The Hidden Attention of Mamba Models

Ameen Ali*, Itamar Zimerman*, and Lior Wolf

School of Computer Science, Tel Aviv University

Abstract. The Mamba layer offers an efficient selective state space model (SSM) that is highly effective in modeling multiple domains including NLP, long-range sequences processing, and computer vision. Selective SSMs are viewed as dual models, in which one trains in parallel on the entire sequence via IO-aware parallel scan, and deploys in an autoregressive manner. We add a third view and show that such models can be viewed as attention-driven models. This new perspective enables us to compare the underlying mechanisms to that of the self-attention layers in transformers and allows us to peer inside the inner workings of the Mamba model with explainability methods. Our code is publicly available[1].

1 Introduction

Recently, Selective State Space Layers [23], also known as Mamba models, have shown remarkable performance in diverse applications including language modeling [4,23,44,56], image processing [33,64], video processing [62], medical imaging [22,32,37,46,57,58,60], tabular data [2], point-cloud analysis [31], graphs [9, 54], N-dimensional sequence modeling [30] and more. Characterized by their linear complexity in sequence length during training and fast RNN-like computation during inference (left and middle panels of Fig. 1), Mamba models offer a 5x increase in the throughput of Transformers for auto-regressive generation and the ability to efficiently handle long-range dependencies.

Despite their growing success, the information-flow dynamics between tokens in Mamba models and the way they learn remain largely unexplored. Critical questions about their learning mechanisms, particularly how they capture dependencies and their resemblance to other established layers like RNNs, CNNs, or attention mechanisms, remain unanswered. Additionally, the lack of interoperability methods for these models may pose a significant hurdle to debugging them and may also reduce their applicability in socially sensitive domains in which explainability is required.

Motivated by these gaps, our research aims to provide insights into the dynamics of the Mamba models and develop methodologies for their interpretation. While the traditional views of state-space models are through the lens of convolutional or recurrent layers [25], we show that selective state-space layers are a form

* These authors contributed equally to this work.

[1] https://github.com/AmeenAli/HiddenMambaAttn

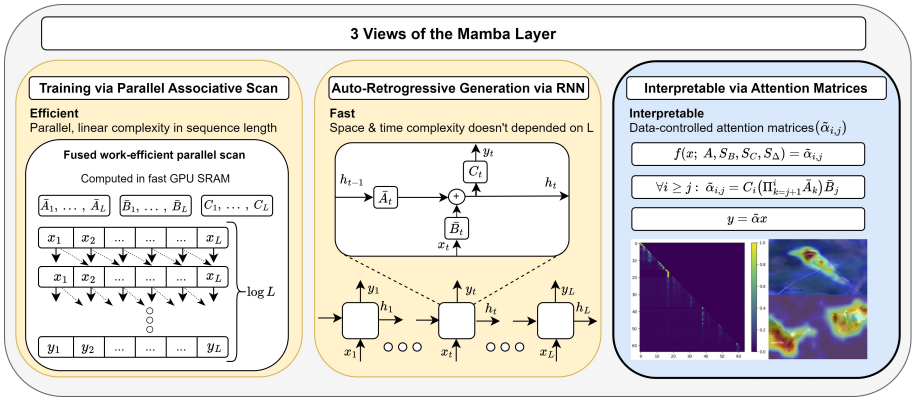


Fig. 1: Three Perspectives of the Selective State-Space Layer:**(Left)** Selective State-Space Models (SSMs) can be efficiently computed with linear complexity using parallel scans, allowing for effective parallelization on modern hardware, such as GPUs. **(Middle)** Similar to SSMs, the selective state-space layer can be computed via a time-variant recurrent rule. **(Right)** A new view of the selective SSM layer, showing that it uses attention similarly to transformers (see Eq. 12). Our view enables the generation of attention maps, offering valuable applications in areas like XAI.

of attention models. This is achieved through a novel reformulation of Mamba computation using a data-control linear operator, unveiling hidden attention matrices within the Mamba layer. This enables us to employ well-established interpretability and explainability techniques, commonly used in transformer realms, to devise the first set of tools for interpreting Mamba models. Furthermore, our analysis of implicit attention matrices offers a direct framework for comparing the properties and inner-representations of transformers [53] and selective-state space layers.

Our main contributions encompass the following main aspects: (i) We shed light on the fundamental nature of Mamba models, by showing that they rely on implicit attention, which is implemented by a unique data-control linear operator, as illustrated in Fig. 1 (right). (ii) Our analysis reveals that Mamba models give rise to three orders of magnitude more attention matrices than transformers. (iii) We provide a set of explainability and interpretability tools based on these hidden attention matrices. (iv) In the domain of computer vision, for comparable model sizes, Mamba model-based attention shows comparable explainability metrics results to that of transformers.

2 Background

Transformers The Transformer architecture [53] is the dominant architecture in the recent NLP and Computer Vision literature. It relies on self-attention to capture dependencies between different tokens. Self-attention allows these models to dynamically focus on different parts of the input sequence, calculating

the relevance of each part to others. It can be computed as follows:

$$Self - Attention(Q, K, V) = \alpha V, \quad \alpha = \text{softmax}\left(\frac{QK^T}{\sqrt{d_k}}\right) \tag{1}$$

where Q, K, and V represent queries, keys, and values, respectively, and d_k is the dimension of the keys. Additionally, the Transformer utilizes H attention heads to parallelly process information, allowing the model to capture various dependencies. The attention matrix α enables the models to weigh the importance of tokens based on their contribution to the context, and they can also used for interpretability [7], explainability [13], and improved classification [14,52].

State-Space Layers State-Space Layers were first introduced in [25] and have seen significant improvements through the seminal work in [24]. These layers have demonstrated promising results across several domains, including NLP [19,40], audio generation [21], image processing [8,42,61], long video understanding [55], RL [15,34], speech recognition [47], and more. Given one channel of the input sequence $x := (x_1, \cdots, x_L)$ such that $x_i \in \mathbb{R}$, these layers can be implemented using either recurrence or convolution. The recurrent formulation, which relies on the recurrent state $h_t \in \mathbb{R}^N$ where N is the state size, is defined as follows: given the discretization functions f_A, f_B, and parameters A, B, C and Δ, the recurrent rule for the SSM is:

$$\bar{A} = f_A(A, \Delta), \quad \bar{B} = f_B(A, B, \Delta), \quad h_t = \bar{A}h_{t-1} + \bar{B}x_t, \quad y_t = Ch_t \tag{2}$$

This recurrent rule can be expanded as:

$$h_t = \bar{A}^t\bar{B}x_0 + \bar{A}^{t-1}\bar{B}x_1 + \cdots + \bar{B}x_t, \quad y_t = C\bar{A}^t\bar{B}x_0 + C\bar{A}^{t-1}\bar{B}x_1 + \cdots + C\bar{B}x_t \tag{3}$$

Since the recurrence is linear, Eq. 3 can also be expressed as a convolution, via a convolution kernel $K := (k_1, \cdots, k_L)$, where $k_i = C\bar{A}^{i-1}\bar{B}$. Thus, allowing sub-quadratic complexity in sequence length. The equivalence between the recurrent and convolution provides a versatile framework that enables parallel and efficient training with sub-quadratic complexity with the convolution view, alongside a faster recurrent view, facilitating the acceleration of autoregressive generation by decoupling step complexity from sequence length. As the layer defined as a map from $\mathbb{R}^{\mathbb{L}}$ to $\mathbb{R}^{\mathbb{L}}$, to processes D channels the layer employ D independent copies of itself.

S6 Layers A recent development in state space layers is selective SSMs [23] (S6), which show outstanding performance in NLP [4,44,56], vision [33,64], graph classification [9,54], and more. These models rely on time-variant state space layers, namely, the discrete matrices $\bar{A}, \bar{B}$, and C of each channel are modified over the L time steps depending on the input sequence. As opposed to traditional state-space layers, which operate individually on each channel, selective state-space layers compute the SSM matrices $\bar{A}_i, \bar{B}_i, C_i$ for all $i \leq L$ based on all the channels, and then apply the time-variant recurrent rule individually for each channel. Hence, we denote the enitre input sequence by $\hat{x} := (\hat{x}_1, \cdots, \hat{x}_L) \in$

$\mathbb{R}^{L\times D}$ where $\hat{x}_i \in \mathbb{R}^D$. The per-time discrete matrices $\bar{A}_i, \bar{B}_i$, and C_i are defined as follows:

$$B_i = S_B\hat{x}_i, \quad C_i = S_C\hat{x}_i, \quad \Delta_i = \text{softplus}(S_\Delta\hat{x}_i) \tag{4}$$

$$f_A(\Delta_i, A) = \exp(\Delta_i A), \quad f_B(\Delta_i, A, B_i) = \Delta_i B \tag{5}$$

$$\bar{A}_i = f_A(\Delta_i, A), \quad \bar{B}_i = f_B(\Delta_i, A, B_i) \tag{6}$$

where f_A, f_B represents the discretization rule, S_A, S_B, S_Δ are linear projection layers, and SoftPlus is an element-wise function that is a smooth approximation of ReLU. While previous state-space layers employ complex-valued SSMs and non-diagonal matrices, Mamba employs real-diagonal parametrization for the system matrices.

The motivation for input-dependent time-variant layers is to make those recurrent layers more expressive and flexible, allowing them to capture more complex dependencies. While other input-dependent time-variant mechanisms have been proposed in previous works through gated RNNs, S5 layer [49], or adaptive filtering via input-dependent IIR filters [36], Mamba significantly improves on these layers by presenting a flexible, yet still efficient, approach. This efficiency was achieved via the IO-aware implementation of associative scans, which can be parallelized on modern hardware via work-efficient parallel scanners [11, 39].

Mamba The Mamba block is built on top of the selective state-space layer, Conv1D and other elementwize operators. Inspired by the architecure of Gated MLP and H3 [19], and given an input $\hat{x}' := (\hat{x}'_1, \cdots \hat{x}'_L)$ it is defined as follows:

$$\hat{x} = \text{SiLU}(\ \text{Conv1D}(\ \text{Linear}(\hat{x}')\ )\ ), \quad \hat{z} = \text{SiLU}(\ \text{Linear}(\hat{x}')\ ) \tag{7}$$

$$\hat{y}' = \text{Linear}(\text{Selective SSM}(\hat{x'}) \otimes \hat{z})), \quad \hat{y} = \hat{y}' + \hat{x}', \quad \ddot{y} = \text{LayerNorm}(\hat{y}) \tag{8}$$

where $\otimes$ is elementwise multiplication. Mamba models contains Λ stacked mamba blocks and D channels per block, and we denote the tensors in the i-th block and j-th channel with a superscript, where the first index refers to the block number.

Inspired by the vision transformer ViT [16], both [33,64] replace the standard self-attention mechanism by two Mamba layers, where each layer is applied in a bidirectional manner. The resulting model achieves favorable results compared to the standard ViT in terms of both accuracy and efficiency, when comparing models with the same number of parameters.

Explainability Explainability methods have been extensively explored in the context of deep neural networks, particularly in domains such as natural language processing (NLP) [1, 3, 5, 12, 13, 63], computer vision [6, 27, 41, 48, 51], and attention-based models [3, 12, 13, 63].

The contributions most closely aligned with ours are those specifically tailored for transformer explainability. Abnar and Zuidema [1] introduce the Attention-Rollout method, which aggregates the attention matrices across different layers by analyzing the paths in the inter-layer pairwise attention graph. Chefer et al. [12,13] combine LRP scores [6] with the attention gradients in order to obtain a class-specific relevance scores. Ali et al. [3] enhanced attributions by treating the non-linear Softmax and LayerNorm operators as a constant, thereby attributing relevance exclusively through the value path, disregarding these operators Yuan et al. [63] treats the output token representations as states in a Markov chain in which the transition matrix is built using attention weights.

Our work performs similar attention-based analysis for Mamba and we derive versions of [1,13] that are suitable for such SSM models.

3 Method

In this section, we detail our methodology. First, in section 3.1, we reformulate selective state-space (S6) layers as self-attention, enabling the extraction of attention matrices from S6 layers. Subsequently, in sections 3.2 and 3.3, we demonstrate how these hidden attention matrices can be leveraged to develop class-agnostic and class-specific tools for explainable AI of Mamba models.

3.1 Hidden Attention Matrices In Selective State Spaces Layers

Given the per-channel time-variant system matrices $\bar{A}_1, \cdots, \bar{A}_L$, $\bar{B}_1, \cdots, \bar{B}_L$, and $C_1, \cdots, C_L$ from Eq. 4 and 6, each channel within the selective state-space layers can be processed independently. Thus, for simplicity, the formulation presented in this section will proceed under the assumption that the input sequence x consists of a single channel.

By considering the initial conditions $h_0 = 0$, unroling Eq. 2 for one channel yields:

$$h_1 = \bar{B}_1 x_1, \quad y_1 = C_1 \bar{B}_1 x_1 \tag{9}$$

$$h_2 = \bar{A}_2 \bar{B}_1 x_1 + \bar{B}_2 x_2, \quad y_2 = C_2 \bar{A}_2 \bar{B}_1 x_1 + C_2 \bar{B}_2 x_2 \tag{10}$$

and in general:

$$h_t = \sum_{j=1}^{t} \left(\Pi_{k=j+1}^{t} \bar{A}_k \right) \bar{B}_j x_j, \quad y_t = C_t \sum_{j=1}^{t} \left(\Pi_{k=j+1}^{t} \bar{A}_k \right) \bar{B}_j x_j \tag{11}$$

By converting Eq. 11 into a matrix form we get:

$$y = \tilde{\alpha} x, \quad \begin{bmatrix} y_1 \\ y_2 \\ \vdots \\ y_L \end{bmatrix} = \begin{bmatrix} C_1\bar{B}_1 & 0 & \cdots & 0 \\ C_2\bar{A}_2\bar{B}_1 & C_2\bar{B}_2 & \cdots & 0 \\ \vdots & \vdots & \ddots & 0 \\ C_L\Pi_{k=2}^{L}\bar{A}_k\bar{B}_1 & C_L\Pi_{k=3}^{L}\bar{A}_k\bar{B}_2 & \cdots & C_L\bar{B}_L \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ \vdots \\ x_L \end{bmatrix} \tag{12}$$

Hence, the S6 layer can be viewed as a data-controlled linear operator [45], where the matrix $\tilde{\alpha} \in \mathbb{R}^{L\times L}$ is a function of the input and the parameters A, S_B, S_C, S_Δ. The element at row i and column j captures how x_j influences y_i, and is computed by:

$$\tilde{\alpha}_{i,j} = C_i\Big(\Pi_{k=j+1}^{i}\bar{A}_k\Big)\bar{B}_j \tag{13}$$

Eq. 12 and 13 links $\tilde{\alpha}$ to the conventional standard attention matrix (Eq. 1), and highlights that S6 can be considered a variant of causal self-attention.

Simplifying and Interpreting the Hidden Matrices Since $\bar{A}_t$ is a diagonal matrix, the different N coordinates of the state h_t in Eq. 11 do not interact when computing h_{t+1}. Thus, Eq. 11 (left) can be computed independently for each coordinate $m \in \{1, 2, \ldots, N\}$:

$$h_t[m] = \sum_{j=1}^{t}\big(\Pi_{k=j+1}^{t}\bar{A}_k[m,m]\big)\bar{B}_j[m]x_j, \quad y_t = \sum_{m=1}^{N} C_t[m]h_t[m] \tag{14}$$

where $C_i[m], A_k[m,m], B_j[m] \in \mathbb{R}$, plugging it into Eq. 13 yiels:

$$\tilde{\alpha}_{i,j} = C_i\Big(\Pi_{k=j+1}^{i}\bar{A}_k\Big)\bar{B}_j = \sum_{m=1}^{N} C_i[m]\Big(\Pi_{k=j+1}^{i}\bar{A}_k[m,m]\Big)\bar{B}_j[m] \tag{15}$$

Note that while equations 11, and 13 contains matrix multiplication, Eq. 14 relies on element-wise multiplication.

An interesting observation arising from Eq. 15 is that a single channel of S6 produces N inner attention matrices, which can be computed by:

$$\forall m \in [N] : C_i[m]\Big(\Pi_{k=j+1}^{i}\bar{A}_k[m,m]\Big)\bar{B}_j[m] \tag{16}$$

These N per-coordinate attention matrices are aggregated in Eq. 15 into a single global attention matrix $\tilde{\alpha}$, per channel. However, in the Transformer, a single attention matrix is produced by each of the H attention heads. Given that the number of channels in Mamba models D is typically a hundred times greater than the number of heads in a transformer (for example, Vision-Mamba-Tiny has $D = 384$ channels, compared to $H = 3$ heads in DeiT-Tiny), the Mamba layer generates approximately $\frac{DN}{H} \approx 100N$ more attention matrices than the original self-attention layer.

To further understand the structure and characterization of these attention matrices, we will express those hidden attention matrices $\tilde{\alpha}$ for each channel d as a direct function of the input $\hat{x}$. To to so, we first substitute Eq.4,5 and Eq.6 into Eq. 13, and obtain:

$$\tilde{\alpha}_{i,j} = S_C\hat{x}_i\Big(\Pi_{k=j+1}^{i}\exp\Big(\text{softplus}(S_\Delta\hat{x}_k)A\Big)\Big)\text{softplus}(S_\Delta\hat{x}_j)S_B\hat{x}_j = \tag{17}$$

$$S_C\hat{x}_i\Big(\exp\Big(\sum_{k=j+1}^{i}\text{softplus}(S_\Delta\hat{x}_k)\Big)A\Big)\text{softplus}(S_\Delta\hat{x}_j)S_B\hat{x}_j \tag{18}$$

For simplicitly, we propose a simplification of Eq. 18 by substituting the softplus function with the ReLU function, and summing only over positive elements:

$$\tilde{\alpha}_{i,j} \approx S_C\hat{x}_i\left(\exp\left(\sum_{\substack{k=j+1\\ S_\Delta\hat{x}_k>0}}^{i}(S_\Delta\hat{x}_k)\right)A\right)\text{ReLU}(S_\Delta\hat{x}_j)S_B\hat{x}_j \tag{19}$$

Consider the following query/key/value notation:

$$\tilde{Q}_i := S_C\hat{x}_i,\ \ \tilde{K}_j := \text{ReLU}(S_\Delta\hat{x}_j)S_B\hat{x}_j,\ \ \tilde{H}_{i,j} := \exp\Big(\sum_{\substack{k=j+1\\ S_\Delta\hat{x}_k>0}}^{i}(S_\Delta\hat{x}_k)\Big)A \tag{20}$$

Eq. 19 can be further simplified to:

$$\tilde{\alpha}_{i,j} \approx \tilde{Q}_i\tilde{K}_j\tilde{H}_{i,j} \tag{21}$$

This formulation enhances our understanding of the Mamba's attention mechanism. Whereas traditional self-attention captures the influence of x_j on x_i through the dot products between Q_i and K_j, Mamba's approach correlates this influence with $\tilde{Q}_i$ and $\tilde{K}_j$, respectively. Additionally, $\tilde{H}_{i,j}$ controls the significance of the recent $i-j$ tokens, encapsulating the continuous aggregated historical context spanning from x_j to x_i.

This distinction between the self-attention and Mamba, captured by $\tilde{H}_{i,j}$ could be a key factor in enabling Mamba-based models to understand and utilize continuous historical context within sequences more efficiently than Transforme models.

Furthermore, Eq. 21, and 20 offer further insights into the characterization of the hidden attention matrices by demonstrating that the only terms modified across channels are A and Δ_i, which influence the values of $\tilde{H}_{i,j}$ and$\tilde{K}_j$ through the discretization rule in Eq. 5. Hence, all the hidden attention matrices follow a common pattern, distinguished by the keys $\tilde{K}_j$ via Δ_i and the significance of the history $\tilde{H}_{i,j}$ via A and Δ_i.

A distinct divergence between Mamba's attention mechanism and traditional self-attention lies in the latter's utilization of a per-row softmax function. It's

essential to recognize that various attention models have either omitted the softmax [35] or substituted it with element-wise neural activations [28,38,59,65], achieving comparable outcomes to the original framework.

3.2 Application to Attention Rollout

As our class-agnostic explainability teqchniqe for Mamba models, we built our method on top of the Attention-Rollout [1] method. For simplicity, we assume that we deal with a vision mamba model, which operates on sequences of size $L+1$, where L is the sequence length obtained from the $\sqrt{L} \times \sqrt{L}$ image patcehs, with a classification (CLS) token appended to the sequence's end.

To do so, for each sample, we first extract the hidden attention matrix $\tilde{\alpha}^{\lambda,d}$ for any channel $d \in [D]$ and layer $\lambda \in [\Lambda]$ according to the formulation in section 3.1 (Eq. 12), such that $\tilde{\alpha}^{\lambda,d} \in \mathbb{R}^{(L+1)\times(L+1)}$

Attention-Rollout is then applied as follows:

$$\forall \lambda \in [\Lambda]: \quad \tilde{\alpha}^{\lambda} = \mathbb{I}_{L+1} + \mathop{\mathbb{E}}_{d\in[D]}(\tilde{\alpha}^{\lambda,d}) \tag{22}$$

where $\tilde{\alpha}^{\lambda} \in \mathbb{R}^{(L+1)\times(L+1)}$ and $\mathbb{I}_{L+1} \in \mathbb{R}^{(L+1)\times(L+1)}$ is an identity matrix utilized to incorporate the influence of skip connections along the layers.

Now, the per-layer global attention matrices $\tilde{\alpha}^{\lambda}$ for all $\lambda \in [\Lambda]$ are aggregated into the final map ρ by:

$$\rho = \Pi_{\lambda=1}^{\Lambda} \tilde{\alpha}^{\lambda}, \quad \rho \in \mathbb{R}^{(L+1)\times(L+1)} \tag{23}$$

Note that each row of ρ corresponds to a relevance map for each token given the other tokens. In the context of this study, which concentrates on classification models, our attention analysis exclusively directs attention to the CLS token, thus, we derive the final relevance map from the raw associated with the CLS token in the output matrix, denoted by $\rho_{\text{CLS}} \in \mathbb{R}^{L}$, which contains the relevance scores evaluating each token's influence on the classification token. Finally, to get the final explanation heatmap we reshape $\rho_{\text{CLS}} \in \mathbb{R}^{L}$ to $\sqrt{L} \times \sqrt{L}$ and upsample it back to the size of the original image using bilinear interpolation.

Although Mamba models are causal by defintion, resulting in causal hidden attention matrices, our method can be straightforwardly extended to a bidirectional setting. This adaptation involves modifying Eq. 22 so that $\tilde{\alpha}^{\lambda,d}$ becomes the outcome of summing the (two) per-direction matrices of the λ-layer and the d-channel.

3.3 Application to Attention-based Attribution

As our class-specific explainability technique for Mamba models, we have tailored the Transformer-Attribution [13] explainability method, which is specifically designed for transformers, to suit Mamba models. This method relies on a combination of LRP scores and attention gradients to generate the relevance scores. Since each Mamba block includes several peripheral layers that are not

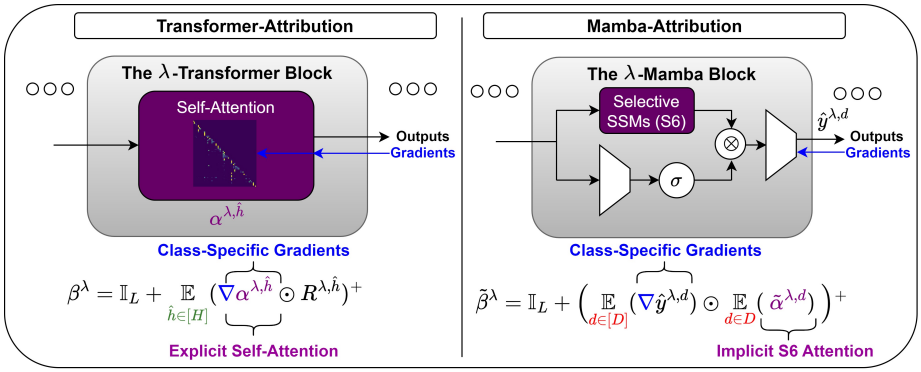


Fig. 2: Comperative Visualization of Transformer-Attribution (Left), and our Mamba-Attribution (Right).

included in transformers, such as Conv1D, additional gating mechanisms, and multiple linear projection layers, a robust mechanism must be carefully designed. For simplicity, we focus on vision Mamba, with a grid of $\sqrt{L}$ patches in each row and column, as in Sec. 3.2.

The Transformer-Attribution method encompasses two stages: (i) generating a relevance map for each attention layer, followed by (ii) the aggregation of these relevance maps across all layers, using the aggregation rule specified in 23, to produce the final map ρ.

The difference from the attention rollout method therefore lies in how step (i) is applied to each Mamba layer $\lambda \in [\Lambda]$. For the $\hat{h} \in [H]$ attention head at layer λ, the transformer method [13] computes the following two maps: (1) LRP [6] relevance scores map $R^{\lambda,\hat{h}}$, and (2) the gradients $\nabla\tilde{\alpha}^{\lambda,\hat{h}}$ with respect to a target class of interest. Then, these two are fused by a Hadamard product:

$$\beta^{\lambda} = \mathbb{I}_L + \mathop{\mathbb{E}}_{\hat{h}\in[\hat{H}]}(\nabla\alpha^{\lambda,\hat{h}} \odot R^{\lambda,\hat{h}})^{+} \tag{24}$$

where $\mathbb{I}_{L+1} \in \mathbb{R}^{(L+1)\times(L+1)}$ is an identity matrix.

Our method, **Mamba-Attribution**, depicted in Fig. 2 (right), deviates from this method by modifying Eq. 24 in the following aspects: (i) Instead of computing the gradients on the per-head attention matrices $\nabla\alpha^{\lambda,\hat{h}}$, we compute the gradients of $\nabla\hat{y}^{\lambda,d}$. The motivation for these modifications is to exploit the gradients of both the S6 mixer and the gating mechanism in Eq. 8 (left), to obtain strong class-specific maps. (ii) We simply replace $R^{\lambda,\hat{h}}$ with the attention matrices $\tilde{\alpha}^{\lambda,d}$ at layer λ and channel d, since we empirically observe that those attention matrices produce better relevance maps. Both of these modifications are manifested by the following form, which defines our method:

$$\tilde{\beta}^{\lambda} = \mathbb{I}_L + \Big(\mathop{\mathbb{E}}_{d\in D}(\nabla\hat{y}^{\lambda,d}) \odot \mathop{\mathbb{E}}_{d\in D}(\tilde{\alpha}^{\lambda,d})\Big)^{+} \tag{25}$$

4 Experiments

In this section, we present an in-depth analysis of the hidden attention mechanism embedded within Mamba models, focusing on its semantic diversity and applicability in explainable AI frameworks. We start by visualize the hidden attention matrices for both NLP and vision models in Sec. 4.1, followed by empirically assessing our xplainable AI techniques via perturbation and segmentation tests in section 4.2.

4.1 Visualization of Attention Matrices

To better understand the hidden attention mechanism employed in Mamba, as described in Sec. 3.1 and manifested in Eq. 18 and 19, Fig. 3 and 4 containes a comparative visualization of attention matrices in Mamba and Transformer on both vision and NLP tasks. For clearer visualization, we apply the Softmax function to each row of the attention matrices and limit our focus to the initial 64 tokens.

NLP In fig. 3 we compare attention matrices extracted from both Mamba (130m) and Transformer (Pythia-160m [10]) language models, trained on the Pile [20] dataset for next token prediction. The attention maps are extracted using examples from the Lambada dataset (preprocessed by OpenAI).

As can be seen, the hidden attention matrices of Mamba (the first 4 columns) appear similar to the attention matrices extracted from transformers (the last column). In both Mamba and transformers, the dependencies between far-away tokens are captured in the deeper layers of the model, as depicted in the lower rows.

Some of the attention matrices demonstrate the ability of selective SSM models and transformers to focus on parts of the input. In those cases, instead of the diagonal patterns, some columns seem to miss the diagonal element and the attention is more diffused (recall that we normalized the Mamba attention maps for visualization purposes. In practice, these columns have little activity).

Vision Fig. 4 contains a comparative visualization of attention matrices in Vision-Mamba and ViT (DeiT), for models of tiny size, trained on ImageNet-1K. The attention maps are extracted using examples from the test set. Each Mamba attention matrix is obtained by combining the two maps of the bidirectional channel.

Evidently, both the Mamba attention matrices and the transformer attention matrices possess similar properties and depict the two-dimensional structure within the data as bands with an offset of $\sqrt{L}$.

4.2 Explainability Metrics

The explainable AI experiments include three types of explainability methods: (1) Raw-Attention, which employs raw attention scores as relevancies. Our findings indicate that averaging the attention maps across layers yields optimal results. (2) Attn-Rollout [1] for Transformers, and its Mamba version, as depicted

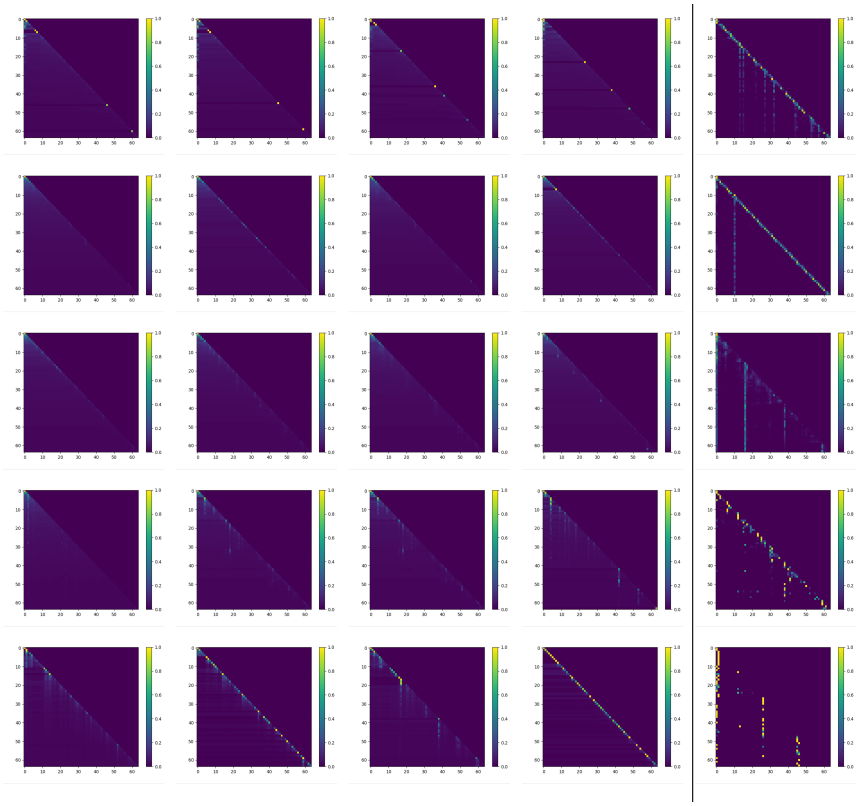


Fig. 3: Hidden Attention Matrices (NLP): The first four columns correspond to the Mamba matrices, while the last column is dedicated to the Transformer. Each row represents a different layer within the models, showcasing the evolution of the attention matrices at 0% (top), 25%, 50%, 75%, and 100% (bottom) of the layer depth.

in section 3.2. Finally, (3) The Transformer Attribution of Chefer et al. [12] and its Mamba Attribution counterpart, detailed in Section 3.3.

Fig. 5 depicts the results of the six attribution methods on typical samples from the ImageNet test set. As can be seen, the Mamba-based heatmaps are often more complete than the transformer-based counterparts. The raw attention of Mamba stands out of the other five heatmaps since it is depicts activity across the entire image. However, the relevant object is highlighted.

Pertubation Tests In this evaluation framework, we employ an input perturbation scheme to assess the efficacy of various explanation methods, following the approach outlined by [12, 13].

These experiments are conducted under two distinct settings. In the positive perturbation scenario, a quality explanation involves an ordered list of pixels, sorted from the most- to the least-relevant. Consequently, when gradually mask-

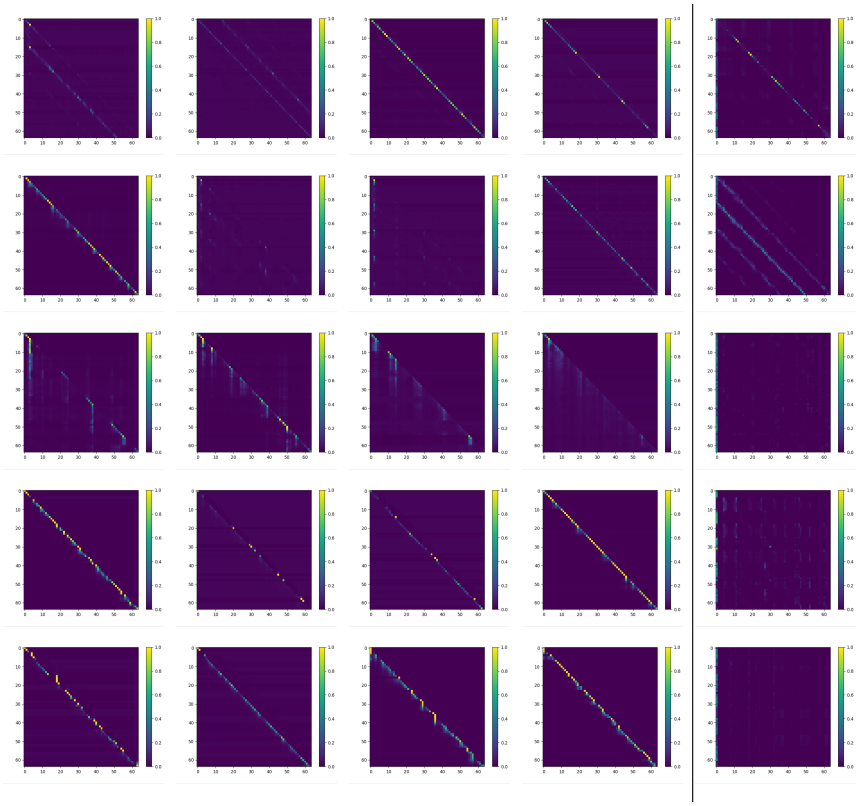


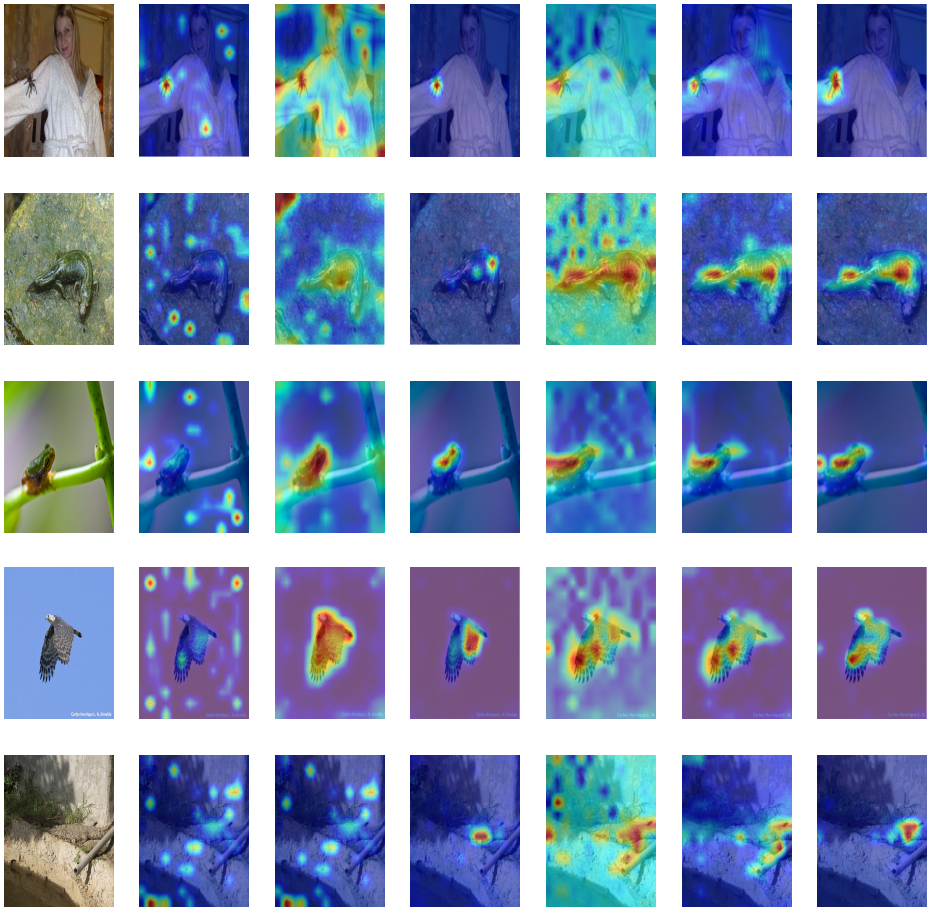
Fig. 4: Hidden Attention Matrices (Vision): The first four columns correspond to the Mamba matrices, while the last column is dedicated to the Transformer. Each row represents a different layer within the models, showcasing the evolution of the attention matrices at 0% (top), 25%, 50%, 75%, and 100% (bottom) of the layer depth.

ing out the pixels of the input image, starting from the highest relevance to the lowest, and measuring the mean top-1 accuracy of the network, one anticipates a notable decrease in performance.

Conversely, in the negative perturbation setup, a robust explanation is expected to uphold the accuracy of the model while systematically removing pixels, starting from the lowest relevance to the highest.

In both cases, the evaluation metrics consider the area-under-curve (AUC) focusing on the erasure of 10% to 90% of the pixels.

The results of the perturbations are presented in Table. 1, depicting the performance of different explanation methods under both positive and negative perturbation scenarios across the two models. In the positive perturbation scenario, where lower AUC values are indicative of better performance, we notice that Raw-Attention, Mamba shows a better AUC (17.268 vs. 20.687) compared to the Vision Transformer. For the Attn-Rollout method, Mamba out-



(a) (b) (c) (d) (e) (f) (g)

Fig. 5: Qualitative results for the different explanation methods for the ViT-small and the Mamba-small models. (a) the original image, (b) the aggregated Raw-Attention of ViT-Small, (c) Attention Rollout for ViT-Small, (d) Transformer-Attribution for ViT-Small, (e) the Raw-Attention of Mamba-Small, (f) Attention-Rollout of Mamba-Small and (g) the Mamba-Attribution method for the Mamba-Small model.

performs the Vision Transformer with an AUC of 18.806 vs. 20.594, while the latter shows a better AUC of 15.351 vs. 16.619 under the Attribution method. In the negative perturbation scenario where higher AUC values are better, the Transformer-based methods consistently outperform Mammba across all three methods: Raw-Attention (40.766 vs. 34.025), Attn-Rollout (43.525 vs. 41.864), and Attribution (48.089 vs. 39.632). The tendency for lower AUC in both positive (where it is desirable) and negative perturbation (where it is undesirable)

	Positive Perturbation		Negative Perturbation	
	Mamba	Transformer	Mamba	Transformer
Raw-Attention	17.268	20.687	34.025	40.766
Attn-Rollout	18.806	20.594	41.864	43.525
Attribution	16.619	**15.351**	39.632	**48.089**

Table 1: Positive and Negative perturbation AUC results (percents) for the predicted class on the ImageNet validation set. For positive perturbation lower is better, and for negative perturbation higher is better.

may indicate that the Mamba model is more sensitive to blacking out of patches, and it would be interesting to add experiments in which the patches are blurred instead, following [18].

Segmentation Tests It is expected that an effective explainability method would produce reasonable foreground segmentation maps. This is assessed for ImageNet classifiers by comparing the obtained heatmap against the ground truth segmentation maps available in the ImageNet-Segmentation dataset [26].

Evaluation is conducted based on pixel accuracy, mean-intersection-over-union (mIoU) and mean average precision (mAP) metrics, aligning with established benchmarks in the literature for explainability [12, 13, 27, 41].

The results are outlined in Table 2. Raw-Attention, Mamba demonstrates significantly higher pixel accuracy (67.64% vs. 59.69%) and mean Intersection over Union (45.09% vs. 36.94%) compared to Vision Transformer, while the latter performs better in mean Average Precision (77.25% vs. 74.88%). Under the Attn-Rollout method, Mamba outperforms Vision Transformer in mean Average Precision (80.78% vs. 80.34%), pixel accuracy (71.01% vs. 66.84%) and mean Intersection over Union (51.51% vs. 47.85). Finally, Transformer-Attribution consistently surpasses Mamba-Attribution, achieving the highest scores in pixel accuracy (79.26% vs. 74.72%), mean Average Precision (84.85% vs. 81.70%), and mean Intersection over Union (60.63% vs. 54.24%), respectively.

These results underscore the potential of Mamba's attention mechanism as approaching and sometimes passing the interoperability level of Transformer models, especially when the attention maps are taken as is. It also highlights the applicability of Mamba models for downstream tasks such as weakly supervised segmentation. It seems, however, that the Mamba-based attribution model, which is modeled closely after the transformer method of Chefer et al. [13] may benefit from further adjustments.

5 Conclusions

In this work, we have established a significant link between Mamba and self-attention layers, illustrating that the Mamba layer can be reformated as an

Model	Method	pixel accuracy	mAP	mIoU
Transformer	Raw-Attention	59.69	**77.25**	36.94
Mamba	Raw-Attention	**67.64**	74.88	**45.09**
Transformer	Attn-Rollout [1]	66.84	80.34	47.85
Mamba	Attn-Rollout (Sec. 3.2)	**71.01**	**80.78**	**51.51**
Transformer	Transformer-Attr [13]	**79.26**	**84.85**	**60.63**
Mamba	Mamba-Attr (Sec. 3.3)	74.72	81.70	54.24

Table 2: Segmentation performance on the ImageNet-Segmentation [26] dataset (percent). Higher is better. The upper part depicts the results for Vision Mamba-small while the lower part contains the results for Vision Transformer-Small

implicit form of causal self-attention mechanism. This directly links the highly effective Mamba layers with the transformer layers.

The parallel perspective plays a crucial role in efficient training and the recurrent perspective is essential for effective causal generation. The attention perspective plays a role in understanding the inner representation of the Mamba model. While "Attention is not Explanation" [29], attention layers have been widely used for transformer explainability. By leveraging the obtained attention matrices, we introduce the first (as far as we can ascertain) explainability techniques for Mamba models, for both task-specific and task-agnostic regimes. This contribution equips the research community with novel tools for examining the performance, fairness, robustness, and weaknesses of Mamba models, thereby paving the way for future improvements, and it also enables weakly supervised downstream tasks. Looking ahead, we plan to delve into the relationships between Mamba, Self-Attention and other recent layers such as RWKV [43], Retention [50] and Hyena [45], and develop XAI methods for LLMs relying on these layers and their corresponding vision variants [17, 66].

References

1. Abnar, S., Zuidema, W.: Quantifying attention flow in transformers. In: Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics. pp. 4190–4197 (2020) 4, 5, 8, 10, 15
2. Ahamed, M.A., Cheng, Q.: Mambatab: A simple yet effective approach for handling tabular data. arXiv preprint arXiv:2401.08867 (2024) 1
3. Ali, A., Schnake, T., Eberle, O., Montavon, G., Müller, K.R., Wolf, L.: Xai for transformers: Better explanations through conservative propagation. In: International Conference on Machine Learning. pp. 435–451. PMLR (2022) 4, 5
4. Anthony, Q., Tokpanov, Y., Glorioso, P., Millidge, B.: Blackmamba: Mixture of experts for state-space models. arXiv preprint arXiv:2402.01771 (2024) 1, 3
5. Arras, L., Montavon, G., Müller, K.R., Samek, W.: Explaining recurrent neural network predictions in sentiment analysis. In: Proceedings of the 8th Workshop on Computational Approaches to Subjectivity, Sentiment and Social Media Analysis. pp. 159–168 (2017) 4

6. Bach, S., Binder, A., Montavon, G., Klauschen, F., Müller, K.R., Samek, W.: On pixel-wise explanations for non-linear classifier decisions by layer-wise relevance propagation. PloS one **10**(7), e0130140 (2015) 4, 5, 9
7. Bahdanau, D., Cho, K., Bengio, Y.: Neural machine translation by jointly learning to align and translate. arXiv preprint arXiv:1409.0473 (2014) 3
8. Baron, E., Zimerman, I., Wolf, L.: 2-d ssm: A general spatial layer for visual transformers. arXiv preprint arXiv:2306.06635 (2023) 3
9. Behrouz, A., Hashemi, F.: Graph mamba: Towards learning on graphs with state space models. arXiv preprint arXiv:2402.08678 (2024) 1, 3
10. Biderman, S., Schoelkopf, H., Anthony, Q.G., Bradley, H., O'Brien, K., Hallahan, E., Khan, M.A., Purohit, S., Prashanth, U.S., Raff, E., et al.: Pythia: A suite for analyzing large language models across training and scaling. In: International Conference on Machine Learning. pp. 2397–2430. PMLR (2023) 10
11. Blelloch, G.E.: Prefix sums and their applications (1990) 4
12. Chefer, H., Gur, S., Wolf, L.: Generic attention-model explainability for interpreting bi-modal and encoder-decoder transformers. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 397–406 (2021) 4, 5, 11, 14
13. Chefer, H., Gur, S., Wolf, L.: Transformer interpretability beyond attention visualization. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. pp. 782–791 (2021) 3, 4, 5, 8, 9, 11, 14, 15
14. Chefer, H., Schwartz, I., Wolf, L.: Optimizing relevance maps of vision transformers improves robustness. Advances in Neural Information Processing Systems **35**, 33618–33632 (2022) 3
15. David, S.B., Zimerman, I., Nachmani, E., Wolf, L.: Decision s4: Efficient sequence-based rl via state spaces layers. In: The Eleventh International Conference on Learning Representations (2022) 3
16. Dosovitskiy, A., Beyer, L., Kolesnikov, A., Weissenborn, D., Zhai, X., Unterthiner, T., Dehghani, M., Minderer, M., Heigold, G., Gelly, S., et al.: An image is worth 16x16 words: Transformers for image recognition at scale. arXiv preprint arXiv:2010.11929 (2020) 4
17. Fan, Q., Huang, H., Chen, M., Liu, H., He, R.: Rmt: Retentive networks meet vision transformers. arXiv preprint arXiv:2309.11523 (2023) 15
18. Fong, R.C., Vedaldi, A.: Interpretable explanations of black boxes by meaningful perturbation. In: Proceedings of the IEEE international conference on computer vision. pp. 3429–3437 (2017) 14
19. Fu, D.Y., Dao, T., Saab, K.K., Thomas, A.W., Rudra, A., Ré, C.: Hungry hungry hippos: Towards language modeling with state space models. arXiv preprint arXiv:2212.14052 (2022) 3, 4
20. Gao, L., Biderman, S., Black, S., Golding, L., Hoppe, T., Foster, C., Phang, J., He, H., Thite, A., Nabeshima, N., et al.: The pile: An 800gb dataset of diverse text for language modeling. arXiv preprint arXiv:2101.00027 (2020) 10
21. Goel, K., Gu, A., Donahue, C., Ré, C.: It's raw! audio generation with state-space models. In: International Conference on Machine Learning. pp. 7616–7633. PMLR (2022) 3
22. Gong, H., Kang, L., Wang, Y., Wan, X., Li, H.: nnmamba: 3d biomedical image segmentation, classification and landmark detection with state space model. arXiv preprint arXiv:2402.03526 (2024) 1
23. Gu, A., Dao, T.: Mamba: Linear-time sequence modeling with selective state spaces. arXiv preprint arXiv:2312.00752 (2023) 1, 3
24. Gu, A., Goel, K., Ré, C.: Efficiently modeling long sequences with structured state spaces. arXiv preprint arXiv:2111.00396 (2021) 3

25. Gu, A., Johnson, I., Goel, K., Saab, K., Dao, T., Rudra, A., Ré, C.: Combining recurrent, convolutional, and continuous-time models with linear state space layers. Advances in neural information processing systems **34**, 572–585 (2021) 1, 3
26. Guillaumin, M., Küttel, D., Ferrari, V.: Imagenet auto-annotation with segmentation propagation. International Journal of Computer Vision **110**, 328–348 (2014) 14, 15
27. Gur, S., Ali, A., Wolf, L.: Visualization of supervised and self-supervised neural networks via attribution guided factorization. In: Proceedings of the AAAI conference on artificial intelligence. vol. 35, pp. 11545–11554 (2021) 4, 14
28. Hua, W., Dai, Z., Liu, H., Le, Q.: Transformer quality in linear time. In: International Conference on Machine Learning. pp. 9099–9117. PMLR (2022) 8
29. Jain, S., Wallace, B.C.: Attention is not explanation. In: Proceedings of NAACL-HLT. pp. 3543–3556 (2019) 15
30. Li, S., Singh, H., Grover, A.: Mamba-nd: Selective state space modeling for multi-dimensional data. arXiv preprint arXiv:2402.05892 (2024) 1
31. Liang, D., Zhou, X., Wang, X., Zhu, X., Xu, W., Zou, Z., Ye, X., Bai, X.: Pointmamba: A simple state space model for point cloud analysis. arXiv preprint arXiv:2402.10739 (2024) 1
32. Liu, J., Yang, H., Zhou, H.Y., Xi, Y., Yu, L., Yu, Y., Liang, Y., Shi, G., Zhang, S., Zheng, H., et al.: Swin-umamba: Mamba-based unet with imagenet-based pretraining. arXiv preprint arXiv:2402.03302 (2024) 1
33. Liu, Y., Tian, Y., Zhao, Y., Yu, H., Xie, L., Wang, Y., Ye, Q., Liu, Y.: Vmamba: Visual state space model. arXiv preprint arXiv:2401.10166 (2024) 1, 3, 4
34. Lu, C., Schroecker, Y., Gu, A., Parisotto, E., Foerster, J., Singh, S., Behbahani, F.: Structured state space models for in-context reinforcement learning. Advances in Neural Information Processing Systems **36** (2024) 3
35. Lu, J., Yao, J., Zhang, J., Zhu, X., Xu, H., Gao, W., Xu, C., Xiang, T., Zhang, L.: Soft: Softmax-free transformer with linear complexity. Advances in Neural Information Processing Systems **34**, 21297–21309 (2021) 8
36. Lutati, S., Zimerman, I., Wolf, L.: Focus your attention (with adaptive iir filters). arXiv preprint arXiv:2305.14952 (2023) 4
37. Ma, J., Li, F., Wang, B.: U-mamba: Enhancing long-range dependency for biomedical image segmentation. arXiv preprint arXiv:2401.04722 (2024) 1
38. Ma, X., Zhou, C., Kong, X., He, J., Gui, L., Neubig, G., May, J., Zettlemoyer, L.: Mega: moving average equipped gated attention. arXiv preprint arXiv:2209.10655 (2022) 8
39. Martin, E., Cundy, C.: Parallelizing linear recurrent neural nets over sequence length. arXiv preprint arXiv:1709.04057 (2017) 4
40. Mehta, H., Gupta, A., Cutkosky, A., Neyshabur, B.: Long range language modeling via gated state spaces. arXiv preprint arXiv:2206.13947 (2022) 3
41. Nam, W.J., Gur, S., Choi, J., Wolf, L., Lee, S.W.: Relative attributing propagation: Interpreting the comparative contributions of individual units in deep neural networks. In: Proceedings of the AAAI conference on artificial intelligence. vol. 34, pp. 2501–2508 (2020) 4, 14
42. Nguyen, E., Goel, K., Gu, A., Downs, G., Shah, P., Dao, T., Baccus, S., Ré, C.: S4nd: Modeling images and videos as multidimensional signals with state spaces. Advances in neural information processing systems **35**, 2846–2861 (2022) 3
43. Peng, B., Alcaide, E., Anthony, Q., Albalak, A., Arcadinho, S., Cao, H., Cheng, X., Chung, M., Grella, M., GV, K.K., et al.: Rwkv: Reinventing rnns for the transformer era. arXiv preprint arXiv:2305.13048 (2023) 15

44. Pióro, M., Ciebiera, K., Król, K., Ludziejewski, J., Jaszczur, S.: Moe-mamba: Efficient selective state space models with mixture of experts. arXiv preprint arXiv:2401.04081 (2024) 1, 3
45. Poli, M., Massaroli, S., Nguyen, E., Fu, D.Y., Dao, T., Baccus, S., Bengio, Y., Ermon, S., Ré, C.: Hyena hierarchy: Towards larger convolutional language models. arXiv preprint arXiv:2302.10866 (2023) 6, 15
46. Ruan, J., Xiang, S.: Vm-unet: Vision mamba unet for medical image segmentation. arXiv preprint arXiv:2402.02491 (2024) 1
47. Saon, G., Gupta, A., Cui, X.: Diagonal state space augmented transformers for speech recognition. In: ICASSP 2023-2023 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP). pp. 1–5. IEEE (2023) 3
48. Selvaraju, R.R., Cogswell, M., Das, A., Vedantam, R., Parikh, D., Batra, D.: Grad-cam: Visual explanations from deep networks via gradient-based localization. In: Proceedings of the IEEE international conference on computer vision. pp. 618–626 (2017) 4
49. Smith, J.T., Warrington, A., Linderman, S.W.: Simplified state space layers for sequence modeling. arXiv preprint arXiv:2208.04933 (2022) 4
50. Sun, Y., Dong, L., Huang, S., Ma, S., Xia, Y., Xue, J., Wang, J., Wei, F.: Retentive network: A successor to transformer for large language models (2023). URL http://arxiv. org/abs/2307.08621 v1 15
51. Sundararajan, M., Taly, A., Yan, Q.: Axiomatic attribution for deep networks. In: International conference on machine learning. pp. 3319–3328. PMLR (2017) 4
52. Touvron, H., Cord, M., Douze, M., Massa, F., Sablayrolles, A., Jégou, H.: Training data-efficient image transformers & distillation through attention. In: International conference on machine learning. pp. 10347–10357. PMLR (2021) 3
53. Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A.N., Kaiser, L.u., Polosukhin, I.: Attention is all you need. In: Advances in Neural Information Processing Systems. vol. 30 (2017) 2
54. Wang, C., Tsepa, O., Ma, J., Wang, B.: Graph-mamba: Towards long-range graph sequence modeling with selective state spaces. arXiv preprint arXiv:2402.00789 (2024) 1, 3
55. Wang, J., Zhu, W., Wang, P., Yu, X., Liu, L., Omar, M., Hamid, R.: Selective structured state-spaces for long-form video understanding. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 6387–6397 (2023) 3
56. Wang, J., Gangavarapu, T., Yan, J.N., Rush, A.M.: Mambabyte: Token-free selective state space model. arXiv preprint arXiv:2401.13660 (2024) 1, 3
57. Wang, Z., Ma, C.: Semi-mamba-unet: Pixel-level contrastive cross-supervised visual mamba-based unet for semi-supervised medical image segmentation. arXiv preprint arXiv:2402.07245 (2024) 1
58. Wang, Z., Zheng, J.Q., Zhang, Y., Cui, G., Li, L.: Mamba-unet: Unet-like pure visual mamba for medical image segmentation. arXiv preprint arXiv:2402.05079 (2024) 1
59. Wortsman, M., Lee, J., Gilmer, J., Kornblith, S.: Replacing softmax with relu in vision transformers. arXiv preprint arXiv:2309.08586 (2023) 8
60. Xing, Z., Ye, T., Yang, Y., Liu, G., Zhu, L.: Segmamba: Long-range sequential modeling mamba for 3d medical image segmentation. arXiv preprint arXiv:2401.13560 (2024) 1
61. Yan, J.N., Gu, J., Rush, A.M.: Diffusion models without attention. arXiv preprint arXiv:2311.18257 (2023) 3

62. Yang, Y., Xing, Z., Zhu, L.: Vivim: a video vision mamba for medical video object segmentation. arXiv preprint arXiv:2401.14168 (2024) 1
63. Yuan, T., Li, X., Xiong, H., Cao, H., Dou, D.: Explaining information flow inside vision transformers using markov chain. In: eXplainable AI approaches for debugging and diagnosis. (2021) 4, 5
64. Zhu, L., Liao, B., Zhang, Q., Wang, X., Liu, W., Wang, X.: Vision mamba: Efficient visual representation learning with bidirectional state space model. arXiv preprint arXiv:2401.09417 (2024) 1, 3, 4
65. Zimerman, I., Baruch, M., Drucker, N., Ezov, G., Soceanu, O., Wolf, L.: Converting transformers to polynomial form for secure inference over homomorphic encryption. arXiv preprint arXiv:2311.08610 (2023) 8
66. Zimerman, I., Wolf, L.: Multi-dimensional hyena for spatial inductive bias. arXiv preprint arXiv:2309.13600 (2023) 15