Continuous Self-Improvement of Large Language Models by Test-time Training with Verifier-Driven Sample Selection

Mohammad Mahdi Moradi
Department of Computer Science, Concordia University
Ascend Team, Huawei Technologies
mohammad.mahdi.moradi@h-partners.com

Hossam Amer
Ascend Team
Toronto Research Center
Huawei Technologies

Sudhir Mudur
Department of Computer Science
Concordia University
mudur@cs.concordia.ca

Weiwei Zhang
Ascend Team
Toronto Research Center
Huawei Technologies

Yang Liu
Ascend Team
Toronto Research Center
Huawei Technologies

Walid Ahmed
Ascend Team
Toronto Research Center
Huawei Technologies

Abstract

Adapting pretrained LLMs to unlabeled, out-of-distribution data remains challenging, especially for structurally novel reasoning tasks. We present VDS-TTT (Verifier-Driven Sample Selection for Test-Time Training), a self-supervised framework that uses a learned verifier to score multiple generated responses and select only high-confidence pseudo-labeled examples for on-the-fly adaptation. For each query, the LLM generates N answers; the verifier picks the most reliable one above a confidence threshold, paired with its query for fine-tuning. We update only low-rank LoRA adapters, enabling efficient and fast adaptation. Across three benchmarks and three state-of-the-art LLMs, VDS-TTT achieves up to 32.29% relative improvement over the base model, showing its effectiveness for continuous test-time self-improvement.

1 Introduction

Large language models (LLMs) are typically trained once and kept fixed at inference, which limits adaptability under distribution shift—i.e., when the test data distribution differs from the training data (Xiao and Snoek [2024]). Test-time training (TTT) mitigates this by updating model parameters on-the-fly for each test instance, enabling transductive adaptation (Bottou and Vapnik [1992], Cleveland [1979], Cleveland and Devlin [1988]). Unlike continual pretraining (Wu et al. [2024]), which incrementally updates models on large corpora, TTT adapts efficiently using unsupervised objectives and no labeled data.

We propose Verifier-Driven Sample Selection for TTT (VDS-TTT), Figure 1, where a verifier, an external model that assesses the reliability of candidate outputs, selects high-confidence pseudo-labels to fine-tune LoRA adapters. This targeted adaptation mitigates noise and forgetting, leading to robust performance in label-scarce settings and outperforming prior TTT baselines. Our main contributions are as follows:

Submitted to 39th Conference on Neural Information Processing Systems (NeurIPS 2025). Do not distribute.

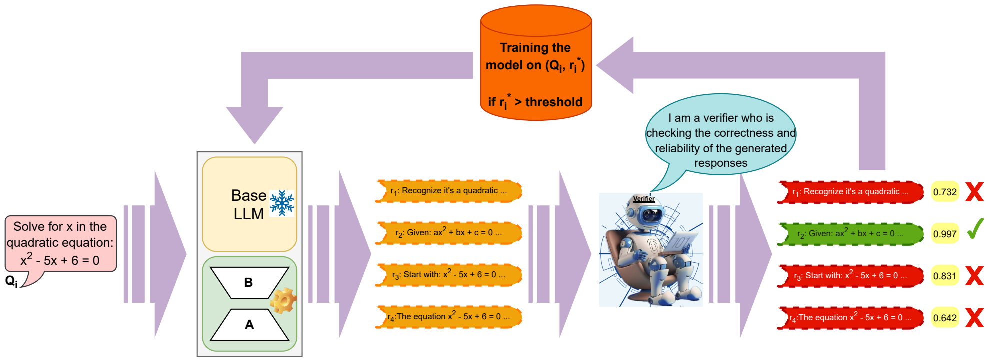

Training the model on (Q_i, r_i^*)
if r_i^* > threshold
I am a verifier who is checking the correctness and reliability of the generated responses
Solve for x in the quadratic equation: $x^2 - 5x + 6 = 0$
Q_i
Base LLM
B
A
r_1: Recognize it's a quadratic ...
r_2: Given: $ax^2 + bx + c = 0$ ...
r_3: Start with: $x^2 - 5x + 6 = 0$ ...
r_4:The equation $x^2 - 5x + 6 = 0$ ...
Verifier
r_1: Recognize it's a quadratic ...
0.732 X
r_2: Given: $ax^2 + bx + c = 0$ ...
0.997 ✓
r_3: Start with: $x^2 - 5x + 6 = 0$ ...
0.831 X
r_4:The equation $x^2 - 5x + 6 = 0$ ...
0.642 X

Figure 1: Our proposed VDS-TTT framework

- We propose a verifier-guided framework for test-time training that adapts LoRA parameters using high-confidence pseudo-labels, enabling efficient and stable self-improvement under distribution shift without requiring human-provided labels, It consistently outperforms the baseline across multiple reasoning benchmarks.
- We show that performance improves steadily with additional test-time training iterations, often matching or exceeding an oracle verifier after only a few iterations.

2 Related Work

A key challenge in TTT is the absence of ground-truth, preventing direct supervised training. Prior works have developed self-supervised or unsupervised signals for adaptation (Bartler et al. [2022]). In language modeling, approaches to inferring missing labels include: **(1) Extra-memory retrieval:** Retrieval-based TTT (Hardt and Sun [2023]) fine-tunes on semantically similar contexts but requires large indices (810GB–2.1TB) and may retrieve redundant samples. SIFT (Hübotter et al. [2024]) reduces redundancy via active selection, but still depends on memory storage. **(2) Internal/external feedback:** Tent (Wang et al. [2020]) minimizes prediction entropy as an uncertainty proxy, though prone to model collapse without regularization (Press et al. [2024]). **(3) RL-based adaptation:** Reward signals from human feedback or external functions are used in methods such as ReST (Gulcehre et al. [2023]) and ReST-MCTS (Zhang et al. [2024]), but these often require labeled data, incur high computational cost, or depend on full reasoning traces. Recent work, TTRL, (Zuo et al. [2025]) applies PPO (Schulman et al. [2017]) and GRPO (Shao et al. [2024]) at test time with majority-vote pseudo-labels from 64 rollouts, but its performance remains sensitive to problem difficulty and the pretrained distribution.

To address these challenges, we propose VDS-TTT, a continuous self-improvement paradigm that: (1) requires no labeled data; (2) is memory- and compute-efficient without expensive rollouts; (3) is simple to implement and stable due to low-variance gradients (Mukobi et al. [2023]); and (4) adapts only LoRA parameters, enabling on-the-fly specialization while improving base model performance.

3 Methodology

VDS-TTT (Algorithm 1) is a fully test-time training framework that adapts LLMs dynamically to distributional shifts using verifier-guided pseudo-labels. For each input Q_i, the pretrained LLM generates N diverse candidate responses $\{r_1, \dots, r_N\}$ via temperature sampling to encourage exploration (Renze [2024]).

A verifier then assigns confidence scores to each candidate. If all scores fall below a threshold τ, the query is discarded; otherwise, the highest-scoring response r^* is paired with Q_i as a pseudo-labeled example. Unlike verifier-free methods, which struggle under heterogeneous solution distributions (Setlur et al. [2025]), this verifier-based filtering ensures high-quality data. The two-step strategy of Best-of-N selection followed by thresholding guarantees that only diverse yet reliable solutions are retained for training.

During test time, the model is fine-tuned on these verifier-approved pairs by minimizing the Supervised Fine-Tuning (SFT) loss function where only LoRA adapter parameters Δ are updated (Hu et al. [2022]) while the base model remains frozen. This lightweight adaptation avoids catastrophic forgetting, converges quickly, and enables continuous self-improvement under domain shifts.

Algorithm 1 Verifier-Driven sample Selection for Test-Time Training (VDS-TTT)

```
Require: Pretrained LLM $f_{\theta_0}$, verifier score function $s(\cdot,\cdot)$, temperature $T$, number of samples $N$,
    score threshold $\tau$, LoRA adapter steps $M$, learning rate $\eta$
Ensure: Adapted adapter parameters $\Delta$
Initialize adapter $\Delta \leftarrow \mathbf{0}$
for all test query $q_i$ do                         ▷ Stage 1: Candidate Generation for Self-Annotation
    $\mathcal{R} \leftarrow \emptyset$
    for $j = 1$ to $N$ do
        Sample response $r_{ij} \sim f_{\theta_0}\big(\cdot \mid q_i; T\big)$        ▷ temperature sampling
        Extract final answer $a_{ij}$ from $r_{ij}$
        $\mathcal{R} \leftarrow \mathcal{R} \cup \{(r_{ij}, a_{ij})\}$
    end for
                                                    ▷ Stage 2: Confidence-Guided High-Quality Annotation
    Compute scores $s_{ij} \leftarrow s\big(r_{ij}, a_{ij}\big)$ for each $(r_{ij}, a_{ij}) \in \mathcal{R}$
    Let $j^* = \arg\max_{1 \le j \le N} s_{ij}$
    if $s_{ij^*} < \tau$ then
        continue                                    ▷ skip low-confidence query
    end if
    Set pseudo-label $(r_i, a_i) \leftarrow (r_{ij^*}, a_{ij^*})$
    Record score $s_i \leftarrow s_{ij^*}$
                                                    ▷ Stage 3: Test-Time Training (LoRA Adaptation)
    for $m = 1$ to $M$ do
        Compute loss
            $$\mathcal{L}(\Delta) = -\sum_{t=1}^{|r_i|} \log\, f_{\theta_0+\Delta}\big(r_{i,t} \mid q_i, r_{i,<t}\big)$$
        $\Delta \leftarrow \Delta - \eta\, \nabla_\Delta\, \mathcal{L}(\Delta)$
    end for
end for
return $\Delta$
```

4 Experiments

Experimental Setup. We evaluate VDS-TTT on four LLMs—Llama-3.2-1B-Instruct, DeepSeek-R1-Distill-Qwen-1.5B, Llama-3.2-3B-Instruct, and LLaMA-3.1-8B-Instruct—across GSM8K Cobbe et al. [2021], Math-500 Hendrycks et al. [2021], and AIME1983-2024. To reduce computational overhead, we employ a lightweight verifier, Skywork-o1-Open-PRM-Qwen-2.5-1.5B. All experiments are conducted on an NVIDIA Tesla V100 GPU (32GB). On Math-500 with LLaMA-3.2-1B-Instruct, runtimes were approximately 1h30m, 2h10m, 3h06m, and 4h30m for $N = 2, 4, 8, 16$, respectively.

Implementation Details. In our experiments, we systematically vary the number of candidate responses $N \in \{2, 4, 8, 16\}$ to evaluate its impact on Best-of-N selection performance. We adopt a stringent verifier confidence threshold $\tau = 0.99$ for GSM8K and Math-500—mirroring pseudo-labeling best practices that apply high thresholds to filter out noisy labels, but relax τ to 0.9 for the more challenging AIME1983-2024 to prevent discarding the vast majority of low-scoring samples if $\tau = 0.99$. Furthermore, we empirically concluded that updating only low-rank LoRA adapter parameters substantially outperforms full fine-tuning of deeper or randomly selected layers. Consequently, we integrate LoRA modules into all key projection components—q_proj, k_proj, v_proj, o_proj—as well as the MLP sublayers (mlp_gate_proj, mlp_up_proj, mlp_down_proj) of our base LLM. We set the LoRA rank to 128 in most cases where sufficient data is available for test-time training. For low-resource scenarios, such as AIME1983-2024, we reduce the LoRA rank to 8 to prevent overfitting and maintain training stability.

4.1 Results and Discussion

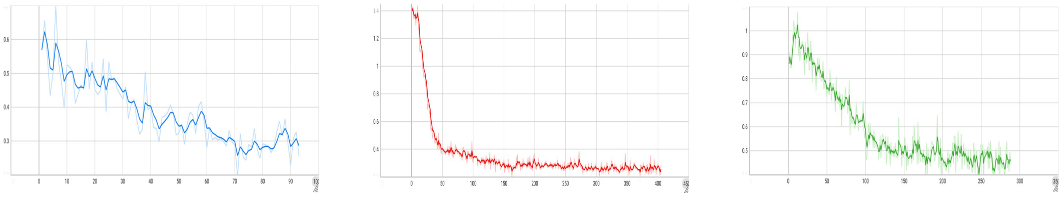
Table 1 reports exact-match accuracy comparing our approach with three baselines: Base (pretrained frozen model), and two verifier-free methods—MV (majority voting) and Ent (entropy minimization). In VDS-TTT, as well as in MV and Ent, LoRA adapters are fine-tuned on verifier-selected high-confidence pseudo-labels. The most notable gain occurs on AIME, where Qwen-1.5B, for which the task is initially unlearned, improves from 0.54% (Base, $N = 2$) to 2.8% (MV), 2.4% (Ent), and 4.22% with VDS-TTT, reaching 6.96% at $N = 16$, indicating that the model begins to acquire the task. Moreover, improvements from $N = 2$ to $N = 4$ are larger than those from $N = 8$ to $N = 16$, suggesting that moderate sampling ($N = 4$) often suffices. Overall, VDS-TTT consistently outperforms both verifier-free and base methods, enabling robust, parameter-efficient adaptation, especially under severe distribution shifts.

Table 1: Accuracy gains of VDS-TTT over base and verifier-free methods across benchmarks.

Model Name	N	Math-500				GSM-8K				AIME1983-2024			
		Base	MV	Ent	VDS-TTT	Base	MV	Ent	VDS-TTT	Base	MV	Ent	VDS-TTT
Llama3-1B	2	20.8	24.8	24.6	**28.00**	40.18	48.1	47.3	**55.88**	3.42	5.2	4.8	**8.37**
	4		26.6	26.4	**30.60**		53.0	52.2	**62.40**		6.4	5.9	**9.29**
	8		30.9	30.7	**36.60**		57.2	56.4	**63.84**		7.8	7.1	**12.35**
	16		31.5	31.2	**37.20**		60.8	59.8	**72.47**		8.2	7.5	**12.63**
Qwen-1.5B	2	19.20	23.0	22.8	**26.60**	21.15	24.4	23.9	**27.56**	0.54	2.8	2.4	**4.22**
	4		25.4	25.2	**29.80**		26.8	26.2	**30.09**		3.1	2.6	**4.31**
	8		28.1	27.8	**34.20**		27.1	26.4	**28.13**		3.7	3.2	**5.11**
	16		30.5	30.2	**36.60**		31.6	30.8	**35.12**		5.2	4.7	**6.96**
Llama3-3B	2	31.80	36.4	36.2	**40.20**	73.09	77.2	76.6	**81.11**	14.47	20.1	19.6	**24.67**
	4		37.5	37.3	**41.60**		79.9	79.2	**85.33**		21.5	20.9	**25.51**
	8		39.2	38.9	**44.80**		80.6	79.8	**85.78**		22.0	21.3	**26.16**
	16		40.5	40.2	**46.40**		83.0	82.2	**88.44**		23.4	22.6	**27.59**

Table 2 compares VDS-TTT with the recently proposed RL-based TTRL method (Zuo et al. [2025]). VDS-TTT consistently outperforms TTRL, despite the latter's reliance on heavy pretraining and its sensitivity to query difficulty and training stability. On the challenging AIME2024 and AMC benchmarks, where the base model is not tuned for math, VDS-TTT significantly outperforms TTRL. Beyond higher accuracy, VDS-TTT is also simpler and far more computationally efficient than RL-based approaches.

Figure 2 plots the test-time training loss incurred by VDS-TTT for three representative model–benchmark–sampling configurations. In all cases, the loss curves exhibit a smooth, monotonic decrease and plateau at low values, confirming that VDS-TTT consistently adapts model parameters to the test-time distribution and achieves reliable convergence under diverse settings.



(a) Llama-3.2-1B-Instruct on Math-500 for $N = 4$

(b) Llama-3.2-3B-Instruct on GSM-8K for $N = 2$

(c) DeepSeek-R1-Distill-Qwen-1.5B on AIME for $N = 8$

Figure 2: Three instances of TTT loss curves.

The cross-task evaluation in Figure 3 shows that VDS-TTT generalizes effectively beyond the task on which it is trained. When test-time training is applied to one benchmark, the method yields consistent accuracy gains on other benchmarks compared to the base model. Notably, these improvements persist under out-of-distribution settings, indicating that VDS-TTT does not rely on task-specific overfitting. Instead, its self-improvement mechanism captures transferable patterns.

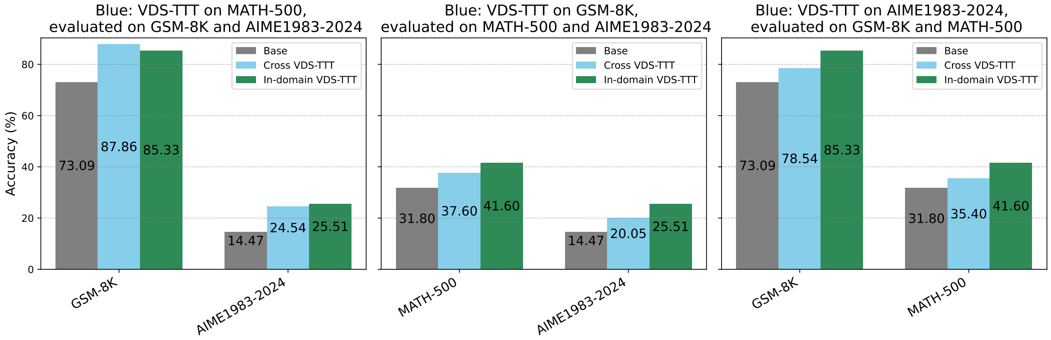
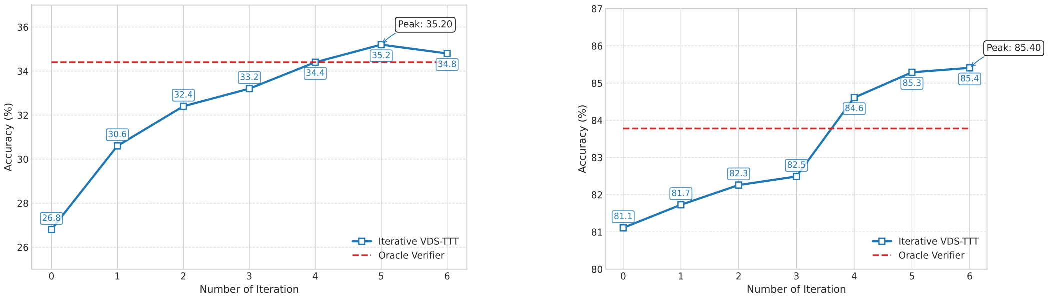


Figure 3: Out-of-distribution (OOD) generalization of Llama-3.2-3B-Instruct with $N = 4$ under VDS-TTT. Each subfigure compares the base frozen model (green) with base models fine-tuned on a source dataset and evaluated on a different target dataset (blue), alongside in-distribution performance where models are trained and evaluated on the same dataset (green). Results are showing robust adaptation of VDS-TTT under cross-domain shifts.

Figure 4 shows the performance trajectory of iterative VDS-TTT, where it is applied multiple times to the same model. Accuracy improves with successive iterations, as the model progressively generates higher-quality outputs, but eventually plateaus due to capacity limits. The red dashed line denotes the Oracle Verifier (an idealized verifier that perfectly judges correctness with access to the labels for choosing the right generated output), serving as an upper bound for best-of-N selection at iteration zero. Notably, iterative VDS-TTT can surpass this bound, demonstrating that repeated self-improvement refines performance beyond static verification. We apply early stopping, halting when accuracy gains fall below ϵ for three steps or stagnate, balancing adaptation and cost.



(a) Llama-3.2-1B-Instruct on Math-500 for $N = 4$

(b) Llama-3.2-3B-Instruct on GSM-8K for $N = 2$

Figure 4: Iterative VDS-TTT results

Table 2: Comparison of our proposed framework, VDS-TTT, with the TTRL (Zuo et al. [2025])

Model Name	Methods	AIME 2024	AMC	Math-500	Average
LLaMA-3.1-8B-Instruct	Base model	3.3	19.3	47.8	23.5
	TTRL	3.3	32.5	**61.8**	32.5
	VDS-TTT	**10.0**	**38.5**	54.2	**38.3**
	Δ_{TTRL}	0.0	+13.2	**+14.0**	+9.0
	$\Delta_{\text{VDS-TTT}}$	**+6.7**	**+19.2**	+6.4	**+10.8**

5 Conclusion

We presented Verifier-Driven Sample Selection for Test-Time Training (VDS-TTT), a framework that integrates a learned verifier into the TTT pipeline to select high-confidence pseudo-labels and adapt only lightweight LoRA parameters. This design enables efficient and stable self-improvement under distribution shift, while avoiding the pitfalls of noisy or low-confidence updates. Empirical results across multiple reasoning benchmarks demonstrate consistent accuracy gains. Furthermore, we showed that performance scales with additional test-time training iterations, rivaling or surpassing oracle verifier performance. Taken together, these findings highlight VDS-TTT as a robust and scalable approach to label-free adaptation, advancing the practical deployment of large language models in dynamic and label-scarce environments.

References

Alexander Bartler, Andre Bühler, Felix Wiewel, Mario Döbler, and Bin Yang. Mt3: Meta test-time training for self-supervised test-time adaption. In *International Conference on Artificial Intelligence and Statistics*, pages 3080–3090. PMLR, 2022.

Léon Bottou and Vladimir Vapnik. Local learning algorithms. *Neural computation*, 4(6):888–900, 1992.

William S Cleveland. Robust locally weighted regression and smoothing scatterplots. *Journal of the American statistical association*, 74(368):829–836, 1979.

William S Cleveland and Susan J Devlin. Locally weighted regression: an approach to regression analysis by local fitting. *Journal of the American statistical association*, 83(403):596–610, 1988.

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.

Caglar Gulcehre, Tom Le Paine, Srivatsan Srinivasan, Ksenia Konyushkova, Lotte Weerts, Abhishek Sharma, Aditya Siddhant, Alex Ahern, Miaosen Wang, Chenjie Gu, et al. Reinforced self-training (rest) for language modeling. *arXiv preprint arXiv:2308.08998*, 2023.

Moritz Hardt and Yu Sun. Test-time training on nearest neighbors for large language models. *arXiv preprint arXiv:2305.18466*, 2023.

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.

Edward J Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, Weizhu Chen, et al. Lora: Low-rank adaptation of large language models. *ICLR*, 1(2):3, 2022.

Jonas Hübotter, Sascha Bongni, Ido Hakimi, and Andreas Krause. Efficiently learning at test-time: Active fine-tuning of llms. *arXiv preprint arXiv:2410.08020*, 2024.

Gabriel Mukobi, Peter Chatain, Su Fong, Robert Windesheim, Gitta Kutyniok, Kush Bhatia, and Silas Alberti. Superhf: Supervised iterative learning from human feedback. *arXiv preprint arXiv:2310.16763*, 2023.

Ori Press, Ravid Shwartz-Ziv, Yann LeCun, and Matthias Bethge. The entropy enigma: Success and failure of entropy minimization. *arXiv preprint arXiv:2405.05012*, 2024.

Matthew Renze. The effect of sampling temperature on problem solving in large language models. In *Findings of the Association for Computational Linguistics: EMNLP 2024*, pages 7346–7356, 2024.

John Schulman, Filip Wolski, Prafulla Dhariwal, Alec Radford, and Oleg Klimov. Proximal policy optimization algorithms. *arXiv preprint arXiv:1707.06347*, 2017.

Amrith Setlur, Nived Rajaraman, Sergey Levine, and Aviral Kumar. Scaling test-time compute without verification or rl is suboptimal. *arXiv preprint arXiv:2502.12118*, 2025.

Zhihong Shao, Peiyi Wang, Qihao Zhu, Runxin Xu, Junxiao Song, Xiao Bi, Haowei Zhang, Mingchuan Zhang, YK Li, Y Wu, et al. Deepseekmath: Pushing the limits of mathematical reasoning in open language models. *arXiv preprint arXiv:2402.03300*, 2024.

Dequan Wang, Evan Shelhamer, Shaoteng Liu, Bruno Olshausen, and Trevor Darrell. Tent: Fully test-time adaptation by entropy minimization. *arXiv preprint arXiv:2006.10726*, 2020.

Tongtong Wu, Linhao Luo, Yuan-Fang Li, Shirui Pan, Thuy-Trang Vu, and Gholamreza Haffari. Continual learning for large language models: A survey. *arXiv preprint arXiv:2402.01364*, 2024.

Zehao Xiao and Cees GM Snoek. Beyond model adaptation at test time: A survey. *arXiv preprint arXiv:2411.03687*, 2024.

Dan Zhang, Sining Zhoubian, Ziniu Hu, Yisong Yue, Yuxiao Dong, and Jie Tang. Rest-mcts*: Llm self-training via process reward guided tree search. *Advances in Neural Information Processing Systems*, 37:64735–64772, 2024.

Yuxin Zuo, Kaiyan Zhang, Shang Qu, Li Sheng, Xuekai Zhu, Biqing Qi, Youbang Sun, Ganqu Cui, Ning Ding, and Bowen Zhou. Ttrl: Test-time reinforcement learning. *arXiv preprint arXiv:2504.16084*, 2025.