftmax}(XW_{QK}X^T, \Lambda)XW_{OV} \oplus [0] \\ &= h(X) \oplus [0] \end{aligned}$$

as desired. Now, creating such $h'_{j,i}$ for each of the original attention heads $h_{j,i}$, we have

$$\begin{aligned} X'_{j+1} &= \text{LayerNorm}(X'_j + \sum h'_{j,i}(X'_j)) \\ &= \text{LayerNorm}((X_j \oplus [1]) + \sum h'_{j,i}(X_j \oplus [1])) \\ &= \text{LayerNorm}((X_j \oplus [1]) + \sum h_{j,i}(X) \oplus [0])) \\ &= \text{LayerNorm}((X_j + \sum h_{j,i}(X))) \oplus [1] \\ &= X_{j+1} \oplus [1] \end{aligned}$$

as desired. This completes the inductive step and the proof.

□

It is instructive to compare this construction to the negative results of Dong et al. (2021), which find that without skip connections or MLPs, a self-attention network converges rapidly to a rank-1 matrix. Since we obviously do away with the MLP layer, our result depends on the use of skip connections. In particular, the "bias term" of $\oplus[1]$ is zeroed out by the construction in Theorem 3, so applying the construction in Theorem 6 without a skip connection results in $X'_0 = X_0 \oplus [1]$, but $X'_1 = X_1 \oplus [0]$. Then, in the $j = 2$ sublayer, the construction in 3 would fail for lack of this bias term, as, without it, the pre-attention matrix $(X')W_{QK}(X')^T$ is 0.

4 Linear Transformations and Activation Functions with Attention Heads

Theorem 3 shows that attention heads can implement an MLP layer, but can they separately implement the components of an MLP, a linear transformation and an activation function? In this section we show that the answer is yes.

We first show that an attention head can perform an arbitrary linear operation row-wise on the matrix.

Theorem 7. *Let $h : M_{N,D} \to M_{N,D}$ be an attention head with masking matrix $\Lambda = I_N$. Then $h(X) = XW_{OV}$.*

Proof. Because $\Lambda = I_n$, after masking, the attention matrix $\text{msoftmax}(XW_{QK}X^T, \Lambda)$ will have nonzero entries only along the diagonal. Since the rows of the attention matrix are normalized to sum to 1, it follows that $\text{msoftmax}(XW_{QK}X^T, \Lambda) = I_n$. Then,

$$h(X) = \text{msoftmax}(XW_{QK}X^T, \Lambda)XW_{OV} = I_nXW_{OV} = XW_{OV}$$

as desired. □

Now we will show that one can apply a generalized SiLU function entrywise.

Theorem 8. *Let α be a generalized SiLU function. Then there are D attention heads $h_1, ..., h_D$ on $M_{N+1,D+1}$ such that*

$$\alpha(X) \oplus [0] = \sum_{i=1}^{D} h_i(X \oplus [1])$$

Proof. This follows immediately from applying Theorem 3 to the MLP $f(X) = \alpha(XI_N)I_N = \alpha(X)$, whose hidden layer is of size $\ell = D$. □

Note that a transformer usually makes use of skip connections, so that the residual stream experiences the transformation $X \mapsto X + sublayer(X)$. Thus, to get the transformation $X \mapsto \alpha(X)$, one can combine these two theorems, using $D + 1$ attention heads to produce $sublayer(X) = \alpha(X) - X$, in which case $X \mapsto X + sublayer(X) = \alpha(X)$.

5 Encoding Masking Patterns in Weight Matrices

Although some previous work has used multiple masking patterns[1], some readers may be disappointed that the attention patterns prescribed in the previous sections are oddly "artificial". In this section, we will show a technique to ameliorate this concern by embedding the masking pattern into the W_{QK} matrix. To do so, we must further augment the residual stream, but our technique allows us to encode an arbitrary masking pattern in the W_{QK} parameters at the cost of arbitrarily small errors and poor training behavior.

Theorem 9. *Let h be a masked attention head on $M_{N,D}$ with mask matrix Λ_1. Then for any mask matrix Λ_2 satisfying $\Lambda_1 \leq \Lambda_2$ entrywise, there is a family of masked attention heads h_Ω, parameterized by $\Omega \in \mathbb{R}$, that use Λ_2 as their mask matrix and such that $h_\Omega([X|I_N]) \to [h(X)|\mathbf{0}]$ uniformly on compacta as $\Omega \to \infty$.*

Proof. Define h_Ω to be the attention head using the mask matrix Λ_2 and parameter matrices

$$\begin{aligned} W_{QK,\Omega} &= W_{QK} \oplus \Omega\Lambda_1 \\ W_{OV,\Omega} &= W_{OV} \oplus \mathbf{0} \end{aligned}$$

Fix some compact set $K \subset M_{N,D}$ and $\epsilon > 0$.

First observe that

$$\begin{aligned} h_\Omega([X|I_N]) &:= \text{msoftmax}([X|I_N]W_{QK,\Omega}[X|I_N]^T, \Lambda_2)[X|I_N]W_{OV,\Omega} \\ &= \text{msoftmax}([X|I_N](W_{QK} \oplus \Omega\Lambda_1)[X|I_N]^T, \Lambda')[X|I_N](W_{OV} \oplus \mathbf{0}) \\ &= \text{msoftmax}(XW_{QK}X^T + \Omega\Lambda_1, \Lambda_2)[XW_{OV}|\mathbf{0}] \end{aligned}$$

Our first task is to show that the attention pattern $A_1 := \text{msoftmax}(XW_{QK}X^T + \Omega\Lambda_1, \Lambda_2)$ converges to the corresponding attention pattern $A_2 := \text{msoftmax}(XW_{QK}X^T, \Lambda_1)$ entrywise as $\Omega \to \infty$. To this end, fix $\epsilon_0 > 0$, and pick $b \in \mathbb{R}$ such that entries of $XW_{QK}X^T$ are bounded in absolute value by b as X ranges over K, and let $\Omega > \ln(N/\epsilon_0) + 2b$. We have three cases depending on whether the corresponding entries in Λ_1 and Λ_2 are 0 or 1:

1. If $\Lambda_{1,(i,j)} = \Lambda_{2(i,j)} = 0$, then $A_{1,(i,j)} = A_{2,(i,j)} = 0$ due to masking.
2. If $\Lambda_{1,(i,j)} = 0$ and $\Lambda_{2,(i,j)} = 1$, then $A_{1,(i,j)} = 0$. Since Λ_1 is a mask matrix, in row i there is a column J such that $\Lambda_{1,(i,J)} = 1$. Then the (i, J)th entry of $\exp(XW_{QK}X^T + \Omega\Lambda_1)$ is at least $\exp(\Omega - b)$, while the (i, j)th entry is at most $\exp(b)$. Thus, after row-normalizing, we have

   $$\begin{aligned} A_{2,(i,j)} &\leq \frac{\exp(b)}{\exp(\Omega - b)} \\ &= \frac{1}{\exp(\Omega - 2b)} \end{aligned}$$

   Since $\Omega > \ln(N/\epsilon_0) + 2b$, we have $\exp(\Omega - 2b) > N/\epsilon_0$, so $A_{2,(i,j)} \leq \frac{1}{N/\epsilon_0} = \epsilon_0/N < \epsilon_0$ as desired.
3. If $\Lambda_{1,(i,j)} = \Lambda_{2,(i,j)} = 1$, then consider the ith row. As shown in the previous two cases, in each entry of this row where $\Lambda_{1,(i,j)} = 0$, we have $A_{2,(i,j)} < \epsilon_0/N$. Since there are N terms in this row, and any row sums to 1 due to normalization, this means that the remaining terms, where $\Lambda_{1,(i,j)} = 1$, sum to some value $S \in [1 - \epsilon_0, 1]$. Since the log ratio between two such terms is the difference of their corresponding entries in $XW_{QK}X^T + \Omega\Lambda_1$, and the Ω terms of those entries will cancel, this shows that the ratio between terms where $\Lambda_{1,(i,j)} = 1$ in A_2 is the same as the corresponding ratio in A_1. That is, the ith row of A_1 concentrates its mass S in the same locations as A_2 at the same ratios, so $A_{1,(i,j)} = SA_{2,(i,j)}$ for all j with $\Lambda_{1,(i,j)} = 1$. Thus $|A_{1,(i,j)} - A_{2,(i,j)}| = A_{1,(i,j)}|1 - S| < \epsilon_0$.

[1] E.g., Brown et al. (2020) uses "alternating dense and locally banded sparse attention patterns".

Rephrasing our partial result, we have shown that $A_1 = A_2 + E_\Omega$, where E_Ω is an error matrix whose entries are bound by ϵ_0 whenever $\Omega > \ln(N/\epsilon_0) + 2b$.

Returning to our expression for $h_\Omega([X|I_N])$, we have

$$\begin{aligned} h_\Omega([X|I_N]) &= A_1[XW_{OV}|\mathbf{0}] \\ &= (A_2 + E_\Omega)[XW_{OV}|\mathbf{0}] \\ &= A_1[XW_{OV}|\mathbf{0}] + E_\Omega[XW_{OV}|\mathbf{0}] \\ &= [h(X)|\mathbf{0}] + [E_\Omega XW_{OV}|\mathbf{0}] \end{aligned}$$

Thus, the entry-wise difference between $h_\Omega([X|I_N])$ and $[h(X)|\mathbf{0}]$ is $[E_\Omega XW_{OV}|\mathbf{0}]$, so it suffices to show that $E_\Omega XW_{OV}$ is entry-wise less than ϵ. To this end, fixing some $\epsilon > 0$, let $\epsilon_0 = \epsilon/K$, where $K = \max(||XW_{OV}||/\sqrt{N}, 1)$ and $||\cdot||$ denotes the operator norm of a matrix. Then, for all $\Omega > \ln(N/\epsilon_0) + 2b$, we have E_Ω is entry-wise less than ϵ_0. Therefore, in the i, jth entry of $E_\Omega XW_{OV}$, we have

$$\begin{aligned} |(E_\Omega XW_{OV})_{i,j}| &= |row_i(E_\Omega) \cdot column_j(XW_{OV})| \\ &\leq \epsilon_0\sqrt{N} \cdot ||XW_{OV}|| \\ &= (\epsilon/K)\sqrt{N}||XW_{OV}|| \\ &\leq (\epsilon/(||XW_{OV}||/\sqrt{N}))\sqrt{N}||XW_{OV}|| \\ &= \epsilon \end{aligned}$$

as desired. □

The above result shows that by augmenting the residual stream with an I_N matrix, one can write the masking pattern into the W_{QK} matrix. Combined with Theorem 6, this shows that one can convert a standard transformer into one using only attention heads and the standard masking pattern.

Remark 10. *Inspecting the relation between ϵ and Ω in the previous theorem allows us to provide a more concrete choice of Ω. We require $\Omega > \ln(N/\epsilon_0) + 2b$, where N is the size of the context window, $\epsilon_0 = \epsilon/\max(||XW_{OV}||/\sqrt{N}, 1)$, and b is a bound on the entries of $XW_{QK}X^T$.*

Using properties of logs, we may simplify our requirement to

$$\Omega > \ln(N/\epsilon) + 2b + \max(\ln(N^{\frac{1}{2}}||XW_{OV}||), 0)$$

Since the entries of a marix are bounded by the matrix's operator norm, we can take $b = ||XW_{QK}X^T|| = ||X||^2||W_{QK}||$. The resulting requirement on Ω is then an increasing function of $||X||$, so we may remove our dependence on it by replacing it with $B = \sup_{X \in K} ||X||$, in which case our bound becomes

$$\Omega > \ln(N/\epsilon) + 2B^2||W_{QK}|| + \max(\ln(N^{\frac{1}{2}}B||W_{OV}||), 0)$$

Notably, Ω grows only in the logarithm of ϵ.

Example 11. *Let's compute a value of Ω that is suitable for a particular language model. Take $\epsilon = 2^{-146}$, the minimum positive value representable by a single-precision floating-point number (IEEE, 2008), and apply this to GPT-2, which has a maximum context window of $N = 1024$ tokens (Radford et al., 2019). According to Millidge & Winsor (2023), individual model weights are normally distributed, falling entirely within $[-1, 1]$. Recall that W_{QK} is in fact stored internally as two matrices W_Q and W_K, with $W_{QK} = W_Q W_K^T$. Such matrices are conventionally of size $N \times D/n_{heads}$, and since $D = 1600$ (Radford et al., 2019), and $n_{heads} = 25$ (Heimersheim & Turner, 2023), we have $W_Q, W_K \in M_{1024,64}$. Combining this with the bound that each entry is in $[-1, 1]$, we get that $||W_Q|| \leq \sqrt{64} = 8$. Similarly, $||W_K|| \leq 8$, so $||W_{QK}|| \leq ||W_Q||||W_K|| \leq 8 \cdot 8 = 64$. By a similar argument, $||W_{OV}|| \leq 64$.*

For the bound B on the norm of the residual stream, we turn to Heimersheim & Turner (2023) who finds that the measured norm of the residual stream increases across layers but does not seem to exceed $B = 10^4$. Combining these into our formula, we find that a sufficient value of Ω is

$$\begin{aligned}\Omega &= \ln(N/\epsilon) + 2B^2||W_{QK}|| + \max(\ln(N^{\frac{1}{2}}B||W_{OV}||_2), 0)\\ &= \ln(1024/2^{-146}) + 2(10^4)^2 \cdot 8 + \max(\ln(1024^{\frac{1}{2}}10^4 \cdot 8), 0)\\ &\approx 1.6 \times 10^9\end{aligned}$$

with almost all of the contribution due to the $2B^2||W_{QK}||$ *term.*

6 Limitations

The technique described in Theorem 6 faces several practical limitations. First is the quantity of attention heads: we use one attention head per dimension of the hidden layer, which can easily increase the number of attention heads by several orders of magnitude, partially offset by the new attention heads having smaller internal dimension. For example, each layer of GPT-3 has 96 attention heads with internal dimension 128 (Brown et al., 2020), and the process we describe would require 49152 additional 1-dimensional attention heads in each layer.

Second, it may be the case that replacing a feedforward network with attention heads slows down model inference or training. In particular, this approach replaces matrix multiplication with many vector-by-vector multiplications. One also computes many terms that are "thrown away" in the masking step. Combined, these suggest that converting an MLP layer to attention heads would increase computational costs.

Finally, the "pseudo-masking" in Theorem 9 introduces a separate set of issues into any training process due to the large Ω terms added to the W_{QK} matrix. Most notably, pseudo-masking would interact poorly with most forms of dropout regularization and with ℓ^2 regularization on the entries of W_{QK}.

7 Discussion

We have proven that attention heads can implement an MLP layer and in particular that any transformer can be converted to an attention-only transformer. One implication of these results is that it is theoretically possible to train an attention-only transformer that matches the performance of an MLP-plus-attention transformer. It remains unknown whether such an architecture would be competitive with the more classical transformer architecture in terms of practical considerations like training or inference speed. Such a test would be a promising future area of research.

Our foremost hope in this work is to facilitate the advancement of mechanistic interpretability approaches such as Elhage et al. (2021), which found the most success in transformers without MLP layers, but found that a complete understanding of transformers "will require progress on MLP layers". Our technique could allow one to reuse the techniques that are successful on attention heads on the MLP layers.

In doing so, the primary impediment is scale since the approach described in this paper increases the number of attention heads in a transformer by several orders of magnitude. However, this is itself a useful new perspective on the difficulty of interpreting MLP layers: MLP layers in a model like GPT-3 are larger than attention layers by a 2:1 margin if one measures by number of parameters but by 500:1 if one measures by number of attention heads. It may be the case that the AI capabilities slogan "scale is all you need" applies equally to mechanistic interpretability.

References

Jimmy Lei Ba, Jamie Ryan Kiros, and Geoffrey E. Hinton. Layer normalization, 2016.

Tom B. Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, Sandhini Agarwal, Ariel Herbert-Voss, Gretchen Krueger, Tom Henighan, Rewon Child, Aditya Ramesh, Daniel M. Ziegler, Jeffrey Wu, Clemens Winter, Christopher Hesse, Mark Chen, Eric Sigler, Mateusz Litwin, Scott Gray, Benjamin Chess, Jack Clark, Christopher Berner, Sam McCandlish, Alec Radford, Ilya Sutskever, and Dario Amodei. Language models are few-shot learners, 2020.

Arthur Conmy, Augustine N Mavor-Parker, Aengus Lynch, Stefan Heimersheim, and Adrià Garriga-Alonso. Towards automated circuit discovery for mechanistic interpretability. *arXiv preprint arXiv:2304.14997*, 2023.

Marcella Cornia, Matteo Stefanini, Lorenzo Baraldi, and Rita Cucchiara. Meshed-memory transformer for image captioning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2020.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. Bert: Pre-training of deep bidirectional transformers for language understanding, 2019.

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*, pp. 2793–2803. PMLR, 2021.

Nelson Elhage, Neel Nanda, Catherine Olsson, Tom Henighan, Nicholas Joseph, Ben Mann, Amanda Askell, Yuntao Bai, Anna Chen, Tom Conerly, Nova DasSarma, Dawn Drain, Deep Ganguli, Zac Hatfield-Dodds, Danny Hernandez, Andy Jones, Jackson Kernion, Liane Lovitt, Kamal Ndousse, Dario Amodei, Tom Brown, Jack Clark, Jared Kaplan, Sam McCandlish, and Chris Olah. A mathematical framework for transformer circuits. *Transformer Circuits Thread*, 2021. https://transformer-circuits.pub/2021/framework/index.html.

Stefan Heimersheim and Alex Turner. Residual stream norms grow exponentially over the forward pass, 2023. URL https://www.alignmentforum.org/posts/8mizBCm3dyc432nK8/residual-stream-norms-grow-exponentially-over-the-forward. Accessed: 2023-09-04.

Dan Hendrycks and Kevin Gimpel. Gaussian error linear units (gelus), 2023.

IEEE. Ieee standard for floating-point arithmetic. *IEEE Std 754-2008*, pp. 1–70, 2008. doi: 10.1109/IEEESTD.2008.4610935.

Salman Khan, Muzammal Naseer, Munawar Hayat, Syed Waqas Zamir, Fahad Shahbaz Khan, and Mubarak Shah. Transformers in vision: A survey. *ACM Comput. Surv.*, 54(10s), sep 2022. ISSN 0360-0300. doi: 10.1145/3505244. URL https://doi.org/10.1145/3505244.

Peter J. Liu, Mohammad Saleh, Etienne Pot, Ben Goodrich, Ryan Sepassi, Lukasz Kaiser, and Noam Shazeer. Generating wikipedia by summarizing long sequences, 2018.

Beren Millidge and Eric Winsor. Basic facts about language model internals, 2023. URL https://www.alignmentforum.org/posts/PDLfpRwSynu73mxGw/basic-facts-about-language-model-internals-1#Weights_Are_Nearly_Gaussian_. Accessed: 2023-09-04.

Niki Parmar, Ashish Vaswani, Jakob Uszkoreit, Lukasz Kaiser, Noam Shazeer, Alexander Ku, and Dustin Tran. Image transformer. In Jennifer Dy and Andreas Krause (eds.), *Proceedings of the 35th International Conference on Machine Learning*, volume 80 of *Proceedings of Machine Learning Research*, pp. 4055–4064. PMLR, 10–15 Jul 2018. URL https://proceedings.mlr.press/v80/parmar18a.html.

Alec Radford, Karthik Narasimhan, Tim Salimans, Ilya Sutskever, et al. Improving language understanding by generative pre-training. *OpenAI blog*, 2018.

Alec Radford, Jeffrey Wu, Rewon Child, David Luan, Dario Amodei, Ilya Sutskever, et al. Language models are unsupervised multitask learners. *OpenAI blog*, 1(8):9, 2019.

Sainbayar Sukhbaatar, Edouard Grave, Guillaume Lample, Herve Jegou, and Armand Joulin. Augmenting self-attention with persistent memory, 2019.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Lukasz Kaiser, and Illia Polosukhin. Attention is all you need, 2023.

Kevin Wang, Alexandre Variengien, Arthur Conmy, Buck Shlegeris, and Jacob Steinhardt. Interpretability in the wild: a circuit for indirect object identification in gpt-2 small, 2022.