Exploring The Effectiveness of Test Time Learning In LLMs for Long Contexts

Nizar Islah
Mila Quebec AI Institute
Université de Montréal
nizar.islah@mila.quebec

Irina Rish
Mila Quebec AI Institute
Université de Montréal
irina.rish@mila.quebec

Eilif Muller
Mila Quebec AI Institute
Université de Montréal
eilif.muller@umontreal.ca

Abstract

Foundation models must keep pace with a changing world, and test-time learning (TTL) promises fast, label-free updates that could make this possible, yet our study shows (a) where that promise breaks, and (b) how to rescue that even under the challenging long context setting. Furthermore, we characterize the effect of TTL on pretrained capabilities from a continual learning perspective via the plasticity–retention trade-off (in our experiments, RULER for long-context plasticity; 3 standard LM downstream tasks for retention). We uncover a sharp pattern: TTL reliably helps at short contexts but stalls or reverses from 8k to 32k sequence lengths, while base knowledge is largely preserved. However, we see a medium-strong (0.77) correlation between input perplexity and long-context plasticity. This connects test-time improvement on long contexts to a single, measurable quantity, and suggests that the TTL objective could be key to moving the needle further and should not be entirely thrown out. Our method, which relies on measuring each token's relevance and weighting the per-token losses, rescues the performance of TTL under longer, noisier contexts. This reframes negative TTL results not as failures of the overall approach but of assuming that all tokens contribute equally; when useful context is sparse, naive test-time updates cannot meaningfully improve the model. At the same time, our method decreases the stability of the model. This work contributes empirical results and a diagnostic that make these trade-offs evident, setting the stage for useful, frequent, and low-cost updates that keep models current without eroding base capabilities.

1 Introduction

Modern LLMs work well on short, well-matched inputs, but accuracy drops under distribution shift and as contexts grow [Biderman et al., 2024, Hsieh et al., 2024, Chatziveroglou et al., 2025]. Re-training with labels is slow and resource-intensive, and retrieval adds systems complexity without adapting the model itself. Most approaches to maintaining model performance rely on labeled feedback or substantial compute. Online fine-tuning and reinforcement/feedback pipelines require targets or human signals; incurring nontrivial compute and supervision. While effective in such settings, these strategies scale poorly when supervision is unavailable or under limited compute.

39th Conference on Neural Information Processing Systems (NeurIPS 2025) Workshop: Continual and Compatible Foundation Model Updates (CCFM).

We study a label-free test-time learning (TTL) protocol following [Hu et al., 2025] that uses the input/prompt itself as supervision by minimizing input perplexity, applying small LoRA-based parameter updates under constrained compute and memory budgets. In the long-context regime, we assess plasticity with the RULER dataset at 4k–32k tokens, measure input perplexity on the same sequences, and compare base model inference to TTL variants. Crucially, we ask whether under long contexts with compute constraints, does minimizing a self-supervised objective with token salience translate into improved retrieval/reasoning performance at test time without hurting base knowledge?

1.1 Test-Time Learning (TTL) for LLMs

Problem setup. Following the setting from [Hu et al., 2025], given unlabeled test inputs $x \sim Q(x)$, adapt a pre-trained LLM by updating a small subset of parameters to better fit the test distribution:

$$\min_{\Theta} \ \mathcal{L}(x; \Theta), \quad x \sim Q(x). \tag{1}$$

Perplexity. For a token sequence $x_{1:T}$, define perplexity

$$\mathrm{PPL}(x_{1:T}; \Theta) = \exp\left(-\frac{1}{T} \sum_{t=1}^{T} \log p(x_t \mid x_{1:t-1}; \Theta) \right). \tag{2}$$

Output vs input perplexity. If y denotes the model's response to x, one can target the response perplexity

$$\min_{\Theta} \ \mathrm{PPL}(y \mid x; \Theta) = \min_{\Theta} \ \exp\left(-\frac{1}{T} \sum_{t=1}^{T} \log p(y_t \mid x, y_{1:t-1}; \Theta) \right), \tag{3}$$

and, in practice, use input perplexity minimization as a surrogate:

$$\min_{\Theta} \ \mathrm{PPL}(x; \Theta). \tag{4}$$

We can do this as [Hu et al., 2025] showed that, under the assumption that input and output are semantically related, the gradients of their respective losses should be aligned.

Lightweight updates via LoRA. Maintain a frozen base Θ and learn a small adapter $\Delta\Theta$ such that $\tilde{\Theta} = \Theta + \Delta\Theta$, optimizing

$$\min_{\Delta\Theta} \ \mathrm{PPL}\big(x; \Theta + \Delta\Theta\big). \tag{5}$$

Algorithmically, initialize $\Delta\Theta$, form $\tilde{\Theta} = \Theta + \Delta\Theta$, then update $\Delta\Theta$ by backpropagating the standard next-token loss on **input**, (minimizing 4), unlike the common supervised finetuning (SFT).

Token salience Under long-context settings where relevant information is *sparse* and distractors dominate, naïve TTL often fails and performs negatively relative to the base model. To address this, we introduce **token salience weighting**, where each token's contribution to the loss is reweighted based on its *last-row attention score* α_t, inspired by GemFilter [Shi et al., 2024]:

$$\mathcal{L} = -\sum_{t=1}^{T} w_t \cdot \log p_\theta(x_t \mid x_{<t}), \quad \text{where} \quad w_t = \frac{\alpha_t}{\max_{j=1}^{T} \alpha_j}. \tag{6}$$

Here, α_t measures the relevance of token t for the given input. Tokens with a low score receive smaller weights, effectively reducing their contribution to gradient updates, while evidence-bearing tokens are emphasized. This simple modification enables TTL to focus parameter updates on *relevant* context, improving performance on sequences with many distractors (see Appendix for more experimental details).

1.2 Key Contributions

- **Long-context TTL evaluation.** We evaluate TTL from 4K to $\mathbf{32}$K tokens across models and sizes, measuring both **RULER** reasoning accuracy and downstream retention.

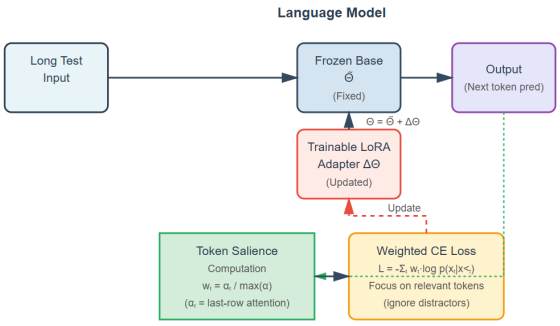


Figure 1: **Overview of label-free test-time learning (TTL) with token salience weighting.** Given a long test input *without any labels*, a frozen base model $\hat{\theta}$ is paired with a trainable **LoRA adapter** $\Delta\theta$. During adaptation, we compute **token salience weights** w_t from α_t where α_t is the **last-row attention score** for token t, inspired by [Shi et al., 2024]. These weights are used in a **weighted cross-entropy loss**, which *down-weights distractors* and *emphasizes evidence-bearing tokens*. Only the adapter parameters $\Delta\theta$ are updated; the frozen base $\hat{\theta}$ remains unchanged. At inference, the adapted model $\theta = \hat{\theta} + \Delta\theta$ is used for next-token prediction.

- **Token salience weighting for stable TTL.** Inspired by GemFilter [Shi et al., 2024], we reweight per-token cross-entropy loss using last-row attention scores, improving reasoning while incurring some forgetting.
- **Diagnostic framework for retention-plasticity trade-off.** We propose a unified analysis combining long-context reasoning and capability preservation, revealing that naïve TTL collapses beyond 8K tokens.

2 Empirical Study

2.1 Experimental Setup

RULER directly probes long-context behavior by placing sparse, relevant evidence amid large amounts of distractors and scaling the context length in a controlled way [Hsieh et al., 2024]. This makes it a good operational measure of plasticity: the model's ability to retrieve, integrate, and reason over long inputs at test time without changing training [Hsieh et al., 2024]. Its standardized tasks and length sweeps let us compare models and settings on the specific failure modes that emerge between 8k–32k+ tokens [Hsieh et al., 2024].

Evaluating general downstream tasks for retention. Retention benchmarks—HellaSwag, ARC-Challenge, and WinoGrande—act as a stable proxy for a model's preserved, general capabilities [Zellers et al., 2019, Clark et al., 2018, Sakaguchi et al., 2020]. Measuring them alongside RULER lets us visualize the plasticity–retention trade-off and observe whether long-context adaptation comes at the cost of core downstream competence [Hsieh et al., 2024].

2.2 Results

TTL consistently improves RULER accuracy at 4k context across Qwen models (Fig. 3). However, at 8k, 16k, and 32k, TTL negatively impacts performance, which is a surprising result relative to [Hu et al., 2025]. The scatter of RULER accuracy vs perplexity (Fig. 4) shows a clear pattern across all models and lengths: lower **input** perplexity aligns with higher retrieval accuracy on short and long contexts. For the Llama 3.2 1B model, we saw an improvement over the baseline at 4k, as well as some improvement at 8k-32k context lengths with TTL. However, when combined with the token-salience weighted loss, we get a boost at all context lengths over the baseline and standard TTL. The frontier-style plot in Fig. 2 shows that increasing (mostly irrelevant) context within a fixed model size mainly reduces RULER accuracy, although TTL + token salience rescues long context performance while also reducing general base knowledge retention, demonstrating the stability-plasticity tradeoff.

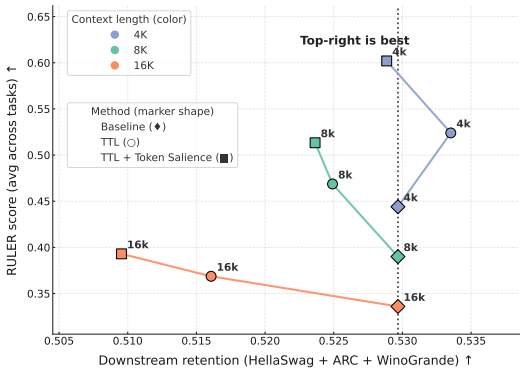
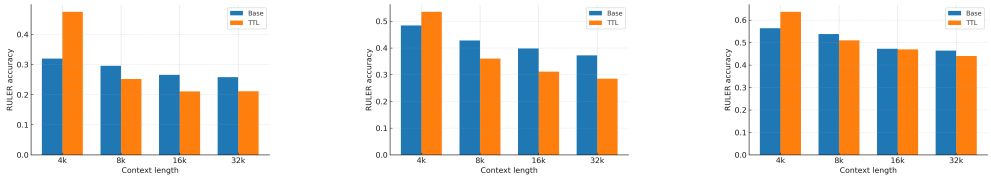


Figure 2: Trade-off between downstream retention and long-context reasoning accuracy. We evaluate test-time learning (TTL) and TTL + token salience weighting across context lengths (4K, 8K, 16K) on LLaMA 3.2 1B. The x-axis measures retention (averaged) after TTL, while the y-axis reports RULER accuracy on long-context reasoning tasks under distractors. Vertical dashed line indicates pretrained knowledge. Connector lines group models by context length, illustrating within-context trade-offs: naïve TTL improves reasoning but incurs retention loss (with the exception of 4k), especially at 8K and 16K, while TTL + token salience further improves plasticity at the cost of stability.



(a) Qwen2.5 0.5B (b) Qwen2.5 1.5B (c) Qwen2.5 7B

Figure 3: RULER accuracy vs context length for Qwen2.5 models: 0.5B (a), 1.5B (b), 7B (c). Blue = base; orange = TTL (averaged over seeds). TTL delivers clear gains at 4k across all sizes, but the advantage vanishes at 8k and often reverses at 16–32k, indicating failure to adapt in long contexts.

3 Discussion

Our results reveal that test-time learning (TTL) operates along a fragile retention–adaptation frontier: while naïve TTL improves local perplexity, it often collapses beyond 8K tokens where relevant signals become sparse and gradients are dominated by distractors. Incorporating token salience weighting shifts models toward the upper-left regime in our diagnostic plots, improving long-context reasoning but sacrificing downstream retention.

This finding complements recent advances in self-adaptive language models such as SEAL [Zweiger et al., 2025], which trains models to generate synthetic data and self-edit their parameters via reinforcement learning. Unlike SEAL's global updates, our approach highlights the importance of local selectivity during adaptation when evidence is sparse. Alternatively, Titans-like models

(a) Input perplexity vs RULER accuracy (b) Output perplexity vs RULER accuracy

Figure 4: Perplexity–accuracy relationship across context lengths. Lower perplexity aligns with higher accuracy on long contexts in both panels. The correlation is surprisingly stronger for input perplexity ($r \sim -0.77$) than for output perplexity ($r \sim -0.43$), indicating that reducing uncertainty on the input side is more predictive of long-context success.

[Behrouz et al., 2024, Zhang et al., 2025] enable persistent, iterative reasoning over million-token contexts by integrating dedicated memory modules at pretraining time. However, this design comes at a cost: memory-augmented reasoning must be architecturally baked in and cannot be retrofitted onto arbitrary pretrained LLMs at test time. Together, these approaches hint at a spectrum of strategies for scalable, stable test-time learning: 1) Global self-adaptation (SEAL), 2) Local, selective TTL (our approach), and 3) Architectural memory integration (Titans). Bridging these paradigms offers a promising path toward foundation models capable of iterative reasoning over extended contexts while maintaining general capabilities.

References

Ali Behrouz, Peilin Zhong, and Vahab Mirrokni. Titans: Learning to memorize at test time. *arXiv preprint arXiv:2501.00663*, 2024. doi: 10.48550/arXiv.2501.00663. URL https://arxiv.org/abs/2501.00663.

Dan Biderman, Jacob Portes, Jose Javier Gonzalez Ortiz, Mansheej Paul, Philip Greengard, Connor Jennings, Daniel King, Sam Havens, Vitaliy Chiley, Jonathan Frankle, Cody Blakeney, and John P. Cunningham. Lora learns less and forgets less. *Transactions on Machine Learning Research*, 2024. URL https://arxiv.org/abs/2405.09673. TMLR journal track (accepted 2024); arXiv preprint arXiv:2405.09673.

Giannis Chatziveroglou, Richard Yun, and Maura Kelleher. Exploring LLM reasoning through controlled prompt variations. arXiv preprint arXiv:2504.02111, 2025. URL https://arxiv.org/abs/2504.02111. arXiv preprint arXiv:2504.02111.

Peter Clark, Isaac Cowhey, Oren Etzioni, Tushar Khot, Ashish Sabharwal, Carissa Schoenick, and Oyvind Tafjord. Think you have solved question answering? try arc, the ai2 reasoning challenge. *arXiv preprint arXiv:1803.05457*, 2018. doi: 10.48550/arXiv.1803.05457. URL https://arxiv.org/abs/1803.05457.

Cheng-Ping Hsieh, Simeng Sun, Samuel Kriman, Shantanu Acharya, Dima Rekesh, Fei Jia, Yang Zhang, and Boris Ginsburg. Ruler: What's the real context size of your long-context language models? *arXiv preprint arXiv:2404.06654*, 2024. doi: 10.48550/arXiv.2404.06654. URL https://arxiv.org/abs/2404.06654.

Jinwu Hu, Zitian Zhang, Guohao Chen, Xutao Wen, Chao Shuai, Wei Luo, Bin Xiao, Yuanqing Li, and Mingkui Tan. Test-time learning for large language models. *arXiv preprint arXiv:2505.20633*, 2025. URL https://arxiv.org/abs/2505.20633.

Meta AI. Llama 3.2: Compact multilingual large language models for diverse applications, 2024. URL https://arxiv.org/abs/2310.12346. Accessed via certification in related LLaMA documentation.

Qwen, An Yang, Baosong Yang, Beichen Zhang, Binyuan Hui, Bo Zheng, Bowen Yu, Chengyuan Li, Dayiheng Liu, Fei Huang, Haoran Wei, Huan Lin, Jian Yang, Jianhong Tu, Jianwei Zhang, Jianxin Yang, Jiaxi Yang, Jingren Zhou, Junyang Lin, Kai Dang, Keming Lu, Keqin Bao, Kexin Yang, Le Yu, Mei Li, Mingfeng Xue, Pei Zhang, Qin Zhu, Rui Men, Runji Lin, Tianhao Li, Tianyi Tang, Tingyu Xia, Xingzhang Ren, Xuancheng Ren, Yang Fan, Yang Su, Yichang Zhang, Yu Wan, Yuqiong Liu, Zeyu Cui, Zhenru Zhang, and Zihan Qiu. Qwen 2.5 technical report, 2024. URL https://arxiv.org/abs/2412.15115. Introduces the Qwen 2.5 family including 0.5B, 1.5B, 7B variants.

Keisuke Sakaguchi, Ronan Le Bras, Chandra Bhagavatula, and Yejin Choi. Winogrande: An adversarial winograd schema challenge at scale. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 34, pages 8732–8740. AAAI Press, 2020. doi: 10.1609/aaai.v34i05.6399. URL https://ojs.aaai.org/index.php/AAAI/article/view/6399.

Zhenmei Shi, Yifei Ming, Xuan-Phi Nguyen, Yingyu Liang, and Shafiq Joty. Discovering the gems in early layers: Accelerating long-context llms with 1000x input token reduction. *arXiv preprint arXiv:2409.17422*, 2024. doi: 10.48550/arXiv.2409.17422. URL https://arxiv.org/abs/2409.17422.

Rowan Zellers, Ari Holtzman, Yonatan Bisk, Ali Farhadi, and Yejin Choi. Hellaswag: Can a machine really finish your sentence? In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 4791–4800, Florence, Italy, 2019. Association for Computational Linguistics. doi: 10.18653/v1/P19-1472. URL https://aclanthology.org/P19-1472/.

Tianyuan Zhang, Sai Bi, Yicong Hong, Kai Zhang, Fujun Luan, Songlin Yang, Kalyan Sunkavalli, William T. Freeman, and Hao Tan. Test-time training done right. *arXiv preprint arXiv:2505.23884*, 2025. URL https://arxiv.org/abs/2505.23884.

Adam Zweiger, Jyothish Pari, Han Guo, Ekin Akyürek, Yoon Kim, and Pulkit Agrawal. Seal: Self-adapting language models. *arXiv preprint arXiv:2506.10943*, 2025.

.1 Experiment Hyperparameters

We ran test time learning (TTL) on the Hotpot QA (QA2) subtask of RULER with various configurations. Unless otherwise specified, we fix most hyperparameters and only vary the `context length`, `random seed`, `adaptation strategy`, and `model family`. For experiments using token salience, we run a preliminary forward pass as in GEMFILTER, fixed at layer 13. Table 1 summarizes the fixed hyperparameters, and Table 2 enumerates the evaluated experimental configurations. All experiments are run for 3 seeds, conducted on a single GPU (NVIDIA L40S, A100, or H100), with the longest runs (`TTL + Token Salience`, 16K context) taking up to 2.5 hours, while shorter runs (4K–8K context) typically complete within 15–45 minutes.

Table 1: Fixed hyperparameters for LLaMA experiments.

Hyperparameter	**Value**	**Description**
Dataset	`RULER (QA2)`	long-context benchmark
Optimizer	`AdamW`	Weight-decay regularized Adam
Epochs	3	Number of training epochs
LoRA rank r	8	Low-rank adaptation dimension
LoRA α	32	Scaling factor for LoRA updates
LoRA dropout	0.1	Dropout applied to LoRA layers
Weight decay	0.01	ℓ_2 regularization
Batch size	4	Number of sequences per step
Learning rate	2×10^{-4}	Peak learning rate
Context length	{4096, 8192, 16384, 32768}	Varied across runs
GemFilter layer	13	Token salience extraction layer
`-input_only`	`True`	TTL enabled

Table 2: Varied experimental configurations. Token salience is applied only for LLaMA-3.2-1B due to limited compute resources.

Model	**Adaptation Strategy**	**Context Lengths**
LLaMA-3.2-1B-Instruct [Meta AI, 2024]	TTL-only	{4K, 8K, 16K}
LLaMA-3.2-1B-Instruct	TTL + Token Salience	{4K, 8K, 16K}
Qwen2.5-0.5B-Instruct [Qwen et al., 2024]	TTL-only	{4K, 8K, 16K, 32K}
Qwen2.5-1.5B-Instruct	TTL-only	{4K, 8K, 16K, 32K}
Qwen2.5-7B-Instruct	TTL-only	{4K, 8K, 16K, 32K}