Incremental Learning of Sparse Attention Patterns in Transformers

Oğuz Kaan Yüksel[*], **Rodrigo Alvarez Lucendo** , **Nicolas Flammarion**
Theory of Machine Learning Lab,
EPFL, Switzerland

Abstract

This paper studies simple transformers on a high-order Markov chain, where the model must incorporate knowledge from multiple past positions, each with different statistical importance. We show that transformers learn the task incrementally, with each stage induced by the acquisition or copying of information from a subset of positions via a sparse attention pattern. Notably, the learning dynamics transition from competitive, where all heads focus on the statistically most important attention pattern, to cooperative, where different heads specialize in different patterns. We explain these dynamics using a set of simplified differential equations, which characterize the stage-wise learning process and analyze the training trajectories. Overall, our work provides theoretical explanations for how transformers learn in stages even without an explicit curriculum and provides insights into the emergence of complex behaviors and generalization, with relevance to applications such as natural language processing and algorithmic reasoning.

1 Introduction

Knowledge is often compositional and hierarchical in nature. As such, understanding complex concepts often requires an *incremental* approach, where simpler concepts are learned first and then combined to form more complex ideas. Such incremental approaches are crucial for various cognitive tasks, including language comprehension, problem-solving, and decision-making in humans and has been recapitulated in machine learning in various settings [Saxe et al., 2019]. In particular, language, is inherently hierarchical, e.g., understanding a sentence requires understanding the meanings of individual words, phrases, and their structure. Consequentially, there has been interest in understanding *incremental* learning behavior of transformers in sequential tasks [Abbe et al., 2023b, Edelman et al., 2024], particularly in how they build upon previously learned information to understand and generate language [Chen et al., 2024a].

The elementary operation that is needed to compose information is *copying*, which is used to duplicate data and then perform downstream computations. In language, copying is essential for tasks such as text generation, where the model must replicate certain phrases or structures from the input to produce coherent and contextually relevant output [Elhage et al., 2021, Olsson et al., 2022], and, as a means to aggregate information from multiple parts of a text to form a comprehensive understanding. Copying is also a fundamental operation in algorithmic reasoning, where it is often necessary to duplicate intermediate results to perform further computations. Transformers implement this operation across different positions via sparse attention patterns which pushes their parameters to diverge. Therefore, the dynamics of how these circuits are established and its implications on reasoning, generalization and emergence are crucial to grasp the inner workings of transformers.

[*]Correspondence to oguz.yuksel@epfl.ch.

39th Conference on Neural Information Processing Systems (NeurIPS 2025) Workshop: Principles of Generative Modeling (PriGM) @ EurIPS2025.

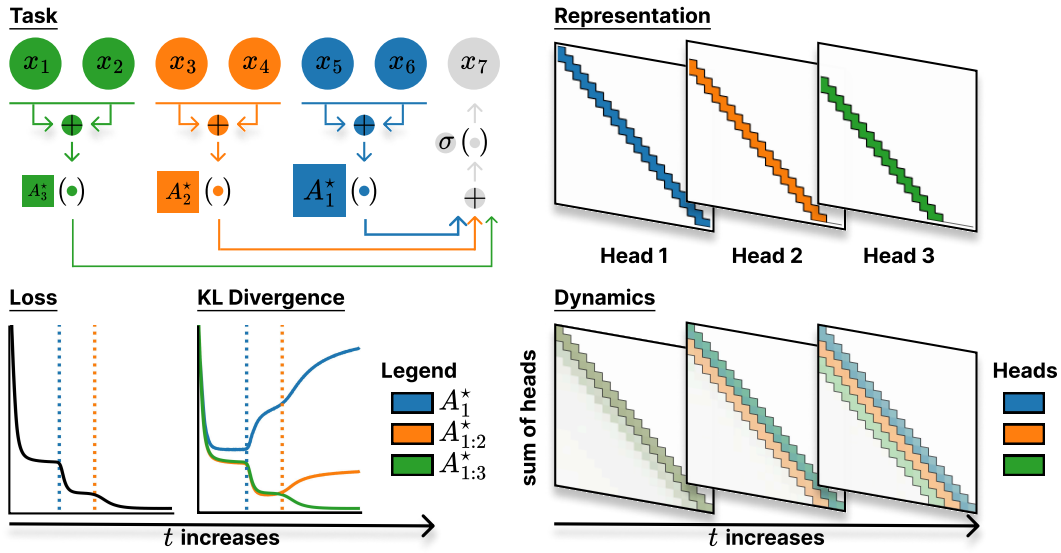

Figure 1: (Top left) The task is based on a high-order Markov chain, where the next token depends on multiple past tokens with different importance weights. The context is divided into different groups of positions, each aggregated and processed by an associated feature matrix $A_k^\star$ of various importance which is represented by the size of the feature matrix. (Top right) An idealized representation of the task in a multi-head single-layer attention. Each head represents an individual sparse attention pattern required to solve the task. (Bottom left) Transformers learn the task incrementally, with each stage corresponding to the acquisition of a sparse attention pattern which is indicated by the KL divergence between predictors $A_{1:i}^\star$ that only depends a subset of relevant positions as defined in Equation (3) and the transformer. (Bottom right) The learning dynamics transition from competitive, where all heads focus on the statistically most important pattern (indicated by high combined attention on the main diagonal), to cooperative, where different heads specialize in different patterns.

In this paper, we study single-block decoder-based transformers and the formation of sparse attention circuits during training. Sparse attention circuits are the building blocks that allow models to duplicate information from one part of the input to another, enabling the integration of information across multiple positions. We show that they are learned incrementally, with the model first acquiring the ability to copy from the most statistically important pattern, as they provide the most significant improvement in prediction accuracy, and then progressively learning the less important patterns. Interestingly, we observe an initial dynamics where all heads compete to learn the most important pattern, followed by a transition to a cooperative phase where different heads specialize in different patterns. We explain these dynamics using a set of simplified differential equations, after simplifications to the architecture and the task. This leads to connections to tensor factorization which is a well-studied problem [Arora et al., 2019, Razin et al., 2021, Li et al., 2021, Jin et al., 2023]. The setting and our main contributions are summarized schematically in Figure 1.

2 Stage-wise Formation of Sparse Attention Patterns

We consider a classification task that is based on a discrete Markov chain of order w with states in a dictionary $\mathcal{D}$ with $|\mathcal{D}| = d$. We treat each element of this dictionary as a one-hot vector in $\mathbb{R}^d$. The sequences are generated as follows:

$$x_{-w+1}, \ldots, x_0 \overset{i.i.d.}{\sim} \mathcal{D}\,, \quad \text{and for all } t \in [T]\,, \quad x_t \sim \text{softmax}\left(\sum_{k=1}^{h} A_k^\star \sum_{i \in I(k)} \alpha_i x_{t-i}\right)\,, \quad (1)$$

where $A_k^\star \in \mathbb{R}^{d\times d}$ are fixed feature matrices, $I(k)$ are disjoint sets that partition $\{0, \ldots, w-1\}$ and α_i are importance weights which verify $\sum_{i\in I(k)} \alpha_i = 1$ for all $k \in [h]$. As $I(k)$ and $A_k^\star$ can be permuted without changing the data generation process, we assume without loss of generality

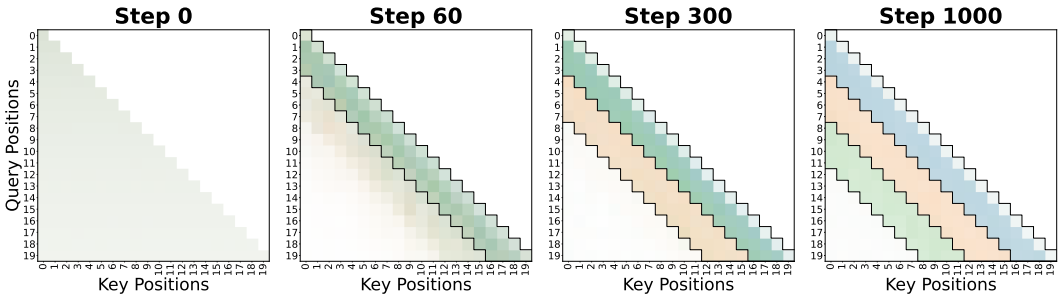

Figure 2: The sum of learned attention patterns at different stages of training where blue, yellow and green colors correspond to different heads. The main diagonal does not have the same intensity as the other positions as it is learned via the skip connection directly from the input.

that $\|A_1^\star\| \geq \|A_2^\star\| \geq \ldots \geq \|A_h^\star\|$ and that $I(1)$ contains the most important positions, i.e., those associated with the largest feature norms.

One particular choice of interest is to have $I(k)$ to be contiguous blocks of indices that start from the most recent position, i.e., for some $0 < i_1 < i_2 < \ldots < i_{h-1} < w-1$,

$$I(1) = \{0, \ldots, i_1\}, I(2) = \{i_1 + 1, \ldots, i_2\}, \ldots, I(h) = \{i_{h-1} + 1, \ldots, w-1\}\,. \tag{2}$$

This choice is inspired by the natural language where nearby tokens that complete the text into a word or a short phrase should have more statistical correlation over the distant tokens. Notably, when each of the $I(k)$ are singletons, the resulting operation is copying from a particular position.

We train single-block decoder-based transformers with h heads on sequences sampled as in Equation (1) by minimizing the cross entropy loss over the full sequence except the initial tokens $x_{-w+1}, \ldots, x_0$ that are not sampled from the process. The details of the architecture, optimization and choice of ground truth are provided in Section B.

We observe that the transformers learn the task incrementally, with each stage corresponding to the acquisition of a sparse attention pattern as in Figure 2. All heads start at uniform due to the initialization. Then, they first mainly focus on the positions in $I(1)$ as they are the most statistically important positions. At this stage, the heads compete to learn from these positions, resulting in overlapping attention patterns. Later, heads gradually specialize in different patterns, with one head learning from the positions in $I(2)$ while the other finally focusing on $I(3)$.

In order to understand the dynamics in the function space, we train models with different maximum context lengths $c = 4, 8, 12$, e.g., when $c = 4$, the model can only access and learn from the positions in $I(1)$. In Figure 3 (right), we plot the Kullback-Leibler (KL) divergence between the predictions of these transformers and the transformer without any context length restriction. We observe that the transformers first approach the model with $c = 4$ and then $c = 8$ before finally reaching the full model with $c = 12$. This indicates that the transformers not only learn the attention patterns but also simultaneously learn the feature matrices associated with these patterns.

Similarly, we study the KL divergence pattern when comparing the predictions of the transformers with restricted context lengths to the ground truths that only depend on restricted positions:

$$f_{A^\star_{1:i}}(x_{t-1}, \ldots, x_{t-w}) = \text{softmax}\left(\sum_{k=1}^{i} A_k^\star \sum_{j \in I(k)} \alpha_j x_{t-j}\right)\,. \tag{3}$$

This is plotted in Figure 3 (left) where we see an identical pattern. These are similar to what Edelman et al. [2024] observed for in-context Markov chain where stages are characterized by sub-n-grams.

2.1 Representation with a Simplified Multi-Head Attention

Here, we construct a simple representation on a single-layer multi-head attention that solves the task. Let $X \in \mathbb{R}^{d \times (T+w)}$ be the input data matrix with columns $x_{-w+1}, \ldots, x_0, x_1, \ldots, x_T$. We assume that the positional information is encoded using one-hot vectors in $\mathbb{R}^T$ and concatenated to the data as follows: $\tilde{X} = \begin{pmatrix} X \\ I_{T+w} \end{pmatrix} \in \mathbb{R}^{(d+T+w) \times (T+w)}$. Then, the transformer takes $\tilde{X}$ as input and produces

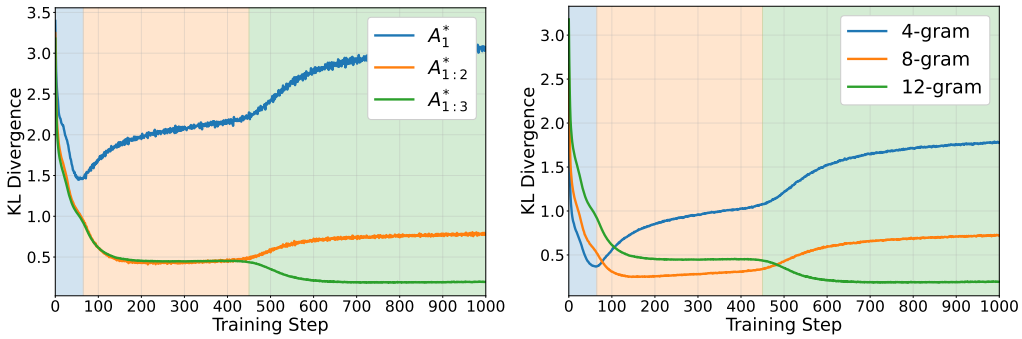

Figure 3: (Left) KL divergence between the ground truths that only depend on the positions in $I(1)$, $I(1) \cup I(2)$ and $I(1) \cup I(2) \cup I(3)$, and the predictions of the transformer with unrestricted context length. (Right) KL divergence between the predictions of the transformers with restricted context lengths $c = 4, 8, 12$ and the transformer without any context length restriction.

the output $Y \in \mathbb{R}^{d \times T}$ with columns $y_0, \ldots, y_{T-1}$ as follows:

$$y_t = \text{softmax}\left(\sum_{k=1}^{H} V_k \tilde{X} a_t^{(k)}\right), \quad a_t^{(k)} = \text{softmax}\left(\mathcal{M}_{T-t}\left(\tilde{X}^\top K_k^\top Q_k x_t\right)\right),$$

where $Q_k, K_k, V_k \in \mathbb{R}^{(d+T+w)\times(d+T+w)}$ are the query, key and value matrices of the head k, respectively and $\mathcal{M}_p$ sets the last p entries to $-\infty$ to apply causal masking.

For head k, we set the value matrix $V_k = A_k^\star$ and $a_t^{(k)}$ to be a positional-only attention corresponding to $I(k)$ with the following sparse pattern

$$\frac{1}{|I(k)|}\left(\underbrace{0, \ldots, 0}_{t \text{ entries}}, \underbrace{\mathbf{1}_{1\in I(k)}, \mathbf{1}_{1\in I(k)}, \ldots, \mathbf{1}_{1\in I(k)}}_{w \text{ entries}}, \underbrace{0, \ldots, 0}_{(T-t) \text{ entries}}\right).$$

Here, the first t entries correspond to the irrelevant tokens in the context and the last $(T-t)$ entries are zeroed out due to the causal masking. Among the relevant tokens in the intermediate w positions, the attention focuses on the indices in $I(k)$ as they can be processed altogether with the same feature matrix $V_k = A_k^\star$. As the target patterns are sparse, the parameters of the attention need to diverge to infinity to exactly learn this operation. In practice, we expect finite values that approximate these sparse attention patterns. These attention patterns can be learned based on the positional information: $K_k^\top Q_k = \lambda \sum_{i\in I(k)} \sum_{p=w}^{T+w} e_{d+p-i} e_{d+p}^\top$, where $\lambda > 0$ is a scaling constant and e_i is the i-th standard basis vector in $\mathbb{R}^{d+T}$. As $\lambda \to \infty$, the attention scores converge to the desired sparse pattern. Note that this construction is not unique as there are many Q_k and K_k that can realize the same attention pattern. In particular, as there are h heads to learn, the construction has a permutation symmetry among the heads. The permutation symmetry is key in understanding the learning dynamics, as we show in Section 3.

3 Training Dynamics on Regression Variant

Consider the following regression task associated to any distribution $\mathcal{P}_X$ and $\mathcal{P}_\xi$

$$(x_1, \ldots, x_T) \sim \mathcal{P}_X, \xi \sim \mathcal{P}_\xi, \quad \text{with} \quad y^\star(X) = \sum_{k=1}^{h} A_k^\star X s_k^\star + \xi, \tag{4}$$

where $s_k^\star \in \mathbb{R}^T$ is the vector with entries α_i for $i \in I(k)$ and zero otherwise. For this section, we set $|I(k)| = 1$ for all k for simplicity. Let $m_k^\star = \|A_k^\star\|_F, V_k^\star = A_k^\star / m_k^\star$ for all $k \in [h]$ with $m_1^\star > m_2^\star > \ldots > m_h^\star$ without loss of generality. We need some assumptions:

Assumption 1. *The noise is zero-mean, i.e.,* $\mathbb{E}[\xi] = 0$ *and the data is normalized, i.e.,*

$$\forall i, j \in [T], \quad \mathbb{E}\left[\langle x_i, x_j \rangle\right] = \mathbf{1}_{i=j}.$$

Assumption 2. *The feature matrices are orthogonal, i.e.,*

$$\forall i, j \in [h]\,, \quad \langle V_i^\star, V_j^\star \rangle = \operatorname{Tr}\left((V_i^\star)^\top V_j^\star\right) = \mathbf{1}_{i=j}\,.$$

We use the following model based on Section 2.1 for learning:

$$y_\theta(X) = \sum_{k=1}^{h} V_k X s_k\,, s_k = \operatorname{softmax}(q_k)\,, \quad \text{where} \quad \theta = (V_1, \ldots, V_h, q_1, \ldots, q_h)\,.$$

We set the loss to the mean square loss and study the gradient flow dynamics of the population loss. The notational details and the proofs are given in Section D.

Proposition 1 reinterprets this dynamics as a gradient flow of a tensor factorization problem.

Proposition 1. *The gradient flow dynamics is equivalent to a gradient flow on the following loss:*

$$\mathcal{L}(\theta) = \frac{1}{2}\|\boldsymbol{G} - \boldsymbol{P}\|_F^2\,, \quad \textit{where} \quad \boldsymbol{P} = \sum_{k=1}^{h} V_k \otimes s_k \;\textit{ and }\; \boldsymbol{G} = \sum_{k=1}^{h} m_k^\star \left(V_k^\star \otimes s_k^\star\right)\,.$$

Attention Reparameterization. Note that due to the softmax operation, $\sum_i q_i$ is always constant and thus we can restrict q_k to have a zero mean. This implies a one-to-one correspondence between q_k and s_k in the subspace of zero-mean vectors. Therefore, we analyze the dynamics in terms of s_k:

$$\dot{V}_k = (\boldsymbol{G} - \boldsymbol{P})\, s_k\,, \dot{s}_k = \Pi(s_k)^2 \left(V_k^\top (\boldsymbol{G} - \boldsymbol{P})\right)\,, \quad \text{where} \quad \Pi(s) = \left(\operatorname{diag}(s) - ss^\top\right)\,. \tag{5}$$

Numerical Simulations. We simulate these differential equations with initialization $V_i = 0$ and $s_i \approx \frac{1}{T} 1_T$. We present the results in Section C.6.

We show that the competitive phase of the learning dynamics can be described by the symmetric initialization $s_1(0) = s_k(0), V_1(0) = V_k(0)$ for all k. This leads to the following coupled dynamics:

$$\dot{V} = \left(\boldsymbol{G} s - H\|s\|^2 V\right)\,, \quad \dot{s} = \Pi(s)^2 \left(V^\top \boldsymbol{G} - H\|V\|_F^2 s\right)\,.$$

Theorem 1. *Assume that the initialization verifies the following for all $k \in [h]$:*

$$\langle V(0), V_1^\star \rangle \geq \langle V(0), V_k^\star \rangle \quad \langle s(0), s_1^\star \rangle \geq \langle s(0), s_k^\star \rangle\,. \tag{6}$$

Then, the dynamics of V and s converge to the following fixed point:

$$V(\infty) = \frac{m_1^\star}{H} V_1^\star\,, \quad s(\infty) = s_1^\star\,. \tag{7}$$

Theorem 1 is based on an ordering argument. As long as the initialization verifies the ordering condition in Equation (6), the dynamics of V and s are such that $\dot{V}$ and $\dot{s}$ reinforces the same order. In Section D.1, we remark that the initialization in Theorem 1 can be further relaxed to a wider basin.

Standalone, Theorem 1 does not explain what happens when the heads do not start with the same initialization. Theorem 2 establishes that when many heads are initialized with a small deviation from the symmetric initialization, the deviation from the symmetric initialization is bounded for a finite time that we can precisely control. Therefore, the initialization determines the coupling time of different heads after which they might start to diverge.

Theorem 2. *Assume that $V(0)$ and $s(0)$ such that $\forall k \in [h]$: $\|V(0) - V_k(0)\|_F \leq \epsilon$ and $\|s(0) - s_k(0)\|_2 \leq \epsilon$, where $\epsilon \ll 1$. Then, there exists a universal constant c_1 such that*

$$\|V_k(t) - V(t)\|_F \leq \epsilon e^{c_1 t} \quad \textit{and} \quad \|s_k(t) - s(t)\|_2 \leq \epsilon e^{c_1 t}\,, \quad \forall t \in \left[0, \frac{1}{-c_1 \log \epsilon}\right]\,.$$

4 Conclusion

In this work, we have provided a simple but rich task in which transformers need to implement multiple sparse attention patterns. We have shown that it captures the essence of position-dependent incremental learning in transformers. The learning dynamics start competitive where all the heads try to learn the most important pattern. We explain this stage via a coupled dynamics of the attention matrices. After this stage, the heads start to collaborate where the offshooting head learns to predict the other patterns. Our results capture the interplay of sparsity of attention patterns and the learning dynamics of transformers. This is crucial for understanding behavior of transformers in real-world tasks such as reasoning and natural language processing.

References

Emmanuel Abbe, Enric Boix Adsera, and Theodor Misiakiewicz. Sgd learning on neural networks: leap complexity and saddle-to-saddle dynamics. In *The Thirty Sixth Annual Conference on Learning Theory*, pages 2552–2623. PMLR, 2023a.

Emmanuel Abbe, Samy Bengio, Enric Boix-Adserà, Etai Littwin, and Joshua M. Susskind. Transformers learn through gradual rank increase. In Alice Oh, Tristan Naumann, Amir Globerson, Kate Saenko, Moritz Hardt, and Sergey Levine, editors, *Advances in Neural Information Processing Systems 36: Annual Conference on Neural Information Processing Systems 2023, NeurIPS 2023, New Orleans, LA, USA, December 10 - 16, 2023*, 2023b. URL http://papers.nips.cc/paper_files/paper/2023/hash/4d69c1c057a8bd570ba4a7b71aae8331-Abstract-Conference.html.

Kwangjun Ahn, Xiang Cheng, Hadi Daneshmand, and Suvrit Sra. Transformers learn to implement preconditioned gradient descent for in-context learning. *Advances in Neural Information Processing Systems*, 36, 2024.

Ekin Akyürek, Bailin Wang, Yoon Kim, and Jacob Andreas. In-context language learning: Arhitectures and algorithms. *arXiv preprint arXiv:2401.12973*, 2024.

Sanjeev Arora, Nadav Cohen, Wei Hu, and Yuping Luo. Implicit regularization in deep matrix factorization. *Advances in Neural Information Processing Systems*, 32, 2019.

Raphaël Berthier. Incremental learning in diagonal linear networks. *arXiv preprint arXiv:2208.14673*, 2022.

Satwik Bhattamishra, Arkil Patel, Phil Blunsom, and Varun Kanade. Understanding in-context learning in transformers and llms by learning to learn discrete functions. *arXiv preprint arXiv:2310.03016*, 2023.

Alberto Bietti, Vivien Cabannes, Diane Bouchacourt, Herve Jegou, and Leon Bottou. Birth of a transformer: A memory viewpoint. *Advances in Neural Information Processing Systems*, 36, 2024.

Enric Boix-Adsera, Etai Littwin, Emmanuel Abbe, Samy Bengio, and Joshua Susskind. Transformers learn through gradual rank increase. *arXiv preprint arXiv:2306.07042*, 2023.

Etienne Boursier, Loucas Pillaud-Vivien, and Nicolas Flammarion. Gradient flow dynamics of shallow reLU networks for square loss and orthogonal inputs. In Alice H. Oh, Alekh Agarwal, Danielle Belgrave, and Kyunghyun Cho, editors, *Advances in Neural Information Processing Systems*, 2022. URL https://openreview.net/forum?id=L74c-iUxQ1I.

Angelica Chen, Ravid Shwartz-Ziv, Kyunghyun Cho, Matthew L. Leavitt, and Naomi Saphra. Sudden drops in the loss: Syntax acquisition, phase transitions, and simplicity bias in mlms. In *The Twelfth International Conference on Learning Representations, ICLR 2024, Vienna, Austria, May 7-11, 2024*. OpenReview.net, 2024a. URL https://openreview.net/forum?id=MO5PiKHELW.

Siyu Chen, Heejune Sheen, Tianhao Wang, and Zhuoran Yang. Unveiling induction heads: Provable training dynamics and feature learning in transformers. In *The Thirty-eighth Annual Conference on Neural Information Processing Systems*, 2024b.

Francesco D'Angelo, Francesco Croce, and Nicolas Flammarion. Selective induction heads: How transformers select causal structures in context. In *The Thirteenth International Conference on Learning Representations, ICLR 2025, Singapore, April 24-28, 2025*. OpenReview.net, 2025. URL https://openreview.net/forum?id=bnJgzAQjWf.

Benjamin L Edelman, Ezra Edelman, Surbhi Goel, Eran Malach, and Nikolaos Tsilivis. The evolution of statistical induction heads: In-context learning markov chains. *arXiv preprint arXiv:2402.11004*, 2024.

Nelson Elhage, Neel Nanda, Catherine Olsson, Tom Henighan, Nicholas Joseph, Ben Mann, Amanda Askell, Yuntao Bai, Anna Chen, Tom Conerly, et al. A mathematical framework for transformer circuits. *Transformer Circuits Thread*, 1(1):12, 2021.

K. Fukumizu and S. Amari. Local minima and plateaus in hierarchical structures of multilayer perceptrons. *Neural Networks*, 13(3):317–327, 2000. ISSN 0893-6080. doi: https://doi.org/10.1016/S0893-6080(00)00009-5. URL https://www.sciencedirect.com/science/article/pii/S0893608000000095.

Shivam Garg, Dimitris Tsipras, Percy S Liang, and Gregory Valiant. What can transformers learn in-context? a case study of simple function classes. *Advances in Neural Information Processing Systems*, 35:30583–30598, 2022.

Gauthier Gidel, Francis Bach, and Simon Lacoste-Julien. Implicit regularization of discrete gradient dynamics in linear neural networks. *Advances in Neural Information Processing Systems*, 32, 2019.

Daniel Gissin, Shai Shalev-Shwartz, and Amit Daniely. The implicit bias of depth: How incremental learning drives generalization. In *International Conference on Learning Representations*, 2020.

Arthur Jacot, François Ged, Berfin Şimşek, Clément Hongler, and Franck Gabriel. Saddle-to-saddle dynamics in deep linear networks: Small initialization training, symmetry, and sparsity. *arXiv preprint arXiv:2106.15933*, 2021.

Liwei Jiang, Yudong Chen, and Lijun Ding. Algorithmic regularization in model-free overparametrized asymmetric matrix factorization. *arXiv preprint arXiv:2203.02839*, 2022.

Jikai Jin, Zhiyuan Li, Kaifeng Lyu, Simon S Du, and Jason D Lee. Understanding incremental learning of gradient descent: A fine-grained analysis of matrix sensing. *arXiv preprint arXiv:2301.11500*, 2023.

Daniel Jurafsky and James H. Martin. *Speech and Language Processing: An Introduction to Natural Language Processing, Computational Linguistics, and Speech Recognition*. Prentice Hall, Upper Saddle River, NJ, 2nd edition, 2009.

Jaeyeon Kim, Sehyun Kwon, Joo Young Choi, Jongho Park, Jaewoong Cho, Jason D. Lee, and Ernest K. Ryu. Task diversity shortens the icl plateau, 2024. URL https://arxiv.org/abs/2410.05448.

Zhiyuan Li, Yuping Luo, and Kaifeng Lyu. Towards resolving the implicit bias of gradient descent for matrix factorization: Greedy low-rank learning. In *International Conference on Learning Representations*, 2021.

Ashok Vardhan Makkuva, Marco Bondaschi, Adway Girish, Alliot Nagle, Martin Jaggi, Hyeji Kim, and Michael Gastpar. Attention with markov: A framework for principled analysis of transformers via markov chains. *arXiv preprint arXiv:2402.04161*, 2024.

Eshaan Nichani, Alex Damian, and Jason D. Lee. How transformers learn causal structure with gradient descent, 2024. URL https://arxiv.org/abs/2402.14735.

Catherine Olsson, Nelson Elhage, Neel Nanda, Nicholas Joseph, Nova DasSarma, Tom Henighan, Ben Mann, Amanda Askell, Yuntao Bai, Anna Chen, Tom Conerly, Dawn Drain, Deep Ganguli, Zac Hatfield-Dodds, Danny Hernandez, Scott Johnston, Andy Jones, Jackson Kernion, Liane Lovitt, Kamal Ndousse, Dario Amodei, Tom Brown, Jack Clark, Jared Kaplan, Sam McCandlish, and Chris Olah. In-context learning and induction heads. *CoRR*, abs/2209.11895, 2022. doi: 10.48550/ARXIV.2209.11895. URL https://doi.org/10.48550/arXiv.2209.11895.

Scott Pesme and Nicolas Flammarion. Saddle-to-saddle dynamics in diagonal linear networks. *Advances in Neural Information Processing Systems*, 36:7475–7505, 2023.

Noam Razin, Asaf Maman, and Nadav Cohen. Implicit regularization in tensor factorization. *CoRR*, abs/2102.09972, 2021. URL https://arxiv.org/abs/2102.09972.

Andrew M Saxe, James L McClelland, and Surya Ganguli. A mathematical theory of semantic development in deep neural networks. *Proceedings of the National Academy of Sciences*, 116(23):11537–11546, 2019.

Anej Svete and Ryan Cotterell. Transformers can represent n-gram language models. *arXiv preprint arXiv:2404.14994*, 2024.

Aditya Varre, Gizem Yüce, and Nicolas Flammarion. Learning in-context n-grams with transformers: Sub-n-grams are near-stationary points. In *International Conference on Machine Learning*, 2025.

Aditya Vardhan Varre, Maria-Luiza Vladarean, Loucas Pillaud-Vivien, and Nicolas Flammarion. On the spectral bias of two-layer linear networks. In *Thirty-seventh Conference on Neural Information Processing Systems*, 2023. URL https://openreview.net/forum?id=FFdrXkm3Cz.

Aditya Vardhan Varre, Margarita Sagitova, and Nicolas Flammarion. Sgd vs gd: Rank deficiency in linear networks. *Advances in Neural Information Processing Systems*, 37:60133–60161, 2024.

Johannes Von Oswald, Eyvind Niklasson, Ettore Randazzo, João Sacramento, Alexander Mordvintsev, Andrey Zhmoginov, and Max Vladymyrov. Transformers learn in-context by gradient descent. In *International Conference on Machine Learning*, pages 35151–35174. PMLR, 2023.

Oğuz Kaan Yüksel and Nicolas Flammarion. On the sample complexity of next-token prediction. In *The 28th International Conference on Artificial Intelligence and Statistics*, 2025. URL https://openreview.net/forum?id=eJkNMwzZzy.

Oğuz Kaan Yüksel, Mathieu Even, and Nicolas Flammarion. Long-context linear system identification. In *The Thirteenth International Conference on Learning Representations, ICLR 2025, Singapore, April 24-28, 2025*. OpenReview.net, 2025. URL https://openreview.net/forum?id=2TuUXtLGhT.

Nicolas Zucchet, Francesco D'Angelo, Andrew K. Lampinen, and Stephanie C. Y. Chan. The emergence of sparse attention: impact of data distribution and benefits of repetition. *CoRR*, abs/2505.17863, 2025. doi: 10.48550/ARXIV.2505.17863. URL https://doi.org/10.48550/arXiv.2505.17863.

Organization of the Appendix

The appendix is organized as follows,

- Section A discusses the related work.
- Section B provides the experimental details.
- Section C presents additional experiments.
- Section D provides the proofs of the theoretical results.

A Related Work

Our work is at the intersection of incremental learning and n-gram models.

Incremental learning. Plateau-like learning curves are a common feature in neural network training. Early analyses, such as Fukumizu and Amari [2000], attributed these behaviors to critical points in supervised learning. Subsequent studies have examined similar dynamics in a variety of simplified settings, including linear networks [Gissin et al., 2020, Saxe et al., 2019, Gidel et al., 2019, Arora et al., 2019, Jacot et al., 2021, Li et al., 2021, Razin et al., 2021, Jiang et al., 2022, Berthier, 2022, Pesme and Flammarion, 2023, Jin et al., 2023, Varre et al., 2023, 2024], ReLU models [Boursier et al., 2022, Abbe et al., 2023a], and simplified transformer architectures [Boix-Adsera et al., 2023]. In transformer training, plateaus followed by sudden capability gains [Chen et al., 2024a, Kim et al., 2024] are often observed in regression tasks [Garg et al., 2022, Von Oswald et al., 2023, Ahn et al., 2024], and formal language recognition [Bhattamishra et al., 2023, Akyürek et al., 2024, D'Angelo et al., 2025].

n-gram models. n-gram language models [Jurafsky and Martin, 2009] serve as a toy setting to understand large language models. This perspective has motivated a range of studies: the optimization landscape has been characterized in Makkuva et al. [2024], expressivity over n-gram distributions has been examined in Svete and Cotterell [2024] and sample complexity has been resolved in Yüksel and Flammarion [2025]. Connections between ICL and the emergence of induction heads [Elhage et al., 2021, Olsson et al., 2022], together with their acquisition via gradient descent [Nichani et al., 2024], are drawn by Bietti et al. [2024]. Training dynamics on n-gram prediction tasks have also been shown to progress in stages: intermediate solutions approximate sub-n-grams [Edelman et al., 2024, Chen et al., 2024b], which later are formalized as near-stationary points by Varre et al. [2025].

B Experimental Details

Ground truth. We sample feature matrices $A_k^\star$ uniformly over orthogonal matrices and then scale with positive scalars m_k. These constants are chosen geometrically, i.e., $m_k = m^{h-k}b_0$ where $m > 1$ is the multiplicative constant and $b_0 > 0$ is the base scale. This results in an importance hierarchy in the feature matrices whereas features within the same matrix has the same importance. For simplicity, we choose $\alpha_i = 1/|I(I^{-1}(i))|$ where I^{-1} is the inverse of I. Lastly, we choose $I(k)$ as in Equation (2) with the same length intervals of size w/h.

Architecture and optimization. The full model has a standard single-layer transformer decoder architecture as discussed in Section 2. It uses absolute positional encodings with learnable embedding and unembedding matrices and has the configuration shown in Table 3. The minimal model, as described in **??**, removes layer normalization, dropout, residual connections, key and output attention matrices and the MLP layer. It uses one-hot positional encodings and does not have embedding and unembedding matrices. Both the full model and the minimal model are trained with the same optimization hyperparameters listed in Table 2, and the same synthetic data generation process described in Table 1. The main difference in the learning task between the two models is the interval lengths $|I(k)|$ of the Markov process: the full model uses intervals of length 4, while the minimal model uses intervals of length 2, as summarized in Table 4.

We train the n-gram models using the same architecture and optimization hyperparameters as the full transformer model but training with windows of size n sliding over the full sequence.

Table 1: Synthetic dataset parameters

Parameter	Value
Heads h	3
Dictionary size d	50
Multiplicative constant m	1.7
Base scale b_0	10
Sequence length T	20
Train samples	9000
Test samples	3000
Seed	0

Table 2: Optimization hyperparameters

Parameter	Value
Steps	2000
Batch size	3000
Gradient clipping	1.0
Optimizer	AdamW
Weight decay	0.01
Learning rate	0.003
Scheduler	ReduceLROnPlateau
Patience	10
Factor	0.5

Table 3: Transformer configuration

Parameter	Value
Hidden dimension	255
Feedforward dimension	64
Dropout	0.1
Initialization scale	1
Number of blocks	1
Number of heads	3

Table 4: Markov process intervals

	Full	Minimal
$\|w\|$	12	6
$I(1)$	$\{1, 2, 3, 4\}$	$\{1, 2\}$
$I(2)$	$\{5, 6, 7, 8\}$	$\{3, 4\}$
$I(3)$	$\{9, 10, 11, 12\}$	$\{5, 6\}$

C Additional Experiments

We run additional experiments to study incremental learning behavior under different settings which we summarize below:

- We identify the minimal architecture that exhibits incremental learning and then execute ablation studies with initialization scale and multiplicative constant in Section C.1,
- We study the impact of incremental learning in generalization performance in Section C.2.

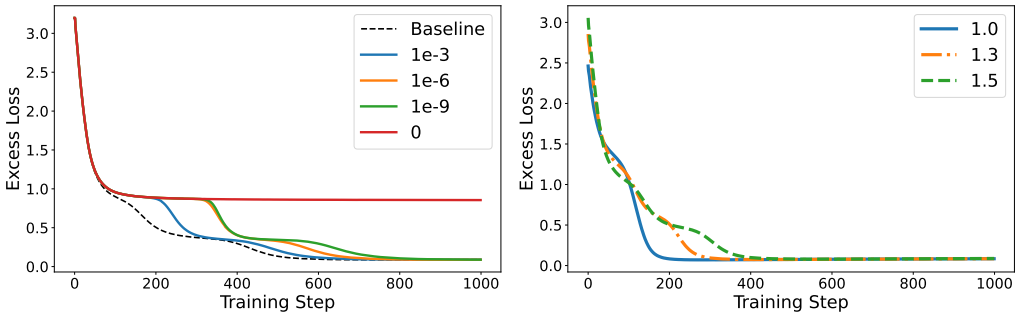

Figure 4: (Left) Excess loss of the minimal architecture with different initialization scales. (Right) Excess loss of the minimal architecture with different multiplicative constants m that determine the importance hierarchy.

- We show that multi-pass training with finite data yields the same behavior as infinite data in Section C.3
- We perform experiments with reversed importance and non-uniform interval lengths in Section C.4.
- We study the impact of weight decay in Section C.5
- Finally, we present our numerical simulations in Section C.6.

C.1 Ablation Studies

In order to isolate the essential components that drive the incremental learning behavior, we simplify the architecture by removing some components. First, we remove any components such as layer normalization and residual connections that are not present in the idealized construction in **??**. Then, we reduce the product $K_k^\top Q_k$ to a single matrix A_k as there is a symmetry between K_k and Q_k. All of these changes individually or combined do not alter the incremental learning behavior. We plot the learning behavior of this simplified model in Figure 1.

We also perform ablation studies with this minimal architecture. We first vary the initialization scale of the attention matrices A_k and set value matrices to be zero. While initializing A_k, we use uniform distribution over $[-u, u]$ where u is the initialization scale. Figure 4 (left) shows that the speed of incremental learning is affected by the initialization scale, with smaller scales resulting in slower learning. At the extreme $u = 0$, we observe that the model only learns a single pattern and does not progress further. This is because of the symmetry between the heads, which requires a small perturbation to break.

We also vary the multiplicative constant m that determines the structure in the data generation process. Figure 4 (right) shows that the number of steps diminish to two for $m = 1$, where there is no importance ordering. Qualitatively, this model first learns a single pattern and then the other two are learned simultaneously. For $m = 1.3$ and $m = 1.5$, we still observe three distinct stages, but the stages are intertwined for $m = 1.3$ where bumps in the loss landscape are less pronounced.

C.2 Dataset Size and Generalization

We study the effect of the dataset size on the incremental learning behavior. As we decrease the dataset size and cross some critical thresholds, we observe that the number of stages that occur in training decreases, as seen in Figure 5 (left). Figure 5 (right) plots the KL divergence between the predictions of the model with different context lengths and the trained transformer. The trend is similar to the one observed in Figure 3 but with different number of bumps for each dataset size.

This points towards a beneficial regularization from the training trajectory which leads to misspecified models, i.e., models that are not able to learn the task perfectly as they have a shorter context length. Yüksel et al. [2025] argue that such misspecification can be beneficial in low-data regimes, making learning statistically feasible. Notably, transformers with early stopping seem to select the misspecification length automatically, hinting at potential sample complexity gains in these settings.

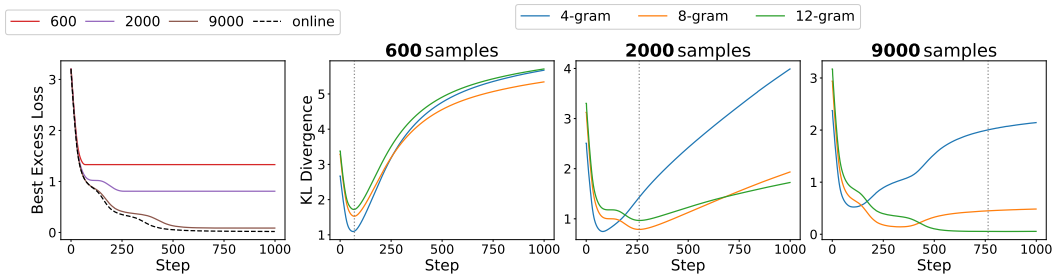

Figure 5: The impact of the dataset size on the incremental learning behavior. (Left) The best validation loss as a function of the dataset size. (Right) The KL divergence between the predictions of the model with different context lengths and the trained transformer. Dashed lines indicate the first step that obtains the best excess loss.

C.3 Infinite Data

Instead of training on a finite dataset of 9000 samples, we train the model with infinite data by sampling a new batch of data at each step. This removes any effect of overfitting in incremental learning. We observe in Figure 6 and Figure 7 that the model still exhibits the same behavior. This experiment is run with the minimal architecture described in Section C.1.

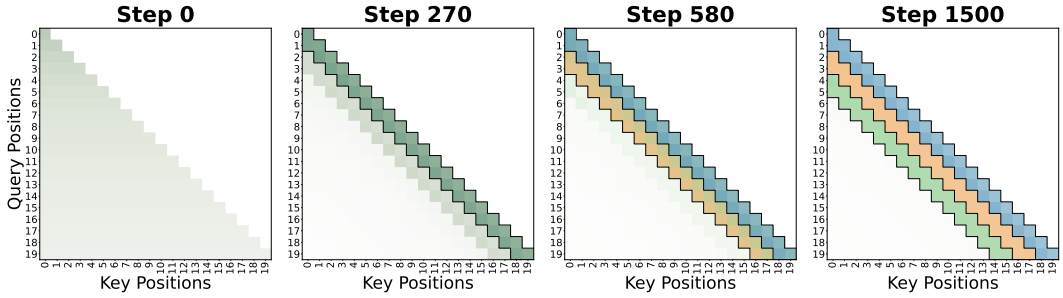

Figure 6: Attention patterns over the training steps with online sampling of data.

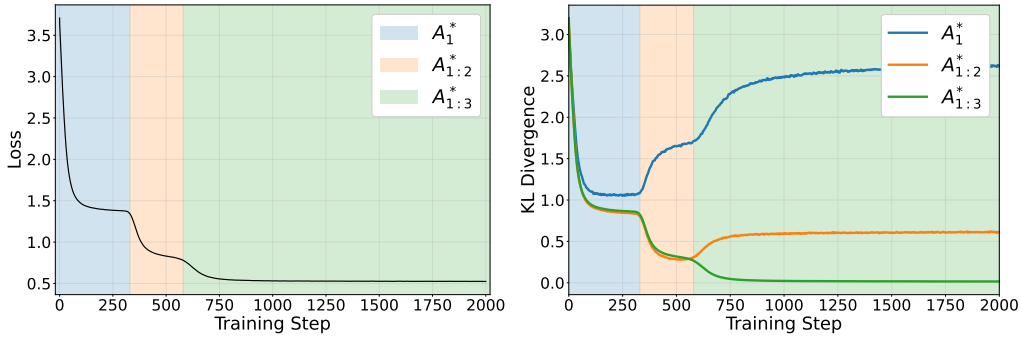

Figure 7: Validation loss and KL divergence over the training steps with online sampling of data.

C.4 Reverse Order

We reverse the order of importance of the intervals such that the most important interval is the furthest one. Figure 8 and Figure 9 show the results when $I(3) = \{12, 13\}$, $I(2) = \{8, 9, 10, 11\}$ and $I(1) = \{0, 1, 2, 3, 4, 5, 6, 7\}$ which reveals the same behavior as the original order. We also note that it is generally easier to observe incremental learning behaviour when the most important interval

is the furthest one. This indicates that the learning dynamics is impacted by the sequential structure of the task. This experiment is run with the full architecture described in Section B.

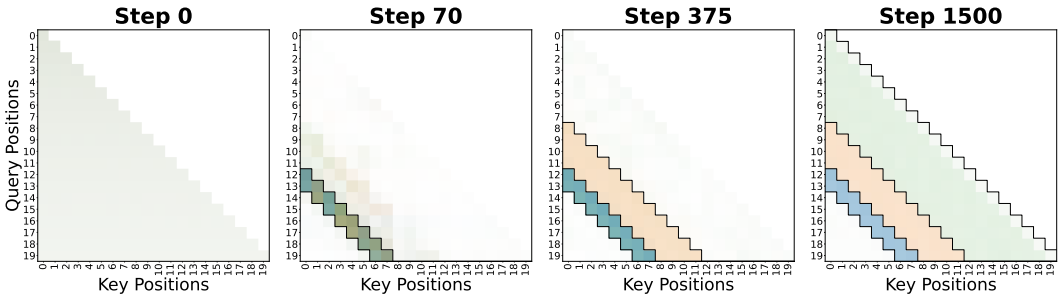

Figure 8: Attention patterns over the training steps with reversed order of importance and varying interval lengths.

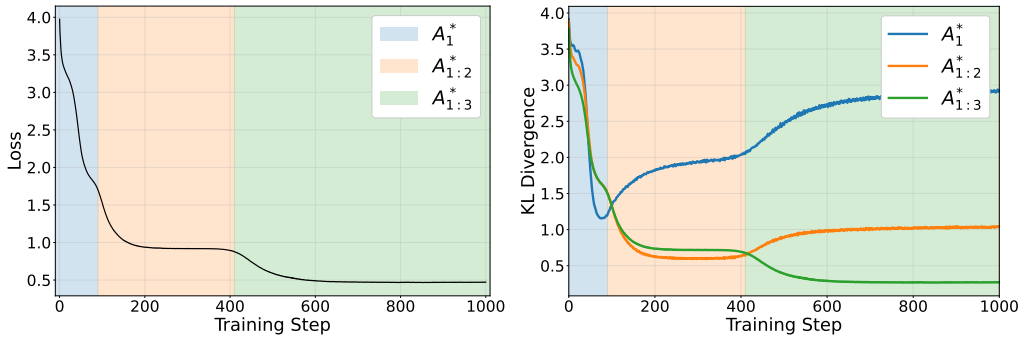

Figure 9: Validation loss and KL divergence over the training steps with reversed order of importance and varying interval lengths.

C.5 Weight Decay

We also study the impact of weight decay on the learning dynamics. We observe almost no difference in the learning dynamics when weight decay is not applied so we do not report the results.

C.6 Simulations

The square loss associated with θ is given as follows:

$$\mathcal{L}(\theta) = \frac{1}{2}\mathbb{E}_{x_1,\ldots,x_T,\xi}\left[\|y_\theta(X) - y^\star(X,\xi)\|^2\right] . \tag{8}$$

We present numerical simulations of the gradient flow dynamics of the loss in Equation (8) with the following parameters: $d = 50$, $T = 40$, $h = 3$, $|I(k)| = 1$ for all $k \in [h]$, $m = 1.7$, $\lambda = 0$. We initialize the value parameters V_i to 0 and the attention patterns s_i to $\frac{1}{T}1_T + \epsilon_i$ where ϵ_i are sampled from Gaussian distribution with zero-mean and ϵI_T covariance with $\epsilon = 10^{-6}$. Figure 10 shows the evolution of the attention patterns s_k, the value parameters V_k and the loss over time.

The results aligns with the transformer experiments in Section 2. Similar to the transformer experiments, the heads first learn from the position (1) and then the position (2) and finally the position (3). The time scales of these stages are clearly separated where the first stage is the fastest and the third stage is the slowest. Notably, at first, all heads tries to learn from the position (1) as it is related to the most important feature. After this competition phase, the heads start to learn from the position (2) and then the position (3) where they specialize in different patterns. Here, they cooperate to learn from the position (3). In particular, the first head offsets feature (3) as the third head's residual attention on the first position results in a cross term.

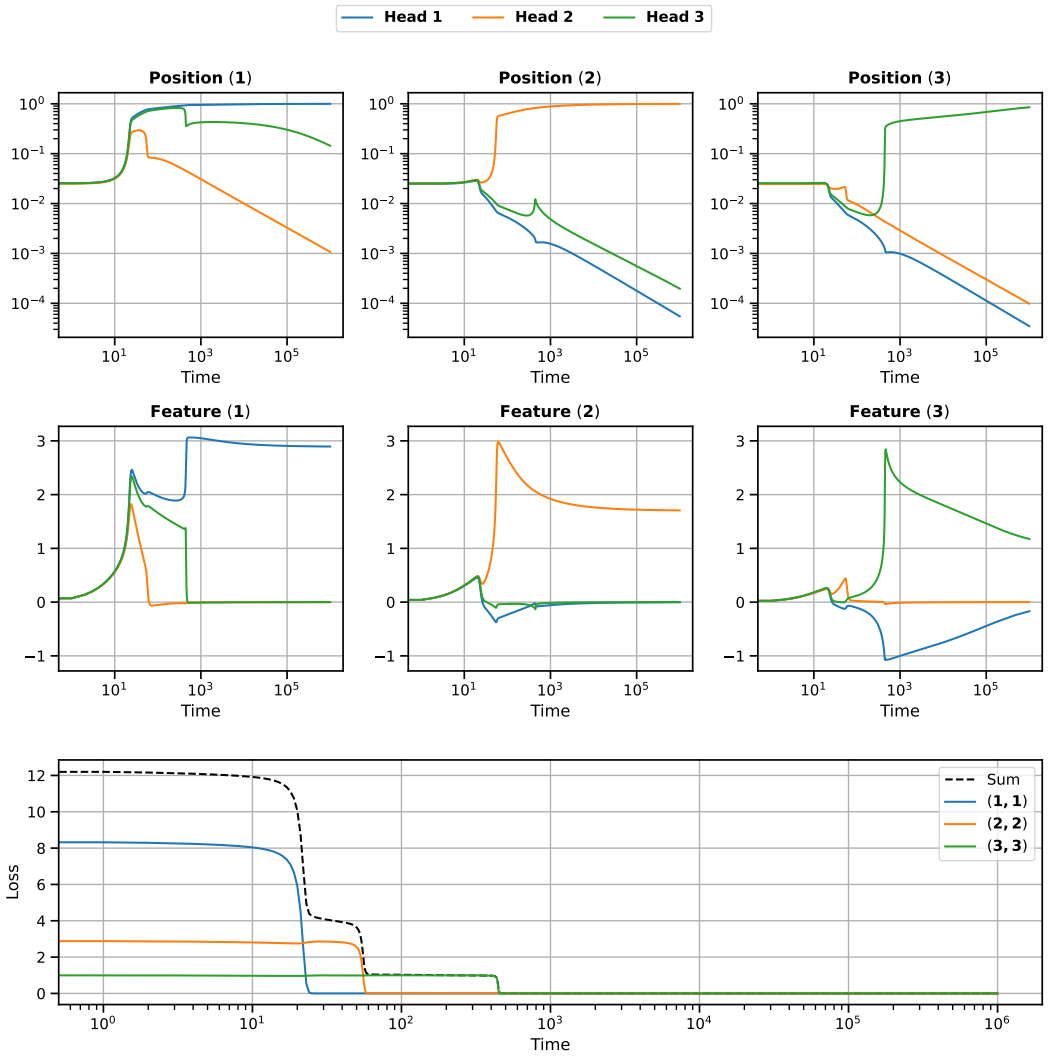

Figure 10: (Top) The evolution of the attention patterns s_k over time. (Middle) The evolution of the value parameters V_k over time. We only plot the relevant coordinates of s_k and V_k for clarity. (Bottom) The evolution of the loss over time. We decompose the loss into the (feature, position) contributions which are plotted in the color of the heads that learn these contributions.

D Missing Proofs

Tensor Notation. We construct tensors that are sum of outer products of matrices and vectors, i.e., $\boldsymbol{M} = \sum_{k=1}^{h} B_k \otimes v_k$ where $B_k \in \mathbb{R}^{d\times d}$ and $v_k \in \mathbb{R}^T$. The product $X^\top \boldsymbol{M}$ denotes $X^\top \boldsymbol{M} = \sum_{k=1}^{h} \langle B_k, X\rangle v_k$ whereas the product $\boldsymbol{M}v$ denotes $\boldsymbol{M}v = \sum_{k=1}^{h} B_k \langle v_k, v\rangle$. The inner product between two tensors $\boldsymbol{M} = \sum_{k=1}^{h} B_k \otimes v_k$ and $\boldsymbol{N} = \sum_{k=1}^{h} B'_k \otimes v'_k$ is denoted by $\langle \boldsymbol{M}, \boldsymbol{N}\rangle = \sum_{k=1}^{h} \langle B_k, B'_k\rangle\langle v_k, v'_k\rangle$. The Frobenius norm of a tensor $\boldsymbol{M}$ is given by $\|\boldsymbol{M}\|_F = \sqrt{\langle \boldsymbol{M}, \boldsymbol{M}\rangle}$.

Proposition 1. *The gradient flow dynamics is equivalent to a gradient flow on the following loss:*

$$\mathcal{L}(\theta) = \frac{1}{2}\|\boldsymbol{G} - \boldsymbol{P}\|_F^2\,, \quad \textit{where} \quad \boldsymbol{P} = \sum_{k=1}^{h} V_k \otimes s_k \;\; \textit{and} \;\; \boldsymbol{G} = \sum_{k=1}^{h} m_k^\star \left(V_k^\star \otimes s_k^\star\right)\,.$$

Proof. We start by some computations. Note that for any vectors $v_1, v_2 \in \mathbb{R}^T$, we have:

$$\begin{aligned}\mathbb{E}\left[(Xv_1)\,(Xv_2)^\top\right] &= \sum_{i=1}^{T}\sum_{j=1}^{T}(v_1)_i(v_2)_j\mathbb{E}\left[x_i x_j^\top\right] \\ &= \langle v_1, v_2\rangle I_d\,.\end{aligned}$$

Also, for any vectors $v_1, v_2 \in \mathbb{R}^T$ and any matrix $Q \in \mathbb{R}^{d\times d}$, we have:

$$\begin{aligned}\mathbb{E}\left[v_1 X^\top Q X v_2\right] &= \sum_{i=1}^{T}\sum_{j=1}^{T}(v_1)_i(v_2)_j\mathbb{E}\left[x_i^\top Q x_j\right] \\ &= \sum_{i=1}^{T}\sum_{j=1}^{T}(v_1)_i(v_2)_j\,\mathrm{Tr}\left(Q\mathbb{E}\left[x_j x_i^\top\right]\right) \\ &= \langle v_1, v_2\rangle\,\mathrm{Tr}(Q)\,.\end{aligned}$$

By selecting $v_2 = e_i$ for all $i \in [d]$, we get:

$$\mathbb{E}\left[v_1 X^\top Q X\right] = \mathrm{Tr}(Q)v_1\,.$$

First, the derivative with respect to V_i is as follows:

$$\begin{aligned}\frac{\partial\mathcal{L}(\theta)}{\partial V_i} &= \mathbb{E}_{X,\xi}\left[\left(f_\theta(X) - f^\star(X,\xi)\right)(Xs_i)^\top\right] \\ &= \sum_{j=1}^{h} V_j\langle s_i, s_j\rangle - \sum_{j=1}^{h} m_j^\star\langle s_i, s_j^\star\rangle V_j^\star\,.\end{aligned}$$

Next, the derivative with respect to q_i is as follows:

$$\begin{aligned}\frac{\partial\mathcal{L}(\theta)}{\partial q_i} &= \left(\mathrm{diag}(s_i) - s_i s_i^\top\right)\mathbb{E}_{X,\xi}\left[X^\top V_i^\top\left(f_\theta(X) - f^\star(X,\xi)\right)\right] \\ &= \left(\mathrm{diag}(s_i) - s_i s_i^\top\right)\left(\sum_{j=1}^{h}\langle V_i, V_j\rangle s_j - \sum_{j=1}^{h} m_j^\star\langle V_i, V_j^\star\rangle s_j^\star\right)\,.\end{aligned}$$

Then, the gradient flow dynamics is as follows:

$$\begin{aligned}\dot{V}_i &= -\nabla_{V_i}\mathcal{L}(\theta) = (\boldsymbol{G} - \boldsymbol{P})\,s_i \\ \dot{q}_i &= -\nabla_{q_i}\mathcal{L}(\theta) = \Pi(s_i)\left(V_i^\top(\boldsymbol{G} - \boldsymbol{P})\right)\,.\end{aligned}$$

This can be seen as a gradient ascent flow on the following loss:

$$\mathcal{L}(\theta) = \frac{1}{2}\|\boldsymbol{G} - \boldsymbol{P}\|_F^2\,.$$

□

Lemma 1. *Let s be a vector on the simplex that verifies $s_1 \geq s_j$ for all $j \in [h]$. Then, for any vector v that verifies $v_1 \geq v_j$ for all $j \in [h]$, we have for all $j \in [h]$:*

$$\left(\Pi(s)v\right)_1 \geq \left(\Pi(s)v\right)_j\,.$$

Proof. We have the following computations:

$$\begin{aligned}\left(\Pi(s)v\right)_1 &= s_1\left(v_1 - \langle s, v\rangle\right) \\ \left(\Pi(s)v\right)_j &= s_j\left(v_j - \langle s, v\rangle\right)\,.\end{aligned}$$

Then, we have:

$$\left(\Pi(s)v\right)_1 - \left(\Pi(s)v\right)_j \geq (s_1 - s_j)\left(v_1 - \langle s, v\rangle\right) \geq 0\,.$$

□

Theorem 1. *Assume that the initialization verifies the following for all $k \in [h]$:*

$$\langle V(0), V_1^\star \rangle \geq \langle V(0), V_k^\star \rangle \quad \langle s(0), s_1^\star \rangle \geq \langle s(0), s_k^\star \rangle \,. \tag{6}$$

Then, the dynamics of V and s converge to the following fixed point:

$$V(\infty) = \frac{m_1^\star}{H} V_1^\star \,, \quad s(\infty) = s_1^\star \,. \tag{7}$$

Proof. Let $\mathcal{R}$ be the following set:

$$\mathcal{R} = \left\{ (V, s) \mid \forall k \in [h], \langle V, V_1^\star \rangle \geq \langle V, V_k^\star \rangle \text{ and } \langle s, s_1^\star \rangle \geq \langle s, s_k^\star \rangle \right\} .$$

We prove that the flow is forward-invariant on $\mathcal{R}$.

Fix any $j \in [h]$. Let $w_j = \langle V, V_1^\star - V_j^\star \rangle$ and $z_j = \langle s, s_1^\star - s_j^\star \rangle$. The flow of w_j and z_j are as follows:

$$\begin{aligned} \dot{w}_j &= m_1^\star \langle s, s_1^\star \rangle - m_j^\star \langle s, s_j^\star \rangle - H \|s\|^2 w_j \,, \\ \dot{z}_j &= \Pi(s)^2 \left(m_1^\star \langle V, V_1^\star \rangle - m_j^\star \langle V, V_j^\star \rangle - H \|s\|^2 z_j \right) . \end{aligned}$$

On the boundary of $\mathcal{R}$, we have $w_j = 0$ or $z_j = 0$. If $w_j = 0$, then $\dot{w}_j \geq 0$ and if $z_j = 0$, then $\dot{z}_j \geq 0$ by Lemma 1. Therefore, a flow that has started in $\mathcal{R}$ will remain in $\mathcal{R}$ for all time.

Now, consider the following Lyapunov function:

$$\phi(V, s) = \langle V, \boldsymbol{G} s \rangle - \frac{H}{2} \|V\|_F^2 \|s\|^2 \,. \tag{9}$$

The derivative of $\phi(V, s)$ is as follows:

$$\begin{aligned} \nabla_V \phi(V, s) &= \boldsymbol{G} s - H \|s\|^2 V \,, \\ \nabla_s \phi(V, s) &= V^\top \boldsymbol{G} - H \|V\|_F^2 s \,. \end{aligned}$$

Therefore, the time derivative of ϕ:

$$\dot{\phi}(V, s) = \|\dot{V}\|^2 + \|\Pi(s) \nabla_s \phi(V, s)\|^2 \, \geq 0 \,.$$

ϕ is optimized when $V = \dfrac{\boldsymbol{G} s}{H \|s\|^2}$ which leads to a finite value upper bound on $\phi(V, s)$. Therefore, $\lim_{t \to \infty} \phi(V(t), s(t))$ is finite and the flow converges to a stationary point of ϕ. That is, the flow converges to a point that verifies:

$$\boldsymbol{G} s - H \|s\|^2 V = 0 \,, \quad V^\top \boldsymbol{G} - H \|V\|_F^2 s \in \ker(\pi(s)) \,. \tag{10}$$

Note that, we have the following equality:

$$(\boldsymbol{G} s)^\top \boldsymbol{G} = \sum_{j=1}^h m_j^\star \left\langle V_j^\star, \sum_{k=1}^h m_k^\star V_k^\star \langle s_k^\star, s \rangle \right\rangle s_j^\star = \sum_{j=1}^h (m_j^\star)^2 \langle s_j^\star, s \rangle s_j^\star \,.$$

Then, the stationary point verifies for any non-zero components of s:

$$\sum_{j=1}^h (m_j^\star)^2 \langle s_j^\star, s \rangle s_j^\star = H^2 \|s\|^2 \|V\|_F^2 \langle s_j^\star, s \rangle s_j^\star + \alpha 1_T \,, \quad \text{for some} \quad \alpha \in \mathbb{R} \,.$$

Any non-zero components of s needs to be processed with the same weight $m_j^\star$ or otherwise this condition is not satisfied. However, we have proven that the trajectories are forward-invariant on $\mathcal{R}$ and that $\mathcal{R}$ is closed. Therefore, the flow converges to the stationary point

$$s = s_1^\star \,, \quad V = \frac{m_1^\star}{H} V_1^\star \,.$$

□

Theorem 2. *Assume that $V(0)$ and $s(0)$ such that $\forall k \in [h]$: $\|V(0) - V_k(0)\|_F \leq \epsilon$ and $\|s(0) - s_k(0)\|_2 \leq \epsilon$, where $\epsilon \ll 1$. Then, there exists a universal constant c_1 such that*

$$\|V_k(t) - V(t)\|_F \leq \epsilon e^{c_1 t} \quad \textit{and} \quad \|s_k(t) - s(t)\|_2 \leq \epsilon e^{c_1 t} \,, \quad \forall t \in \left[0, \frac{1}{-c_1 \log \epsilon} \right] .$$

Proof. We write the flow of V_i and s_i in terms of the flow of V and s by new variables:

$$W_i = V_i - V\,, \quad z_i = s_i - s\,.$$

Let ϵ be the following quantity:

$$\epsilon = \max_{j\in[h]} \max\{\|W_j\|_F, \|z_j\|\}\,.$$

We are interested in the regime where $\epsilon \ll 1$.

Recall that, $\phi(V, s)$ defined in Equation (9) is always non-decreasing. Therefore, V cannot grow larger than $\dfrac{\boldsymbol{G}s}{H\|s\|^2}$ in norm or otherwise $\phi(V, s)$ would decrease. This is the optimal value of V for a particular s. Thus, we have a time-independent upper bound $|V| \le \max_s \dfrac{\boldsymbol{G}s}{H\|s\|^2} = \dfrac{m_1^\star}{H}$.

Then, the flow of W_i and z_i is as follows:

$$\begin{aligned}
\dot{W}_i &= \boldsymbol{G}z_i - \boldsymbol{P}s_i + H\|s\|^2 V\,,\\
\dot{z}_i &= \Pi(s_i)^2\left(V_i^\top(\boldsymbol{G}-\boldsymbol{P})\right) - \Pi(s)^2\left(V^\top\boldsymbol{G} - H\|V\|^2 s\right)\,.
\end{aligned}$$

Note that, $\boldsymbol{P}$ can be rewritten as follows:

$$\boldsymbol{P} = \sum_{j=1}^h V_j \otimes s_j = HV\otimes s + \left(\sum_{j=1}^h W_j\right)\otimes s + V\otimes\left(\sum_{j=1}^h z_j\right) + \left(\sum_{j=1}^h W_j\otimes z_j\right)\,.$$

This implies that:

$$V^\top\boldsymbol{P} = H\|V\|^2 s + \mathcal{O}\left(\epsilon+\epsilon^2\right)\,, \quad \boldsymbol{P}s = H\|s\|^2 V + \mathcal{O}\left(\epsilon+\epsilon^2\right)\,.$$

We can rewrite the flow of z_i as follows:

$$\dot{z}_i = \left(\Pi(s_i)^2 - \Pi(s)^2\right)\left(V_i^\top(\boldsymbol{G}-\boldsymbol{P})\right) + \Pi(s)^2\left(W_i^\top\boldsymbol{G} - V_i^\top\boldsymbol{P} + H\|V\|^2 s\right)\,.$$

Therefore, we have:

$$\dot{W}_i = \mathcal{O}(\epsilon), \quad \dot{z}_i = \mathcal{O}(\epsilon)\,.$$

The norm of W_i and z_i are then evolve as follows:

$$\dot{\widehat{\|W_i\|}} = \frac{\dot{W_i}^\top W_i}{\|W_i\|} \le \|\dot{W}_i\| = \mathcal{O}(\epsilon)\,.$$

We similarly derive that $\|\dot{z}_i\| = \mathcal{O}(\epsilon)$.

This implies that ϵ verifies the equation:

$$\dot{\epsilon} \le C\epsilon\,, \quad \text{as long as } \epsilon \ll 1\,,$$

where C is a constant that depends on the problem parameters H and $\boldsymbol{G}$. From the Grönwall's inequality, we have:

$$\epsilon(t) \le \epsilon(0)e^{Ct}\,, \quad \text{as long as } t \in \left[0, \frac{1}{-C\log\epsilon(0)}\right]\,.$$

□

D.1 Initialization of Theorem 1

We can relax the initialization condition of Theorem 1 via a Taylor expansion around $t = 0$. This follows the approach of Zucchet et al. [2025], who have studied the escape time from this initialization when the data covariance has $\frac{1}{d}$ and the loss has $\frac{1}{T}$ scaling, leading to a slowdown of order $\frac{1}{dT}$ compared to our setting.

Remark 1. *The initialization of interest is $s_k(0) \approx \frac{1}{T}1_T$ for all $k \in [h]$ as seen in Figure 2. By expanding the dynamics around this initialization with $V_k \approx 0$ for all $k \in [h]$, we get: $\dot{V}_k(0) \approx \frac{1}{T}\boldsymbol{G}1_T\,, \quad \dot{s}_k(0) \approx 0$. Similarly, second-order local approximation shows that s_k has the largest increase towards the direction $s_1^\star$. Therefore, we can quantify a wider basin of attraction for Theorem 1 as all V_k and s_k move towards the initialization space defined by Equation* (6).