DISTINCT COMPUTATIONS EMERGE FROM COMPOSITIONAL CURRICULA IN IN-CONTEXT LEARNING

Anonymous authors
Paper under double-blind review

ABSTRACT

In-context learning (ICL) typically presents a function through a uniform sample of input-output pairs. Here, we investigate how presenting a compositional subtask curriculum in context may alter the computations that the model learns. We design a compositional algorithmic task based on the modular exponential—a double exponential task composed of two single exponential subtasks—and train transformer models to learn the task in-context. We compare the model when trained (a) using an in-context curriculum consisting of single exponential subtasks and, (b) the model trained directly on the double exponential task without such a curriculum. We show that the model trained with a subtask curriculum can perform zero-shot inference on unseen compositional tasks and is more robust given the same context length. We study how the task is represented across the two training regimes, in particular whether subtask information is represented. We find that the model employs different mechanisms, possibly changing through training, in a way modulated by the data properties of the in-context curriculum.

1 INTRODUCTION

Many complex real-world tasks consist of the composition of intermediate functions or subtasks. This notion of systematic compositionality has been extensively studied (Chomsky, 1999; Frege, 1948; Szabó, 2024) and is a key in flexible intelligence, enabling "infinite use from finite means." Nevertheless, it has been a central controversy whether neural networks can exhibit human-like compositionality Fodor & Pylyshyn (1988); Smolensky (1988); Lake & Baroni (2018). The recent success of Large Language Models (LLMs) has only brought this controversy to a head, given their often inscrutable nature yet remarkable generalization capabilities, especially their ability to adapt to context (Brown et al., 2020; Wei et al., 2022; Lampinen et al., 2024).

On the other hand, breaking down a complex task into its intermediate components not only makes the task easier but also supports identifying the correct 'components' of the task, which is crucial for robustness and generalization. There is a vast space of tasks where models can perform well on the training data distribution by learning surface correlations rather than identifying the true task structure, a phenomenon extensively studied as *shortcut learning* (Arjovsky et al., 2019; McCoy, 2019; Geirhos et al., 2020; Hermann et al., 2023). Similarly, for tasks defined by composition of several functions or subtasks, the model can either find a surrogate strategy rather than learning underlying true compositional structure (Dziri et al., 2024).

Inspired by the above observation, we investigate whether a transformer can infer and leverage information about the task structure from an in-context *curriculum*—in-context examples of component functions—to perform more robustly on a compositional task. We design an algorithmic task utilizing the composition of two modular exponential tasks defined by two exponent bases, (a, b). The curriculum is defined by examples of single exponentials from each base. We train the model with example sequences sampled from training task sets and evaluate its generalization ability on unseen combinations of (a, b). We first demonstrate that a curriculum-trained model is capable of zero-shot inference on unseen compositional task queries and shows higher robustness compared to a model trained without a curriculum (Section 3.1). We show correlative evidence that the curriculum enables the model to represent and combine the task parameters to be composed (Section 3.2). Finally, we study how the length of the curriculum affects the model's learning and potentially which strategy is used to solve the compositional task (Section 3.3).

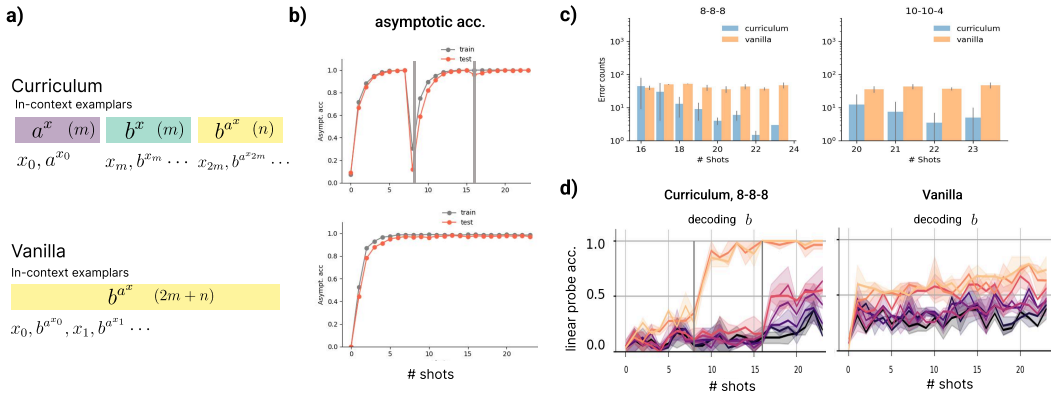

Figure 1: Overview of the setup and key results. **a)** Task schema. In curriculum training, each training sequence is composed of m exemplars for two single-exponential tasks defined by a and b, respectively, followed by n composite double-exponential task exemplars. In vanilla training, the model is trained with a sequence of $2m + n$ in-context exemplars for the double-exponential task defined by task parameters (a, b). **b)** Example asymptotic accuracy at the end of training on training and evaluation tasks for curriculum training ($m = 8$, $n = 8$) (top) and vanilla training (bottom). **c)** Comparison of the error counts for the last compositional block for curriculum training and vanilla training (left: $m = 8, n = 8$, right: $m = 10, n = 4$). While both models perform fairly well, the curriculum condition enhances robustness on unseen compositional task block. **d)** In-context curriculum can promotes the utilization of the task parameter. Linear probe decoding accuracy of task parameter b from unseen evaluation sequences. The curriculum trained model represents the task parameter in compositional task block while vanilla model does not.

2 Experimental Setup

2.1 Task

We use a modular arithmetic task of composition of an exponential function, namely $b^{a^x} \mod P$, referred to as the modular double exponential task. Inspired by well-studied linear modular arithmetic tasks such as summation, multiplication or both He et al. (2024); Nanda et al. (2023a), we chose the modular double exponential function for its greater complexity while still offering a deterministic functional mapping and effectively constraining the vocabulary size. We design *curriculum* in-context exemplars, which provide blocked examples for $a^x \mod P$ and $b^x \mod P$, followed by $b^{a^x} \mod P$. In contrast, *vanilla* in-context exemplars consist only of $b^{a^x} \mod P$. We train transformer architecture on these sequences using a next token prediction task for every (x, y) pair in the sequence, rather than only on the final query.

To make fair information gain during training, we provide single exponential task exemplars in *vanilla* training as well, but the main difference is that they are not given in-context together with the compositional task. In every example sequence, we randomly sample task parameters (a, b), and the model needs to learn to adapt its answer in-context according to (a, b). The task combinations (a, b) seen during the training include all possible individual a, b, but not all pairs. We evaluate the trained model on unseen combinations of (a, b). We permit integers $x \in [0, P)$, and a, b are sampled from the primitive roots of P. Throughout the main experiments, we focus on $P = 37$, but we extend our findings to another P value in Appendix A.2.

In the *curriculum* setting, we use an equal curriculum length m for each exponential task (a^x, b^x) and n for the compositional task. We use the same total length, $2m + n$, for the exemplars in the *vanilla* setting. While varying the compositional task length in the *curriculum*, we maintain the importance of the compositional task equal to each single exponential task in the curriculum by controlling the weighting factor for the loss contribution from the compositional task (namely, making it such that the loss from the compositional task is 1/3 of the total loss). Similarly, for a fair comparison, in the vanilla setting the network sees some sequences for a single exponential task, with a ratio of 2 to 1 (to match the overall weight of the compositional task to the curriculum setting). By doing

this, we effectively make it so that the same (exemplar, label) pairs are seen in both curriculum and vanilla settings, with the same loss weight to single vs. double exponential tasks. The key difference between the two settings is the in-context correlations: In the curriculum setting, these correlations are more complex/hierarchical (possibly leading to in-context compositional), while in the vanilla setting, they focus on single function learning (the standard few-shot ICL setting).

We fix the total context length to 48, equivalent to 24 pairs of (x, y), and ensure that all x in context are unique.

2.2 Model and Training

We train 8-layer transformers with sinusoidal positional embeddings with time constant of 120, a hidden dimension size of 128, and 8 heads, using the Adam optimizer, a learning rate of 7.5×10^{-4}, and a batch size of 512. All results we report are based on 2 data seeds and trained with 2×10^8 sequences unless otherwise mentioned. Specifically, for the vanilla model, we further trained it up to 3×10^8 sequences to ensure that the model's performance is saturated. See also Appendix A.1 for the total error counts of the vanilla and curriculum models and example loss curve and performance evolution.

3 Results

3.1 Subtask curriculum can increase the robustness of in-context learning of modular double exponential task

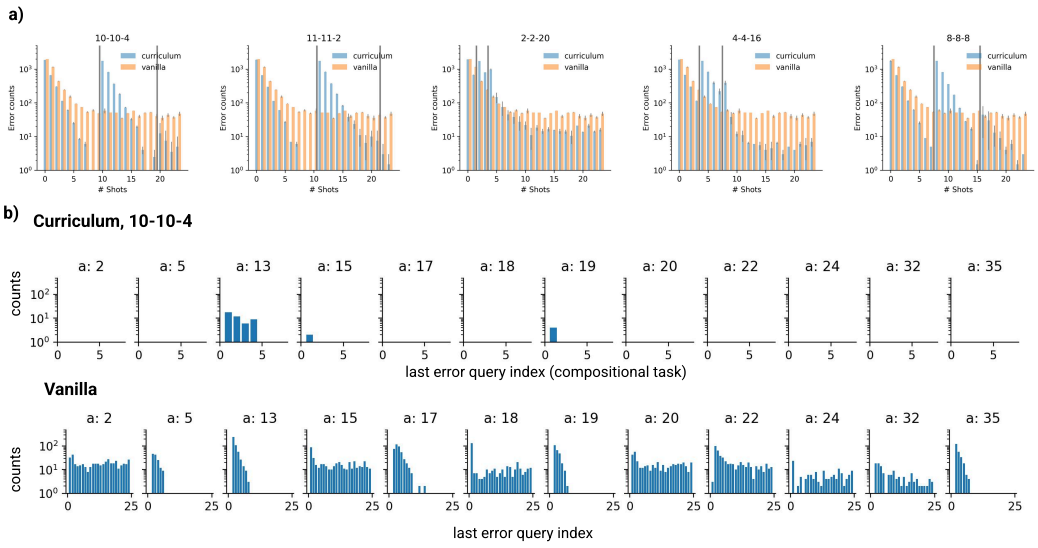

Figure 2: In-context error counts of vanilla-trained model vs. in-context curriculum-trained model. **a)** Mean error counts after each number of examples in 2K evaluation sequences with unseen task parameter combinations (a, b) over 2 seeds. Each panel title indicates a different curriculum length m-m-n, and the gray vertical lines denote the curriculum task boundaries. **b)** The last index of error for the compositional task (if an error occurred).

First, we demonstrate that the subtask curriculum can make in-context learning of unseen compositional task more robust. In both cases, the models tend to reach near-perfect performance on the task, as shown in Figure 1b. In Figure 2a, we examine the performance closely with the error counts after each number of in-context exemplars for 2K evaluation sequences sampled with unseen task parameter combinations (a, b). Notice that the curriculum enables the model to zero-shot infer the correct answer for the compositional task after single-task blocks. This effect is more prominent with the longer curriculum blocks, which allow the model to correctly identify the task parameter (a, b). In contrast, as more examples are shown, vanilla training also results in fewer errors, but the error counts eventually saturate.

In Figure 2b, we show the distribution of the index where the last error was made for different a task parameters. A left-skewed histogram indicates that the model requires only a few examples to make a correct inference of the task. A histogram that is more distributed to the right suggests that the model tends to have higher uncertainty and requires more examples to make correct inference.

3.2 Curriculum Changes the Representation of Compositional Tasks in the Model

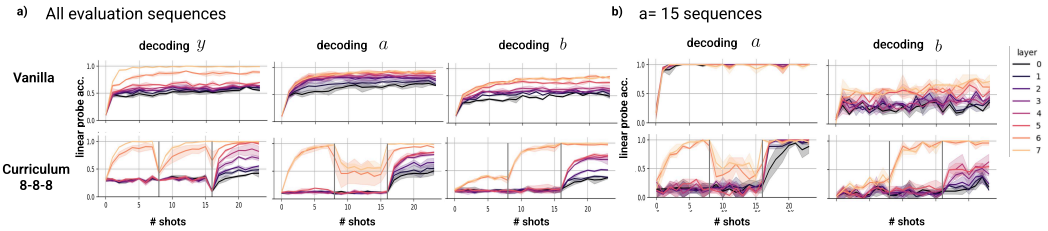

Figure 3: Linear probing of task parameters (a, b). **a)** Linear probe decoding accuracy of $y = b^{a^x}$, task parameters b and a for the vanilla model and the curriculum model ($m = 8$, $n = 8$) across all evaluation sequences. **b)** Linear probe decoding accuracy of b and a for the $a = 15$ evaluation sequences, where the vanilla model is more susceptible to failure (see Figure 2).

In the previous section, we observed that the model can benefit from having a subtask curriculum for solving compositional tasks in-context, particularly its zero-shot inference ability on the compositional task and higher robustness given the same context window, compared to that of the vanilla model. In this section, we ask *how* does in-context curriculum enable this compositional solution.We hypothesize that the in-context curriculum of the single exponential tasks provides unambiguous information about the task parameters a and b, which can be used to solve the double exponential task.

To verify this hypothesis, we use a linear probe to decode the subtask parameters in-context from the internal representation of the trained model. Yet simple, linear probing method has been widely used to study the internal representations of transformers Gurnee et al. (2023); Nanda et al. (2023b). We trained a linear classifier on the output of each layer from evaluation sequences to decode a task parameter (a, b) for each y position. We used $80/20$ split of the 1K unseen test sequences for linear probe training testing of the decoding accuracy. [1].

In both the vanilla and curriculum models, decoding of y values becomes near perfect at the final layer, as expected from the high accuracy (Figure 1b). However, we found a noticeable difference in decoding of the task parameters. With in-context curriculum, task parameters (a, b) are faithfully decoded in the corresponding curriculum task block (Figure 3a, bottom panel). The decoding accuracy of both (a, b) is also high in the compositional task block (shots 16-24), suggesting that the representation of the task parameters inferred from the curriculum is utilized in the compositional task. In contrast, the vanilla-trained model shows lower decoding accuracy for the task parameters, especially parameter b. The difference becomes clearer when we examine the failure cases of the vanilla model, namely $a = 15$ (see Figure 2b, with widely spread errors across the context window). Since we train and test on $a = 15$ evaluation sequences, decoding of a becomes trivial, but decoding of b is much worse for the vanilla model, while it is maintained near perfect in the curriculum model in layers 6 and 7.

Additionally, we visualized the attention pattern of the heads in layers 6 and 7, where we can find heads that attend to the earlier curriculum block from the compositional task block. This aligns with the linear probe result and indicates that the curriculum-trained model encodes and utilizes the task parameters inferred from the curriculum block.

In brief, we show correlative evidence that the model trained with in-context curriculum encodes the task parameters (a, b) in its internal representation and is capable of using them for the compositional

[1]We performed a control experiment using shuffled labels in Appendix A.3, which aligns with the baseline performance in Figure 3.

task using linear probes. In contrast, the vanilla model does not necessarily encode the correct task parameters and we show the correlation between the decoding accuracy of the task parameters and the robustness using targeted examples.

3.3 Which strategy to use? Curriculum length changes the model's learning strategy on compositional tasks

We showed that the in-context curriculum can enhance the model's robustness on compositional generalization and promotes encoding of the task parameters. However, since the compositional task can be learned well from the training data without curriculum (as seen in the vanilla model), it is unclear why the model learns to use the task parameters inferred from the single exponential task examples. In other words, the model can learn the compositional task independently, even when the curriculum sequence is provided, without utilizing the task parameters.

In this section, we take the first step to answer this question by looking closely at the loss evolution and linear decoding probe across the training phase. In Figure 4, we observe a noticeable difference in the order in which each task is learned when the curriculum length is varied. When the compositional task sequence is long enough (Figure 4a), the model first learns the compositional task without mastering each single exponential task, and thus without being fully able to utilize the task parameters (black arrow). Only after learning each single exponential task does the model become capable of zero-shot inference, possibly using the carried information of (a, b), as shown in the linear probe decoding of a and b in layers 6 and 7 (gray arrow)[2]. With a shorter compositional task length (Figure 4b), the single exponential tasks and compositional task are learned concurrently (black arrow), and the convergence of loss at the zero-shot happens simultaneously with the rest of the compositional task (gray arrow).

Note that since we control the weighting of the loss from the compositional task to be invariant to the number of examples (i.e., the compositional task block and each single exponential task block always have the same importance), we effectively control the information gain from the varying number of examples only, rather than differences in loss contribution stemming from the number of examples.

All these observations serve as the first evidence that the different designs of the curriculum affect the model's learning mechanism for the compositional task, which requires more thorough investigation.

4 Related Works

In-context learning has brought significant interest recently, particularly due to the emergent capabilities of LLMs (Wei et al., 2022). Broadly, in-context learning can be seen as a special case of few-shot learning, where the model adapts and generalizes to unseen input examples without requiring gradient updates. In earlier works on meta-learning (Santoro et al., 2016; Vinyals et al., 2016; Wang et al., 2016), it was shown that neural networks trained with specific objectives or data can perform few-shot learning. In the recent works on transformer architecture (Vaswani, 2017) in scale, researchers found that in-context learning can emerge from auto-regressive next-token prediction tasks without specific tuning of the training objective. Many studies (Chan et al., 2022; Xie et al., 2022; Raventós et al., 2024) have highlighted the importance of data properties for in-context learning. Furthermore, the transient and non-monotonic nature of in-context learning has been investigated (Singh et al., 2024a;b). A few studies (Hendel et al., 2023; Todd et al., 2023) have explored how different in-context tasks can be represented in LLMs in the form of task vectors.

Curriculum learning is critical in learning of humans and animals, well-attested in a body of literature (Skinner, 2019; Elio & Anderson, 1984; Clerkin et al., 2017; Dekker et al., 2022). While its potential importance has long been acknowledged in machine learning community (Bengio et al., 2009; Wang et al., 2021), the benefit from curriculum has been shown marginal in standard supervised learning benchmarks (Wu et al., 2021). However, the right curricula show greater significance in the context of reinforcement learning (Karpathy & Van De Panne, 2012; Tessler et al., 2017;

[2]See Appendix A.6 for more detailed analysis indicating that the model develops independent strategy for compositional task in the beginning.

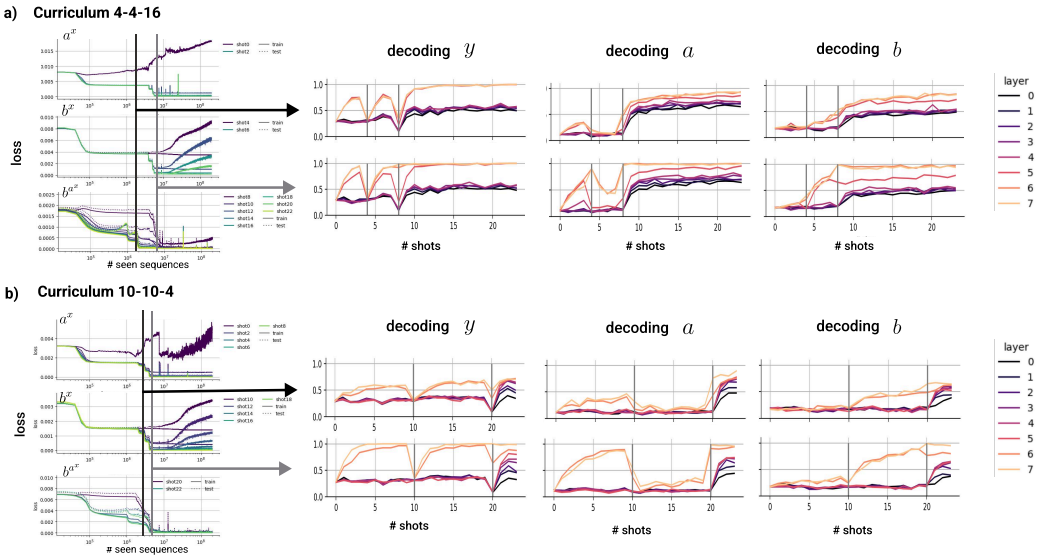

Figure 4: Different curriculum lengths lead to different learning strategies. **a)** When the compositional task sequence is longer ($m = 4, n = 16$), the model first learns the compositional task independently (top panel, black arrow), but later learns to compose task parameters for zero- or first-shot inference (bottom panel, gray arrow). **b)** When the compositional task sequence is shorter ($m = 10, n = 4$), the model relies on task parameters inferred from the curriculum block. Before learning each exponential task, the compositional task performance is limited by the single exponential task performance (top panel, black arrow). The compositional task block loss decreases simultaneously with the single exponential task loss (bottom panel, gray arrow).

Narvekar et al., 2020). In particular, Lee et al. (2024) shows theoretical evidence of importance of subtask curricula in reinforcement learning of compositional tasks.

Modular arithmetic tasks have been used in a rich body of literature to understand how sequence models, such as transformers, can implement the internal mechanisms required to solve these tasks. For example, Power et al. (2022); Zhong et al. (2023) used simple modular addition tasks to investigate the grokking phenomenon and demonstrate that transformers can implement multiple solutions. He et al. (2024) studied how transformers can learn skill composition in-context with out-of-distribution tasks. Our work builds on these findings by exploring how transformers can utilize a curriculum of subtasks given in-context to achieve compositional generalization.

5 Discussion

We investigated how transformers can leverage inferred subtask information from an in-context curriculum to generalize to unseen compositional tasks, using a modular double exponential task as a case study. We demonstrated that incorporating a curriculum enables zero-shot inference on compositional tasks and increases model robustness. As an initial step to understand the model's internal workings, we used a linear probe to explore how the model processes the curriculum. We found that the internal representation encodes task parameters from the curriculum blocks, and these parameters are effectively decoded as the compositional task sequence is processed. By analyzing targeted failure cases, we showed that the decoding accuracy of the task parameters is correlated with model performance, which may explain why the curriculum-trained model is more robust. Finally, we observed that the amount of compositional task information provided in-context (controlled by curriculum length) affects both the learning strategy and the evolution of task representations during training. Our observations suggest the importance of the data property present in-context, such as curriculum, can impact compositional generalization.

Our analysis in this paper is limited to correlational evidence. Further analysis with causal manipulation would be necessary to gain a more precise understanding of the mechanisms behind the observed model behavior.

REFERENCES

Martin Arjovsky, Léon Bottou, Ishaan Gulrajani, and David Lopez-Paz. Invariant risk minimization. *arXiv preprint arXiv:1907.02893*, 2019.

Yoshua Bengio, Jérôme Louradour, Ronan Collobert, and Jason Weston. Curriculum learning. In *Proceedings of the 26th annual international conference on machine learning*, pp. 41–48, 2009.

Tom Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared D Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, et al. Language models are few-shot learners. *Advances in neural information processing systems*, 33:1877–1901, 2020.

Stephanie Chan, Adam Santoro, Andrew Lampinen, Jane Wang, Aaditya Singh, Pierre Richemond, James McClelland, and Felix Hill. Data distributional properties drive emergent in-context learning in transformers. *Advances in Neural Information Processing Systems*, 35:18878–18891, 2022.

Noam Chomsky. Derivation by phase. 1999. URL `https://api.semanticscholar.org/CorpusID:118158028`.

Elizabeth M Clerkin, Elizabeth Hart, James M Rehg, Chen Yu, and Linda B Smith. Real-world visual statistics and infants' first-learned object names. *Philosophical Transactions of the Royal Society B: Biological Sciences*, 372(1711):20160055, 2017.

Ronald B Dekker, Fabian Otto, and Christopher Summerfield. Curriculum learning for human compositional generalization. *Proceedings of the National Academy of Sciences*, 119(41): e2205582119, 2022.

Nouha Dziri, Ximing Lu, Melanie Sclar, Xiang Lorraine Li, Liwei Jiang, Bill Yuchen Lin, Sean Welleck, Peter West, Chandra Bhagavatula, Ronan Le Bras, et al. Faith and fate: Limits of transformers on compositionality. *Advances in Neural Information Processing Systems*, 36, 2024.

Renee Elio and John R Anderson. The effects of information order and learning mode on schema abstraction. *Memory & cognition*, 12(1):20–30, 1984.

Jerry A Fodor and Zenon W Pylyshyn. Connectionism and cognitive architecture: A critical analysis. *Cognition*, 28(1-2):3–71, 1988.

Gottlob Frege. Ueber sinn und bedeutung. *Philosophical Review*, 57(n/a):209, 1948.

Robert Geirhos, Jörn-Henrik Jacobsen, Claudio Michaelis, Richard Zemel, Wieland Brendel, Matthias Bethge, and Felix A Wichmann. Shortcut learning in deep neural networks. *Nature Machine Intelligence*, 2(11):665–673, 2020.

Wes Gurnee, Neel Nanda, Matthew Pauly, Katherine Harvey, Dmitrii Troitskii, and Dimitris Bertsimas. Finding neurons in a haystack: Case studies with sparse probing. *arXiv preprint arXiv:2305.01610*, 2023.

Tianyu He, Darshil Doshi, Aritra Das, and Andrey Gromov. Learning to grok: Emergence of in-context learning and skill composition in modular arithmetic tasks. *arXiv preprint arXiv:2406.02550*, 2024.

Roee Hendel, Mor Geva, and Amir Globerson. In-context learning creates task vectors. *arXiv preprint arXiv:2310.15916*, 2023.

Katherine L Hermann, Hossein Mobahi, Thomas Fel, and Michael C Mozer. On the foundations of shortcut learning. *arXiv preprint arXiv:2310.16228*, 2023.

Andrej Karpathy and Michiel Van De Panne. Curriculum learning for motor skills. In *Advances in Artificial Intelligence: 25th Canadian Conference on Artificial Intelligence, Canadian AI 2012, Toronto, ON, Canada, May 28-30, 2012. Proceedings 25*, pp. 325–330. Springer, 2012.

Brenden Lake and Marco Baroni. Generalization without systematicity: On the compositional skills of sequence-to-sequence recurrent networks. In *International conference on machine learning*, pp. 2873–2882. PMLR, 2018.

Andrew Kyle Lampinen, Stephanie C. Y. Chan, Aaditya K. Singh, and Murray Shanahan. The broader spectrum of in-context learning, 2024. URL https://arxiv.org/abs/2412.03782.

Jin Hwa Lee, Stefano Sarao Mannelli, and Andrew Saxe. Why do animals need shaping? a theory of task composition and curriculum learning. *arXiv preprint arXiv:2402.18361*, 2024.

RT McCoy. Right for the wrong reasons: Diagnosing syntactic heuristics in natural language inference. *arXiv preprint arXiv:1902.01007*, 2019.

Neel Nanda, Lawrence Chan, Tom Lieberum, Jess Smith, and Jacob Steinhardt. Progress measures for grokking via mechanistic interpretability, 2023a. URL https://arxiv.org/abs/2301.05217.

Neel Nanda, Andrew Lee, and Martin Wattenberg. Emergent linear representations in world models of self-supervised sequence models. *arXiv preprint arXiv:2309.00941*, 2023b.

Sanmit Narvekar, Bei Peng, Matteo Leonetti, Jivko Sinapov, Matthew E Taylor, and Peter Stone. Curriculum learning for reinforcement learning domains: A framework and survey. *Journal of Machine Learning Research*, 21(181):1–50, 2020.

Alethea Power, Yuri Burda, Harrison Edwards, Igor Babuschkin, and Vedant Misra. Grokking: Generalization beyond overfitting on small algorithmic datasets. *CoRR*, abs/2201.02177, 2022. URL https://arxiv.org/abs/2201.02177.

Allan Raventós, Mansheej Paul, Feng Chen, and Surya Ganguli. Pretraining task diversity and the emergence of non-bayesian in-context learning for regression. *Advances in Neural Information Processing Systems*, 36, 2024.

Adam Santoro, Sergey Bartunov, Matthew Botvinick, Daan Wierstra, and Timothy Lillicrap. Meta-learning with memory-augmented neural networks. In *International conference on machine learning*, pp. 1842–1850. PMLR, 2016.

Aaditya Singh, Stephanie Chan, Ted Moskovitz, Erin Grant, Andrew Saxe, and Felix Hill. The transient nature of emergent in-context learning in transformers. *Advances in Neural Information Processing Systems*, 36, 2024a.

Aaditya K Singh, Ted Moskovitz, Felix Hill, Stephanie CY Chan, and Andrew M Saxe. What needs to go right for an induction head? a mechanistic study of in-context learning circuits and their formation. *arXiv preprint arXiv:2404.07129*, 2024b.

Burrhus Frederic Skinner. *The behavior of organisms: An experimental analysis*. BF Skinner Foundation, 2019.

Paul Smolensky. On the proper treatment of connectionism. *Behavioral and brain sciences*, 11(1): 1–23, 1988.

Zoltán Gendler Szabó. *Compositionality*. Metaphysics Research Lab, Stanford University, fall 2024 edition, 2024. URL https://plato.stanford.edu/archives/fall2024/entries/compositionality/.

Chen Tessler, Shahar Givony, Tom Zahavy, Daniel Mankowitz, and Shie Mannor. A deep hierarchical approach to lifelong learning in minecraft. In *Proceedings of the AAAI conference on artificial intelligence*, volume 31, 2017.

Eric Todd, Millicent L Li, Arnab Sen Sharma, Aaron Mueller, Byron C Wallace, and David Bau. Function vectors in large language models. *arXiv preprint arXiv:2310.15213*, 2023.

A Vaswani. Attention is all you need. *Advances in Neural Information Processing Systems*, 2017.

Oriol Vinyals, Charles Blundell, Timothy Lillicrap, Daan Wierstra, et al. Matching networks for one shot learning. *Advances in neural information processing systems*, 29, 2016.

Jane X Wang, Zeb Kurth-Nelson, Dhruva Tirumala, Hubert Soyer, Joel Z Leibo, Remi Munos, Charles Blundell, Dharshan Kumaran, and Matt Botvinick. Learning to reinforcement learn. *arXiv preprint arXiv:1611.05763*, 2016.

Xin Wang, Yudong Chen, and Wenwu Zhu. A survey on curriculum learning. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 44(9):4555–4576, 2021.

Jason Wei, Yi Tay, Rishi Bommasani, Colin Raffel, Barret Zoph, Sebastian Borgeaud, Dani Yogatama, Maarten Bosma, Denny Zhou, Donald Metzler, et al. Emergent abilities of large language models. *arXiv preprint arXiv:2206.07682*, 2022.

Xiaoxia Wu, Ethan Dyer, and Behnam Neyshabur. When do curricula work? In *International Conference on Learning Representations*, 2021. URL https://openreview.net/forum?id=tW4QEInpni.

Sang Michael Xie, Aditi Raghunathan, Percy Liang, and Tengyu Ma. An explanation of in-context learning as implicit bayesian inference. In *International Conference on Learning Representations*, 2022.

Ziqian Zhong, Ziming Liu, Max Tegmark, and Jacob Andreas. The clock and the pizza: Two stories in mechanistic explanation of neural networks, 2023. URL https://arxiv.org/abs/2306.17844.

A Additional results

A.1 Loss curve and total error counts

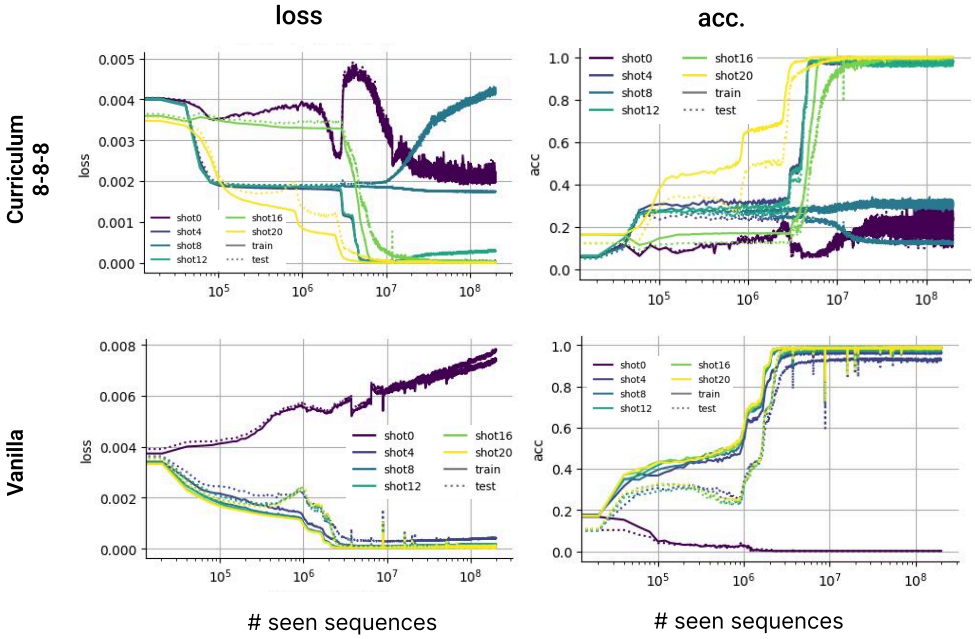

Figure 5: Example loss and performance curve.

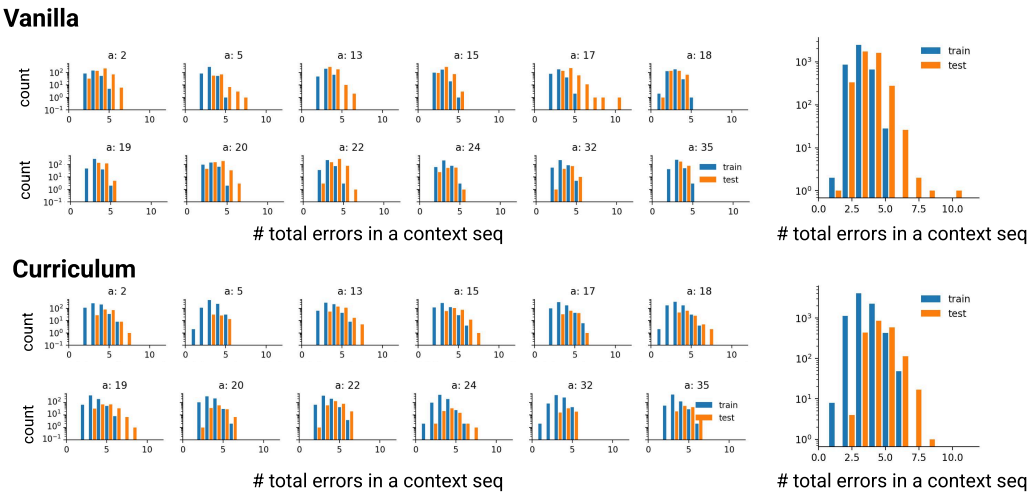

Figure 6: Total error counts histogram for train/test sequences in vanilla and curriculum (8-8-8) trained model (2 seeds). Left: Error counts for sequences of corresponding a. Right: Total errors in all evaluation sequences.

A.2 P=41

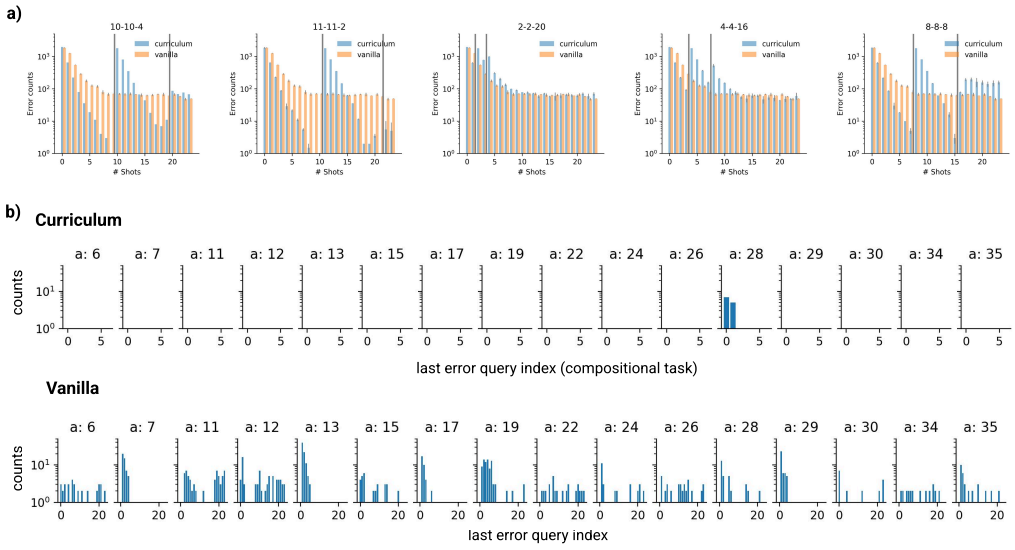

Figure 7: Analysis of robustness on vanilla vs. curriculum model on P=41.**a)** Error counts of after each number of exemplars in the sequence. For P=41, only the shortest curriculum (11-11-2) was better than vanilla. **b)** Distribution of the last error query index for curriculum trained model (11-11-2) and vanilla trained model.

A.3 Linear probe - control baseline

We performed a control experiment with shuffled labels to find the baseline performance. Since the pair of (x, y) are not uniformly distributed in single and double exponential tasks, the baseline performance are different as shown in the following figure of control decoding of y in curriculum.

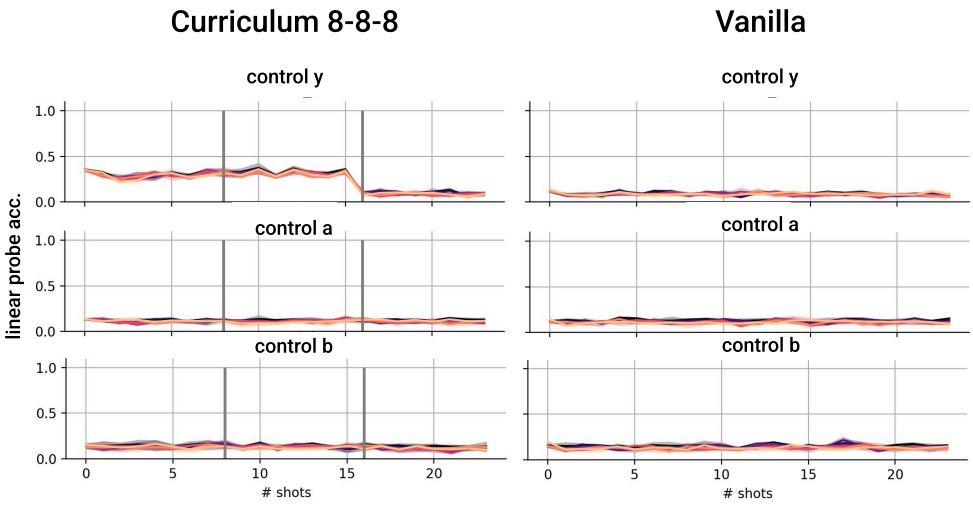

Figure 8: Control linear probe decoding. We used shuffled labels for linear probe training to validate the baseline performance. The baseline performance for decoding of y in single exponential task is slightly higher as (x, y) mapping for single exponential task is not uniformly distributed.

A.4 Linear probe - Varying curriculum length

We report linear probe results from other curriuculum lengths-$(10 - 10 - 4)$ and $(4 - 4 - 16)$. The main findings are aligned with the main figure.

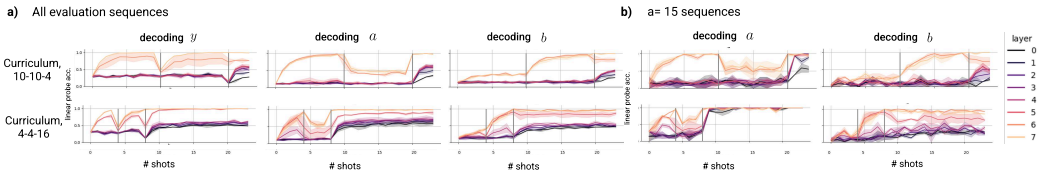

Figure 9: Linear probe decoding results for other curriculum lengths.

A.5 Attention pattern analysis

We visualize the attention pattern from layer 6 and 7 from curriculum trained model and vanilla trained model, averaged on 2K evaluation sequences. In vanilla trained model, the attention pattern is continuous without block structure. On the other hand, curriculum trained model develops atention heads that show strong attention from compositional task block to curriculum block, which is aligns with other results.

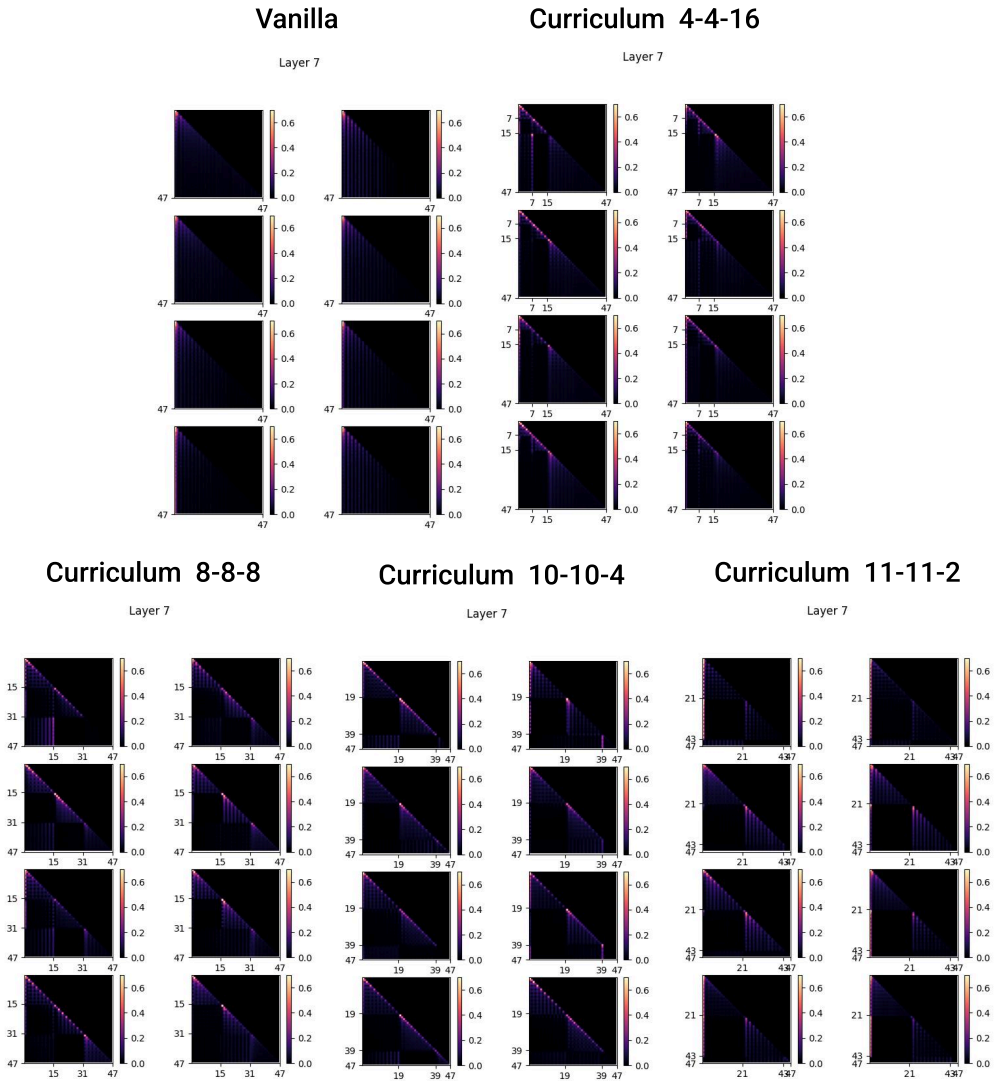

Figure 10: Attention pattern analysis for layer 7 in models trained with different curriculum length.

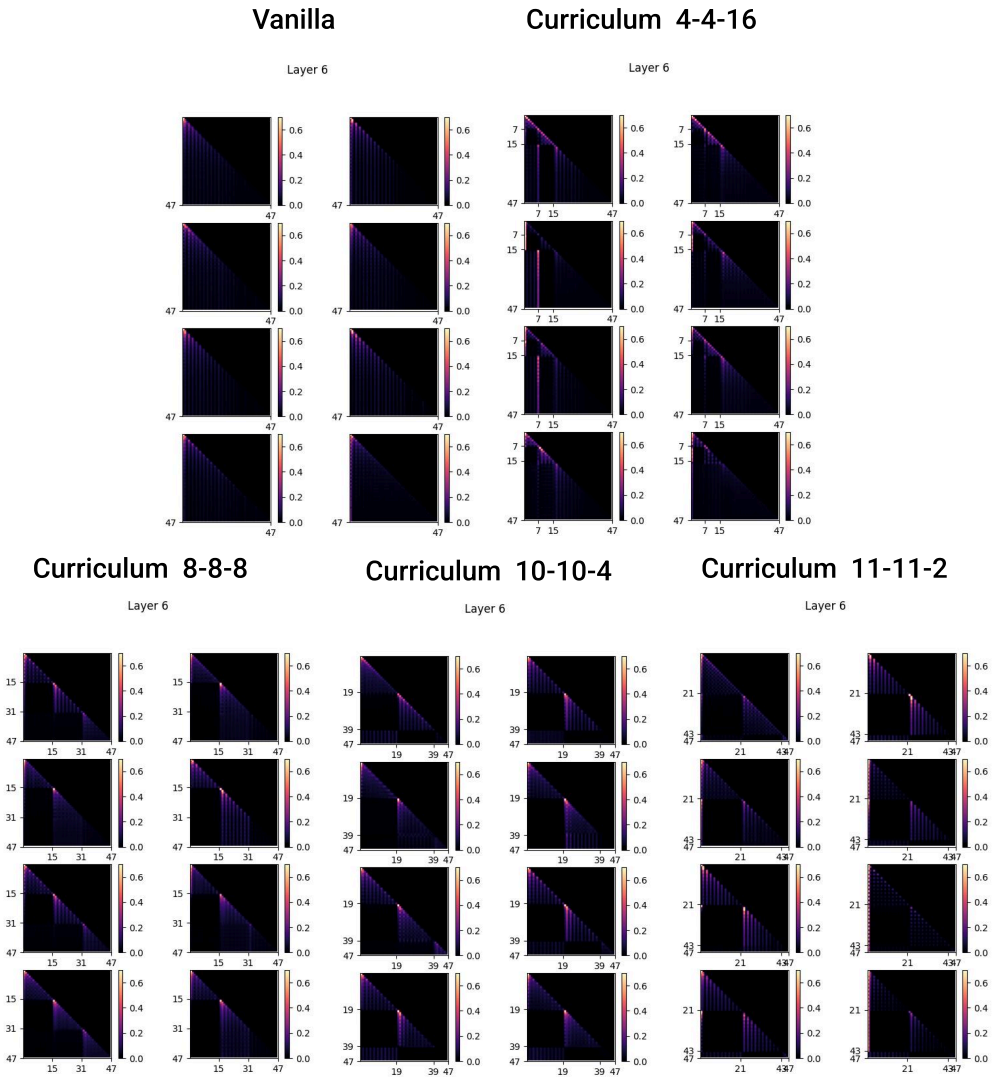

Figure 11: Attention pattern analysis for layer 6 in models trained with different curriculum length.

A.6 Independent learning strategy of compositional task

As an observational analysis to better know that the model initially learns the compositional task independently without utilizing the information from curriculum block, we closely looked at the error pattern through out the training and compare it to the that of the models at the end of vanilla training or curriculum training. We hypothesized that if the model use the information from the curriculum even before it fully learns the task, we could see the error pattern being different from vanilla model.

Specifically, we focus on the 4 checkpoints during the training (indicated with the bars in Figure 12a), which are in the order of 1) the model learned compositional task independently, 2) the model is in the process of learning single exponential tasks, 3) the model learned single exponential tasks and utilizes them for the compositional task and 4) the training is further maintained.

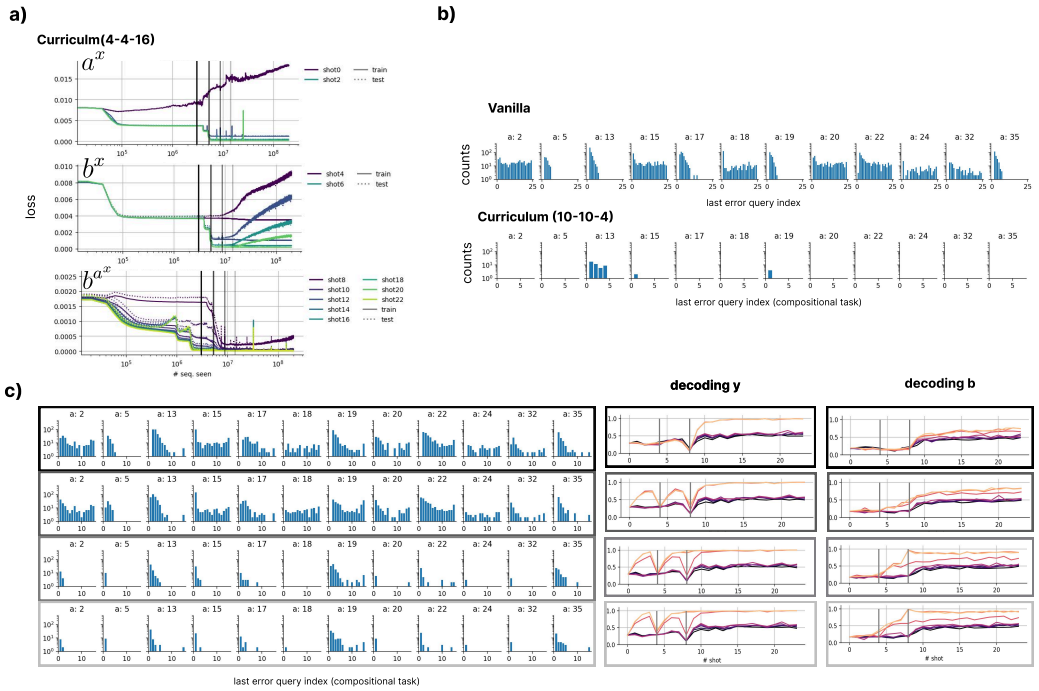

Figure 12: Detailed analysis over training periodm of the model strategy on compositional task. **a)** Loss curve. In panel c we present the error pattern and decoding analysis of the colored bars which indicates different learning phase. **b)** Example error distribution pattern in vanilla and curriculum trained model. **c)** Left: Last error position distribution over training phase. The model's error pattern is very similar to that of vanilla trained model before the model acquires single exponential task. Right: Linear probe decoding on task parameter b. The utilization of task information from the curriculum block happens only after model acquires almost perfect b.