TEST-TIME SCALING MEETS ASSOCIATIVE MEMORY: CHALLENGES IN SUBQUADRATIC MODELS

Hamza Tahir Chaudhry*, Mohit Kulkarni* & Cengiz Pehlevan

School of Engineering & Applied Sciences

Harvard University

Cambridge, MA 02138, USA

{hchaudhry, mkulkarni, cpehlevan}@g.harvard.edu

ABSTRACT

The emerging paradigm of scaling test-time compute–enhancing model performance by scaling up chain of thought reasoning–is gaining significant traction in the deep learning community. While effective, these methods incur substantial computational costs at inference time due to the quadratic memory complexity of Transformers with respect to sequence length. Recently, subquadratic architectures such as Mamba have emerged which approach the performance of Transformers on language tasks while showcasing significant improvements in computational efficiency on long sequences. In this paper, we present the first empirical investigation into test-time compute scaling for subquadratic architectures. Our findings reveal that while these models do benefit from increase test-time compute, their gains are consistently lower than those observed in Transformers. We find that this limitation is correlated with their reduced capabilities for in-context associative memory, which hinder reasoning over extended sequences. These results shed light on the trade-offs between computational efficiency and reasoning capabilities in modern architectures, providing a foundation for future research on designing models for both test-time compute scalability and long-chain reasoning.

1 INTRODUCTION

Just as neural scaling laws for training compute were beginning to plateau, OpenAI's o1 pointed the community to a novel paradigm based on test-time scaling, also referred to as inference scaling Kaplan et al. (2020); Henighan et al. (2020); Hoffmann et al. (2022); OpenAI et al. (2024). DeepSeek's subsequent release of R1 further invigorated research by providing thorough details on the training process of reasoning models DeepSeek-AI et al. (2025). As such, there has been a rapidly growing list of strategies for scaling test-time compute, the vast majority of which hinge on leveraging a model's internal chain of thought to "reason" before providing a final answer Wei et al. (2022). These methods have all been shown to drastically improve the performance of models, especially on tasks such as mathematics and coding with verifiable rewards Shao et al. (2024).

These test-time scaling methods can be categorized broadly into two main branches: sequential and parallel. Sequential scaling refers to increasing the length of a single chain-of-thought, where the model is trained to utilize or independently discovers techniques such as backtracking and self-reflection to improve the quality of the response Kumar et al. (2024); Muennighoff et al. (2025). This can achieved in a number of different ways, most notably through the use of reinforcement fine-tuning in DeepSeek-R1 and budget forcing in s1. Parallel scaling refers to increasing the number of chains of thought, with a fixed or learned strategy to combine answers by searching through completed responses to identify the best response candidates or average to reduce variance in responses Snell et al. (2024); Brown et al. (2024). More advanced methods attempt to combine these methods together by encouraging the model to search over multiple chains of thought simultaneously during generation and allow each chain-of-thought to learn from the others' mistakes/reasoning Xie et al. (2024). Further yet, there exist some additional approaches to test-time scaling which allow the model to "reason" on internal representations without outputting tokens altogether Hao et al.

(2024). These methods have various advantages and disadvantages, and it is an area of active research to determine which methods are best-suited for a given task, model, and compute budget.

However, regardless of the specific test-time scaling approach, it remains the case that these boosts in reasoning capabilities always come at a great computational cost. For instance, OpenAI's o3 model was able to score an unprecedented 87.5% on the famous ARC-AGI benchmark, but this came at the cost of more than $3000 per problem OpenAI (2024); Chollet (2019). This inference cost is directly tied to the quadratic complexity of the Attention mechanism, which computes the relation between every token in a sequence resulting in a memory complexity of $\mathcal{O}(L^2)$ and time complexity of $\mathcal{O}(L^2D)$, compared to feed-forward layers which scale as $\mathcal{O}(LD^2)$, where L and D are the sequence length and embedding dimension respectively. Thus, as we scale to more or longer chains-of-thought, the total amount of compute spent on each new token within each chain scales quadratically.

To circumvent this, we propose the usage of Attention-free architectures such as Mamba or Mamba2, also referred to as subquadratic architectures due to their complexity scaling subquadratically with sequence length Gu & Dao (2023); Dao & Gu (2024). These architectures have been shown to closely match the performance of Transformers on many language tasks, and have been shown to be performant in a slew of other tasks Liu et al. (2025); Ma et al. (2024); Nguyen et al. (2024); Yan et al. (2024). The development and popularization of this architecture has reinvigorated research into recurrent neural networks, resulting in a flurry of new subquadratic architectures Yang et al. (2023); Sun et al. (2023); Peng et al. (2023); Yang et al. (2024c); Beck et al. (2024). Additionally, recent methods have demonstrated that one can effectively distill from pre-trained Transformers into Mamba and other subquadratic architectures using supervised fine-tuning Wang et al. (2024); Bick et al. (2025); Zhang et al. (2024). These distilled models outperform subquadratic models trained from scratch by leveraging the performance and optimization of high-quality models such as Meta's Llama series to develop highly performant subquadratic models with a limited compute and data budget Dubey et al. (2024).

While these architectures can match Transformers on most tasks, recent works have identified a clear performance gap between Mamba and Transformers on in-context learning tasks. In particular, these architectures struggle on language tasks requiring associative memory such as selective copying and multi-query associative recall Arora et al. (2023); Jelassi et al. (2024), where the model must be able to effectively recall information seen only at inference time. This calls back to research directly connecting Attention to the Hopfield Network Hopfield (1982), the canonical associative memory device Ramsauer et al. (2020). Empirical researchers have developed a suite of synthetic tasks that highlight differences between Mamba and Transformers while theoretical researchers have formally proven that transformers are strictly more expressive Arora et al. (2024); Poli et al. (2024). Simply put, the finite size of Mamba's hidden states limits its memory capacity. However, it is still unclear what impact this associative recall gap has on the performance of the model in real-world applications. Furthermore, researchers have found that hybrid models composed of Mamba layers interleaved with Attention layers can alleviate this associative recall gap in language tasks while retaining much of the computational gains Lieber et al. (2024); Ren et al. (2024); Glorioso et al. (2024).

In this work, we provide the first exploration into the efficacy of test-time scaling on these subquadratic architectures. To our surprise, we find that subquadratic architectures systematically underperform Attention-based architectures on mathematical reasoning tasks, something that has not been addressed in the literature. We experiment with hybrid models interleaving Attention with Mamba layers and find that increasing the percentage of Attention in the architecture monotonically improves performance. Through a series of synthetic tasks, we find that this poor performance is closely tied to the models' poor performance on in-context associative recall, highlighting a close relationship between associative memory and effective chain-of-thought reasoning. This suggests that these subquadratic architectures would be poor candidates for reasoning tasks, and highlights the need for subquadratic architectures with better associative memory capabilities in the era of test-time scaling.

2 Test-Time Scaling of Subquadratic Architectures

We evaluate the efficacy of test-time scaling by increasing the number of parallel chains of thought. We put a specific focus on majority voting, also known in the literature as self-consistency, to increase model accuracy by selecting the majority response over N samples thereby reducing variance

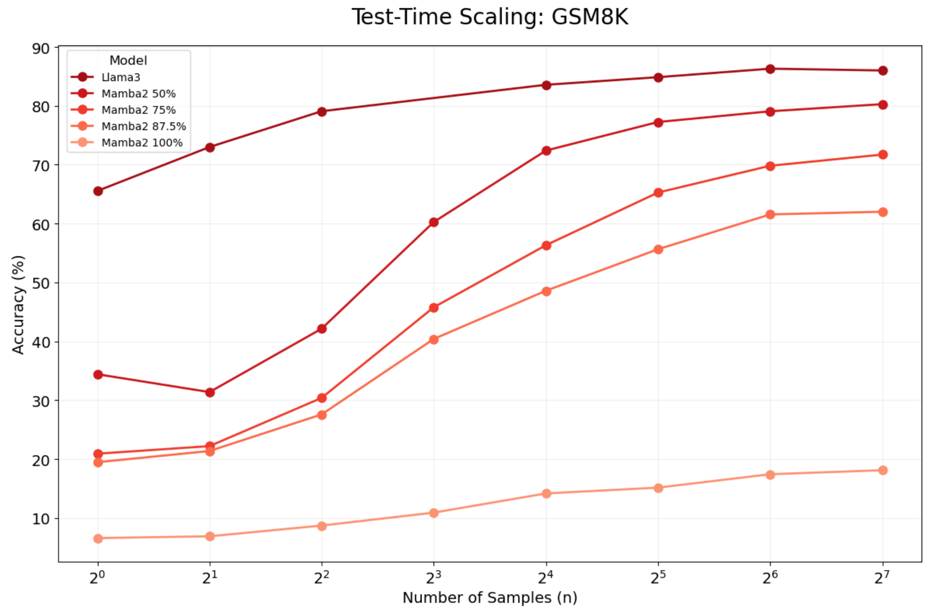

Figure 1: Test-Time Sample Scaling of Llama3, Mamba2, and hybrid models on GSM-8K.

in the responses and masking individual mistakes Wang et al. (2022). We also show results for other parallel test-time scaling methods such as weighted voting using an outcome reward model in Appendix A.2. We examine 8B Mamba / Mamba2 based architectures distilled from Meta's Llama3-8B-Instruct model Dubey et al. (2024). The distillation procedure consists of distilling the weights from the teacher to the student, supervised fine-tuning Kim & Rush (2016), and direct preference optimization Rafailov et al. (2023). This method of distilling architectures has been shown to outperform Mamba / Mamba2 models trained from scratch and closely matches the performance of Llama3-8B-Instruct on language tasks Wang et al. (2024). Furthermore, it enables a fair comparison between architectures based on Mamba2 and Attention. To systematically probe the importance of the Attention mechanism, we also assess 3 hybrid architectures respectively interleaving Mamba2 and Attention layers in the following ratios: 50% / 50%, 75% / 25%, and 87.5% / 12.5%. For a fair comparison, we also compare these models to the performance of Llama3-8B-Instruct which has 100% Attention layers. Results for Mamba models are shown in Appendix A.2.

We assess the performance of these models on standard mathematical reasoning datasets GSM-8K, high-quality grade school math word problems Cobbe et al. (2021). We also show results for the MATH-500 dataset, composed of more difficult competition-level math problems, in Appendix A.2 Hendrycks et al. (2021). We gauge performance as a function of the number of samples (per question). We present our key results in Figure 1, demonstrating a monotonic increase in performance as we increase the amount of Attention layers. We find that the pure Mamba2 architecture performs poorly on the GSM-8K dataset, with its normal prediction performance hovering below 10%. Even as we scale test-time compute by increasing the number of samples, majority voting is only able to improve prediction performance to below 20%. On the other hand, the hybrid architecture with 12.5% outperforms the pure Mamba2 model even with a single sample, with significant improvements as we increase the number of samples up to above 60% at 128 samples. Increasing the ratio of Attention strictly increases performance, with the lower Attention models seeming to saturate in performance at lower accuracy values.

In order to account for the inference efficiency of Mamba2, we also assess the performance of the model as a function of compute. Due to the difficulty of computing FLOPs for the hybrid architectures, we elect to utilize wall-clock time as a surrogate for time complexity. This enables us to assess whether the computational gains of Mamba2 can compensate for any performance deficits, and understand whether its test-time scaling with respect to number of samples might differ from its test-time scaling with respect to compute. We omit the plots for Llama3 as it was implemented with the inference package vLLM, which was not available for Mamba2 and thus prevents a fair comparison Kwon et al. (2023). Note that the generated sequence lengths are limited to 768 tokens

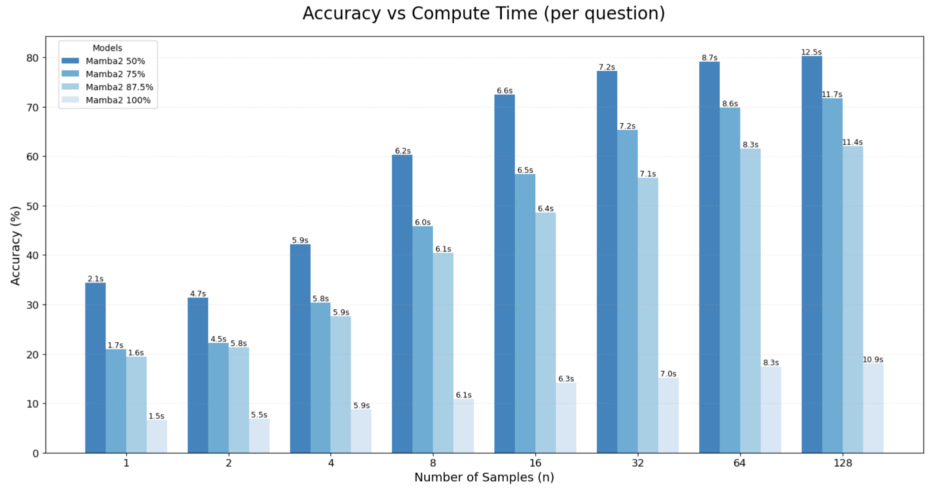

Figure 2: Test-Time Compute Scaling of Mamba2 and hybrid models on GSM-8K.

so for 8B models, the time complexity of the feedforward layers dominates over that of the Attention layers. Therefore, while increasing the ratio of Mamba2 does boost efficiency, the gains would only be relevant at longer context lengths. However, we see the models' performance begin to plateau already at this scale, indicating that Attention-based architectures will scale better with compute.

3 Associative Memory is Important for Mathematical Reasoning

To probe the importance of associative memory in reasoning, we design a set of synthetic mathematical reasoning tasks. In particular, we test the models' performance on two similar tasks, one which explicitly requires associative recall (AR) to solve and the other which does not. For each problem, we generate a set of key-value pairs $\{(K_1, V_1), (K_2, V_2), \ldots, (K_L, V_L)\}$ where each key is a randomly chosen word from the tokenizer and each value is a random integer. We select unique natural numbers $a, b, c \in \{1, \ldots, L\}$ as indices. Then, we ask the model two separate questions:

1. AR Task: Given the list of key-value pairs, return the value of $K_a + K_b + K_c$.
2. Non-AR Task: Given the list of key-value pairs, return the value of $V_a + V_b + V_c$.

The first question requires associative memory to identify the values associated with each key, and then do some basic arithmetic. The latter question only requires the model to do arithmetic. We provide the same list of key-value pairs and indices a, b, c in both questions in order to control for sequence length, and ensure the model is provided the same information in both tasks. This enables us to clearly disambiguate arithmetic / logical mistakes from associative recall mistakes. We test the aforementioned Llama3, Mamba2, and hybrid models on these tasks, assessing the performance of the models as a function of the number of KV-pairs in Table 1 and number of samples in Figure 3. Note that across all models, the model performs worse on the AR task than the non-AR task. Furthermore, pure Mamba2 models seem to perform terribly on both the AR and non-AR tasks.

		Llama3	**Mamba2-50%**	**Mamba2-75%**	**Mamba2-87.5%**	**Mamba2-100%**
10 Pairs	No AR	100.0%	98.0%	82.6%	98.8%	4.2%
	AR	100.0%	97.4%	71.0%	62.0%	2.0%
20 Pairs	No AR	100.0%	99.4%	91.0%	99.8%	2.2%
	AR	99.8%	97.8%	70.8%	33.6%	0.4%
50 Pairs	No AR	100%	99.8%	97.8%	97.8%	2.8%
	AR	99.6%	93.8%	53.8%	11.8%	0.6%

Table 1: Performance comparison of Llama3, Mamba2, and hybrid models on synthetic math tasks (with/without associative recall) with increasing numbers of KV pairs to increase difficulty.

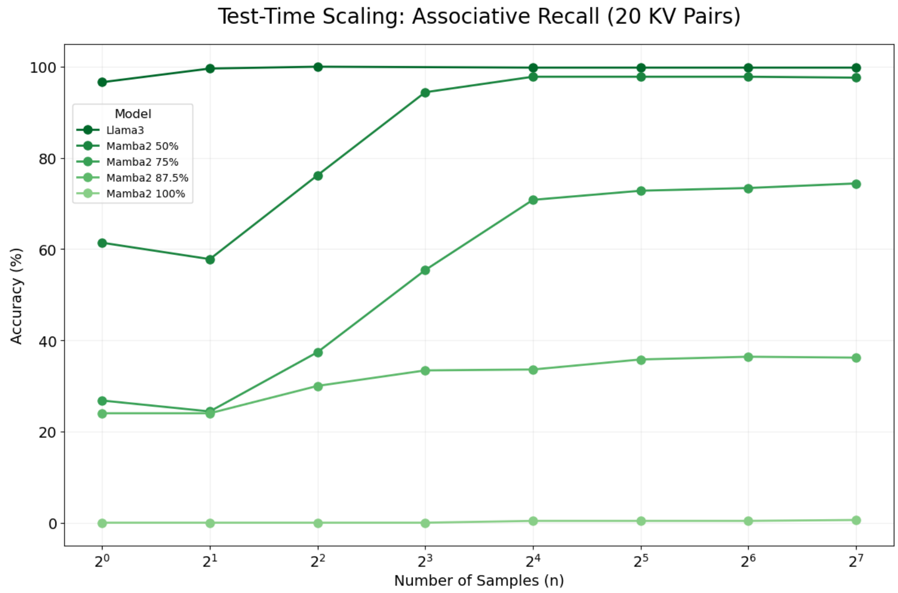

Figure 3: Test-Time Scaling of Llama3, Mamba2, and hybrid models on AR task with 20 KV-pairs.

4 Discussion

To our knowledge, this paper represents the first exploration of test-time scaling strategies applied to subquadratic architectures. Shortly after our submission, another paper appeared exploring test-time scaling in Mamba which we consider concurrent Paliotta et al. (2025). We find that pure Mamba2 models not only have worse baseline performance on mathematical reasoning tasks across the board, they also barely benefit from test-time scaling methods compared to Transformers. However, we found that interleaving Attention layers between Mamba2 layers monotonically improved performance at baseline and improved test-time scaling, with even the 12.5% Attention model showing substantial performance increases. Synthetic mathematics tasks were designed to identify the source of the errors, and we found that Mamba-based models systematically underperformed Attention-based models on mathematical tasks requiring associative recall. We plan to extend this analysis.

We would like to evaluate the models' performance on a more fine-grained suite of mathematical tasks, scaling difficulty via increasing the number and types of operations beyond a simple sum of three numbers. We would also like to explore distilling from more performant reasoning models, similar to how small Llama models distilled from DeepSeek-R1 match the performance of o1-mini. We will also explore alternate reasoning strategies, especially including those based on sequential scaling of chain-of-thought such as budget forcing or reinforcement fine-tuning to encourage the model to produce longer, more structured chains of thought. Furthermore, chain-of-thought reasoning can be generalized to tree-of-thoughts or graph-of-thoughts for better performance on mathematical tasks Yao et al. (2023); Besta et al. (2024). Intuitively, we expect these methods to perform even worse in subquadratic architectures as increasing the sequence length should lead to even lossier memory. Furthermore, there are other reasoning methods that scale up test-time compute internal representations without outputting additional tokens Barrault et al. (2024).

We have begun experimenting with reasoning capabilities of other subquadratic architectures mentioned in the introduction. Recent works have shown that these architectures can be unified under the lens of test-time associative recall Yang et al. (2024b); Wang et al. (2025). Among these architectures, we are interested in exploring architecture which replace the vector or matrix valued hidden states with nonlinear MLPs trained via a test time associative memory loss, as these seem to match the accuracy of Transformers while preserving much of the efficiency of recurrent neural networks Sun et al. (2024); Liu et al. (2024); Behrouz et al. (2024). Through the use of dense associative memory, there might be ways to even exceed the associative memory capabilities Transformers Krotov & Hopfield (2016); Chaudhry et al. (2024). We plan to gain insights from these different models to unpack the relationship between associative memory and reasoning, thereby designing performant architectures with better test-time scaling laws.

REFERENCES

Simran Arora, Sabri Eyuboglu, Aman Timalsina, Isys Johnson, Michael Poli, James Zou, Atri Rudra, and Christopher Ré. Zoology: Measuring and improving recall in efficient language models. *arXiv preprint arXiv:2312.04927*, 2023.

Simran Arora, Sabri Eyuboglu, Michael Zhang, Aman Timalsina, Silas Alberti, Dylan Zinsley, James Zou, Atri Rudra, and Christopher Ré. Simple linear attention language models balance the recall-throughput tradeoff. *arXiv preprint arXiv:2402.18668*, 2024.

Loïc Barrault, Paul-Ambroise Duquenne, Maha Elbayad, Artyom Kozhevnikov, Belen Alastruey, Pierre Andrews, Mariano Coria, Guillaume Couairon, Marta R Costa-jussà, David Dale, et al. Large concept models: Language modeling in a sentence representation space. *arXiv e-prints*, pp. arXiv–2412, 2024.

Maximilian Beck, Korbinian Pöppel, Markus Spanring, Andreas Auer, Oleksandra Prudnikova, Michael Kopp, Günter Klambauer, Johannes Brandstetter, and Sepp Hochreiter. xlstm: Extended long short-term memory. *arXiv preprint arXiv:2405.04517*, 2024.

Edward Beeching, Lewis Tunstall, and Sasha Rush. Scaling test-time compute with open models. URL `https://huggingface.co/spaces/HuggingFaceH4/blogpost-scaling-test-time-compute`.

Ali Behrouz, Peilin Zhong, and Vahab Mirrokni. Titans: Learning to memorize at test time. *arXiv preprint arXiv:2501.00663*, 2024.

Maciej Besta, Nils Blach, Ales Kubicek, Robert Gerstenberger, Michal Podstawski, Lukas Gianinazzi, Joanna Gajda, Tomasz Lehmann, Hubert Niewiadomski, Piotr Nyczyk, et al. Graph of thoughts: Solving elaborate problems with large language models. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 38, pp. 17682–17690, 2024.

Aviv Bick, Kevin Li, Eric Xing, J Zico Kolter, and Albert Gu. Transformers to ssms: Distilling quadratic knowledge to subquadratic models. *Advances in Neural Information Processing Systems*, 37:31788–31812, 2025.

Bradley Brown, Jordan Juravsky, Ryan Ehrlich, Ronald Clark, Quoc V Le, Christopher Ré, and Azalia Mirhoseini. Large language monkeys: Scaling inference compute with repeated sampling. *arXiv preprint arXiv:2407.21787*, 2024.

Hamza Chaudhry, Jacob Zavatone-Veth, Dmitry Krotov, and Cengiz Pehlevan. Long sequence hopfield memory. *Advances in Neural Information Processing Systems*, 36, 2024.

François Chollet. On the measure of intelligence. *arXiv preprint arXiv:1911.01547*, 2019.

Karl Cobbe, Vineet Kosaraju, Mohammad Bavarian, Mark Chen, Heewoo Jun, Lukasz Kaiser, Matthias Plappert, Jerry Tworek, Jacob Hilton, Reiichiro Nakano, et al. Training verifiers to solve math word problems. *arXiv preprint arXiv:2110.14168*, 2021.

Tri Dao and Albert Gu. Transformers are ssms: Generalized models and efficient algorithms through structured state space duality. *arXiv preprint arXiv:2405.21060*, 2024.

DeepSeek-AI, Daya Guo, Dejian Yang, Haowei Zhang, Junxiao Song, Ruoyu Zhang, Runxin Xu, Qihao Zhu, Shirong Ma, Peiyi Wang, Xiao Bi, Xiaokang Zhang, Xingkai Yu, Yu Wu, Z. F. Wu, Zhibin Gou, Zhihong Shao, Zhuoshu Li, Ziyi Gao, Aixin Liu, Bing Xue, Bingxuan Wang, Bochao Wu, Bei Feng, Chengda Lu, Chenggang Zhao, Chengqi Deng, Chenyu Zhang, Chong Ruan, Damai Dai, Deli Chen, Dongjie Ji, Erhang Li, Fangyun Lin, Fucong Dai, Fuli Luo, Guangbo Hao, Guanting Chen, Guowei Li, H. Zhang, Han Bao, Hanwei Xu, Haocheng Wang, Honghui Ding, Huajian Xin, Huazuo Gao, Hui Qu, Hui Li, Jianzhong Guo, Jiashi Li, Jiawei Wang, Jingchang Chen, Jingyang Yuan, Junjie Qiu, Junlong Li, J. L. Cai, Jiaqi Ni, Jian Liang, Jin Chen, Kai Dong, Kai Hu, Kaige Gao, Kang Guan, Kexin Huang, Kuai Yu, Lean Wang, Lecong Zhang, Liang Zhao, Litong Wang, Liyue Zhang, Lei Xu, Leyi Xia, Mingchuan Zhang, Minghua Zhang, Minghui Tang, Meng Li, Miaojun Wang, Mingming Li, Ning Tian, Panpan Huang, Peng Zhang, Qiancheng Wang, Qinyu Chen, Qiushi Du, Ruiqi Ge, Ruisong Zhang, Ruizhe Pan, Runji Wang,

R. J. Chen, R. L. Jin, Ruyi Chen, Shanghao Lu, Shangyan Zhou, Shanhuang Chen, Shengfeng Ye, Shiyu Wang, Shuiping Yu, Shunfeng Zhou, Shuting Pan, S. S. Li, Shuang Zhou, Shaoqing Wu, Shengfeng Ye, Tao Yun, Tian Pei, Tianyu Sun, T. Wang, Wangding Zeng, Wanjia Zhao, Wen Liu, Wenfeng Liang, Wenjun Gao, Wenqin Yu, Wentao Zhang, W. L. Xiao, Wei An, Xiaodong Liu, Xiaohan Wang, Xiaokang Chen, Xiaotao Nie, Xin Cheng, Xin Liu, Xin Xie, Xingchao Liu, Xinyu Yang, Xinyuan Li, Xuecheng Su, Xuheng Lin, X. Q. Li, Xiangyue Jin, Xiaojin Shen, Xiaosha Chen, Xiaowen Sun, Xiaoxiang Wang, Xinnan Song, Xinyi Zhou, Xianzu Wang, Xinxia Shan, Y. K. Li, Y. Q. Wang, Y. X. Wei, Yang Zhang, Yanhong Xu, Yao Li, Yao Zhao, Yaofeng Sun, Yaohui Wang, Yi Yu, Yichao Zhang, Yifan Shi, Yiliang Xiong, Ying He, Yishi Piao, Yisong Wang, Yixuan Tan, Yiyang Ma, Yiyuan Liu, Yongqiang Guo, Yuan Ou, Yuduan Wang, Yue Gong, Yuheng Zou, Yujia He, Yunfan Xiong, Yuxiang Luo, Yuxiang You, Yuxuan Liu, Yuyang Zhou, Y. X. Zhu, Yanhong Xu, Yanping Huang, Yaohui Li, Yi Zheng, Yuchen Zhu, Yunxian Ma, Ying Tang, Yukun Zha, Yuting Yan, Z. Z. Ren, Zehui Ren, Zhangli Sha, Zhe Fu, Zhean Xu, Zhenda Xie, Zhengyan Zhang, Zhewen Hao, Zhicheng Ma, Zhigang Yan, Zhiyu Wu, Zihui Gu, Zijia Zhu, Zijun Liu, Zilin Li, Ziwei Xie, Ziyang Song, Zizheng Pan, Zhen Huang, Zhipeng Xu, Zhongyu Zhang, and Zhen Zhang. Deepseek-r1: Incentivizing reasoning capability in llms via reinforcement learning, 2025. URL `https://arxiv.org/abs/2501.12948`.

Abhimanyu Dubey, Abhinav Jauhri, Abhinav Pandey, Abhishek Kadian, Ahmad Al-Dahle, Aiesha Letman, Akhil Mathur, Alan Schelten, Amy Yang, Angela Fan, et al. The llama 3 herd of models. *arXiv preprint arXiv:2407.21783*, 2024.

Paolo Glorioso, Quentin Anthony, Yury Tokpanov, James Whittington, Jonathan Pilault, Adam Ibrahim, and Beren Millidge. Zamba: A compact 7b ssm hybrid model. *arXiv preprint arXiv:2405.16712*, 2024.

Albert Gu and Tri Dao. Mamba: Linear-time sequence modeling with selective state spaces. *arXiv preprint arXiv:2312.00752*, 2023.

Shibo Hao, Sainbayar Sukhbaatar, DiJia Su, Xian Li, Zhiting Hu, Jason Weston, and Yuandong Tian. Training large language models to reason in a continuous latent space, 2024. URL `https://arxiv.org/abs/2412.06769`.

Dan Hendrycks, Collin Burns, Saurav Kadavath, Akul Arora, Steven Basart, Eric Tang, Dawn Song, and Jacob Steinhardt. Measuring mathematical problem solving with the math dataset. *arXiv preprint arXiv:2103.03874*, 2021.

Tom Henighan, Jared Kaplan, Mor Katz, Mark Chen, Christopher Hesse, Jacob Jackson, Heewoo Jun, Tom B Brown, Prafulla Dhariwal, Scott Gray, et al. Scaling laws for autoregressive generative modeling. *arXiv preprint arXiv:2010.14701*, 2020.

Jordan Hoffmann, Sebastian Borgeaud, Arthur Mensch, Elena Buchatskaya, Trevor Cai, Eliza Rutherford, Diego de Las Casas, Lisa Anne Hendricks, Johannes Welbl, Aidan Clark, et al. Training compute-optimal large language models. *arXiv preprint arXiv:2203.15556*, 2022.

John J Hopfield. Neural networks and physical systems with emergent collective computational abilities. *Proceedings of the national academy of sciences*, 79(8):2554–2558, 1982.

Samy Jelassi, David Brandfonbrener, Sham M Kakade, and Eran Malach. Repeat after me: Transformers are better than state space models at copying. *arXiv preprint arXiv:2402.01032*, 2024.

Jared Kaplan, Sam McCandlish, Tom Henighan, Tom B Brown, Benjamin Chess, Rewon Child, Scott Gray, Alec Radford, Jeffrey Wu, and Dario Amodei. Scaling laws for neural language models. *arXiv preprint arXiv:2001.08361*, 2020.

Yoon Kim and Alexander M Rush. Sequence-level knowledge distillation. *arXiv preprint arXiv:1606.07947*, 2016.

Dmitry Krotov and John J. Hopfield. Dense associative memory for pattern recognition. In D. Lee, M. Sugiyama, U. Luxburg, I. Guyon, and R. Garnett (eds.), *Advances in Neural Information Processing Systems*, volume 29. Curran Associates, Inc., 2016. URL `https://proceedings.neurips.cc/paper/2016/file/eaae339c4d89fc102edd9dbdb6a28915-Paper.pdf`.

Aviral Kumar, Vincent Zhuang, Rishabh Agarwal, Yi Su, John D Co-Reyes, Avi Singh, Kate Baumli, Shariq Iqbal, Colton Bishop, Rebecca Roelofs, Lei M Zhang, Kay McKinney, Disha Shrivastava, Cosmin Paduraru, George Tucker, Doina Precup, Feryal Behbahani, and Aleksandra Faust. Training language models to self-correct via reinforcement learning, 2024. URL https://arxiv.org/abs/2409.12917.

Woosuk Kwon, Zhuohan Li, Siyuan Zhuang, Ying Sheng, Lianmin Zheng, Cody Hao Yu, Joseph E. Gonzalez, Hao Zhang, and Ion Stoica. Efficient memory management for large language model serving with pagedattention. In *Proceedings of the ACM SIGOPS 29th Symposium on Operating Systems Principles*, 2023.

Opher Lieber, Barak Lenz, Hofit Bata, Gal Cohen, Jhonathan Osin, Itay Dalmedigos, Erez Safahi, Shaked Meirom, Yonatan Belinkov, Shai Shalev-Shwartz, et al. Jamba: A hybrid transformer-mamba language model. *arXiv preprint arXiv:2403.19887*, 2024.

Bo Liu, Rui Wang, Lemeng Wu, Yihao Feng, Peter Stone, and Qiang Liu. Longhorn: State space models are amortized online learners. *arXiv preprint arXiv:2407.14207*, 2024.

Yue Liu, Yunjie Tian, Yuzhong Zhao, Hongtian Yu, Lingxi Xie, Yaowei Wang, Qixiang Ye, Jianbin Jiao, and Yunfan Liu. Vmamba: Visual state space model. *Advances in neural information processing systems*, 37:103031–103063, 2025.

Jun Ma, Feifei Li, and Bo Wang. U-mamba: Enhancing long-range dependency for biomedical image segmentation. *arXiv preprint arXiv:2401.04722*, 2024.

Niklas Muennighoff, Zitong Yang, Weijia Shi, Xiang Lisa Li, Li Fei-Fei, Hannaneh Hajishirzi, Luke Zettlemoyer, Percy Liang, Emmanuel Candès, and Tatsunori Hashimoto. s1: Simple test-time scaling, 2025. URL https://arxiv.org/abs/2501.19393.

Eric Nguyen, Michael Poli, Matthew G Durrant, Brian Kang, Dhruva Katrekar, David B Li, Liam J Bartie, Armin W Thomas, Samuel H King, Garyk Brixi, et al. Sequence modeling and design from molecular to genome scale with evo. *Science*, 386(6723):eado9336, 2024.

OpenAI. o3 model family technical report. Technical report, OpenAI, 2024. URL https://openai.com/index/openai-o3-mini/. Retrieved from OpenAI's official model documentation.

OpenAI, :, Aaron Jaech, Adam Kalai, Adam Lerer, Adam Richardson, Ahmed El-Kishky, Aiden Low, Alec Helyar, Aleksander Madry, Alex Beutel, Alex Carney, Alex Iftimie, Alex Karpenko, Alex Tachard Passos, Alexander Neitz, Alexander Prokofiev, Alexander Wei, Allison Tam, Ally Bennett, Ananya Kumar, Andre Saraiva, Andrea Vallone, Andrew Duberstein, Andrew Kondrich, Andrey Mishchenko, Andy Applebaum, Angela Jiang, Ashvin Nair, Barret Zoph, Behrooz Ghorbani, Ben Rossen, Benjamin Sokolowsky, Boaz Barak, Bob McGrew, Borys Minaiev, Botao Hao, Bowen Baker, Brandon Houghton, Brandon McKinzie, Brydon Eastman, Camillo Lugaresi, Cary Bassin, Cary Hudson, Chak Ming Li, Charles de Bourcy, Chelsea Voss, Chen Shen, Chong Zhang, Chris Koch, Chris Orsinger, Christopher Hesse, Claudia Fischer, Clive Chan, Dan Roberts, Daniel Kappler, Daniel Levy, Daniel Selsam, David Dohan, David Farhi, David Mely, David Robinson, Dimitris Tsipras, Doug Li, Dragos Oprica, Eben Freeman, Eddie Zhang, Edmund Wong, Elizabeth Proehl, Enoch Cheung, Eric Mitchell, Eric Wallace, Erik Ritter, Evan Mays, Fan Wang, Felipe Petroski Such, Filippo Raso, Florencia Leoni, Foivos Tsimpourlas, Francis Song, Fred von Lohmann, Freddie Sulit, Geoff Salmon, Giambattista Parascandolo, Gildas Chabot, Grace Zhao, Greg Brockman, Guillaume Leclerc, Hadi Salman, Haiming Bao, Hao Sheng, Hart Andrin, Hessam Bagherinezhad, Hongyu Ren, Hunter Lightman, Hyung Won Chung, Ian Kivlichan, Ian O'Connell, Ian Osband, Ignasi Clavera Gilaberte, Ilge Akkaya, Ilya Kostrikov, Ilya Sutskever, Irina Kofman, Jakub Pachocki, James Lennon, Jason Wei, Jean Harb, Jerry Twore, Jiacheng Feng, Jiahui Yu, Jiayi Weng, Jie Tang, Jieqi Yu, Joaquin Quiñonero Candela, Joe Palermo, Joel Parish, Johannes Heidecke, John Hallman, John Rizzo, Jonathan Gordon, Jonathan Uesato, Jonathan Ward, Joost Huizinga, Julie Wang, Kai Chen, Kai Xiao, Karan Singhal, Karina Nguyen, Karl Cobbe, Katy Shi, Kayla Wood, Kendra Rimbach, Keren Gu-Lemberg, Kevin Liu, Kevin Lu, Kevin Stone, Kevin Yu, Lama Ahmad, Lauren Yang, Leo Liu, Leon Maksin, Leyton Ho, Liam Fedus, Lilian Weng, Linden Li, Lindsay McCallum, Lindsey Held, Lorenz Kuhn, Lukas Kondraciuk, Lukasz Kaiser, Luke Metz, Madelaine Boyd, Maja Trebacz, Manas Joglekar, Mark Chen, Marko Tintor, Mason Meyer, Matt Jones, Matt Kaufer, Max Schwarzer, Meghan Shah, Mehmet Yatbaz, Melody Y. Guan, Mengyuan Xu, Mengyuan Yan, Mia Glaese, Mianna Chen, Michael Lampe, Michael Malek, Michele Wang, Michelle Fradin, Mike McClay, Mikhail Pavlov, Miles Wang, Mingxuan Wang, Mira Murati, Mo Bavarian, Mostafa Rohaninejad, Nat McAleese, Neil Chowdhury, Neil Chowdhury, Nick Ryder, Nikolas Tezak, Noam Brown, Ofir Nachum, Oleg Boiko, Oleg Murk, Olivia Watkins, Patrick Chao, Paul Ashbourne, Pavel Izmailov, Peter Zhokhov, Rachel Dias, Rahul Arora, Randall Lin, Rapha Gontijo Lopes, Raz Gaon, Reah Miyara, Reimar Leike, Renny Hwang, Rhythm Garg, Robin Brown, Roshan James, Rui Shu, Ryan Cheu, Ryan Greene, Saachi Jain, Sam Altman, Sam Toizer, Sam Toyer, Samuel Miserendino, Sandhini Agarwal, Santiago Hernandez, Sasha Baker, Scott McKinney, Scottie Yan, Shengjia Zhao, Shengli Hu, Shibani Santurkar, Shraman Ray Chaudhuri, Shuyuan Zhang, Siyuan Fu, Spencer Papay, Steph Lin, Suchir Balaji, Suvansh Sanjeev, Szymon Sidor, Tal Broda, Aidan Clark, Tao Wang, Taylor Gordon, Ted Sanders, Tejal Patwardhan, Thibault Sottiaux, Thomas Degry, Thomas Dimson, Tianhao Zheng, Timur Garipov, Tom Stasi, Trapit Bansal, Trevor Creech, Troy Peterson, Tyna Eloundou, Valerie Qi, Vineet Kosaraju, Vinnie Monaco, Vitchyr Pong, Vlad Fomenko, Weiyi Zheng, Wenda Zhou, Wes McCabe, Wojciech Zaremba, Yann Dubois, Yinghai Lu, Yining Chen, Young Cha, Yu Bai, Yuchen He, Yuchen Zhang, Yunyun Wang, Zheng Shao, and Zhuohan Li. Openai o1 system card, 2024. URL `https://arxiv.org/abs/2412.16720`.

Daniele Paliotta, Junxiong Wang, Matteo Pagliardini, Kevin Y Li, Aviv Bick, J Zico Kolter, Albert Gu, François Fleuret, and Tri Dao. Thinking slow, fast: Scaling inference compute with distilled reasoners. *arXiv preprint arXiv:2502.20339*, 2025.

Bo Peng, Eric Alcaide, Quentin Anthony, Alon Albalak, Samuel Arcadinho, Stella Biderman, Huanqi Cao, Xin Cheng, Michael Chung, Matteo Grella, et al. Rwkv: Reinventing rnns for the transformer era. *arXiv preprint arXiv:2305.13048*, 2023.

Michael Poli, Armin W Thomas, Eric Nguyen, Pragaash Ponnusamy, Björn Deiseroth, Kristian Kersting, Taiji Suzuki, Brian Hie, Stefano Ermon, Christopher Ré, et al. Mechanistic design and scaling of hybrid architectures. *arXiv preprint arXiv:2403.17844*, 2024.

Rafael Rafailov, Archit Sharma, Eric Mitchell, Christopher D Manning, Stefano Ermon, and Chelsea Finn. Direct preference optimization: Your language model is secretly a reward model. *Advances in Neural Information Processing Systems*, 36:53728–53741, 2023.

Hubert Ramsauer, Bernhard Schäfl, Johannes Lehner, Philipp Seidl, Michael Widrich, Thomas Adler, Lukas Gruber, Markus Holzleitner, Milena Pavlović, Geir Kjetil Sandve, et al. Hopfield networks is all you need. *arXiv preprint arXiv:2008.02217*, 2020.

Liliang Ren, Yang Liu, Yadong Lu, Yelong Shen, Chen Liang, and Weizhu Chen. Samba: Simple hybrid state space models for efficient unlimited context language modeling. *arXiv preprint arXiv:2406.07522*, 2024.

Zhihong Shao, Peiyi Wang, Qihao Zhu, Runxin Xu, Junxiao Song, Xiao Bi, Haowei Zhang, Mingchuan Zhang, Y. K. Li, Y. Wu, and Daya Guo. Deepseekmath: Pushing the limits of mathematical reasoning in open language models, 2024. URL `https://arxiv.org/abs/2402.03300`.

Charlie Snell, Jaehoon Lee, Kelvin Xu, and Aviral Kumar. Scaling llm test-time compute optimally can be more effective than scaling model parameters, 2024. URL `https://arxiv.org/abs/2408.03314`.

Yu Sun, Xinhao Li, Karan Dalal, Jiarui Xu, Arjun Vikram, Genghan Zhang, Yann Dubois, Xinlei Chen, Xiaolong Wang, Sanmi Koyejo, Tatsunori Hashimoto, and Carlos Guestrin. Learning to (learn at test time): Rnns with expressive hidden states. 2024. URL `https://arxiv.org/abs/2407.04620`.

Yutao Sun, Li Dong, Shaohan Huang, Shuming Ma, Yuqing Xia, Jilong Xue, Jianyong Wang, and Furu Wei. Retentive network: A successor to transformer for large language models. *arXiv preprint arXiv:2307.08621*, 2023.

Junxiong Wang, Daniele Paliotta, Avner May, Alexander Rush, and Tri Dao. The mamba in the llama: Distilling and accelerating hybrid models. In A. Globerson, L. Mackey, D. Belgrave, A. Fan, U. Paquet, J. Tomczak, and C. Zhang (eds.), *Advances in Neural Information Processing Systems*, volume 37, pp. 62432–62457. Curran Associates, Inc., 2024. URL https://proceedings.neurips.cc/paper_files/paper/2024/file/723933067ad315269b620bc0d2c05cba-Paper-Conference.pdf.

Ke Alexander Wang, Jiaxin Shi, and Emily B Fox. Test-time regression: a unifying framework for designing sequence models with associative memory. *arXiv preprint arXiv:2501.12352*, 2025.

Xuezhi Wang, Jason Wei, Dale Schuurmans, Quoc Le, Ed Chi, Sharan Narang, Aakanksha Chowdhery, and Denny Zhou. Self-consistency improves chain of thought reasoning in language models. *arXiv preprint arXiv:2203.11171*, 2022.

Jason Wei, Xuezhi Wang, Dale Schuurmans, Maarten Bosma, Fei Xia, Ed Chi, Quoc V Le, Denny Zhou, et al. Chain-of-thought prompting elicits reasoning in large language models. *Advances in neural information processing systems*, 35:24824–24837, 2022.

Yuxi Xie, Anirudh Goyal, Wenyue Zheng, Min-Yen Kan, Timothy P. Lillicrap, Kenji Kawaguchi, and Michael Shieh. Monte carlo tree search boosts reasoning via iterative preference learning, 2024. URL https://arxiv.org/abs/2405.00451.

Jing Nathan Yan, Jiatao Gu, and Alexander M Rush. Diffusion models without attention. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 8239–8249, 2024.

An Yang, Beichen Zhang, Binyuan Hui, Bofei Gao, Bowen Yu, Chengpeng Li, Dayiheng Liu, Jianhong Tu, Jingren Zhou, Junyang Lin, et al. Qwen2. 5-math technical report: Toward mathematical expert model via self-improvement. *arXiv preprint arXiv:2409.12122*, 2024a.

Songlin Yang, Bailin Wang, Yikang Shen, Rameswar Panda, and Yoon Kim. Gated linear attention transformers with hardware-efficient training. *arXiv preprint arXiv:2312.06635*, 2023.

Songlin Yang, Jan Kautz, and Ali Hatamizadeh. Gated delta networks: Improving mamba2 with delta rule. *arXiv preprint arXiv:2412.06464*, 2024b.

Songlin Yang, Bailin Wang, Yu Zhang, Yikang Shen, and Yoon Kim. Parallelizing linear transformers with the delta rule over sequence length. *arXiv preprint arXiv:2406.06484*, 2024c.

Shunyu Yao, Dian Yu, Jeffrey Zhao, Izhak Shafran, Tom Griffiths, Yuan Cao, and Karthik Narasimhan. Tree of thoughts: Deliberate problem solving with large language models. *Advances in neural information processing systems*, 36:11809–11822, 2023.

Michael Zhang, Simran Arora, Rahul Chalamala, Alan Wu, Benjamin Spector, Aaryan Singhal, Krithik Ramesh, and Christopher Ré. Lolcats: On low-rank linearizing of large language models. *arXiv preprint arXiv:2410.10254*, 2024.

A Appendix

A.1 Experimental Details

We adapt the Huggingface Search and Learn repository for our scaling methods. Beeching et al.. We also tried a weighted best-of-n scaling method using a Process Reward Model Shao et al. (2024) but find it scales similar to majority voting. We speculate that this is because the verifier tends to filter out incorrect answers, which are typically more diverse, while correct answers naturally cluster together.

We adapt the Qwen2.5-Math repository Yang et al. (2024a) to verify correct answers. This is done by specifically prompting the language model to reason step by step and answer in a \boxed{} format.

We use a cluster of H100 GPUs with 80GB of memory. All of these experiments can run on a single H100 GPU.

A.2 Additional Experiments

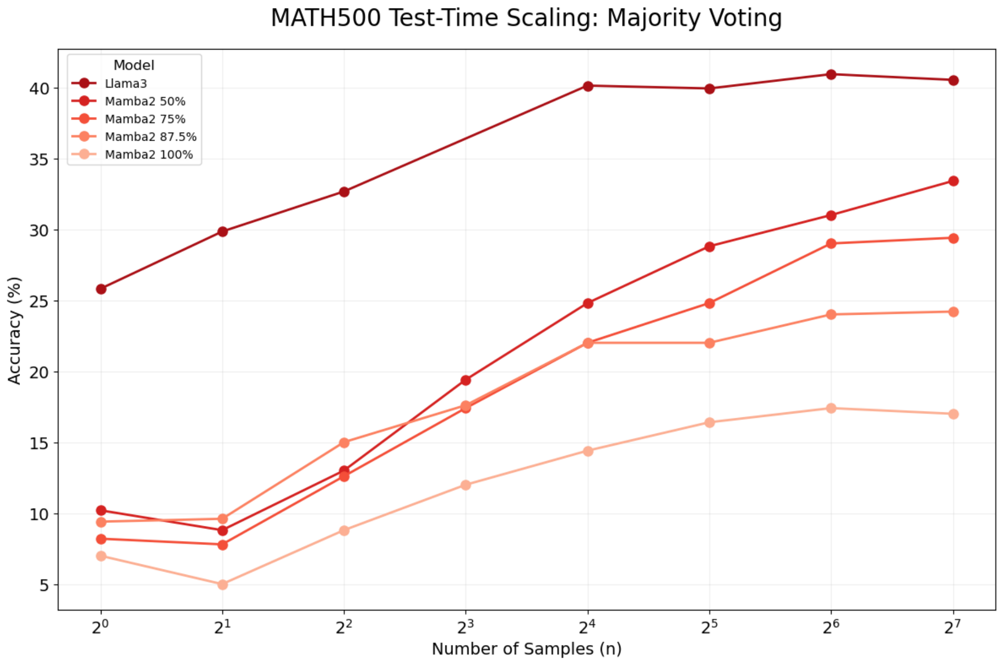

Figure 4: Test-Time Sample Scaling of Llama3, Mamba2, and hybrid models on MATH500.

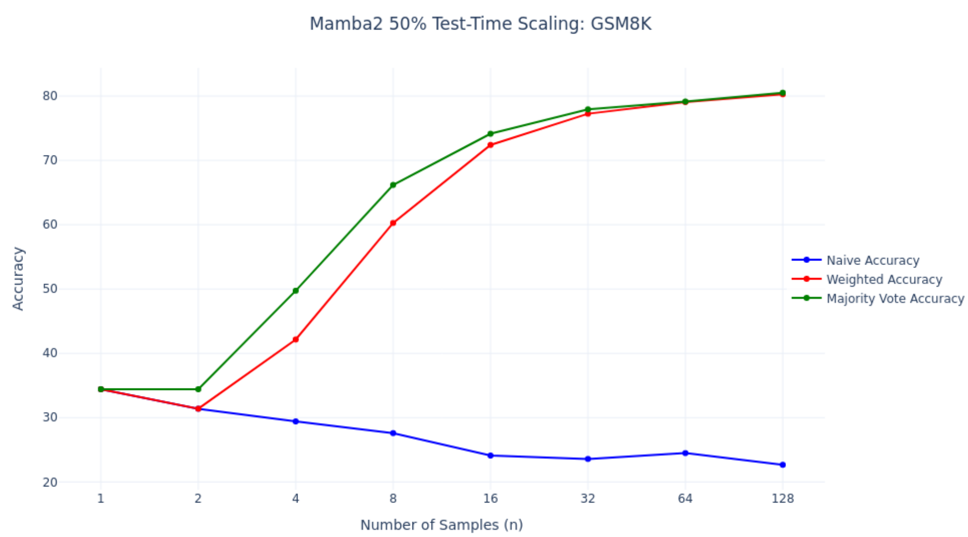

Figure 5: Test-Time Sample Scaling of Mamba2-50% on GSM8K with different strategies. Using a Outcome Reward Model based on Llama3, we score the outputs and retrieve a weighted average in Weighted Voting or return the answer with the maximum score in Naive Accuracy

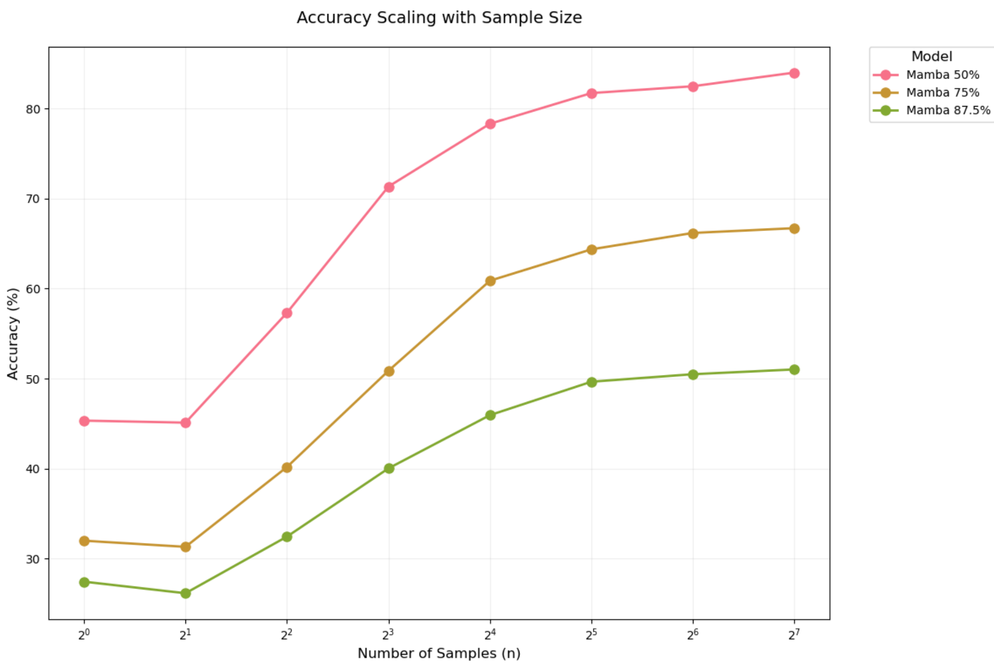

Figure 6: Test-Time Sample Scaling of Mamba hybrid models on GSM8K.