MAMBA Explain: an SSM Explainability Project

Yuval Ran-Milo
Tel Aviv University
yuvalmilo@mail.tau.ac.il

Yoni Slutzky
Tel Aviv University
slutzky1@mail.tau.ac.il

Itay Tshuva
Tel Aviv University
itaytshuva@mail.tau.ac.il

ABSTRACT

This paper presents MAMBA Explain, a project focused on explaining the information flow and behavior of Structured State Space Models (SSMs) through an interpretation of Mamba-2 as an attention-based framework. Leveraging the Structured State Space Duality (SSD) framework, we transformed Mamba-2's selective state-space layers into linear attention mechanisms, enabling an in-depth comparison with transformer-based models. Through experiments inspired by previous explainability studies on transformers, we examined both macro and micro levels of information flow within the adapted model. Our results reveal that Mamba-2 processes subject and relation information similarly to transformers but lacks the first-position bias typical of transformer architectures. These insights highlight both similarities and unique aspects of SSMs in processing factual information, paving the way for further research in SSM interpretability.

1 INTRODUCTION

In recent years, transformer-based models have become the dominant architecture for numerous natural language processing (NLP) tasks, especially in language modeling. These models, exemplified by architectures such as GPT and BERT, utilize the attention mechanism to capture long-range dependencies in sequences, demonstrating remarkable performance across a wide array of applications. However, these models often suffer from quadratic complexity in sequence length, making them computationally expensive for processing long inputs.

In parallel, Structured State Space Models (SSMs), such as Mamba [4] and Mamba-2 [2], have emerged as efficient alternatives to transformers, particularly for long-range sequence modeling tasks. SSMs leverage linear recurrence mechanisms to achieve linear scaling with sequence length, offering potential improvements in efficiency while maintaining competitive performance.

In this work, we build upon the insights from [2], which introduces a framework for viewing SSMs as a variant of linear attention layers through the lens of Structured State Space Duality (SSD). Inspired by this approach, we converted the Mamba-2 architecture into a model based on attention mechanisms, while preserving its functionality and outputs. The transition to an attention-based model allows us to revisit a set of explainability experiments performed in [3], comparing the results achieved for transformer models to the ones achieved for SSM models.

Our experiments were designed to investigate how information propagates through different layers and positions during inference; The first experiment examines the "macro" behavior of information flow within the model, while the second delves deeper into the "micro" interactions between tokens within sentences.

Our results provide new insights into the relationship between SSMs and attention layers, offering a deeper understanding of how critical information flows in SSM-based models. Notably, we observe similarities in information flow compared to transformers, while also identifying key differences, such as the absence of the first-position bias phenomenon Mamba-2 model.

2 RELATED WORK

This study builds on two key areas of recent research: the theoretical framework linking SSMs and attention mechanisms, and empirical methods for understanding information flow in language models.

The first area is explored in [2], which presents the Structured State Space Duality (SSD) framework. This work establishes a connection between Structured State Space Models (SSMs) and attention layers, enabling SSMs to be viewed as variants of attention mechanisms. This theoretical insight allows us to transform Mamba's recurrence-based architecture into one based on attention layers while retaining the original model's functionality.

The second foundation for our work comes from [3], which investigates the flow of factual associations within transformer-based language models. An important method used is the Attention Knockout, which examines the performance of the model as individual tokens are blocked from attending to others in certain layers. This technique provides a basis for understanding information flow in our attention-based Mamba-2 model, allowing us to compare its behavior with transformer-based models.

3 EXPERIMENTAL SETUP

3.1 Attention Knockout

We begin by introducing the **Attention Knockout** methodology from [3] and apply it to identify critical points in the flow of information essential for factual predictions. For successful next-token prediction, a model must process the input tokens so that the next-token can be inferred from the last position. In [3], the authors investigate this process internally by "knocking out" parts of the computation and measuring the effect on the prediction. To this end, they propose a fine-grained intervention on the attention layers, which serve as a crucial module for communicating information between positions. By disrupting critical information transfer through these layers, they demonstrate that factual predictions are constructed in stages, with essential information reaching the prediction position at specific layers during inference.

Intuitively, critical attention connections are those whose disruption results in a marked decline in prediction quality. To test whether essential information flows between two hidden representations at a specific layer, the authors zero out all attention connections between them. Formally, given two positions $r, c \in [1, N]$ with $r \leq c$, they prevent x_l^r from attending to x_c^l at layer $l \leq L$ by zeroing the attention weights for that layer in the relevant indices.

3.2 Implicit Linear Attention of Mamba-2

We began by implementing the selective State Space Model (SSM) layer within Mamba-2 to a form of linear attention. Specifically, the SSM layer processes an input tensor $\mathbf{X} \in \mathbb{R}^{L \times H}$, where L represents the sequence length and H the dimensionality of each input token. The layer transforms $\mathbf{X}$ through the operation $\mathbf{L} \circ (\mathbf{X}\mathbf{M}\mathbf{X}^\top)\mathbf{X}$, where $\mathbf{M}$ and $\mathbf{L}$ are fixed matrices determined by the layer's weights, and $\mathbf{L}$ is lower triangular. This formulation can be viewed as an attention mechanism where $\mathbf{L} \circ (\mathbf{X}\mathbf{M}\mathbf{X}^\top)$ serves as the attention matrix. In this matrix, the entry at position (i, j) quantifies the degree to which the i-th token attends to the j-th token, analogous to the attention scores in traditional transformer-based models. This transformation maintained the core functionality of the original Mamba-2 model, ensuring that both the attention-based Mamba-2 and the original model produced identical outputs for the same inputs.

3.3 Datasets

To evaluate the performance of the new model, we utilized the COUNTERFACT dataset, which contains a variety of factual triplets in the form (subject, relation, attribute). The task at hand requires models to predict a factual attribute (e.g., "Beats Music is owned by ___"). We selected 1000 random queries for which the original model predicted the true next-token to focus on understanding the flow of information within the model.

3.4 Hardware and Implementation Details

All experiments were conducted on a single NVIDIA RTX A6000 GPU, utilizing PyTorch as the primary deep learning framework. To ensure consistency, the attention-based Mamba-2 model was evaluated within the same environment across all experiments.

The original Mamba implementation does not explicitly materialize the attention matrix, relying instead on a more efficient computation strategy. For our analysis, we modified an existing implementation [5] based on the $mamba2-1.3b$ model. Specifically, we re-implemented the selective SSM layer to explicitly construct the attention matrix required for our experiments. While the functionality remains unchanged, the implementation approach was adapted to facilitate attention-based analysis. The model weights and tokenizer, $gpt-neox-20b$ [1], were both imported from Hugging Face. Code for reproducing our experiments can be found at https://github.com/YoniSlutzky98/MAMBA-explain.

4 EXPERIMENTS

In this section, we present the findings from our experiments on the attention-based Mamba-2 model. The results offer insights into how information flows through the model during inference, with comparisons drawn to the behaviors observed in transformer-based models.

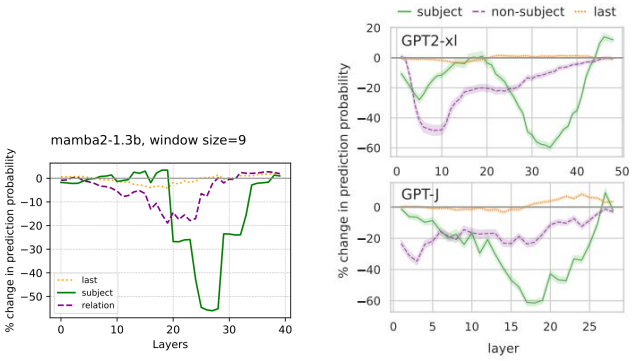
4.1 Information Flow on a Macro Level

This experiment employs Attention Knockout to determine if, and where, information from the subject and relation positions directly reaches the final prediction. Let S and $R \subset [1, N)$ denote the subject and non-subject positions for a given prompt. For each layer l, we block the attention connections from the last position to each of S, R, and the final N-th position, within a window of k layers around layer l, and measure the resulting change in prediction probability of the true next-token. This approach allows us to replicate the information flow experiments conducted on transformer architectures within the context of Mamba's linear attention framework.

Figure 1 presents the results. In the graphs, the X-axis indicates the starting layer for blocking attention, while the Y-axis shows the change in relative prediction probability. The GPT-related graphs are taken from section 5 of [3]. Blocking attention to subject tokens (solid green lines) in the middle and upper layers leads to a significant drop in prediction probability, underscoring the crucial role of subject information in reaching the final position for accurate predictions.

Additionally, blocking attention to relation tokens (dashed purple lines) causes a smaller decrease, mainly in the lower layers, indicating that relation information is integrated earlier in the model. This pattern suggests a distinct information flow: relation information reaches the prediction layer initially, followed by subject information in the deeper layers.

These findings align with those of [3], showing that both models refine subject information in the later layers and process relation information earlier, consistent with transformer-based handling of factual queries.



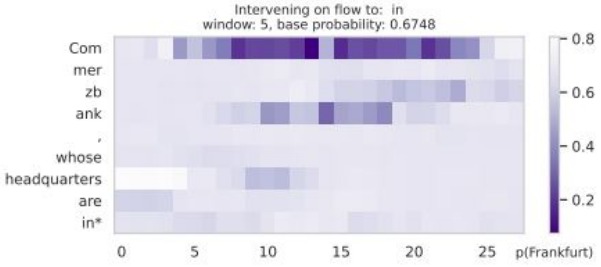
(a) Mamba-2 ($k = 9$) **(b) GPT2-2 ($k = 9$), GPT-J ($k = 5$)**

Figure 1: Macro-level comparison of Mamba-2 and GPT.

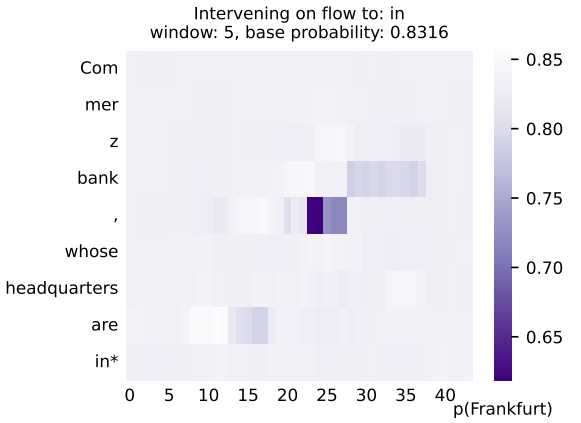
4.2 Information Flow on a Micro Level

In this experiment, we focus on individual tokens to assess their impact on the model's predictions. Using the Mamba-2 model, we analyze the model's ability to predict the next token for correct prompts while blocking attention from each token to the last token across consecutive layers.

Figure 2 presents the results. For the transformer-based model GPT-J as described [3], blocking the last position from attending to the first position—regardless of which token is in that position—significantly reduces the prediction probability. This effect, known as *first-position bias* or *attention sync*, is a well-known phenomenon in transformer-based models. However, this behavior does not appear in the Mamba-2 model. The Mamba-2 model's predictions remain robust even when attention from the first position is blocked, indicating that other positions, particularly those corresponding to the subject, play a more dominant role in generating



(a) GPT-J ($k = 5$)



(b) Mamba-2 ($k = 5$)

Figure 2: Micro-level comparison of Mamba-2 and GPT.

the final prediction. This suggests that the first token in Mamba-2 does not hold special significance in determining the model's output.

This difference in behavior highlights an intriguing distinction in information processing between Mamba-2 and traditional transformers. The absence of first-position bias in Mamba-2 could be attributed to the model's inherent tendency to assign diminishing attention weights to tokens as their distance from each other increases. This tendency is evident in the mask matrix $\mathbf{L}$ (see 3.2) as the rows are monotonically decreasing below the diagonal.

5 CONCLUSION

In this work, we successfully converted the Mamba-2 architecture into an attention-based model using the framework of Structured State Space Duality (SSD). This transformation maintained the core capabilities of the original Mamba model while allowing us to investigate how information flows through the architecture using techniques inspired by attention mechanisms.

Our experiments, based on the methodology of [3], provided significant insights into the behavior of the attention-based Mamba-2 model. On the one hand, we found macro-level similarities to the behaviors of transformer-based models:

- subject-related information plays a crucial role in generating correct predictions and is propagated through the deeper layers of the model
- Relation-related information is incorporated earlier in the model, with the lower layers primarily responsible for processing relational context

However on a micro level and unlike traditional transformer-based models, Mamba-2 model did not exhibit a first-position bias, suggesting that it does not heavily rely on the first token in the sequence to make predictions. These findings highlight important similarities and distinctions in information storage and flow between SSM-based models and transformer-based, paving the way for future research in these areas.

6 LIMITATIONS

6.1 Fixed Window Size

In both sets of experiments, we tested the propagation of information by blocking attention within a fixed window size across different layers. While this approach allowed us to localize information flow effectively, we only explored a limited range of window sizes. Larger or smaller window sizes might reveal different patterns of information propagation, particularly in deeper layers or when dealing with longer sequences. Without a comprehensive analysis of various window sizes, there may be subtle effects that we have not fully captured. Future experiments should explore a wider range of window sizes to gain a more nuanced understanding of the information flow dynamics within the model.

6.2 Absence of Detailed Analysis on Micro Level Experiments

In the second set of experiments, we observed that the Mamba-2 model did not exhibit the first-position bias phenomenon seen in transformers. However, we did not conduct a deeper analysis to explain why this bias was absent in Mamba-2. A more thorough investigation is required to understand the underlying structural differences between Mamba-2 and traditional transformers that contribute to this behavior.

6.3 Generalization Beyond Factual Queries

Our experiments focused primarily on attribute prediction, where factual knowledge retrieval is key. While this provided useful insights into how information is propagated through the attention-based Mamba model, these results may not fully generalize to other types of tasks, such as generative text modeling or tasks requiring more complex reasoning. Further testing across a wider variety of tasks is necessary to understand how well the model performs in different contexts.

REFERENCES

[1] Sid Black, Stella Biderman, Eric Hallahan, Quentin Anthony, Leo Gao, Laurence Golding, Horace He, Connor Leahy, Kyle McDonell, Jason Phang, et al. 2022. Gpt-neox-20b: An open-source autoregressive language model. *arXiv preprint arXiv:2204.06745* (2022).

[2] Tri Dao and Albert Gu. 2024. Transformers are SSMs: Generalized models and efficient algorithms through structured state space duality. *arXiv preprint arXiv:2405.21060* (2024).

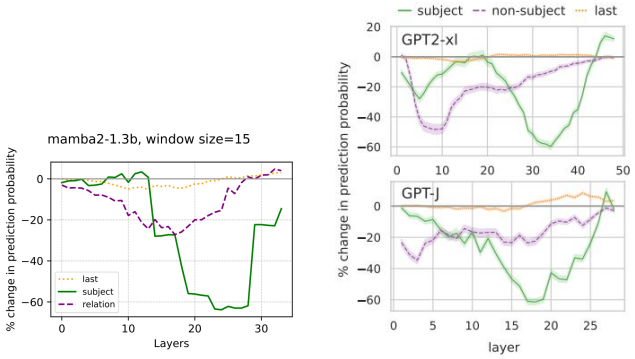
[3] Mor Geva, Jasmijn Bastings, Katja Filippova, and Amir Globerson. 2023. Dissecting recall of factual associations in auto-regressive language models. *arXiv preprint arXiv:2304.14767* (2023).

[4] Albert Gu and Tri Dao. 2023. Mamba: Linear-time sequence modeling with selective state spaces. *arXiv preprint arXiv:2312.00752* (2023).

[5] Tommy Ip. 2023. GitHub Repository for Mamba Implementation. https://github.com/tommyip/mamba2-minimal/blob/main/mamba2.py Accessed: 2024-10-26.

A ADDITIONAL MACRO-LEVEL INFORMATION FLOW EXPERIMENTS

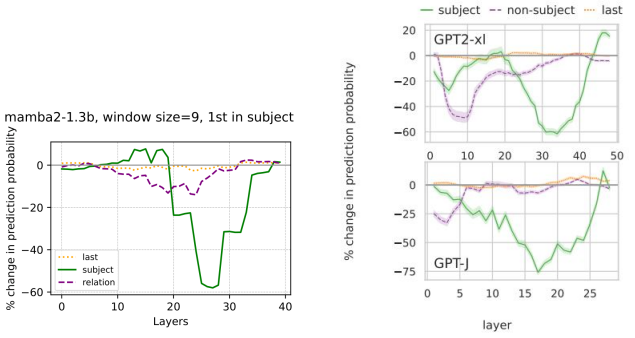
This appendix presents extended macro-level information flow experiments. Figure 3 presents the results for an experiment with a window size of $k = 15$, allowing for a comparison with the $k = 9$ results in figure 1.



(a) Mamba-2 ($k = 15$) **(b) GPT2-2 ($k = 9$), GPT-J ($k = 5$)**

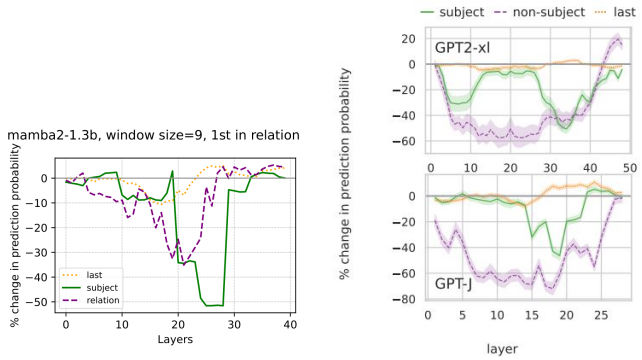
Figure 3: Macro-level comparison of Mamba-2 and GPT.

Figures 4 and 5 compare the macro-level results for the cases where the subject precedes the relation and when it succeeds it, i.e. they differentiate between prompts where the subject contains the first token and prompts where the relation contains the first token. These figures demonstrate our findings are robust on the macro-level to the first-position bias, similarly to the results in [3].



(a) Mamba-2 ($k = 9$) **(b) GPT2-2 ($k = 9$), GPT-J ($k = 5$)**

Figure 4: Macro-level comparison of Mamba-2 and GPT when 1st token is in subject.

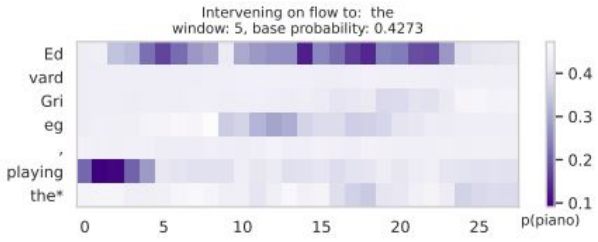


(a) Mamba-2 ($k = 9$) **(b) GPT2-2 ($k = 9$), GPT-J ($k = 5$)**

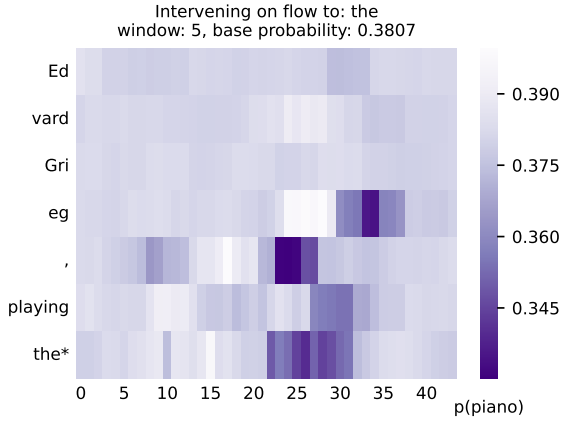
Figure 5: Macro-level comparison of Mamba-2 and GPT when 1st token is in relation.

B ADDITIONAL MICRO-LEVEL INFORMATION FLOW EXPERIMENTS

This appendix presents additional micro-level information flow experiments, showcasing the lack of first-position bias in our model, compared to its promotion in transformer-based models. The results are illustrated in figures 6, 7 and 8. One can observe the trend shared with figure 2, where the first token doesn't carry significant influence over the output for the Mamba-2 model.

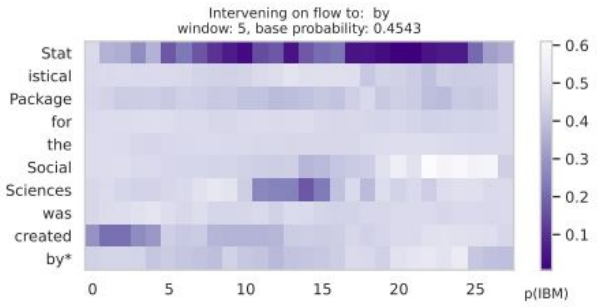


(a) GPT-J ($k = 5$)

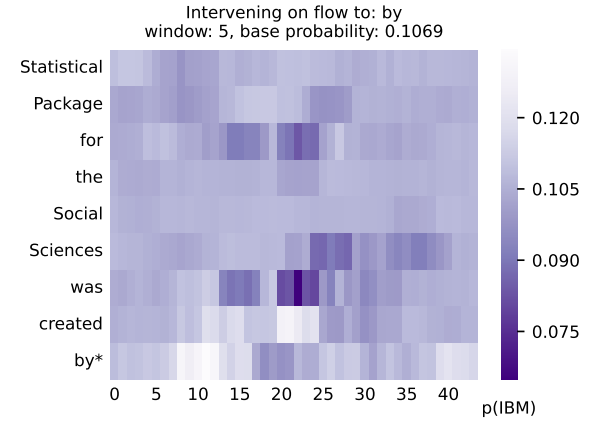


(b) Mamba-2 ($k = 5$)

Figure 6: Micro-level comparison of Mamba-2 and GPT.

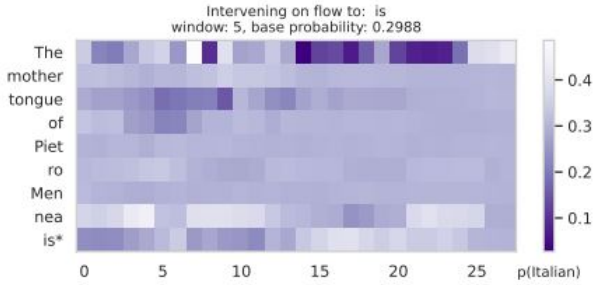


(a) GPT-J ($k = 5$)

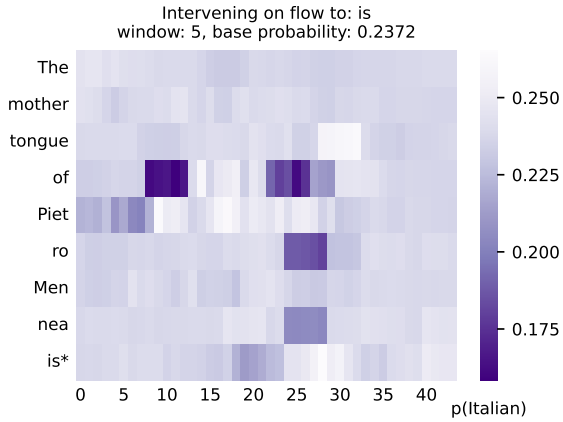


(b) Mamba-2 ($k = 5$)

Figure 7: Micro-level comparison of Mamba-2 and GPT.



(a) GPT-J ($k = 5$)



(b) Mamba-2 ($k = 5$)

Figure 8: Micro-level comparison of Mamba-2 and GPT.