Binding and Selection

A two-mechanism architecture for an open-ended language system on modest hardware: a small Transformer that learns to bind, and a selectively-forgetting memory that carries what it knows

Marcel Langjahr

Pantareon GmbH
https://pantareon.io/
langjahr@pantareon.io

June 2026

Abstract

We separate two capabilities that a language model conflates and that a count-based substrate cannot both provide. The first is *sequential binding*: connecting content across distance—subject to verb, pronoun to referent, a narrative thread across a paragraph. The second is open-ended *knowledge selection*: keeping what is worth keeping from a stream that never ends, without unbounded growth. A companion study establishes, by holding architecture fixed and varying only the memory regime, that a variable-order context graph perfects local statistics and yet meets a ceiling exactly at binding—at 11.3M characters it reaches 1.83 held-out bits-per-character and emits orthographically perfect, locally fluent text with no long-range coherence. We argue that this ceiling has two orthogonal faces, that the brain solves them with two different *kinds* of mechanism rather than one refined representation, and we propose an architecture that does the same: a small decoder Transformer (the *binder*), trained from scratch on a curated, capacity-matched corpus, learns the binding mechanic in a critical-period-like window; a selectively-forgetting count graph (the *counter*) carries open-ended knowledge as a datastore keyed on the binder's own representations, read back by retrieval. Forgetting is not housekeeping but metabolism: in a system whose representations drift, selective forgetting is the only way the memory stays aligned and finite while everything else moves. This is a design and hypotheses paper. We state precisely what is already established (the substrate's ceiling; forgetting as a sample-efficiency regulariser; a gradient-verified from-scratch binder), what is hypothesised (the value of the composition; forgetting as self-healing against representational drift), and what is open (the drift rate; continual competence acquisition). The empirical sections carry explicit placeholders; their results are forthcoming on stronger hardware and will be folded into this document as they arrive.

Scope of this paper: This is a design paper, not a results paper. Its purpose is to model an architecture and justify it from measured precursors and prior work, and to fix, in advance, the experiments that would confirm or falsify it. One artifact exists and is verified: a zero-dependency, from-scratch decoder Transformer whose hand-written backward pass passes a central-difference gradient check (max relative error 4.8×10^{-5} across all parameter tensors). One control is measured: the same-corpus count-model baseline on TinyStories (1.306 prequential bits/char; locally fluent, narratively incoherent generation). Everything else marked *hypothesised* or *open* below is exactly that. The intended regime is deliberately modest: own weights, no inherited pre-trained model, no datacenter, a curated corpus rather than the open web, and a single workstation. Absolute figures are to be read relative to the stated baselines, as in the companion work. The epistemic status of every claim is summarised in Table 1, which doubles as the scaffold for later revision: as runs complete, *open* rows become *established* rows.

1 Introduction

The substrate behind this work descends from Emergent Graph Theory: function carried by a discrete graph grown from data under a conserved quantity, rather than by a fixed parametric model fitted by global optimisation. Realised as a character-level language model, this is a variable-order context graph with PPM-C backoff [12]: a node is a context, an edge carries the count of a next symbol, prediction is a relaxation down the suffix

chain, and there are no learned weights—the topology is the model. A companion paper studies this substrate directly and reports two things [13]. First, bounded forgetting is a regulariser: an energy-conservation regime (decay-funded deposits, floor pruning, eviction of the lowest-energy contexts at a memory ceiling) lowers held-out bits-per-character whenever capacity exceeds data, and the margin grows as data grows scarcer. Second, and decisively, no memory regime lifts a ceiling that the substrate hits by construction: perfected local statistics produce words that are right and sentences that are almost right, but no meaning that lives beyond the window.

A second companion paper locates language capability itself by a different controlled variation—holding architecture fixed and varying the learning rule—and concludes that the capability lives in gradient credit assignment, not in any substrate's native dynamics [14]. The present paper takes both results as given and asks the constructive question they leave open: if local statistics cannot bind, and binding is what a learned model supplies, what is the smallest, most honest architecture that binds *and* keeps learning over time without retraining—on hardware one person owns?

Two faces of the ceiling. The "meaning ceiling" is not one wall but two orthogonal ones, and conflating them has cost the field clarity. The first is *similarity*, or connectionless generalisation: recognising that two contexts that share no surface suffix nevertheless behave alike, so that evidence can be pooled across them. The second is *sequential binding*: carrying a specific dependency across distance, where the relevant token may sit far back in the current sentence. A count graph with maximum order D is structurally blind to the second: its window *is* D, and a dependency longer than D cannot be observed, let alone represented. Sparse codes, locality hashing, and spectral embeddings—all the machinery one might bolt on—address the first face only. The second is what attention does, and only what attention does: content-addressed reading across distance.

2 Two mechanisms for two axes

The separation is not an artifact of our substrate; the brain exhibits it, and solves the two faces with two different kinds of mechanism. The *similarity* axis is carried by distributed, overlapping codes: a concept is a sparse pattern over many units, similar concepts overlap, and overlap *is* similarity. This is the cortical, representational-geometry picture, and it emerges from Hebbian learning over overlapping inputs more or less for free. It is, in essence, the sparse-distributed-representation idea the field has long circled [10].

The *binding* axis is not a richer static representation but a *temporally dynamic* mechanism. The correlation theory of brain function [9]—binding by synchrony—proposes that features of one object are bound by firing in temporal coincidence, the synchrony itself the label "these belong together," avoiding the combinatorial explosion of a dedicated binding unit. For sequence specifically, a theta–gamma phase code holds ordered items by relative timing within an oscillatory cycle, and phase precession compresses an experienced sequence into a window short enough for local plasticity to capture. And arbitrary relational binding is delegated to a fast, dedicated structure—the hippocampal–entorhinal system—whose computation has been argued to be formally close to the Transformer's attention [8]. The convergence is the point: binding, however implemented, converges on content-addressed reading across distance, and it is *separate machinery* from the cortical similarity code, not a refinement of it. We take this as the licence for a two-mechanism architecture.

A critical-period analogy, used carefully. Humans acquire the binding mechanic of language—grammar, agreement, long-range structure—in an early window, after which native syntax is hard to attain, while *vocabulary* and *fact* remain acquirable for life [7]. This is the same competence/knowledge asymmetry our substrate study draws, now with a temporal signature: the *mechanic* has a window; the *content* does not. The brain does not keep the syntax window open; it closes it and freezes the mechanic, accepting that re-priming is thereafter costly. The analogy is suggestive, not probative—a brain is not a Transformer, and adult plasticity is real if limited—but it reframes our hardest open problem (Section 5) from "keep the binder forever plastic without forgetting" to "prime the mechanic once, freeze it, and extend reach by external memory rather than by re-wiring."

3 The architecture

Table 1: Epistemic status of the paper's claims. This table is the scaffold for revision: as the experiments of Section 6 complete, *open* and *hypothesised* rows are intended to move to *established*, with the evidence column updated to the measured result.

Claim	**Status**	**Evidence / where**
The count substrate perfects local statistics but cannot bind across distance	Established	Companion [13]: 1.83 held-out bpc, no long-range coherence at 11.3M chars
Bounded forgetting is a sample-efficiency regulariser (capacity > data)	Established	Companion [13]: lower held-out bpc at every size, margin grows as data scarcer
On a curated, regular corpus the counter reaches a lower bpc and its incoherence is *exposed* rather than hidden	Established	This work, §6: TinyStories 1.306 prequential bpc; fluent surface, no narrative thread
A from-scratch decoder Transformer with hand-written, gradient-verified backprop can be built zero-dependency	Established	This work: gradient check max rel. err. 4.8×10^{-5}; trains and generates
A small Transformer binds on capacity-matched curated data	Hypothesised (lit.)	TinyStories [4]; our binding test pending (§6, **[pending — run]**)
A count graph keyed on the binder's representations is a useful, online, selectively-forgetting datastore	Hypothesised	Retrieval-augmentation lineage [2, 3]; our composition untested
Selective forgetting keeps the datastore aligned under representational drift (self-healing)	Hypothesised	Argued in §3.1; drift rate unmeasured
The frozen mechanic applies to new knowledge built from familiar representations, with no retraining	Hypothesised	In-context learning as precedent; reach not yet bounded
Representational drift is slower than content recurrence (so migration keeps up)	Open	Measurable; §5, **[pending — drift curve]**
The counter can guide the binder against catastrophic forgetting for *competence* expansion	Open	Research question; §5, **[pending — A/B experiment]**

The architecture is two mechanisms composed as siblings, not as stacked layers.

The binder. A small decoder-only Transformer—multi-head causal attention, pre-norm residual blocks, next-token prediction [1] —trained from scratch on a curated, capacity-matched corpus. The existence proof that this is enough is the TinyStories result: models of a few million parameters produce coherent, grammatical English when the data's complexity is matched to their capacity, whereas the same models on open-web text produce only mush [4]. We treat binding as an *ablatable mechanic*: the question is not "does a large model bind" (it does) but "does a small one, from scratch, on curated data, on modest hardware"—and the test is the read generation, not the bits-per-character, because the companion shows bpc near 1.8–2.2 *while* meaning decays. Our implementation is complete and its gradient is verified; what remains is to run it (Section 6).

The memory. The counter, repurposed. Crucially it need not *be* semantic; it borrows the binder's key. As in kNN language models [2], the datastore is keyed on the model's hidden representation: the binder supplies a meaningful key for free (it is run regardless), and the store contributes what it measurably does well—fast, exact, incremental write and lookup, with a selective-forgetting retention policy. This dissolves the similarity problem we circled for many pages: the metric is not *built*, it is *inherited* from the binder. The counter keeps its spirit (exact, online, sample-efficient, forgetting) but gives up its surface-suffix *addressing*—it becomes a representation-keyed store, not a character-suffix tree.

The composition. The binder runs on raw text and *queries* the memory: either by interpolating its output distribution with a nearest-neighbour lookup [2], or by attending to retrieved chunks [3]. The cortex→hippocampus picture breaks here in an instructive way: the binder receives raw tokens, not the counter's output, as input. The counter is a *side* store the binder reads from, not a foundation it sits on. This is why "counter first" is the wrong ordering: the binder is primed first; the memory is what is added; and in the representation-keyed variant the binder must be present to generate the keys at all.

3.1 Forgetting as metabolism

A finite store that writes over time must delete, or it overflows—but that is the trivial half, satisfiable by any FIFO. The half that matters is specific to a *dynamic* system, one whose keys move

as the binder learns. There, forgetting is coherence maintenance, not space reclamation. Without deletion the store would fill not merely with nodes but with *stale* entries written under representations the binder no longer produces; the system would fail not from exhaustion but from incoherence—a growing layer of dead keys through which the live ones are ever harder to find. Selective forgetting, evicting the cold and non-reactivated, removes exactly these drift-stale entries and keeps the store aligned to the binder's current representation space. A dumb FIFO does not do this; "what is no longer hit, out" does.

Two properties follow, and they are the reason the design tolerates a moving key where the standard retrieval models freeze their encoder to avoid it. First, a drift-stale entry is doubly harmless: it is not *retrieved*, because queries use the binder's current key while the stale entry sits under the old one; and it is not *kept*, because being un-reactivated it goes cold and is evicted. Second, for recurring content the store *migrates*: as the binder improves, the same content recurs, is re-written under the better key, and the old entry dies—forgetting is what makes the migration clean rather than letting old and new encodings clutter the space together. This elevates the companion's empirical finding—forgetting as a regulariser—to an architectural principle: in a dynamic, learning system with moving keys, selective forgetting is the only way the memory stays both aligned and finite while everything else moves. It is the system's metabolism, not its janitor.

4 Why this fits the constraint

The design is shaped by a hard limit: own weights, no inherited pre-trained model, no datacenter, a single workstation. The constraint is survivable only because two costs that "moon-shot" rhetoric conflates are in fact separated by orders of magnitude. Pre-training general, open competence into weights is the moon price—unaffordable here, and no hybrid avoids it. But *this architecture does not pay it*: it acquires only the binding mechanic, on a curated corpus matched to a small model, which is cheap. Knowledge is the cent price—one online pass into an index, not a training run—and is exactly what the counter does best.

Open as process, not as state. The decisive reframing is that "open-ended" is not a frozen breadth held in weights (the moon price) but a process: start narrow, and let the boundary move over time through the selectively-forgetting memory. The binder alone is *solid state*: to learn anything new it must be retrained, expensively and with catastrophic forgetting. The hybrid keeps the binder's competence fixed and lets knowledge flow through the memory. This is why the binder alone is not a cheaper version of the same thing—it is a frozen, fixed-domain model, the very thing the project sets out to avoid.

Does the hybrid reduce training cost? Honestly: not for binding. Retrieval offloads *knowledge* from the weights, not *competence*; the binding floor—the parameters and steps at which grammar and agreement appear—is irreducible and must be paid once. What the hybrid eliminates is the *recurring* cost of folding new knowledge into a model: new knowledge into the counter is one pass; into the binder it is retraining. In the kNN-LM variant the binder is trained exactly as it would be alone and the datastore is attached with near-zero additional training, so open-ended knowledge costs almost nothing on the training side [2]. The hybrid is not cheaper to *start*; it is cheaper to *keep current*, forever.

5 Open problems

We name the open problems rather than absorb them into the design, because they are research questions, not engineering.

Drift rate versus recurrence interval. The self-healing of Section 3.1 keeps up only if representations drift more slowly than content recurs: knowledge survives migration if it is re-written under a new key before the old entry decays; a domain that goes quiet while the binder drifts heavily can evaporate before it is refreshed. This is biological forgetting—you lose what you do not revisit—and is the selection the project wants, except in the case of rare-but-important-once knowledge, the inherent hard case of any finite forgetting memory. The operative quantity is therefore not "is it evicted" (it is) but the ratio of drift rate to recurrence interval, which is measurable (§6).

Continual competence and catastrophic forgetting. Knowledge built from *familiar* representations is the solved, effortless case: the frozen mechanic applies in-context to new names, facts, and combinations it never saw in training, the way a frozen large model does. A genuinely *new* domain—one demanding representations the binder never

learned to form—is not: its embedding is uninformed, and storing an uninformed key captures nothing useful. Extending competence there requires the binder to learn further, and there sits catastrophic forgetting—a robust, decades-old phenomenon [5], not an open mystery. The *naive* path (just keep training) is known-broken. What is open is whether non-naive methods—replay, elastic consolidation [6]—tame it, and here our stack has a candidate, not a solution: the selective counter already knows *which* old knowledge is salient and recurring, which is exactly the signal such methods need to protect. We offer this as a hypothesis. The honest framing, after the critical-period analogy, is a question of *degree*: how far does the effortless reach (frozen mechanic on a familiar representation space, extended by external memory) extend before re-priming is required—and the selective memory is the machinery for pushing that reach by retrieval rather than by re-wiring.

6 Experimental plan

The experiments are fixed here in advance; their results will be folded into this section as they complete.

1. The binding test (the control is set). The counter and the binder are run on the *same* curated corpus (TinyStories [4]), so that the difference between their generations isolates binding with nothing else varying. The counter baseline is already measured: 1.306 prequential bits/char, with generation that is fluent at the surface (real names, fairy-tale register, clean short fragments) and incoherent as narrative (characters morph mid-sentence, no thread). The binder must beat 1.306 on the boring axis *and*, on the axis that matters, hold a narrative thread on reading where the counter produces a textureful cloud. Success is read, not scored. **[pending — binder bpc and generation samples; side-by-side reading against the counter baseline.]**

2. Drift rate. A small binder is trained; at fixed checkpoints a fixed probe set of contexts is embedded and the cosine similarity to the same context's embedding at the previous checkpoint is recorded, giving a drift curve. A curve that falls quickly licenses the simple design (warm up, then freeze keys, classical fixed-encoder retrieval); a curve that persists forces continual re-indexing or the forgetting–drift coupling of §3.1. This does not require a *good* binder, only a *training* one, and so can be run on modest hardware now. **[pending — drift curve across checkpoints; characteristic time of stabilisation.]**

3. Catastrophic forgetting. Train on domain A, then on domain B, and measure the degradation of A. Compare the naive baseline against counter-guided protection (replay weighted by the counter's salience/recurrence signal). Relative degradation, not absolute quality, is the readout. **[pending — A-degradation under naive vs. counter-guided continual training.]**

On hardware. The reference implementation is pure-`f64`, single-threaded, and zero-dependency, which is correct but not fast; the binding test at a binding-capable size is hours on a modest mobile CPU. The verified implementation is therefore the *specification* against which a faster variant (thread-parallel matmuls, or a GPU framework's autodiff) is validated: the gradient check and identical bits-per-character are the acceptance test, so speed is bought without surrendering the correctness that the from-scratch, hand-checked backward pass was built to guarantee.

7 Related work

The retrieval-augmentation skeleton is established and we claim none of it: nearest-neighbour language models interpolate a parametric model with a lookup over a datastore and improve the long tail and freshly added knowledge without retraining [2]; RETRO lets a comparatively small Transformer attend to chunks retrieved from a very large store and reach what would otherwise need a far larger model [3]. Our contribution is narrower and specific: (i) a *selectively-forgetting* datastore as the retention policy, in place of a static pre-built index—the counter's measured strength as a store discipline; (ii) the reading of forgetting as *coherence maintenance under representational drift*, which is what lets the key move where the standard models freeze the encoder; and (iii) a *prime-then-freeze* binder with the memory in the role of adult plasticity, extending reach by retrieval rather than by re-priming. The binding-at-small-scale premise rests on TinyStories [4]; the two-mechanism reading on the hippocampal–cortical literature [9, 8] and the critical-period account of syntax [7]; catastrophic forgetting and its mitigations on the connectionist literature [5, 6]; and distributed/holographic representation on its classics [10, 11]. The substrate, its

forgetting regime, and its measured ceiling are the present author's companion papers [13, 14].

8 Conclusion

We have set out, as a falsifiable design, a two-mechanism language system: a small from-scratch Transformer that learns the binding mechanic in a critical-period-like window, and a selectively-forgetting count graph that carries open-ended knowledge by retrieval keyed on the binder's representations, with forgetting as the metabolism that keeps the store aligned and finite under drift. The architecture is grounded in measured precursors—a substrate whose ceiling is binding, a forgetting regime that is a regulariser, and a gradient-verified binder—and in a two-mechanism reading the neuroscience supports. Its central virtue under the project's constraint is that it pays the binding cost once and the knowledge cost per-pass, making "open" a process rather than a frozen state, on hardware one person owns. Its central honesty is that the hardest step remains open: continual competence acquisition is a real research problem, the naive path is known-broken, and our selective memory is a candidate hebel within it, not a proof. But the value does not hinge on that step. A binder that binds in its domain, plus a memory that absorbs fresh knowledge built from familiar representations without retraining, is already "open as process" for the large class of knowledge made of familiar building-blocks. The order is therefore clean: break the certain ceiling first (show the binder binds), build the usable floor (binder plus memory in the mastered space), then take the open bet (the counter against catastrophic forgetting). Each step stands on its own, and none depends on the next succeeding.

References

[1] Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., Kaiser, Ł., Polosukhin, I., "Attention Is All You Need", *Advances in Neural Information Processing Systems* **30** (2017).

[2] Khandelwal, U., Levy, O., Jurafsky, D., Zettlemoyer, L., Lewis, M., "Generalization through Memorization: Nearest Neighbor Language Models", *Proc. International Conference on Learning Representations (ICLR)* (2020).

[3] Borgeaud, S., Mensch, A., Hoffmann, J., et al., "Improving Language Models by Retrieving from Trillions of Tokens" (RETRO), *Proc. International Conference on Machine Learning (ICML)* (2022).

[4] Eldan, R., Li, Y., "TinyStories: How Small Can Language Models Be and Still Speak Coherent English?", arXiv:2305.07759 (2023).

[5] McCloskey, M., Cohen, N. J., "Catastrophic Interference in Connectionist Networks: The Sequential Learning Problem", *Psychology of Learning and Motivation* **24**, 109–165 (1989).

[6] Kirkpatrick, J., Pascanu, R., Rabinowitz, N., et al., "Overcoming Catastrophic Forgetting in Neural Networks", *Proceedings of the National Academy of Sciences* **114**(13), 3521–3526 (2017).

[7] Lenneberg, E. H., *Biological Foundations of Language*, Wiley, New York (1967).

[8] Whittington, J. C. R., Muller, T. H., Mark, S., Chen, G., Barry, C., Burgess, N., Behrens, T. E. J., "The Tolman–Eichenbaum Machine: Unifying Space and Relational Memory through Generalization in the Hippocampal Formation", *Cell* **183**(5), 1249–1263 (2020).

[9] von der Malsburg, C., "The Correlation Theory of Brain Function", in *Models of Neural Networks II*, E. Domany, J. L. van Hemmen, K. Schulten (eds.), Springer, New York, pp. 95–119 (1994); orig. MPI für Biophysikalische Chemie, Internal Report 81-2 (1981).

[10] Kanerva, P., *Sparse Distributed Memory*, MIT Press, Cambridge, MA (1988).

[11] Plate, T. A., "Holographic Reduced Representations", *IEEE Transactions on Neural Networks* **6**(3), 623–641 (1995).

[12] Cleary, J. G., Witten, I. H., "Data Compression Using Adaptive Coding and Partial String Matching", *IEEE Transactions on Communications* **32**(4), 396–402 (1984).

[13] Langjahr, M., "Forgetting as Regularisation in an Energy-Conserving Context Graph", Pantareon Foundations (2026).

[14] Langjahr, M., "The Source of Language Capability in an Energy-Based Associative Substrate", Pantareon Foundations (2026).