ESANN 2026 proceedings, European Symposium on Artificial Neural Networks, Computational Intelligence and Machine Learning. Bruges (Belgium) and online event, 22-24 April 2026, i6doc.com publ., ISBN 9782875870964. Available from http://www.i6doc.com/en/.

# Neuromodulated Delta Adapters: Stabilizing Test-Time Adaptation via Gated Error Correction

Jiarui Zhang[1,2] and Yifan Deng[1,2]

1- Institute of Information Engineering, Chinese Academy of Sciences, No. 19 Shu Cun Road, Haidian District, Beijing, China

2- School of Cyber Security, University of Chinese Academy of Sciences

**Abstract**. Static neural networks degrade under distribution shift, and existing Test-Time Adaptation (TTA) either backpropagates during inference or relies on unstable Hebbian dynamics. We propose the **Neuromodulated Delta Adapter (NDA)**, a plug-and-play PEFT module that inserts a rank-$r$ fast-weight bottleneck into frozen Transformers. NDA couples a gated Delta rule with a three-factor "surprise" signal, providing adaptive gain control that keeps fast weights Lyapunov-stable. On FLORES-101 (continuous) NDA surpasses TENT by +0.9 spBLEU and remains stable on English-to-Yoruba. On PG-19 it lowers perplexity to 36.9 at 8k tokens and recalls 94% of long-range "needle" facts.

## 1 Introduction

Real-world inference streams, from low-resource NMT to long-context dialogue, exhibit continuous distributional drift, so frozen pretrained Transformers degrade. TTA updates parameters online, yet current methods either backpropagate during inference (TENT [1]), paying high cost and risking collapse, or rely on unconstrained Hebbian rules ($\Delta W \propto yx^T$) that diverge on long contexts [2, 3]. Dynamic-weight architectures such as Fast Weight Programmers, Mamba, or Gated DeltaNet [4, 5, 6] improve stability but demand backbone retraining, limiting deployment.

We retrofit frozen Transformers with the **Neuromodulated Delta Adapter (NDA)**: a PEFT module that confines fast weights to a rank-$r$ bottleneck, gates them with a three-factor signal [3], and updates them via a rank-one Delta rule. NDA leaves the backbone untouched, adds <0.2% parameters, and keeps dynamics bounded through adaptive gain control, outperforming Hebbian, TENT, and SAR on FLORES+ and PG-19.

## 2 Related Work

**Dynamic Evaluation & TTA:** Dynamic evaluation [7] adapts via backpropagation but costs roughly 3× inference time and needs careful tuning. NDA achieves similar adaptation with $O(1)$ forward updates.

**Fast Weight Programmers:** Linear Transformers [8], Fast Weight Programmers [2], Mamba [5], and Gated DeltaNet [6] stabilize long sequences by redesigning backbone layers. Unlike their input-only gates, NDA conditions

ESANN 2026 proceedings, European Symposium on Artificial Neural Networks, Computational Intelligence and Machine Learning. Bruges (Belgium) and online event, 22-24 April 2026, i6doc.com publ., ISBN 9782875870964. Available from http://www.i6doc.com/en/.

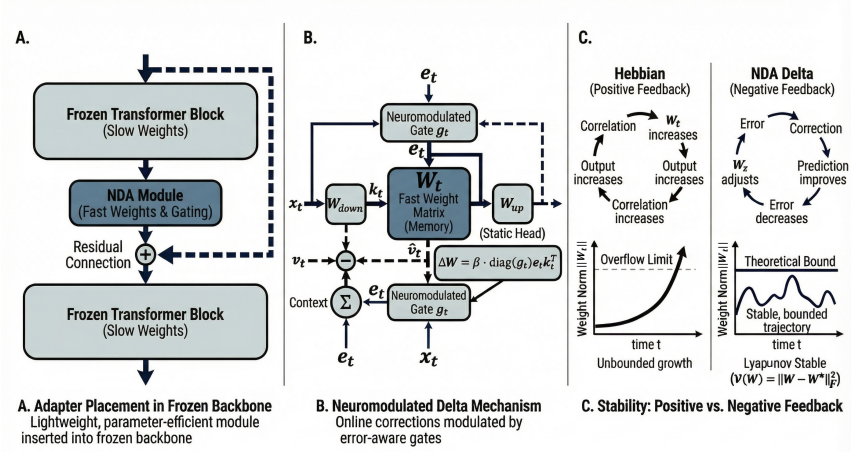


Fig. 1: Architecture of the Neuromodulated Delta Adapter. Unlike positive-feedback Hebbian rules, a gated Delta update uses the error $e_t$ to modulate $g_t$, forming a negative-feedback loop that preserves Lyapunov stability.

updates on prediction errors, acting as a content-addressable correction while staying a drop-in adapter for frozen models.

**PEFT:** LoRA [9] and classical adapters [10] add light trainable modules yet stay static after fine-tuning. NDA keeps the PEFT footprint but endows the adapter itself with dynamic weights for online deployment.

## 3 The Neuromodulated Delta Adapter

NDA threads through the frozen Transformer as shown in Figure 1: tokens enter a shared down-projection, interact with the gated fast weights, and rejoin the residual stream via the up-projection before each component is detailed below.

### 3.1 Adapter Placement and Data Flow

NDA inserts between frozen Transformer blocks: each hidden state $x_t \in \mathbb{R}^d$ forms a bottleneck key $k_t = W_{down} x_t$, retrieves $\hat{v}_t = W_t^{(t-1)} k_t$, and feeds the correction through a residual adapter.

$$h_t = x_t + W_{up}\left(\text{LayerNorm}\left(\sigma(\hat{v}_t + b_v) + k_t\right)\right) \tag{1}$$

with $W_{up} \in \mathbb{R}^{d \times r}$ the up-projection, $b_v$ a static bias, and $\sigma$ the SiLU activation.

### 3.2 Error-Conditioned Neuromodulation (Three-Factor Rule)

Adaptation uses a local target $v_t$ from the static head and error $e_t = v_t - \hat{v}_t$. Following three-factor learning [3], we gate the update via

$$g_t = \sigma(\text{MLP}_{gate}([x_t; e_t])), \tag{2}$$

ESANN 2026 proceedings, European Symposium on Artificial Neural Networks, Computational Intelligence and Machine Learning. Bruges (Belgium) and online event, 22-24 April 2026, i6doc.com publ., ISBN 9782875870964. Available from http://www.i6doc.com/en/.

so the meta-learned "surprise" detector opens the learning window only for confident mistakes.

### 3.3 Fast-Weight Delta Update

Fast weights follow a gated, rank-one Delta rule [11]:

$$W_t^{(t)} = W_t^{(t-1)} + \beta \, \mathrm{diag}(g_t) \, e_t k_t^\top, \tag{3}$$

where $\beta$ is a learnable rate, approximating online Recursive Least Squares (RLS) in the bottleneck with $O(r^2)$ complexity.

### 3.4 Stability Analysis: Adaptive Gain Control

Standard Hebbian updates ($\Delta W \propto v_t k_t^\top$) [12] reinforce their own predictions and blow up. NDA keeps the loop closed: the Delta rule [11] supplies negative feedback while the gate shrinks the effective rate $\eta_{eff} = \beta g_t$ whenever $\|k_t\|$ or $\|e_t\|$ spike. By enforcing the condition

$$0 < \beta \|g_t\|_\infty \|k_t\|^2 < 2 \tag{4}$$

we remain inside the stability region of the Recursive Least Squares (RLS) approximation, preventing the divergence typically seen in Hebbian learning [2]. Using $V(W_t) = \|W_t - W^*\|_F^2$ as the Lyapunov function yields the decrement:

$$\|W_t - W^*\|_F^2 - \|W_{t-1} - W^*\|_F^2 = -2\eta_{eff} \langle e_t k_t^\top, W_{t-1} - W^* \rangle + \eta_{eff}^2 \|e_t k_t^\top\|_F^2, \tag{5}$$

stays non-positive under that bound, so the fast weights contract even under adversarial streams. We clip $\|W_t\|_F$ at $R = 5$ to handle rare corrupted tokens.

### 3.5 Implementation Details and Hyperparameters

We insert NDA blocks after every decoder feed-forward and cross-attention in mBART-50 (24 adapters) and every other MLP in OPT-1.3B. The backbone stays frozen: we train $(W_{down}, W_{up})$, the two-layer SiLU gate MLP (width $4r$ shared across layers), and the scalar $\beta$, while $W_t$ exists only at inference and reuses frozen LayerNorm stats.

Fast weights run in 16-bit precision, reset at document boundaries on FLORES and once per book on PG-19. We fuse the gated Delta update into the residual adapter kernel, making the per-token cost a rank-one outer product plus a diagonal gate. Both datasets use $r = 64$ and a gate hidden size of 256, with $\beta = 0.05$ for translation streams and $\beta = 0.08$ for PG-19.

## 4 Experimental Setup

We evaluate NDA on FLORES-101 translation and PG-19 long-context modeling.

ESANN 2026 proceedings, European Symposium on Artificial Neural Networks, Computational Intelligence and Machine Learning. Bruges (Belgium) and online event, 22-24 April 2026, i6doc.com publ., ISBN 9782875870964. Available from http://www.i6doc.com/en/.

**Datasets.** With a frozen mBART-50 [13] backbone we cover high (En→De), mid (En→Zh), and zero-shot (En→Yo) FLORES regimes; Hausa (En→Ha) is used for ablations. PG-19 streams books through a frozen OPT-1.3B decoder using a 2048-token window and 512 stride.

**Baselines.** We compare to Static LoRA [9], Vanilla Hebbian [2], Sliding Window Hebbian (SWH), TENT [1], and SAR [14]; PG-19 additionally reports the static OPT-1.3B model.

**Evaluation Protocol.** All FLORES runs keep fast weights across each document (Continuous Adaptation), and the News → Bible → Medical curriculum reuses the same state.

## 5 Results and Analysis

### 5.1 Multilingual Translation Performance

Table 1 summarizes performance across resource levels.

| Method | High (En-De) | Mid (En-Zh) | Low (En-Yo) | Stability |
|---|---|---|---|---|
| mBART-50 (Zero-shot) | 32.4 | 20.1 | 1.8 | N/A |
| Static LoRA | 33.1 | 21.5 | 16.5 | Stable |
| Vanilla Hebbian | NaN | NaN | NaN | Unstable |
| TENT (Entropy Min.) | 32.8 (-0.3) | 21.0 (-0.5) | 14.2 (-2.3) | Stable |
| Sliding Window Hebbian | 33.2 | 21.8 | 16.8 | Stable |
| SAR (Adapted) | 33.0 (-0.1) | 21.4 (-0.1) | 16.1 (-0.4) | Stable |
| **NDA (Ours)** | **33.5** | **22.4** | **17.4** | Stable |

Table 1: Continuous Test-Time Adaptation results on FLORES-101 (spBLEU). **High**: En-De, **Mid**: En-Zh, **Low**: En-Yo. The term "Diverges" denotes numerical instability resulting in floating-point overflow (NaN).

En→Yo is our zero-shot focus; Hausa is shown only in ablations. SAR improves stability over Hebbian updates yet still trails NDA by ∼1.3 BLEU on En→Yo.

**Stability vs. Plasticity.** Vanilla Hebbian diverges under positive feedback. NDA stays bounded, matching the Lyapunov analysis, while SAR relies on rescaling and plateaus near SWH.

**Robustness to Calibration Error (vs. TENT).** TENT stays near-static on En→De but collapses on En→Yo because its entropy objective amplifies miscalibrated, high-confidence mistakes. NDA instead updates on reconstruction discrepancies and preserves its +0.9 BLEU gap over Static LoRA.

### 5.2 Long-Context Modeling (PG-19)

On PG-19, Table 2 shows that Sliding Window Hebbian plateaus as its fixed horizon forgets early entities, whereas NDA behaves as a content-addressed memory whose perplexity keeps dropping with context.

ESANN 2026 proceedings, European Symposium on Artificial Neural Networks, Computational Intelligence and Machine Learning. Bruges (Belgium) and online event, 22-24 April 2026, i6doc.com publ., ISBN 9782875870964. Available from http://www.i6doc.com/en/.

| Method | PPL @2k | PPL @8k | Needle Recall (%) |
|---|---|---|---|
| Static OPT-1.3B | 41.5 | 43.9 | 0 |
| Sliding Window Hebbian | 40.6 | 40.8 | 21 |
| **NDA (Ours)** | **39.2** | **36.9** | **94** |

Table 2: PG-19 streaming perplexity (lower is better) using a 2048-token window and 512 stride. Needle recall measures the percentage of synthetic definitions retrieved at 8k tokens.

The needle-in-a-haystack probe shows that NDA retains almost all injected definitions while SWH rapidly forgets them, indicating that the error-driven gate protects rare facts.

### 5.3 Ablation Study: Unpacking the Stability

We validate design choices on the En→Ha continuous protocol (Table 3).

| Variant | En→Ha BLEU | Stability (after 10k tokens) |
|---|---|---|
| **NDA (Full Model)** | **14.1** | **Stable** |
| *Gating Mechanisms* | | |
| w/o error-conditioned gate ($g_t = 1$) | 11.8 (−2.3) | Diverges (NaN) |
| w/o gating entirely (pure Delta) | 12.4 (−1.7) | Stable but slow convergence |
| input-only gate ($g_t = \sigma(\text{MLP}(x_t))$) | 13.2 (−0.9) | Stable |
| error-only gate ($g_t = \sigma(W_e e_t)$) | 13.8 (−0.3) | Stable |
| *Update Rules & Architecture* | | |
| Hebbian instead of Delta ($e_t \to \hat{v}_t$) | 13.5 (−0.6) | Diverges (Positive Feedback) |
| static bias ($\theta = 0$) | 13.9 (−0.2) | Stable |
| full-dimensional fast weights ($r = d$) | 13.9 (−0.2) | **OOM** on long docs |
| no norm clipping | 13.7 (−0.4) | Occasional overflow |

Table 3: Ablation study on En→Ha (Continuous). Dropping the error-conditioned gate diverges outright, while input-only gating—as in Mamba-2/Gated DeltaNet—still sacrifices BLEU in this low-resource regime.

Table 3 shows that removing the error-conditioned gate immediately diverges, input-only gating still incurs a noticeable BLEU drop, and growing the fast weights to full dimension causes OOM, underscoring the need for selective gating and a compact bottleneck.

### 5.4 Continual Learning and Complexity

On the News → Bible → Medical curriculum, NDA limits forgetting (FM = −0.8 vs. −3.2 for Static LoRA) while keeping the recurrent update $O(Lr^2)$ and adding only about 5% latency.

ESANN 2026 proceedings, European Symposium on Artificial Neural Networks, Computational Intelligence and Machine Learning. Bruges (Belgium) and online event, 22-24 April 2026, i6doc.com publ., ISBN 9782875870964. Available from http://www.i6doc.com/en/.

## 6 Conclusion

We introduced the Neuromodulated Delta Adapter, a PEFT drop-in that couples a gated Delta rule with low-rank fast weights to keep frozen Transformers stable during test-time adaptation. Across continuous FLORES-101 and PG-19 streams it stays bounded while outperforming TENT, Hebbian adapters, and SAR, offering a practical recipe for deploying neuromodulated fast weights on existing multilingual and long-context systems. We will release code/models to support follow-up work.

## 7 Acknowledgements

We extend our sincere gratitude to the anonymous reviewers for their invaluable and insightful feedback. This research was supported by the National Natural Science Foundation of China (Grant No.U21B2009).

## References

[1] Wang et al. Tent: Fully test-time adaptation by entropy minimization. In *International Conference on Learning Representations*, 2021.

[2] Ba et al. Using fast weights to attend to the recent past. *Advances in neural information processing systems*, 29, 2016.

[3] Miconi et al. Backpropamine: training self-modifying neural networks with differentiable neuromodulated plasticity. In *International Conference on Learning Representations*, 2018.

[4] Schlag et al. Linear transformers are secretly fast weight programmers. In *International conference on machine learning*, pages 9355–9366. PMLR, 2021.

[5] Gu et al. Mamba: Linear-time sequence modeling with selective state spaces. In *First conference on language modeling*, 2024.

[6] Yang et al. Gated delta networks: Improving mamba2 with delta rule. In *The Thirteenth International Conference on Learning Representations*, 2025.

[7] Krause et al. Dynamic evaluation of neural sequence models. In *International Conference on Machine Learning*, pages 2766–2775. PMLR, 2018.

[8] Katharopoulos et al. Transformers are rnns: Fast autoregressive transformers with linear attention. In *International conference on machine learning*. PMLR, 2020.

[9] Hu et al. Lora: Low-rank adaptation of large language models. In *International Conference on Learning Representations*, 2022.

[10] Houlsby et al. Parameter-efficient transfer learning for nlp. In *International conference on machine learning*, pages 2790–2799. PMLR, 2019.

[11] Widrow et al. Adaptive switching circuits. In *Neurocomputing: foundations of research*, pages 123–134. 1988.

[12] Oja et al. Simplified neuron model as a principal component analyzer. *Journal of mathematical biology*, 15(3):267–273, 1982.

[13] Tang et al. Multilingual translation with extensible multilingual pretraining and finetuning. *arXiv preprint arXiv:2008.00401*, 2020.

[14] Niu et al. Towards stable test-time adaptation in dynamic wild world. In *The Eleventh International Conference on Learning Representations*, 2023.