# CONTINUUMFORMER: PERSISTENT SOFT STRUCTURAL MEMBERSHIP FOR INCREMENTAL SEQUENCE AND DEPTH COMPUTATION

**Anonymous authors**
Paper under double-blind review

## ABSTRACT

Transformers repeatedly construct input-dependent relationships through attention, routing, and layer aggregation, yet these structural associations are usually treated as transient. We introduce ContinuumFormer, a Transformer architecture built around persistent soft structural membership: token-to-sequence-structure and layer-to-depth-structure associations are maintained as a persistent state through sparse, overlapping assignments. The state induces a weighted structural topology that parameterizes sparse structural reconstruction, and is updated through candidate retrieval, change detection, hysteresis, local correction, and an explicit update budget. We formalize this mechanism as a state-transition system and derive conditional properties for topology construction, membership-to-topology stability, fixed-state local sensitivity, and budgeted structural maintenance. We evaluate the architecture on a 150M-parameter language model using matched variants, controlled ablations, and long-context validation. The experiments show improved language-modeling performance over the matched Vanilla baseline across the evaluated standard context lengths, while axis-wise ablations reveal asymmetric behavior between sequence- and depth-side organization. An additional extended-context evaluation further probes robustness of the full persistent-state configuration beyond the standard context range. The study is designed as controlled architectural validation rather than compute-saturated language-model pretraining. Overall, the results provide controlled empirical evidence that persistent structural membership can serve as an additional organizational abstraction for Transformer computation.

## 1 INTRODUCTION

Transformers construct flexible input-dependent interactions between representations through attention, routing, and depth-wise aggregation (Vaswani et al., 2017). A recurring computational pattern is

$$x \rightarrow \text{association scores} \rightarrow \text{weights} \rightarrow \text{aggregation}. \tag{1}$$

These associations can encode useful information about the model's current structural organization, but conventional implementations usually recompute them for subsequent computation rather than maintaining them as an explicit persistent object.

We investigate a different abstraction:

$$\boxed{\text{transient computational association}} \quad \longrightarrow \quad \boxed{\text{persistent structural membership.}} \tag{2}$$

The goal is not to assume that caching an intermediate tensor automatically makes a Transformer faster. Instead, the architectural question is whether a model can maintain a compact structural state and selectively modify the regions affected by new information, thereby turning structural organization into an incremental computational process. For evaluation, we use controlled language-model NLL/PPL as the primary metrics to isolate the architectural effect while holding the base Transformer and training protocol fixed; downstream task performance is outside this evaluation scope.

We instantiate this abstraction along two orthogonal axes:

$$\boxed{\text{Token} \rightarrow \text{Sequence Structure}} \qquad \boxed{\text{Layer} \rightarrow \text{Depth Structure}}. \tag{3}$$



Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

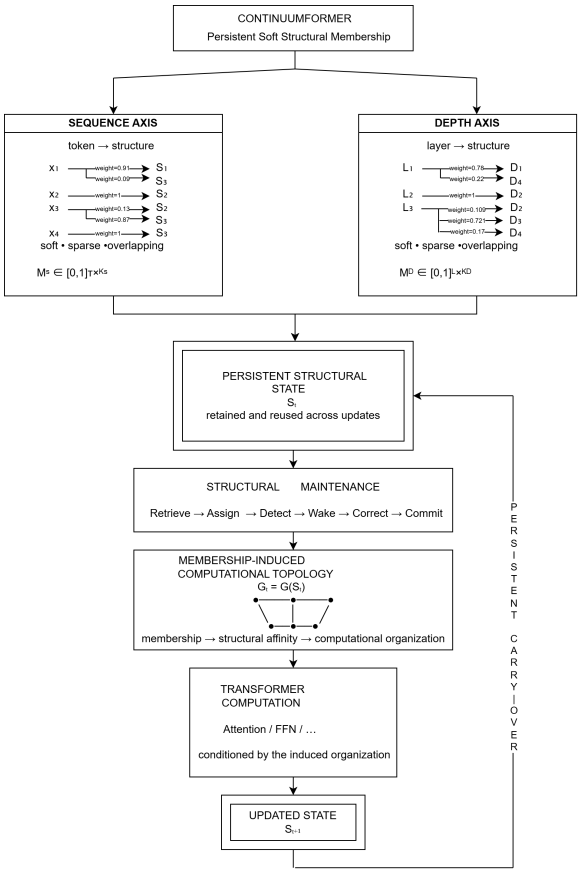


Figure 1: ContinuumFormer architecture. Sequence- and depth-axis memberships form a persistent structural state. The state is maintained through the shared lifecycle Retrieve–Assign–Detect–Wake–Correct–Commit, with bounded updates and reset at sequence boundaries.

The sequence axis organizes tokens into learned, soft structural units, while the depth axis organizes layers into learned virtual blocks. Both use the same state semantics while retaining domain-specific compatibility functions.

Our core hypothesis is that structural organization itself can be a persistent computational state: rather than recomputing all structural associations afresh, the model can retain soft memberships, selectively detect changes, and locally update the induced organization. This hypothesis separates the proposed mechanism from conventional attention weights or one-step routing scores, which need not persist as an explicit state across successive updates.

This formulation is related to dynamic sequence organization such as H-Net and Byte Latent Transformer (Hwang et al., 2026; Pagnoni et al., 2025), depth-wise aggregation such as DenseFormer and Attention Residuals (Pagliardini et al., 2024; Kimi Team et al., 2026), and conditional computation such as Mixture-of-Depths (Raposo et al., 2024). It is also related to persistent memory mechanisms such as Transformer-XL, Compressive Transformers, Memorizing Transformers, and Infini-attention We do not claim novelty for soft weighting, sparse routing, dynamic grouping, or persistence individually. The contribution is a persistent membership state whose induced weighted relations parameterize structural reconstruction across both sequence and depth axes.

This paper makes four contributions:

1. We formulate sequence and depth organization as instances of a common persistent soft-membership operator.

2. We formalize the induced weighted structural topology and an incremental maintenance lifecycle: Retrieve-Assign-Detect-Wake-Correct-Commit, which uses persistent membership to identify regions for maintenance and correction.

3. We derive conditional properties of this structural representation under explicit assumptions; they characterize topology construction, stability, local sensitivity, and budgeted maintenance rather than empirical performance.

4. We empirically characterize quality, axis-wise behavior, robustness, and the modeling-systems trade-off of the resulting architecture.

## 2 Persistent Soft Structural Membership

### 2.1 State and topology

Let $N$ denote the number of sequence entities and $K_S$ the number of sequence structures. At each assignment step, the provisional sequence membership is

$$\widehat{M}_t^S \in [0,1]^{N\times K_S}, \qquad \sum_{k=1}^{K_S} \widehat{M}_{t,ik}^S = 1. \tag{4}$$

Similarly, for $L$ Transformer layers and $K_D$ virtual depth structures, the provisional depth membership is

$$\widehat{M}_t^D \in [0,1]^{L\times K_D}, \qquad \sum_{b=1}^{K_D} \widehat{M}_{t,lb}^D = 1. \tag{5}$$

The committed persistent matrices $M_t^{\mathrm{seq}}$ and $M_t^{\mathrm{depth}}$ store the maintained state after EMA updates and therefore need not remain row-normalized when historical support is retained. The reference configuration uses sparse top-$k$ membership with $k = 2$ for each provisional assignment on both axes.

A generic membership matrix induces an affinity topology

$$A(M) = MM^\top, \tag{6}$$

with

$$A_t^{seq} = M_t^{\mathrm{seq}}(M_t^{\mathrm{seq}})^\top, \qquad A_t^{depth} = M_t^{\mathrm{depth}}(M_t^{\mathrm{depth}})^\top. \tag{7}$$

The topology is an analytical object; the implementation does not materialize the dense $N \times N$ adjacency matrix.

We define the persistent structural state as

$$S_t = \left(M_t^{\mathrm{seq}}, M_t^{\mathrm{depth}}, \theta_t^{str}, R_t, Q_t, I_t, \ell_t\right), \tag{8}$$

where $R_t$ denotes retrieved candidates, $Q_t$ confidence/change statistics, $I_t$ structural indices, and $\ell_t$ auxiliary state. New information updates the state according to

$$S_{t+1} = \mathcal{U}_\theta(S_t, x_{t+1}), \qquad G_t = \mathcal{G}_\theta(S_t). \tag{9}$$

Thus the central abstraction is
Membership -> persistent state -> induced topology -> structural reconstruction.

Forward-pass interface. Here topology denotes an analytical, membership-induced weighted relation rather than an executable dense adjacency tensor. It is not injected as a dense attention-logit bias. In the reference implementation, sequence and depth membership instead parameterize post-stack sparse structural reconstruction; the exact tensor-level equations and gradient path are given in Appendix E.3.

The persistence claimed here is intra-stream structural persistence: the state is reused across successive updates of the same sequence or autoregressive stream and is reset at sequence boundaries. It is not persistent model memory carried across unrelated examples, nor is it a cache of past hidden activations in the sense of Transformer-XL or a content memory in Memorizing/Infini-attention.

### 2.2 INCREMENTAL STRUCTURAL MAINTENANCE

For a new representation $x_{t+1}$, the maintenance lifecycle first retrieves a bounded candidate set $\mathcal{C}_t$, computes a sparse assignment $\widehat{M}$, and evaluates a change score:

$$D_t = \mathrm{ChangeScore}(M_t, \widehat{M}). \tag{11}$$

Hysteresis uses separate activation and deactivation thresholds,

$$\tau_{\mathrm{enter}} = 0.15, \qquad \tau_{\mathrm{exit}} = 0.08, \tag{12}$$

and the previous activation state. For row $i$, with change score $d_{t,i}$ and prior state $a_{t-1,i}$,

$$a_{t,i} = \mathrm{Wake}(d_{t,i}, a_{t-1,i}) = \begin{cases} 1, & d_{t,i} \geq \tau_{\mathrm{enter}}, \\ 0, & d_{t,i} \leq \tau_{\mathrm{exit}}, \\ a_{t-1,i}, & \tau_{\mathrm{exit}} < d_{t,i} < \tau_{\mathrm{enter}}. \end{cases} \tag{13}$$

to reduce repeated switching near a single threshold. Candidate structural changes are then filtered by an update budget over maintained entity rows. Let $E$ denote the number of maintained rows ($E = N$ on the sequence axis and $E = L$ on the depth axis). The selected correction mask satisfies

$$|\Omega_t^\star| \leq \rho_{\max} E, \qquad \rho_{\max} = 0.125. \tag{14}$$

Candidate retrieval remains separately bounded by $|\mathcal{C}_t| \leq C$. Thus the budget limits which entity rows are mutated, not the number of retrieved candidates and not standard Transformer attention or feed-forward computation. The persistent state is finally committed and both forward and reverse indices are updated. The provisional assignments are Top-$k$ sparse, but the released implementation applies EMA directly to selected persistent rows and does not perform a post-EMA hard Top-$k$ projection. Consequently, a persistent row may accumulate support over successive updates; Top-$k$ sparsity applies to the newly computed assignments, while the committed state retains historical mass.

Persistence is therefore defined as

$$\boxed{\text{Persistence} = \text{retention} + \text{reuse} + \text{selective mutation.}} \tag{15}$$

### 2.3 SEQUENCE AND DEPTH AS ONE OPERATOR

Let

$$\mathcal{M} : \mathcal{X} \to \Delta^{K-1} \tag{16}$$

be a normalized membership operator. The sequence and depth cases are instances of

$$\text{Token} \to \text{Chunk}, \qquad \text{Layer} \to \text{Block}. \tag{17}$$

Although the compatibility functions differ, both induce A(M) = MM^T and share the same persistent maintenance lifecycle. The abstraction is shared state maintenance, not identical optimization behavior.

## 3 THEORY

We give a compact set of conditional formal properties that characterize the structural state representation and its induced relation. The results are intentionally scoped: they establish representation-level properties under explicit assumptions, not empirical performance guarantees, and they do not assume that those assumptions necessarily hold for natural-language streams.

**Proposition 1 (topology construction).** For any membership matrix $M$, $A(M) = MM^\top$ is symmetric positive semidefinite. In particular,

$$v^\top A(M) v = \left\| M^\top v \right\|_2^2 \geq 0. \tag{18}$$

For nonnegative row-normalized membership $M$, $A_{ij}$ is the inner product of the membership profiles of entities $i$ and $j$. We use this matrix as an analytical, membership-induced structural affinity; the implementation does not materialize the dense adjacency matrix.

**Proposition 2 (membership-to-topology stability).** For compatible membership matrices $M, M'$,

$$\left\|MM^\top - M'M'^\top\right\|_F \le (\|M\|_F + \|M'\|_F)\,\|M - M'\|_F\,. \tag{19}$$

Thus a bounded perturbation of membership produces a bounded perturbation of the induced topology. This statement is algebraic and does not assume that membership itself changes smoothly with the input.

**Proposition 3 (fixed-state local sensitivity).** For this proposition, define the membership-induced topology as $G(S,x) := A(M(S,x)) = M(S,x)M(S,x)^\top$. Let $M(S,x)$ denote the membership assignment produced from representation $x$ with structural state $S$ held fixed. Within a region in which the selected Top-$k$ support is unchanged, suppose the assignment map is $L_M$-Lipschitz:

$$\|M(S,x) - M(S,x')\|_F \le L_M\,\|x - x'\|_2\,. \tag{20}$$

Then the corresponding induced topologies satisfy

$$\|G(S,x) - G(S,x')\|_F \le (\|M(S,x)\|_F + \|M(S,x')\|_F)L_M\,\|x - x'\|_2\,. \tag{21}$$

The fixed-state qualification is essential: this proposition does not absorb the separate state-transition difference $\mathcal{U}_\theta(S,x') - \mathcal{U}_\theta(S,x)$ into the bound. At Top-$k$ support boundaries, the local smoothness assumption may fail, and the maintenance mechanism is instead responsible for detecting and correcting such structural changes.

**Proposition 4 (budgeted structural maintenance).** Suppose candidate retrieval returns at most $C$ candidates per maintained query and the correction policy selects at most a fraction $\rho_{\max}$ of the $E$ maintained entity rows on the axis. Then

$$N_t^{\mathrm{upd}} \le \rho_{\max} E, \tag{22}$$

and the structural-maintenance cost can be written as

$$C_{\mathrm{maint}} = C_{\mathrm{retrieve}} + O(Cd) + O(\rho_{\max} E)c_{\mathrm{corr}}, \tag{23}$$

where $d$ is the representation dimension and $c_{\mathrm{corr}}$ is the per-correction cost. This is an accounting bound for structural maintenance, not a bound on total Transformer inference time.

**Hysteresis and dual-axis scope.** The enter/exit rule with $\tau_{\mathrm{exit}} < \tau_{\mathrm{enter}}$ creates an explicit retention interval: scores inside the interval do not trigger either transition. This suppresses switching caused by small score fluctuations that remain inside the margin, but it does not prevent switching when a genuine change crosses the thresholds. The sequence and depth axes are dual instances of the same normalized membership-state abstraction, not a claim that their optimization behavior is identical.

These results are deliberately conditional. They establish properties of the specified operator under stated assumptions; they do not establish that natural-language streams always satisfy locality, smoothness, or bounded-change conditions, nor do they imply constant end-to-end inference complexity.

# 4 Experimental Setup

## 4.1 Models and training

The primary detailed experiments use a 150M-parameter Transformer with 24 layers, hidden size 512, 8 attention heads, 2 KV heads, FFN dimension 1536, and a 4096-token training context for the primary training runs. The model configuration supports a maximum sequence length of 32768. All primary models are trained with a 4096-token context; therefore, the 4K evaluation is within the training length, whereas the 8K, 16K, and 32K evaluations test length extrapolation beyond the training context. The reference optimizer is AdamW with $\beta_1 = 0.9$, $\beta_2 = 0.95$, weight decay 0.1, gradient clipping 1.0, BF16 arithmetic, cosine learning-rate decay, and 2% warmup. The 150M learning rate is $3 \times 10^{-4}$.

The structural configuration uses $K_S = 1024$ sequence structures, top-$k = 2$ sequence membership, four depth structures, top-$k = 2$ depth membership, EMA coefficient $\eta = 0.1$, hysteresis thresholds $(0.15, 0.08)$, correction budget $\rho_{\max} = 0.125$, and refresh interval 2048 cycles.

The broader archive contains 150M and 3B configurations. The detailed ablation and long-context analysis focuses on 150M. A valid 3B run is reported only as a preliminary feasibility check, not as a scaling comparison. The corresponding quantitative details are reserved for the appendix because they are not a matched scale comparison.

### 4.2 Baselines and ablations

The main baseline is a matched Vanilla Transformer. We evaluate:

1. Vanilla;
2. Sequence-only persistent membership;
3. Depth-only persistent membership;
4. Soft Membership;
5. Soft Overlap;
6. Persistent;
7. Persistent + Local Correction;
8. Full-CF (budget-controlled);
9. No-depth, No-sequence, and No-hysteresis controls.

All principal 150M ablations use the same base Transformer architecture, approximately 200.08M training tokens, seed 17, one GPU, and the same optimizer configuration; the controlled variants differ in the structural mechanism being ablated. The approximately 0.20B-token budget is intentionally used as a controlled architectural-probing regime rather than as a compute-saturated language-model pretraining run: holding data exposure, optimization, and model scale fixed allows many matched variants to be compared under the same optimization budget and reduces confounding from unequal training exposure. Accordingly, the 150M results are interpreted as evidence about architectural behavior and mechanism viability, not as a claim of converged language-modeling capability. Training PPL and long-context validation PPL are reported separately because they are different evaluation protocols.

### 4.3 Long-context protocol

Standard long-context validation is reported at 4K, 8K, 16K, and 32K. The reported evaluation uses the held-out streaming protocol with document-boundary state resets, and loss is averaged over prediction tokens rather than padding. The 4K point is within the training context, whereas 8K–32K are length-extrapolation evaluations. An additional 102,400-token validation evaluation is reported for Full-CF and Vanilla as a separate extended protocol.

The released reference attention implementation uses causal scaled dot-product attention without an explicit RoPE, ALiBi, or learned absolute positional-embedding module. We use this NoPE configuration as a controlled mechanism-isolation setting, not as a production-model recommendation. Removing an explicit positional module makes the setting useful for studying whether positional and structural organization can emerge from the causal computation and hidden-state dynamics themselves. Prior work shows that causal Transformers without explicit positional encodings can acquire implicit positional information through the network, including information recoverable from hidden representations and attention statistics (Haviv et al., 2022; Chi et al., 2023). We therefore do not interpret NoPE as an absence of positional information; rather, it removes an explicit positional inductive bias so that the proposed persistent structural state can be studied without an additional positional module. Thus the reported 102,400-token comparison is not based on extrapolating a rotary frequency beyond 32,768 positions. The extended result is stored as a separate reported validation artifact rather than generated by the standard 4K–32K driver, so we do not infer a unique positional mechanism or attribute the absolute PPL level to NoPE alone.

Table 1: 150M standard long-context validation PPL for the seed-17 models. Lower is better. Percentage reductions are relative to Vanilla.

| Method | 4K | 8K | 16K | 32K |
|---|---|---|---|---|
| Vanilla | 916.01 | 1328.68 | 1282.72 | 429.40 |
| Full-CF | 463.68 | 749.87 | 750.19 | 239.48 |
| Depth-only | **454.41** | **663.22** | **667.30** | **231.79** |
| Full-CF reduction | 49.38% | 43.56% | 41.52% | 44.23% |
| Depth-only reduction | 50.39% | 50.08% | 47.98% | 46.02% |

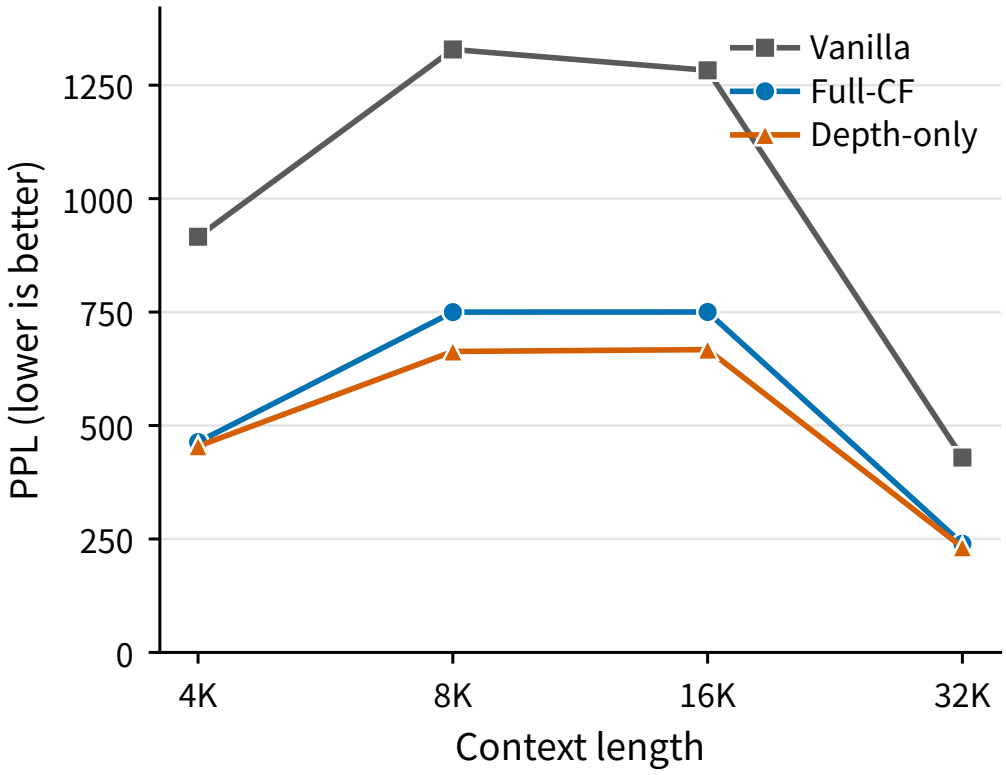


Figure 2: 150M long-context validation PPL at 4K–32K for the seed-17 models. Full-CF reduces PPL relative to Vanilla at every evaluated context, while Depth-only achieves lower PPL at all four points.

## 5 Main Results

### 5.1 Long-context language modeling

Table 1 summarizes the standard long-context PPL results. NLL is the underlying mean negative log-likelihood, and PPL is its exponential transform, $\mathrm{PPL} = \exp(\mathrm{NLL})$. We use NLL as the primary probabilistic quantity and PPL for interpretability; a separate loss metric would be redundant because token-level cross-entropy equals mean NLL under the causal prediction protocol. Exact NLL values are retained in the supplementary evidence record. Full-CF reduces both quantities relative to Vanilla at every reported context length.

The Full-CF reductions are 49.38% at 4K, 43.56% at 8K, 41.52% at 16K, and 44.23% at 32K. The improvement is therefore not restricted to a single context length.

The 32K point is informative but should be interpreted cautiously. The relative Full-CF improvement increases from 41.52% at 16K to 44.23% at 32K, while the absolute Full-CF–Depth-only gap falls

Table 2: Extreme-context evaluation for the reported seed-17 models. The 102,400-token result uses the reported validation protocol and is not treated as a standard context-length point.

| Method | Validation tokens | PPL | Reduction |
|---|---|---|---|
| Vanilla | 102,400 | 60016.43 | – |
| Full-CF | 102,400 | 370.69 | 99.38% |

Table 3: 150M training ablation PPL. All runs use approximately 200.08M tokens and seed 17.

| Variant | PPL | Tok/s | Update ratio |
|---|---|---|---|
| Vanilla | 464.17 | 23723 | 0 |
| Soft Membership | 440.97 | 20353 | 0 |
| Soft Overlap | 449.47 | 20384 | 0 |
| Persistent | 439.07 | 20334 | 0.591 |
| Local Correction | 448.62 | 20328 | 0.606 |
| Full-CF | 453.26 | 20594 | 0.125 |

from 82.884 to 7.69. Because PPL contracts sharply for all three configurations at 32K, the absolute gap contraction is partly a metric-scale effect. We therefore treat it as an observational hypothesis about possible context-dependent sequence-side contribution, not evidence of a causal crossover; Depth-only remains lower-PPL at every standard length.

### 5.2 Extreme-context evaluation

We additionally use the reported 102,400-token validation as an extreme-context evaluation rather than a conventional converged-model benchmark. Vanilla and Full-CF use the same base Transformer computation in this evaluation; the controlled architectural difference is the persistent structural membership mechanism. The reference implementation does not include an explicit positional-encoding module in the attention stack. Full-CF obtains PPL 370.69, compared with 60016.43 for Vanilla, corresponding to a 99.38% reduction (approximately a 161.9x PPL ratio). The absolute Vanilla value should be interpreted cautiously: length extrapolation from training on shorter sequences to evaluation on substantially longer ones is a documented difficulty for Transformers, with positional representations and attention organization playing important roles (Sun et al., 2023; Zhao et al., 2024; Chi et al., 2024). The result is therefore consistent with known extrapolation degradation, but it does not establish that vanilla attention collapses or identify a unique causal source of the degradation. Under the same extended protocol, Full-CF is substantially more robust than Vanilla. This supports the broader long-context hypothesis, but the evaluation does not isolate whether the robustness arises from the sequence axis, depth axis, or their interaction. Because the protocol differs from the standard 4K-32K sweep, we do not place it on the main context-length curve.

## 6 Ablation and Axis Analysis

### 6.1 Incremental structural mechanisms

Table 3 reports the principal A0-A5 150M component-wise ablations (A0 Vanilla -> A1 Soft Membership -> A2 Soft Overlap -> A3 Persistent -> A4 Local Correction -> A5 Full-CF). The component effect is non-monotonic under the fixed training budget: Persistence improves training PPL, whereas local correction and budget control change the quality-maintenance trade-off. The separate axis ablation below compares Sequence-only and Depth-only directly.

The persistent configuration improves training PPL from 464.17 to 439.07, while Depth-only reaches 429.85. The Sequence-only configuration reaches 469.10 and therefore does not provide a standalone training-PPL advantage under the current budget; its available long-context results are reported in Appendix Table 8 and are descriptive from a single run. This result should not be interpreted as evidence that sequence structure is intrinsically ineffective; rather, it establishes an important asymmetry

in the current regime. Sequence-only also records substantially more structural maintenance activity than Depth-only under the same training protocol, with 512 corrections versus 2. This difference is consistent with the larger structural domain of the sequence axis, although the present ablation does not isolate optimization difficulty as the causal explanation.

### 6.2 Depth–sequence asymmetry

The strongest individual 150M configuration on the standard 4K–32K validation is Depth-only. This is scientifically useful because it separates the two proposed structural axes rather than attributing all gains to the dual-axis design.

One possible explanation is that depth-side structure is easier to optimize in the current setting because the structural domain contains only a small number of layers and virtual blocks, whereas sequence-side structure must organize a substantially larger and more context-dependent set of token relationships. The higher Sequence-only maintenance activity is consistent with this explanation, but it does not identify optimization difficulty as the cause.

At the same time, Full-CF retains a large advantage over Vanilla and becomes much closer to Depth-only at 32K. Overall, depth-side persistence provides the strongest immediate benefit in the standard regime. The narrowing Full-CF-Depth-only gap is consistent with a context-dependent sequence-side contribution, but the current ablations do not isolate that contribution. The 102.4K evaluation therefore supports robustness of the full persistent-state configuration under the extended protocol, not a causal attribution to the sequence axis.

### 6.3 Robustness across seeds

Five Full-CF robustness runs (seeds 101, 137, 173, 199, 223) have training-PPL mean 452.11 and standard deviation 4.09. This provides seed-level training robustness evidence; no independent long-context sweep is claimed for every seed.

## 7 Systems Behavior

Vanilla reaches 23.72K tokens/s versus 20.59K for Full-CF (13.2% lower throughput). Sequence-only reaches 20.37K tok/s versus 20.69K for Depth-only, suggesting sequence-side computation contributes to incremental cost, but this is not a causal kernel decomposition. Full-CF and Depth-only have similar throughput despite 513 versus 2 corrections; end-to-end throughput also depends on fixed attention/FFN work, memory traffic, and synchronization, so correction count need not scale with runtime. We make no wall-clock speedup, multi-GPU scaling, or monotonic context-length overhead claim; matched DenseFormer/Mixture-of-Depths baselines are also absent.

## 8 Limitations and Future Work

The detailed analysis is concentrated at 150M; the approximately 0.20B-token budget is a controlled architectural-probing regime rather than compute-saturated pretraining. Evaluation uses NLL/PPL under matched protocols, with structural diagnostics reported separately; downstream tasks are not evaluated. NoPE is used for mechanism isolation, not as an absence of positional information; interaction between positional information and persistent structural state is not independently isolated. Sequence-only has one complete 4K-32K run; the 102,400-token protocol omits Depth-only; matched DenseFormer/Mixture-of-Depths and systems-scaling controls are absent. The causes of Sequence-only collapse and Depth-only superiority under the current budget remain open questions; future work can address these attribution limits with downstream long-context benchmarks, positional-encoding variants, and GPU-efficient structural execution.

## 9 Conclusion

We introduced ContinuumFormer, a Transformer architecture based on persistent soft structural membership across sequence and depth. The state evolves incrementally and induces a weighted structural topology that parameterizes sparse structural reconstruction, with conditional formal properties under explicit assumptions. Controlled experiments show improved long-context PPL over matched Vanilla and asymmetric axis behavior: Depth-only provides the strongest immediate benefit in the standard regime, while longer-context observations are consistent with increasing sequence-side relevance. The available evidence does not causally isolate the sequence axis; the 102.4K Full-CF result supports the broader long-context robustness hypothesis for the full persistent-state configuration.

## Reproducibility Statement

The reported environment is Python 3.11.16, PyTorch 2.13.0+cu130, CUDA 13.0, Linux x86-64, BF16, FineWeb-Edu streaming, and the Qwen/Qwen2.5-3B tokenizer. Principal 150M ablations use 200.08M tokens, seed 17, one GPU, and the fixed optimizer/structural configuration summarized in Appendix A; the robustness study uses five additional seeds. The supplementary artifact preserves anonymous source code, configurations, environment manifests, run records, and regeneration scripts, with training PPL and long-context validation kept as distinct evidence types.

## AI Use Statement

Generative AI tools were used as auxiliary research and writing tools for hypothesis refinement, experiment/methodology feedback, result analysis and interpretation, theorem and mathematical-claim consistency checking, manuscript organization, language refinement, scientific figure preparation/editing, and code/document review. The authors determined the research questions, architecture, theoretical assumptions, theorem selection, experimental execution, data collection, result verification, and final scientific conclusions. AI-assisted suggestions were treated as hypotheses or review inputs, not authoritative evidence. The authors independently verified all mathematical statements, assumptions, experimental configurations and results, numerical values, citations, and final manuscript content, and take full responsibility for the scientific validity and integrity of the work.

## References


Daniel Bolya, Cheng-Yang Fu, Xiaoliang Dai, Peizhao Zhang, Christoph Feichtenhofer, and Judy Hoffman. Token merging: Your ViT but faster. In *International Conference on Learning Representations*, 2023.

Ta-Chung Chi, Ting-Han Fan, Li-Wei Chen, Alexander Rudnicky, and Peter Ramadge. Latent positional information is in the self-attention variance of transformer language models without positional embeddings. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pp. 1183–1193. Association for Computational Linguistics, 2023.

Ta-Chung Chi, Ting-Han Fan, and Alexander I. Rudnicky. Attention alignment and flexible positional embeddings improve transformer length extrapolation. In *Findings of the Association for Computational Linguistics: NAACL 2024*, pp. 132–148, 2024.

Zihang Dai, Zhilin Yang, Yiming Yang, Jaime Carbonell, Quoc V. Le, and Ruslan Salakhutdinov. Transformer-xl: Attentive language models beyond a fixed-length context. *ACL*, 2019.

William Fedus, Barret Zoph, and Noam Shazeer. Switch transformers: Scaling to trillion parameter models with simple and efficient sparsity. *Journal of Machine Learning Research*, 23:1–39, 2022.

Albert Gu and Tri Dao. Mamba: Linear-time sequence modeling with selective state spaces. In *First Conference on Language Modeling*, 2024.

Adi Haviv, Ori Ram, Ofir Press, Peter Izsak, and Omer Levy. Transformer language models without positional encodings still learn positional information. In *Findings of the Association for Computational Linguistics: EMNLP 2022*, pp. 1382–1390. Association for Computational Linguistics, 2022.

Sukjun Hwang, Brandon Wang, and Albert Gu. Dynamic chunking for end-to-end hierarchical sequence modeling. In *International Conference on Learning Representations*, 2026.

Kimi Team et al. Attention residuals. *arXiv preprint arXiv:2603.15031*, 2026.

Tsendsuren Munkhdalai, Manaal Faruqui, and Siddharth Gopal. Leave no context behind: Efficient infinite context transformers with infini-attention. *arXiv preprint arXiv:2404.07143*, 2024.

Matteo Pagliardini, Amirkeivan Mohtashami, Francois Fleuret, and Martin Jaggi. Denseformer: Enhancing information flow in transformers via depth weighted averaging. In *Advances in Neural Information Processing Systems*, 2024.

Artidoro Pagnoni, Ramakanth Pasunuru, Pedro Rodriguez, John Nguyen, Benjamin Muller, Margaret Li, Chunting Zhou, Lili Yu, Jason E. Weston, Luke Zettlemoyer, Gargi Ghosh, Mike Lewis, Ari Holtzman, and Srini Iyer. Byte latent transformer: Patches scale better than tokens. In *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 9238–9258. Association for Computational Linguistics, 2025.

Jack W. Rae, Anna Potapenko, Siddhant M. Jayakumar, Chloe Hillier, and Timothy P. Lillicrap. Compressive transformers for long-range sequence modelling. In *International Conference on Learning Representations*, 2020.

David Raposo, Sam Ritter, Blake Richards, Timothy P. Lillicrap, Peter Conway Humphreys, and Adam Santoro. Mixture-of-depths: Dynamically allocating compute in transformer-based language models. *arXiv preprint arXiv:2404.02258*, 2024.

Yutao Sun, Li Dong, Barun Patra, Shuming Ma, Shaohan Huang, Alon Benhaim, Vishrav Chaudhary, Xia Song, and Furu Wei. A length-extrapolatable transformer. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 14590–14604, 2023.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. In *Advances in Neural Information Processing Systems*, 2017.

Yuhuai Wu, Markus N. Rabe, DeLesley Hutchins, and Christian Szegedy. Memorizing transformers. *International Conference on Learning Representations*, 2022.

Liang Zhao, Xiachong Feng, Xiaocheng Feng, Weihong Zhong, Dongliang Xu, Qing Yang, Hongtao Liu, Bing Qin, and Ting Liu. Length extrapolation of transformers: A survey from the perspective of positional encoding. In *Findings of the Association for Computational Linguistics: EMNLP 2024*, pp. 9959–9977. Association for Computational Linguistics, 2024.

# A Reference Configuration

Table 4: Reference structural and optimization hyperparameters.

| Parameter | Value |
|---|---|
| Sequence structures | 1024 |
| Sequence top-$k$ | 2 |
| Depth structures | 4 |
| Depth top-$k$ | 2 |
| EMA coefficient $\eta$ | 0.1 |
| Wake threshold $\tau_{\text{enter}}$ | 0.15 |
| Exit threshold $\tau_{\text{exit}}$ | 0.08 |
| Correction budget $\rho_{\max}$ | 0.125 |
| Refresh interval | 2048 cycles |
| Precision | BF16 |
| Optimizer | AdamW |
| $\beta_1, \beta_2$ | 0.9, 0.95 |
| Weight decay | 0.1 |
| Gradient clipping | 1.0 |
| Warmup | 2% |

## B Additional Training Metrics

This appendix reports the structural-maintenance measurements recorded during the principal 150M training runs. The metrics complement the main-text PPL results by exposing how often the persistent state is updated, how many structural corrections are performed, and the observed retrieval cost. The tables below report the recorded values without introducing additional evaluation claims.

Table 5: Selected structural diagnostics from the 150M training runs. "Corrections" counts selected entity-row corrections; "Active" records the corresponding active maintenance entries in the current logging implementation, so the two counters coincide for these runs.

| Variant | PPL | Update ratio | Corrections | Entropy | Active | Retrieval (ms) |
|---|---|---|---|---|---|---|
| Vanilla | 464.17 | 0 | 0 | 0 | 0 | 0 |
| Persistent | 439.07 | 0.5910 | 2435 | 0.467 | 2435 | 4.073 |
| Local Correction | 448.62 | 0.6056 | 2495 | 0.590 | 2495 | 4.376 |
| Full-CF | 453.26 | 0.125 | 513 | 0.505 | 513 | 4.262 |
| Depth-only | **429.85** | 0.0833 | 2 | 0.602 | 2 | 1.841 |
| Sequence-only | 469.10 | 0.1250 | 512 | 0.598 | 512 | 2.746 |

Table 6: Full-CF seed robustness at 150M; detailed per-seed systems measurements.

| Seed | PPL | Tok/s | Update ratio | Retrieval (ms) |
|---|---|---|---|---|
| 101 | 450.94 | 20236 | 0.1250 | 3.754 |
| 137 | 452.24 | 19681 | 0.1250 | 4.574 |
| 173 | 451.81 | 19753 | 0.1248 | 4.357 |
| 199 | 447.10 | 19472 | 0.125 | 22.734 |
| 223 | 458.46 | 19700 | 0.1250 | 4.378 |
| Mean | 452.11 | – | – | – |
| Std. | 4.09 | – | – | – |

Table 7: 150M axis ablation under the current seed-17 training regime.

| Variant | PPL | Update ratio | Corrections |
|---|---|---|---|
| Vanilla | 464.17 | 0 | 0 |
| Sequence-only | 469.10 | 0.125 | 512 |
| Depth-only | **429.85** | 0.0833 | 2 |
| No Sequence | 432.36 | 0.0833 | 2 |
| No Depth | 468.32 | 0.125 | 512 |
| No Hysteresis | 467.19 | 0.125 | 515 |

Table 8: Sequence-only long-context validation PPL from the available complete run. A second independent 4K–32K run is unavailable, so no variance estimate is reported.

| Method | 4K | 8K | 16K | 32K |
|---|---|---|---|---|
| Sequence-only | 158891.20 | 952852.81 | 789771.00 | 72483.17 |

Table 9: Training throughput and peak GPU memory for selected configurations; the 150M rows are matched under the principal training protocol.

| Configuration | Train tok/s | Peak GPU memory (GiB) |
|---|---|---|
| 150M Vanilla | 23,723 | 9.60 |
| 150M Full-CF | 20,594 | 9.65 |
| 150M Depth-only | 20,691 | 9.65 |
| 150M Sequence-only | 20,365 | 9.66 |
| 3B Full-CF | 14,486 | 68.86 |

The Sequence-only values show that the variant does not establish a competitive standalone long-context result in the available run. Because only one complete run is available, these values are reported descriptively rather than used for a variance-based comparison.

## C Larger-Scale Feasibility Validation

We additionally evaluate a 3B-parameter Full-CF model as a larger-scale feasibility check. The completed run processes approximately 1.00B training tokens and reaches a training PPL of 121.54. On the reported validation evaluation containing 1,003,520 tokens, it obtains an NLL of 4.8901 and a PPL of 132.96. This experiment is intended to verify that the proposed structural mechanism remains operational at substantially larger model scale; it is not a matched scaling comparison because an independently trained 3B Vanilla baseline is not available.

Table 10: Larger-scale feasibility validation for Full-CF.

| Metric | 3B Full-CF |
|---|---|
| Parameters | 2.999B |
| Training tokens | 1,000,341,504 |
| Training PPL | 121.54 |
| Validation tokens | 1,003,520 |
| Validation NLL | 4.8901 |
| Validation PPL | 132.96 |
| Train tok/s | 14,485.5 |
| Peak GPU memory | 68.86 GiB |

## D Data Interpretation Notes

The consolidated training PPL is distinct from standard long-context validation PPL. Run1 and Run2 for Vanilla, Full-CF, and Depth-only reproduce the reported long-context PPL/NLL values pointwise. The 102,400-token Vanilla/Full-CF result is the source of the approximately 60K versus 370 PPL comparison; the available record identifies it as a 102,400-token validation protocol. The larger-model validation reported in the main text is intentionally described as a preliminary feasibility check rather than a full scaling study.

## E Formal Assumptions and Proof Details

The main text deliberately keeps the theoretical contribution compact. This appendix makes explicit the assumptions needed by Propositions 1–4 and gives proof details sufficient to audit their scope. None of these assumptions is intended as a claim that natural-language streams satisfy the corresponding regularity condition globally.

### E.1 Assumptions used by the four propositions

**A1: Valid membership.** Provisional assignment matrices have finite dimensions, nonnegative entries, and row sums equal to one, so each provisional row lies in the probability simplex. The committed EMA state remains nonnegative but need not remain row-normalized. The nonnegativity part is used when interpreting $A(M)$ as a nonnegative affinity; the algebraic positive-semidefinite statement itself does not require nonnegativity.

**A2: Fixed-state local regularity.** For Proposition 3, the persistent structural state $S$ is held fixed and the selected Top-$k$ support remains unchanged in the neighborhood considered. The assignment map satisfies

$$\|M(S,x) - M(S,x')\|_F \le L_M \|x - x'\|_2. \tag{24}$$

This is a local condition. Hard Top-$k$ support changes are structural events rather than points at which the local smoothness statement is assumed to continue to hold.

**A3: Bounded candidate retrieval.** At a maintenance cycle, Retrieve returns at most $C$ candidates per maintained structural query. This is an implementation-level bound on the candidate set used by the maintenance analysis.

**A4: Budgeted correction policy.** Let $E$ be the number of maintained entity rows on an axis. The correction policy selects no more than a fraction $\rho_{\max}$ of these rows in a maintenance cycle, i.e., $|\Omega_t^\star| \le \rho_{\max} E$. Candidate retrieval is separately bounded by A3.

**A5: Hysteresis margin.** The enter and exit thresholds satisfy $\tau_{\text{exit}} < \tau_{\text{enter}}$, and the wake decision receives the previous activation state. Scores above the enter threshold activate, scores below the exit threshold deactivate, and scores inside the interval retain the previous decision. This defines a retention interval; it does not imply that arbitrary perturbations cannot eventually cross a threshold.

Table 11: Assumption-to-proposition dependency for the formal claims in the main text.

| Result | Required condition | Scope |
|---|---|---|
| Prop. 1 | finite $M$; A1 for nonnegativity | topology construction |
| Prop. 2 | compatible finite $M, M'$ | membership perturbation |
| Prop. 3 | A2 | fixed-state local sensitivity |
| Prop. 4 | A3–A4 | row-level maintenance count and cost |
| Hysteresis rule | A5 | threshold retention only |

### E.2 Proof details for Propositions 1–4

**Proposition 1.** For any compatible $M$, $A(M) = MM^\top$ satisfies

$$A(M)^\top = (MM^\top)^\top = MM^\top. \tag{25}$$

For any vector $v$,

$$v^\top A(M) v = v^\top MM^\top v = \|M^\top v\|_2^2 \ge 0. \tag{26}$$

Hence $A(M)$ is symmetric positive semidefinite. Under A1, $A_{ij} = \sum_k M_{ik} M_{jk} \ge 0$, so the induced affinity is also nonnegative. The result is an algebraic property of the induced object; it does not require materializing the dense matrix in the implementation.

**Proposition 2.** Add and subtract $M'M^\top$:

$$MM^\top - M'M'^\top = (M - M')M^\top + M'(M - M')^\top. \tag{27}$$

Taking Frobenius norms and applying triangle inequality and submultiplicativity gives

$$\|MM^\top - M'M'^\top\|_F \le (\|M\|_F + \|M'\|_F)\|M - M'\|_F. \tag{28}$$

Therefore membership perturbations induce bounded topology perturbations whenever the matrices remain in a bounded finite-dimensional region. The proposition does not state that the input-to-membership map is itself smooth.

**Proposition 3.** Within the fixed-support region of A2, apply the membership perturbation bound from Proposition 2 to $M(S, x)$ and $M(S, x')$, then substitute the local Lipschitz condition in A2:

$$\|G(S, x) - G(S, x')\|_F \leq \big(\|M(S, x)\|_F + \|M(S, x')\|_F\big) \tag{29}$$

$$\cdot L_M \|x - x'\|_2. \tag{30}$$

The state transition is intentionally excluded from this bound. Thus the result describes local sensitivity with $S$ fixed, while structural changes caused by state updates are handled by the maintenance mechanism.

**Proposition 4.** By A3, at most $C$ candidates enter a maintenance cycle. By A4, at most $\rho_{\max} E$ entity rows are corrected. If retrieval costs $C_{\text{retrieve}}$, candidate scoring costs $O(Cd)$, and each correction costs at most $c_{\text{corr}}$, summing these terms gives

$$C_{\text{maint}} = C_{\text{retrieve}} + O(Cd) + O(\rho_{\max} E) c_{\text{corr}}. \tag{31}$$

This is a bound on the structural-maintenance component only. Attention, MLP, communication, memory traffic, kernel-launch, and other Transformer costs remain outside this accounting expression.

**Forward-pass interface.** The induced topology is an analytical representation rather than a dense tensor injected into the attention logits. In the reference implementation, the structural operators condition the post-stack hidden representation through sparse reconstruction. Let $H \in \mathbb{R}^{N \times d}$ denote the final hidden states after the ordinary causal Transformer stack, and let $Z \in \mathbb{R}^{L \times d}$ collect the mean-pooled hidden state of each layer. The sequence and depth routers are learned linear maps,

$$r^S = H W_S^\top, \qquad r^D = Z W_D^\top, \tag{32}$$

followed by candidate-restricted normalized Top-$k$ assignments $M^S$ and $M^D$. For a persistent stream, let $B_t^S$ and $B_t^D$ denote the previously committed sequence and depth structural summaries. The sequence-side reconstruction is

$$\widehat{H}^S = M^S B_t^S, \tag{33}$$

while the depth-side reconstruction is formed for every layer and the final-layer reconstruction is used for the language-model output,

$$\widehat{Z}^D = M^D B_t^D, \qquad \widehat{H}^D = \widehat{Z}^D_{L:}. \tag{34}$$

When correction is enabled, selected rows are gated by the maintenance state and the output hidden representation is

$$H_{\text{out}} = H + \eta \left[ A^S (\widehat{H}^S - H) + A^D (\widehat{H}^D - H) \right], \tag{35}$$

where A^S and A^D are the selected-row gates inherited from the committed activation state. The language-model head then operates on Hout. On the first state initialization, no previous structural summaries exist, so the correction term is absent; subsequent compatible updates reuse the committed summaries. When correction is enabled, the membership tensors remain on the differentiable path to the language-model loss and therefore train the learned router parameters; the committed runtime state is detached before persistence. Thus the induced topology describes structural relations, while the executed forward operation is sparse structural reconstruction followed by residual correction. The depth router is evaluated once per forward pass from mean-pooled layer representations, and the physical layer order is unchanged. Thus A(M) is not itself an attention-logit bias, block-sparse attention mask, or layer-skipping operator. This explicit interface also clarifies that the theoretical topology bounds concern the structural object; they do not by themselves bound attention activations or loss perturbations.

### E.3 Budgeted maintenance lifecycle

The implementation realizes the conceptual lifecycle used in the main text as follows. Retrieve obtains a bounded candidate set from the current structural bank; Score computes the learned compatibility scores for those candidates; Assign converts them into sparse normalized membership; Detect computes the row-wise membership change score; Wake applies the enter/exit hysteresis rule using the previous activation state; Correct selects the largest eligible entity-row changes under the structural update budget and applies the configured EMA correction; Commit rebuilds the forward and reverse structural indices and records diagnostics. The following pseudocode abstracts these operations without assuming a particular retrieval backend.

**Algorithm 1** Budgeted persistent structural maintenance

**Require:** Previous state $S_t$, current representations $x_{t+1}$, budget $\rho_{\max}$

$C_t \leftarrow \textsc{Retrieve}(S_t, x_{t+1})$
$r_{t+1} \leftarrow \textsc{Score}(x_{t+1}, C_t)$
$M_{t+1} \leftarrow \textsc{Assign}(r_{t+1}, C_t)$
$d_{t,i} \leftarrow \frac{1}{2}\|M_{t+1,i:} - M_{t,i:}\|_1$ ▷ per entity row $i$
$a_{t,i} \leftarrow \textsc{Wake}(d_{t,i}, a_{t-1,i}, \tau_{\text{enter}}, \tau_{\text{exit}})$
$a_t \leftarrow (a_{t,i})_{i=1}^{E}$
**if** budget control is enabled **then**
    $a_t \leftarrow \textsc{TopFraction}(d_t, a_t, \rho_{\max})$
**end if**
$S'_t \leftarrow \textsc{Correct}(S_t, M_{t+1}, a_t, \eta)$
update forward/reverse indices from $S'_t$
record changed-row count, candidate count, and topology diagnostics
**return** $S_{t+1} \leftarrow \textsc{Commit}(S'_t)$

The sequence and depth axes invoke the same maintenance pattern with different representations and structural-bank sizes. In the released implementation, the sequence axis uses 1024 structures with Top-$k = 2$, while the depth axis uses four structures with Top-$k = 2$. The physical order of Transformer layers is unchanged by the depth membership state; the state only represents layer-to-structure association.

### E.4 Persistent-state lifecycle and reset semantics

Persistence is scoped to a stream identity. The implementation stores the membership matrices, structural summaries, forward indices, reverse indices, stream identifiers, and maintenance diagnostics as runtime state rather than model parameters. A state is reused only when batch size, hidden dimension, and stream identifiers remain compatible. When the stream identity changes or the state shape becomes incompatible, the state and associated retrieval index are cleared before a new state is initialized. This prevents structural candidates from leaking between independent streams.

The persistent membership is updated by the configured EMA coefficient $\eta = 0.1$ on selected rows. The correction budget therefore limits which structural entity rows are mutated during maintenance; it does not limit the dense Transformer forward pass. Because the EMA is not followed by hard Top-$k$ projection, committed membership can retain historical support beyond the Top-$k$ support of each provisional assignment. The released tests include checks for state reset on stream changes and shape changes, retrieval-cache isolation, budget limits, variant semantics, and state-shape consistency.

### E.5 Complexity and implementation scope

The formal maintenance accounting should be read as a decomposition rather than as an end-to-end complexity claim. Candidate retrieval and scoring operate on structural representations; local correction updates only the selected structural rows and corresponding summaries. The implementation keeps forward and reverse indices explicitly and does not construct $MM^\top$ as a dense runtime adjacency matrix. The measured training results nevertheless show overhead from retrieval and irregular maintenance, so the paper makes no claim of wall-clock acceleration.

The current reference implementation uses an exact GPU dot-product retrieval backend for the main training path, while the package also contains a local-cached approximate backend and retrieval-sensitivity utilities. These are implementation options, not additional modeling contributions. The reported main results should be reproduced with the frozen reference configuration rather than by substituting a different retrieval backend. No matched DenseFormer or Mixture-of-Depths run is included in the released empirical results; the Depth-only result should therefore be interpreted as an internal axis ablation rather than evidence that persistent depth maintenance dominates dedicated depth-aggregation or conditional-computation baselines.

### E.6 Reproducibility mapping

Table 12: Mapping from paper concepts to the released implementation.

| Paper concept | Implementation realization |
|---|---|
| Persistent state | Runtime state containing membership, summaries, stream IDs, and structural indices |
| Sequence axis | Sequence router, 1024 structures, Top-$k = 2$ |
| Depth axis | Depth router, 4 structures, Top-$k = 2$ |
| Retrieve | Retriever interface with per-stream/per-axis index keys |
| Assign | Candidate-restricted sparse softmax membership |
| Detect | Half-$L_1$ membership change score |
| Wake | Enter/exit thresholds $(0.15, 0.08)$ |
| Correct | EMA update with $\eta = 0.1$ on selected rows |
| Budget | Top-fraction selection with $\rho_{\max} = 0.125$ |
| Commit | Forward/reverse index rebuild and diagnostic update |
| Reset | Stream-ID or shape incompatibility clears state and retrieval cache |

## F Additional Figures and Diagnostics

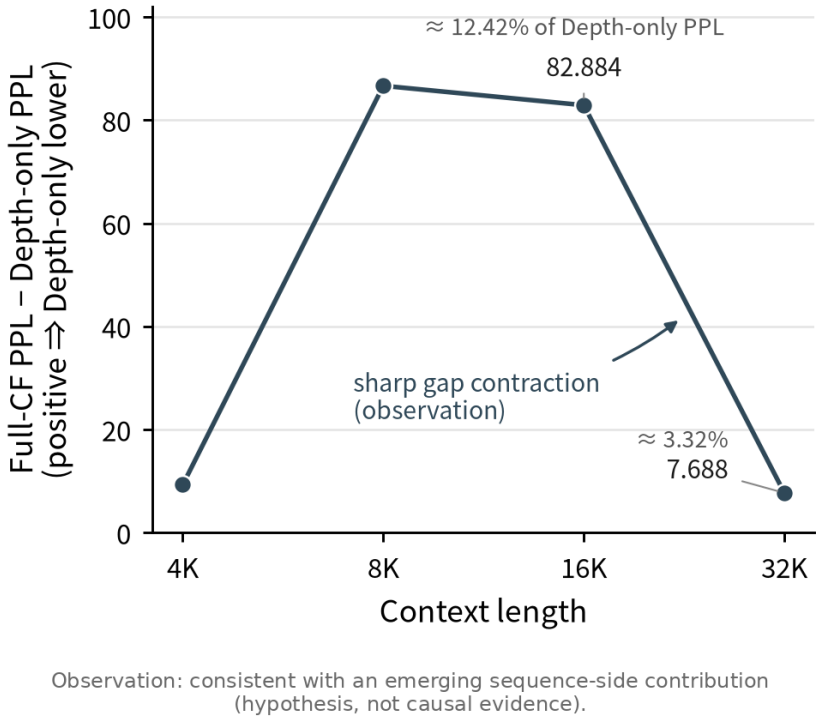


Figure 3: Absolute PPL gap between Full-CF and Depth-only across context length. The sharp contraction from 16K to 32K is an observation, not proof of a causal sequence-axis crossover.

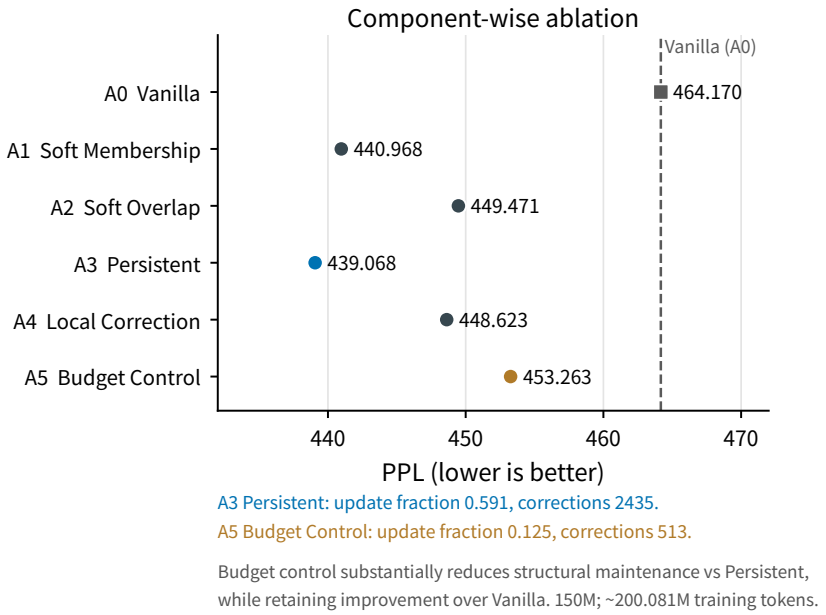


Figure 4: Component-wise 150M training ablation. The figure separates soft membership, overlap, persistence, local correction, and budget control under the reported training protocol.

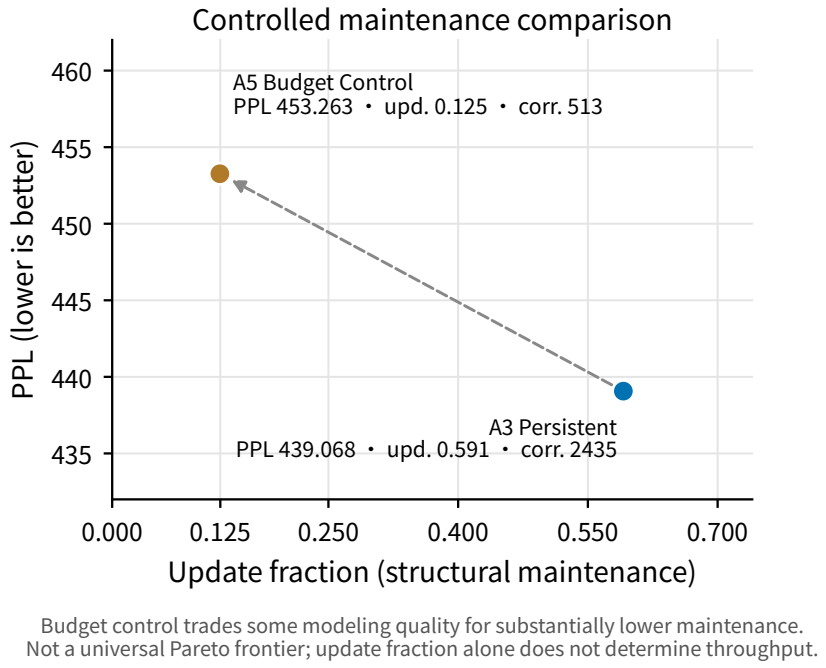


Figure 5: Measured quality–maintenance trade-off for Persistent and budgeted Full-CF. The plotted configurations are discrete measurements; the figure is not a hardware Pareto frontier.