Brain-Inspired Embedding, Dynamic Scaling, and Deployment System for Neural Language Models

Open research specification 7 of 7

Krrish Choudhary

Verace Pvt. Ltd.

This document was first drafted in the form of a patent specification, which is why it speaks of an invention and of claims. No patent application was filed and none is intended. Where the text mentions a related or co-pending patent application, it means a companion document in this set, and none of them was filed. The work is published as open research.

Published under the PolyForm Noncommercial License 1.0.0

Copyright (c) 2026 Verace Pvt. Ltd. (https://verace.in)

Contents

1 FIELD **3**

2 BACKGROUND **3**

2.1 Prior Art: Token Embeddings in Neural Language Models 3

2.2 Prior Art: Network Scaling 4

2.3 Prior Art: Deployment Optimisation 4

2.4 Limitations of the Prior Art 4

2.4.1 Single-Stage Embedding 4

2.4.2 Linguistic Blindness of Statistical Tokenisation 4

2.4.3 Fixed Position Encoding 5

2.4.4 Static Network Architecture 5

2.4.5 Scale-Blind Hyperparameter Transfer 5

2.4.6 Uncertainty-Blind Deployment 5

2.5 Need for Improvement 5

3 SUMMARY **6**

4 DESCRIPTION OF THE FIGURES **9**

5 DETAILED DESCRIPTION **27**

5.1 Four-Stage Brain Embedding Pipeline 27

5.1.1 Stage 1: Base Embedding Lookup 28

5.1.2 Stage 2: Dual-Route Morphological Composition (Theorem 58) 28

5.1.3 Stage 3: Hub-and-Spoke Semantic Integration (Theorem 56) 30

5.1.4 Stage 4: Sparse Distributed Projection (Theorem 57) 31

5.2 Grid Cell Position Encoding 33

5.2.1 Learnable Grid Cell Module Periods (Theorem 59) 33

5.2.2 Per-Head Scale Factors (Theorem 62) 34

5.2.3 Boundary-Aware Phase Remapping (Theorem 61) 34

5.3 Dual-Route Morphological Tokeniser 34

5.3.1 Lexical Route 35

5.3.2 Sublexical Route 35

5.3.3 Byte-Level Fallback 36

5.3.4 Oscillatory Alignment 36

5.4 Dynamic Column Scaling System (AutoScale) 36

5.4.1 Multi-Scale σ^2 Depth Bands 37

5.4.2 Health Check Protocol 38

5.4.3 Operation 1: Kill and Rebirth 38

5.4.4 Operation 2: Merge Redundant Columns 39
5.4.5 Operation 3: Shrink Underutilized Columns 40
5.4.6 Operation 4: Split Overloaded Columns (Net2Net) 41
5.4.7 Operation 5: Grow New Columns 42
5.5 Maximal Update Parameterisation (muP) Transfer 43
5.5.1 Scale-Invariant Parameters 44
5.5.2 Dimension-Proportional Parameters 44
5.6 σ^2-Guided Deployment Methods 45
5.6.1 σ^2-Guided Early Exit 45
5.6.2 σ^2-Adaptive Direct Preference Optimisation (DPO) 46
5.6.3 σ^2-Adaptive Supervised Fine-Tuning (SFT) 47
5.6.4 σ^2-Driven Tool Selection 47
5.7 Computational Kernel Optimisations 48
5.7.1 Parallel Prefix Scan with Log-Space Arithmetic 48
5.7.2 Algebraic Windowed Context Decomposition 49
5.7.3 NMDA Parallel Scan Decomposition 50
5.7.4 Tsodyks-Markram Linearisation as Dual Sequential Scans 51

6 CLAIMS 52

TITLE

Brain-Inspired Embedding, Dynamic Scaling, and Deployment System for Neural Language Models

1. FIELD

The present invention relates generally to neural network embedding systems, dynamic network scaling methods, and deployment infrastructure for biologically-grounded neural language models, and more particularly to: (a) a multi-stage brain-inspired embedding pipeline comprising dual-route morphological composition modelled on the Coltheart Dual-Route Cascaded (DRC) reading architecture, hub-and-spoke semantic integration modelled on anterior temporal lobe cross-modal binding, sparse distributed projection modelled on neocortical population coding, and grid-cell-inspired position encoding with boundary-aware phase remapping; (b) a morphological tokenizer implementing dual-route lexical access with oscillatory alignment; (c) a dynamic scaling system for cortical column networks comprising automated grow, split, merge, kill, and shrink operations driven by multi-scale prediction uncertainty (σ^2); (d) a maximal update parameterisation (muP) transfer method for brain-inspired architectures; (e) σ^2-guided early exit, alignment, and tool-use deployment methods; and (f) computational kernel optimisations including parallel prefix scan, algebraic windowed context decomposition, and recurrence linearisation for working memory systems.

2. BACKGROUND

2.1. Prior Art: Token Embeddings in Neural Language Models

Modern neural language models represent input tokens as dense continuous vectors through a learned embedding table. In the dominant transformer architecture (Vaswani et al., 2017), a single lookup operation maps each integer token identifier to a fixed-dimensional vector $\mathbf{e} \in \mathbb{R}^d$. Position information is injected either through additive sinusoidal encoding (Vaswani et al., 2017) or through Rotary Position Embeddings (RoPE; Su et al., 2021) that encode position as a rotation in a complex vector space. The embedded representations are then processed through a stack of uniform self-attention blocks.

Byte Pair Encoding (BPE; Sennrich et al., 2016) and its variants (SentencePiece, WordPiece) have become the standard tokenisation approach, using statistical co-occurrence rather than linguistic structure to segment text into subword units.

2.2. Prior Art: Network Scaling

Scaling neural networks has historically required either training an entirely new model from scratch at each scale, or employing computationally expensive hyperparameter search. The Net2Net approach (Chen et al., 2016) introduced function-preserving transformations for widening and deepening networks, but provided no principled method for deciding when or where to scale. Maximal Update Parameterisation (muP; Yang and Hu, 2021) enables hyperparameter transfer from small proxy models to large target models for transformer architectures, but has not been adapted to biologically-grounded cortical column networks where local learning rules and column-level dynamics create fundamentally different scaling requirements.

2.3. Prior Art: Deployment Optimisation

Early exit mechanisms (Schwartz et al., 2020; Xin et al., 2020) allow neural networks to terminate computation at intermediate layers when a confidence criterion is met. However, existing approaches rely on auxiliary classifiers attached at each layer, and the confidence threshold is a single global scalar with no multi-scale uncertainty decomposition. Alignment methods such as Direct Preference Optimisation (DPO; Rafailov et al., 2023) and supervised fine-tuning (SFT) treat model confidence as uniform, with no mechanism to modulate preference strength based on the model's own uncertainty estimate. Tool-use decisions in existing language model agents (Schick et al., 2023) rely on prompt engineering or fine-tuning rather than intrinsic neural signals.

2.4. Limitations of the Prior Art

The prior art exhibits the following deficiencies that the present invention addresses:

2.4.1. Single-Stage Embedding

Transformer embeddings perform a single lookup followed by optional position encoding. The brain, by contrast, processes lexical access through a temporally staged pipeline: obligatory morphological decomposition at approximately 170 milliseconds post-stimulus onset (Rastle and Davis, 2008), followed by cross-modal semantic integration in the anterior temporal lobe (Patterson et al., 2007), and sparse population coding in neocortical circuits (Ahmad and Hawkins, 2016). No existing neural network embedding system replicates this multi-stage cascade.

2.4.2. Linguistic Blindness of Statistical Tokenisation

BPE and related methods segment text based on statistical frequency without respecting morphological boundaries. The string "unhappiness" may be split as "un/happ/iness" (correct morphological boundaries) or "unh/appin/ess" (linguistically arbitrary), depending entirely on corpus statistics. This violates the psycholinguistic finding that the brain decomposes words along morpheme boundaries (Coltheart et al., 2001). Furthermore, BPE provides no mechanism

for dual-route access where high-frequency words are accessed as wholes while low-frequency words undergo compositional analysis.

2.4.3. Fixed Position Encoding

RoPE provides fixed sinusoidal frequencies determined by a single base parameter ($\theta = 10000$). Grid cells in the mammalian entorhinal cortex (Hafting et al., 2005) instead use learnable multi-scale periodic modules with boundary-aware phase remapping (Fyhn et al., 2007). No existing position encoding system provides learnable periods, per-head scale factors, or boundary-triggered phase reset.

2.4.4. Static Network Architecture

Transformer architectures are fixed at initialisation. The brain, by contrast, undergoes postnatal cortical development (growing new functional areas), synaptic pruning (removing underutilised connections), and adult neuroplasticity (reorganising after injury). No existing neural network system provides a unified framework for dynamically growing, splitting, merging, killing, and shrinking computational units during training, guided by multi-scale uncertainty signals.

2.4.5. Scale-Blind Hyperparameter Transfer

Existing muP methods are designed for transformer architectures with global backpropagation. Cortical column networks with local Hebbian learning, predictive coding, and per-column dynamics require a fundamentally different treatment: local learning rates, oscillatory parameters, and neuromodulatory scales are dimension-independent and transfer directly, while structural parameters (head count, inhibitory dimension, working memory slot dimension) must scale proportionally with the model dimension.

2.4.6. Uncertainty-Blind Deployment

Existing early exit, alignment, and tool-use methods do not leverage the multi-scale prediction uncertainty (σ^2) that biologically-grounded architectures provide at every processing level. An early exit decision based on a single layer's confidence risks terminating before both syntactic (token-level) and semantic (phrase-level) processing are complete. Alignment methods that apply uniform preference strength regardless of model uncertainty produce overconfident preferences on ambiguous content. Tool-use decisions based on prompt-level heuristics ignore the rich uncertainty signals available within the model's own computation.

2.5. Need for Improvement

There exists a need in the art for: (a) a multi-stage embedding pipeline grounded in the neuroscience of lexical access; (b) a morphological tokeniser that respects linguistic structure; (c) a

position encoding system with learnable periods and boundary-aware phase remapping; (d) a dynamic scaling framework that uses multi-scale uncertainty to grow, split, merge, kill, and shrink computational units; (e) muP hyperparameter transfer adapted for biologically-grounded architectures; (f) deployment methods that exploit multi-scale uncertainty for early exit, alignment, and tool use; and (g) computational kernel optimisations that enable efficient execution of the biological recurrence equations. The present invention addresses all of these needs.

3. SUMMARY

The present invention provides a computer-implemented system and method for brain-inspired embedding, dynamic scaling, and deployment of cortical column neural language models.

In a first aspect, the invention provides a **multi-stage brain embedding pipeline** comprising four cascaded stages, each adding a gated residual correction to an evolving token representation:

(i) a base embedding stage performing token-to-vector lookup;

(ii) a dual-route morphological composition stage implementing the Coltheart Dual-Route Cascaded (DRC) model, wherein a per-token frequency gate $g_{\text{lex}} = \text{sigmoid}(f_i)$ blends a lexical route (direct embedding lookup) with a sublexical route (causal depthwise-separable convolution over local context), the blend being modulated by a master identity-at-init(Stale from transformer era check with /nct-research) gate;

(iii) a hub-and-spoke semantic integration stage wherein the embedding is decomposed into S modality-specific spoke subvectors (visual, auditory, motor, verbal, emotional, encyclopedic), each spoke independently projected and then integrated through a shared bottleneck hub implementing anterior temporal lobe (ATL) cross-modal binding; and

(iv) a sparse distributed projection stage expanding the embedding to $d \times r$ dimensions, enforcing exact-k sparsity ($k = \lceil d \cdot r \cdot 0.02 \rceil$) via top-$k$ selection with a straight-through estimator, then projecting back to d dimensions, producing sparse population codes with combinatorial pattern capacity $\binom{n}{k}$ exceeding 10^{81} for typical parameter settings.

All four stages satisfy an identity-at-init(Stale from transformer era check with /nct-research)ialisation property: each stage's gate parameter is initialised to yield an effective gate value near zero ($g \approx 0.12$), ensuring the complete pipeline outputs the base embedding at initialisation and gradually learns corrections during training.

In a second aspect, the invention provides a **grid-cell-inspired position encoding** system replacing fixed Rotary Position Embeddings with:

(i) learnable grid cell module periods via $\tau_i = \text{softplus}(\theta_i^{\text{raw}})$, initialised to recover standard RoPE frequencies;

(ii) per-head scale factors $s_h = \text{softplus}(s_h^{\text{raw}})$ enabling multi-resolution conjunctive coding; and

(iii) boundary-aware phase remapping: a boundary detector $p_{\text{boundary}} = \text{sigmoid}(\mathbf{W}_b \cdot \mathbf{x})$ computes cumulative boundary-triggered phase shifts $\phi_{\text{cum}} = \text{cumsum}(p_{\text{boundary}} \cdot \phi_{\text{boundary}})$, which are added to the base angles through a learnable gate, enabling position reset at detected boundaries.

In a third aspect, the invention provides a **dual-route morphological tokeniser** implementing the Coltheart DRC model at the tokenisation level:

(i) a lexical route performing whole-word lookup for known high-frequency words;

(ii) a sublexical route performing greedy affix stripping (prefixes sorted by descending length, suffixes sorted by descending length) with morphophonemic normalisation and root validation against a known-word dictionary; and

(iii) a byte-level fallback for novel non-alphabetic input, ensuring lossless deterministic encoding.

Each token is aligned to approximately one gamma cycle ($\sim$1–2 temporal steps), yielding oscillatory alignment wherein words span theta cycles and phrases span alpha cycles.

In a fourth aspect, the invention provides a **dynamic column scaling system** (AutoScale) for cortical column networks, comprising five operations driven by multi-scale prediction uncertainty:

(i) **Grow**: progressively adding new cortical columns with thalamic routing expansion, lateral connection expansion, and workspace expansion, while freezing existing column parameters;

(ii) **Split (Net2Net)**: deep-copying an overloaded column, scaling output weights by 0.5 in each copy with symmetry-breaking noise, so that the sum of the two children preserves the parent's function at the moment of split;

(iii) **Merge**: detecting redundant columns via cosine similarity computed only on functional weights (association, output, prediction weights — not working memory, oscillatory, or gate parameters), merging via running average $\text{lerp}(1/(n+2))$ to give equal weight to all absorbed columns;

(iv) **Kill and rebirth**: detecting dead columns (mean gate value below threshold) and replacing them with a noisy deep copy of the best-performing column; and

(v) **Shrink**: detecting underutilized columns (low gates and low σ^2 persisting for multiple checks) with depth-band-aware thresholds (token-band columns require lower gate activity to be considered underutilised than discourse-band columns), absorbing their learned knowledge into nearest neighbours before removal.

All scaling operations include subsystem rebuilding (thalamus, lateral connections, workspace, self-assessment) to maintain architectural coherence after column count changes.

In a fifth aspect, the invention provides a **muP hyperparameter transfer method** for cortical column networks, comprising extraction of over 40 scale-invariant hyperparameters (local learning rates, oscillatory parameters, neuromodulatory scales, gate dynamics, working memory parameters, structural plasticity thresholds) from a trained small proxy model, and construction of a scaled target configuration wherein dimension-independent parameters transfer directly while structural parameters (head count, inhibitory dimension, working memory slot dimensions, relay dimension) are scaled proportionally to the d_{model} ratio.

In a sixth aspect, the invention provides σ^2**-guided deployment methods**:

(i) **Early exit**: exiting column processing when both token-band σ^2 (from shallow columns) and phrase-band σ^2 (from early-mid columns) fall below a threshold, ensuring both syntactic and semantic processing are complete before termination;

(ii) σ^2**-adaptive DPO**: computing an effective preference temperature $\beta_{\text{eff}} = \beta / \max(\bar{\sigma}^2, 0.1)$, reducing preference enforcement when the model is genuinely uncertain; and

(iii) σ^2**-adaptive SFT**: weighting the supervised fine-tuning loss by $w = 1/(1+0.1 \cdot \max(0, \sigma^2_{\text{sentence}} - 2))$, with auxiliary loss preservation (predictive coding, cerebellar, thalamic reticular nucleus losses) to prevent brain mechanism atrophy during alignment; and

(iv) σ^2**-driven tool selection**: deciding tool use when $\sigma^2 > \theta / \max(0.3 + \text{tonic_DA}, 0.1)$, where dopamine modulates the threshold such that high dopamine (exploration tendency) lowers the barrier to tool invocation.

In a seventh aspect, the invention provides a non-transitory computer-readable storage medium storing instructions that, when executed by one or more processors, cause the processors to implement the systems and methods described above.

The architecture provides the following advantages over the prior art:

- Biologically-grounded multi-stage lexical access with distinct morphological, semantic, and sparse coding stages, each with identity-at-init(Stale from transformer era check with /nct-research)ialisation guarantees;
- Morphologically faithful tokenisation respecting linguistic structure rather than statistical co-occurrence;
- Learnable multi-scale position encoding with boundary-aware phase remapping, recovering standard RoPE at initialisation;
- Automated architecture evolution driven by multi-scale uncertainty rather than manual intervention;

- Scale-invariant hyperparameter transfer exploiting the independence of local learning rules from total column count;
- Deployment optimisations that leverage the model's own prediction uncertainty for early exit, alignment, and tool use; and
- Computational kernel innovations achieving up to 9.8× speedup through parallel prefix scan, algebraic score decomposition, and recurrence linearisation.

4. DESCRIPTION OF THE FIGURES

The invention will be described with reference to the accompanying drawings, in which:

Figure 1 is a block diagram of the four-stage brain embedding pipeline, showing the sequential cascade from base embedding through morphological composition, hub-and-spoke semantic integration, and sparse distributed projection, with gated residual connections at each stage.

Figure 2 illustrates the dual-route morphological composition gate (Theorem 58), showing the parallel lexical route (direct lookup) and sublexical route (causal depthwise-separable convolution), the per-token frequency gate g_{lex} blending the two routes, and the master identity-at-init(Stale from transformer era check with /nct-research) gate controlling the overall correction magnitude.

Figure 3 illustrates the hub-and-spoke semantic integration module (Theorem 56), showing the decomposition of an embedding vector into $S = 6$ modality-specific spoke subvectors, independent spoke projections, concatenation into the hub bottleneck (Linear $\rightarrow$ GELU $\rightarrow$ Linear), and the gated residual correction.

Figure 4 illustrates the sparse distributed projection module (Theorem 57), showing the expansion from d to $d \times r$ dimensions via W_{up}, exact-k sparsity enforcement via top-k selection, the straight-through estimator gradient path, and the projection back to d dimensions via W_{down}.

Figure 5 illustrates the grid-cell-inspired position encoding system, showing the learnable period module (Theorem 59), per-head scale factors (Theorem 62), and boundary-aware phase remapping (Theorem 61) with the cumulative boundary phase shift mechanism.

Figure 6 is a block diagram of the dual-route morphological tokeniser, showing the lexical route (whole-word lookup), sublexical route (greedy affix stripping with morphophonemic normalisation), byte-level fallback, and the oscillatory alignment property.

Figure 7 is a state diagram of the AutoScale dynamic column scaling system, showing the five operations (grow, split, merge, kill, shrink), their trigger conditions based on multi-scale σ^2 and gate statistics, the priority ordering (kill $\rightarrow$ merge $\rightarrow$ shrink $\rightarrow$ split $\rightarrow$ grow), and the mandatory subsystem rebuild step.

Figure 8 illustrates the Net2Net column split operation, showing the parent column, the two daughter columns with halved output weights and symmetry-breaking noise, and the function-preservation property at the moment of split.

Figure 9 illustrates the merge operation, showing cosine similarity computation on functional weights only (W_assoc, W_out, W_pred), the running average lerp$(1/(n+2))$ merge strategy, and the subsystem rebuild after column removal.

Figure 10 illustrates the shrink operation with depth-band-aware thresholds, showing the different shrink sensitivities for token-band (0.5×), phrase-band (0.7×), sentence-band (0.9×), and discourse-band (1.0×) columns, the patience mechanism, and knowledge absorption into nearest neighbours.

Figure 11 illustrates the muP hyperparameter transfer method, showing the extraction of scale-invariant parameters from a small proxy model, the dimension-proportional scaling of structural parameters, and the construction of the target configuration.

Figure 12 illustrates the σ^2-guided early exit mechanism, showing the column-by-column processing, the multi-scale σ^2 aggregation (token-band and phrase-band), the dual-threshold exit condition, and the compute savings on easy inputs.

Figure 13 illustrates the σ^2-adaptive DPO alignment method, showing the computation of effective beta $\beta_{\text{eff}} = \beta / \max(\bar{\sigma}^2, 0.1)$, the chosen and rejected log-probability computation, and the modulated preference loss.

Figure 14 illustrates the σ^2-driven tool selection adapter, showing the extraction of σ^2, tonic dopamine, and ACC conflict signals from the model's forward pass, the DA-modulated threshold computation, and the binary tool-use decision.

Figure 15 illustrates the parallel prefix scan kernel, showing the associative operator $(a, b) \oplus (a', b') = (a \cdot a', a' \cdot b + b')$, the chunked computation strategy (vectorised within chunks, sequential across chunks), and the log-space arithmetic for numerical stability.

Figure 16 illustrates the algebraic windowed context decomposition, showing the key insight that $\text{score}[t, j] = c[t] - c_{\text{pad}}[j + t]$ where $c = (x \cdot W_{\text{dist}}).\text{sum}(-1)$ is a scalar, reducing score computation from $O(T \times w \times D)$ to $O(T \times D + T \times w)$.

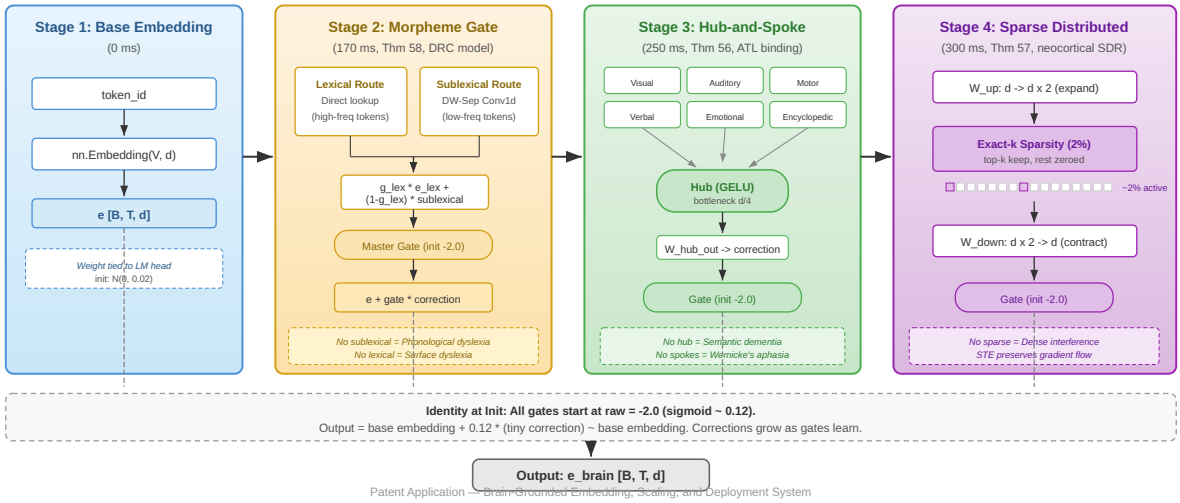
Figure 17 illustrates the NMDA parallel scan decomposition, showing the pre-computation of all write probabilities and contents, the K independent per-slot parallel scans, and the batch read step.

Figure 18 illustrates the silent trace parallel scan decomposition, showing the Tsodyks-Markram linearisation into dual sequential scans: facilitation u (first scan) and depression x (second scan, dependent on u).

Note: In the actual patent filing, each figure described above shall be provided as a separate drawing sheet conforming to Rule 15 of the Patents Rules, 2003.

ARCHITECTURAL OVERVIEW FIGURES

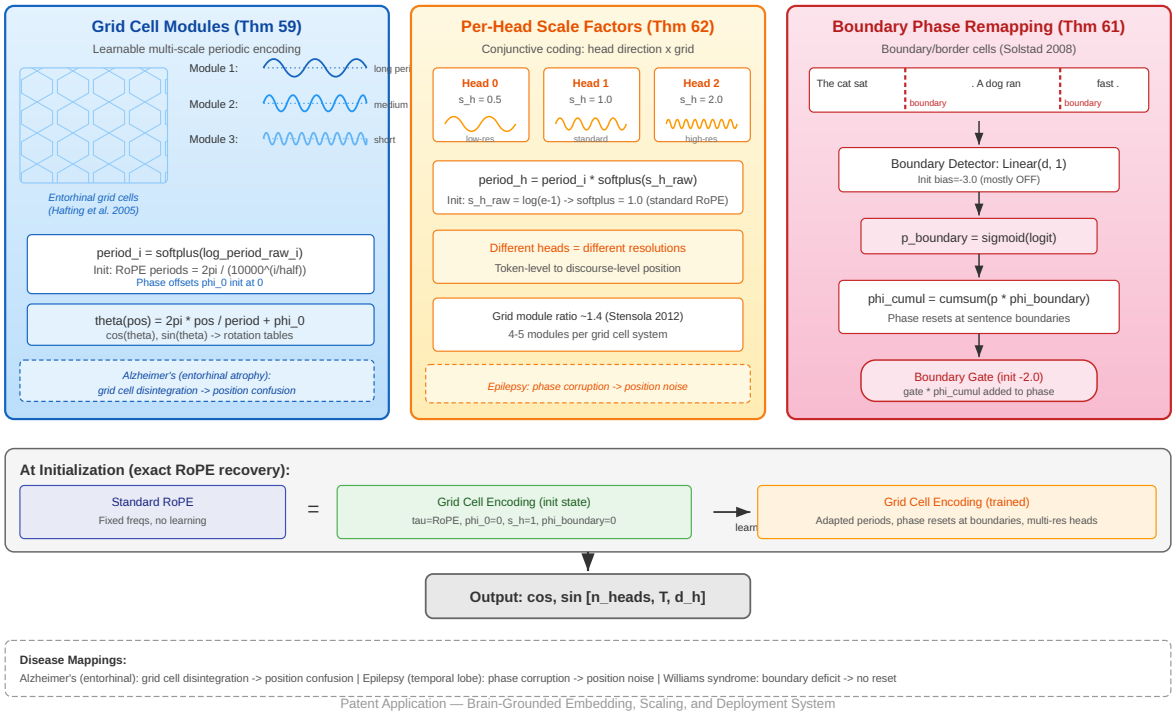
FIG. 21



Patent Application — Brain-Grounded Embedding, Scaling, and Deployment System

Figure 1: Figure 1 — Four-Stage Brain Embedding Pipeline (Claim 1 overview)

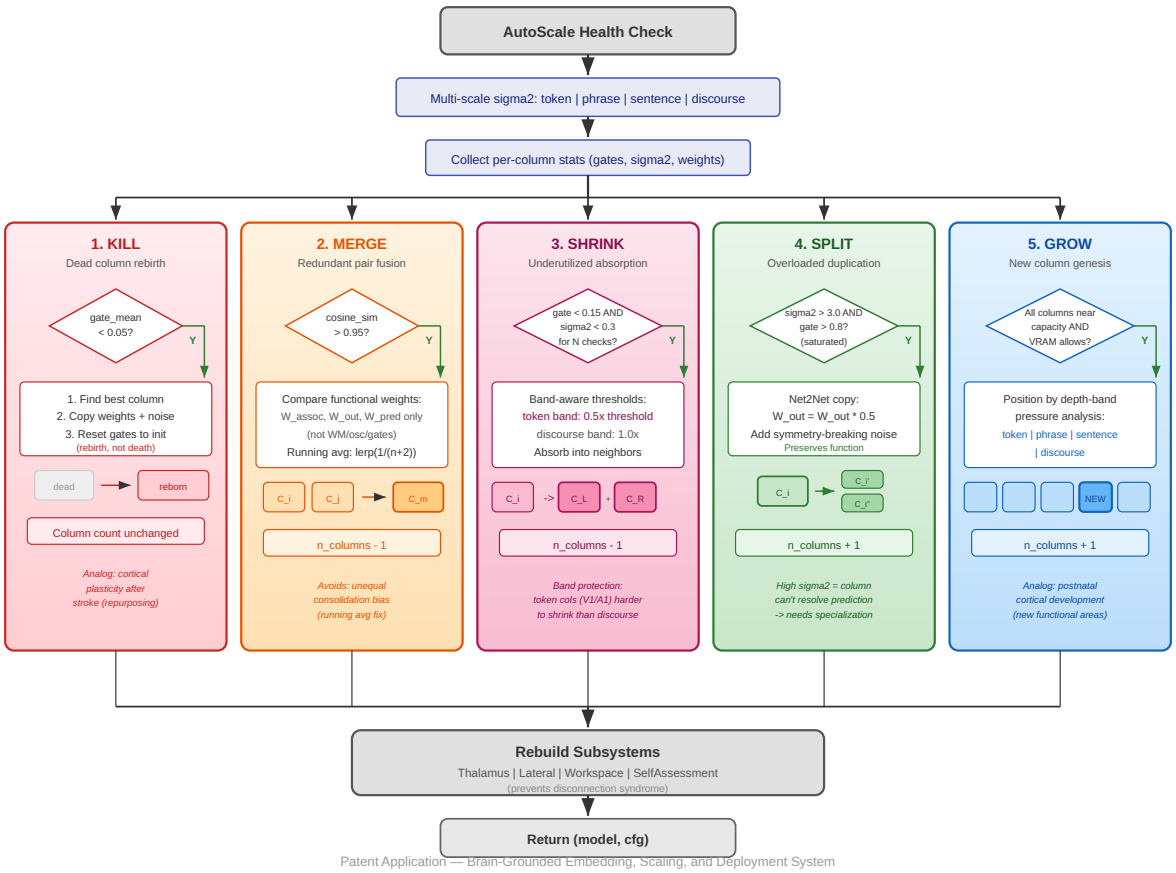
FIG. 22



Patent Application — Brain-Grounded Embedding, Scaling, and Deployment System

Figure 2: Figure 5 — Grid-Cell-Inspired Position Encoding System (Claims 5–7)

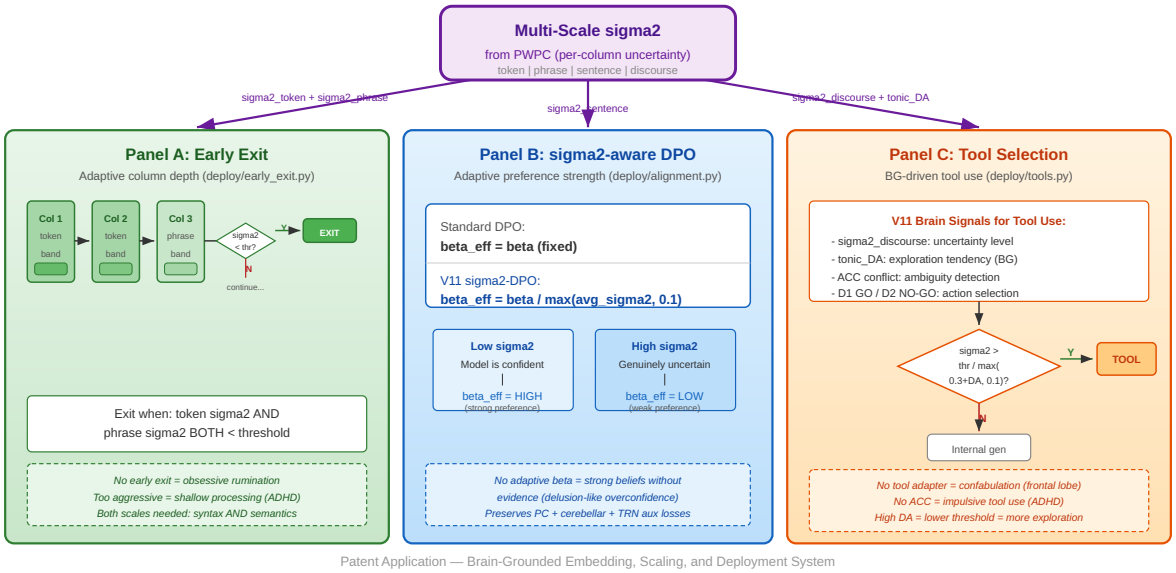
FIG. 23



Patent Application — Brain-Grounded Embedding, Scaling, and Deployment System

Figure 3: Figure 7 — AutoScale Dynamic Column Scaling System (Claims 9–15)

FIG. 24



Patent Application — Brain-Grounded Embedding, Scaling, and Deployment System

Figure 4: Figure 12 — σ^2-Guided Deployment System (Claims 16–21)

CLAIM ILLUSTRATION SHEETS

FIG. P7-1: Multi-Stage Brain Embedding Pipeline (Claim 1)

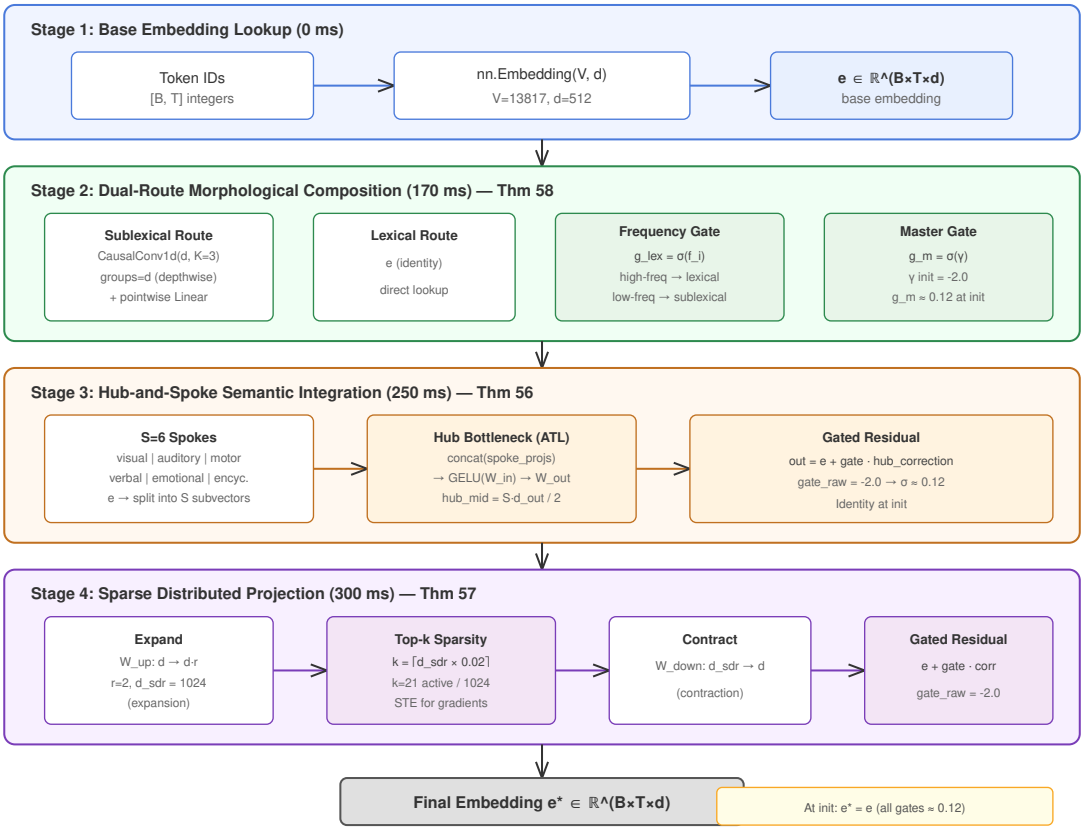


Figure 5: Claim 1 — Multi-Stage Brain Embedding Pipeline

FIG. P7-2: Per-Token Frequency Bias Initialisation (Claim 2)

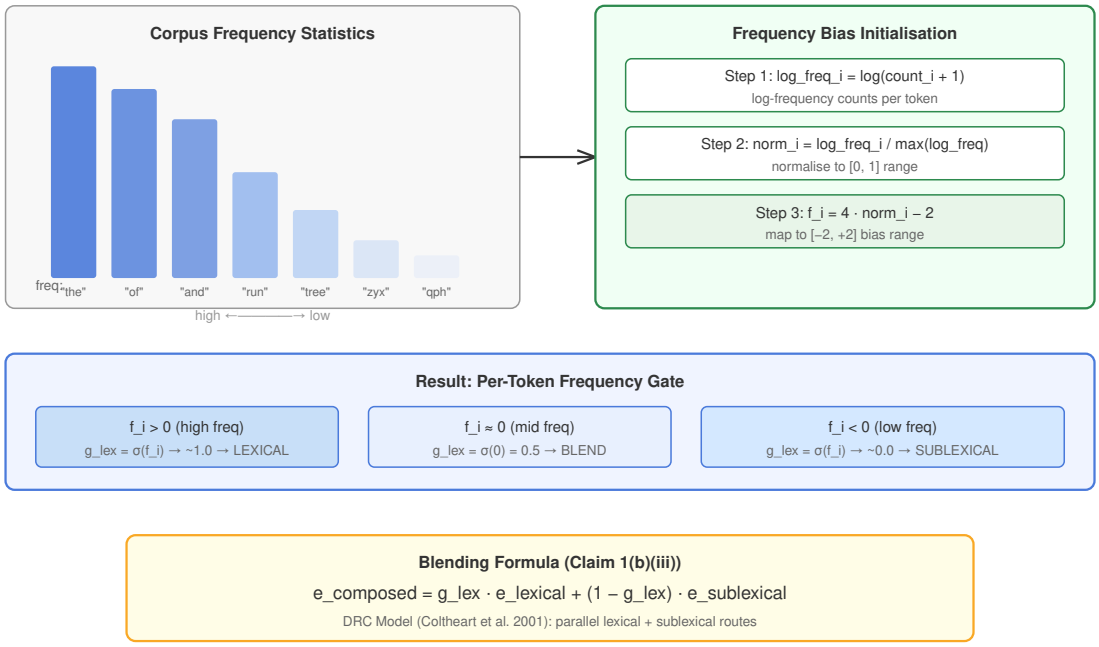


Figure 6: Claim 2 — Per-Token Frequency Bias Initialisation

FIG. P7-3: Causal Depthwise-Separable Convolution (Claim 3)

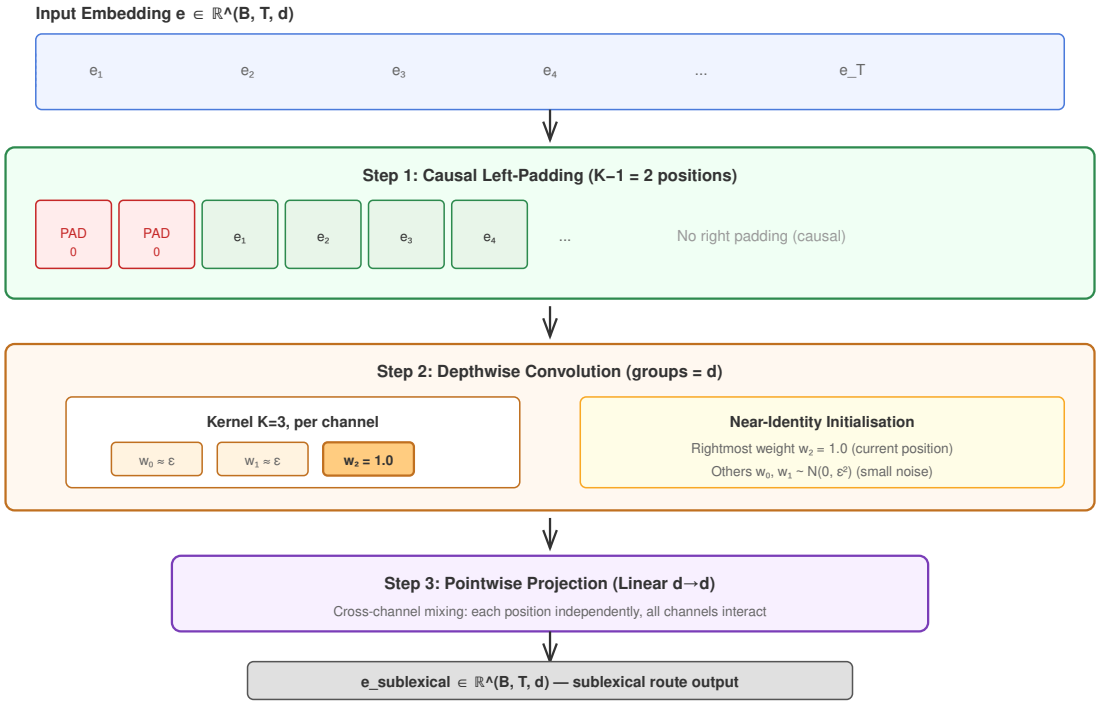


Figure 7: Claim 3 — Hub-and-Spoke Semantic Integration

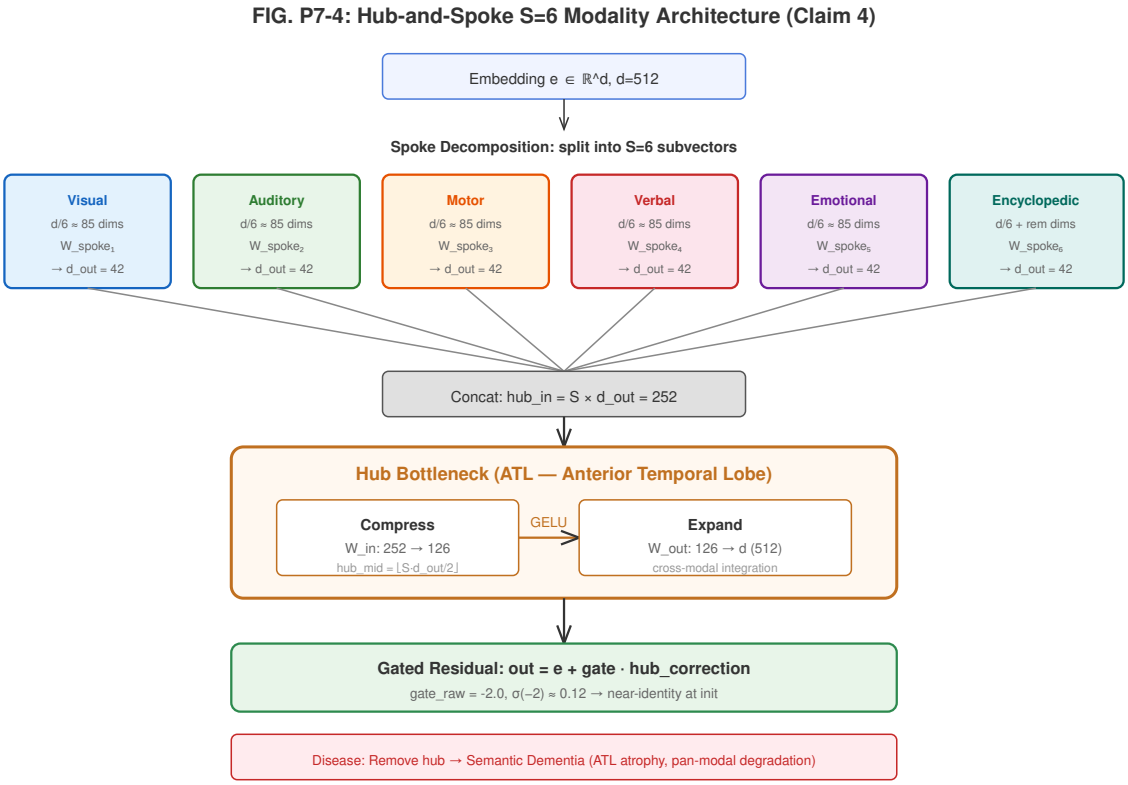


Figure 8: Claim 4 — Sparse Distributed Projection

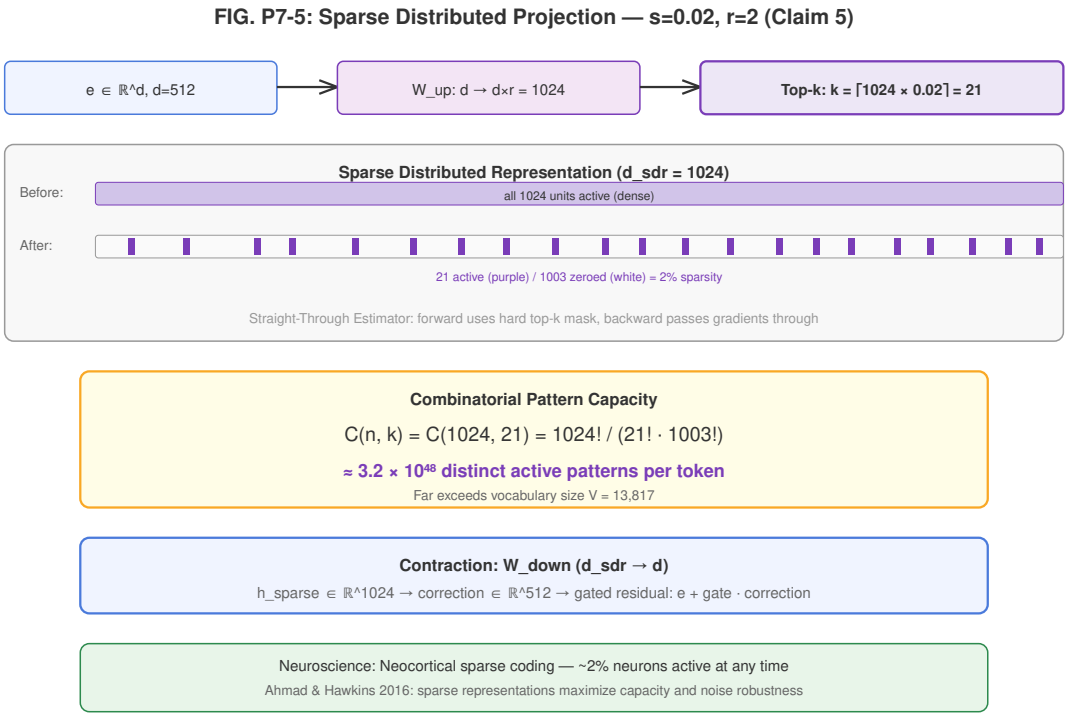


Figure 9: Claim 5 — Grid Cell Learnable Periods

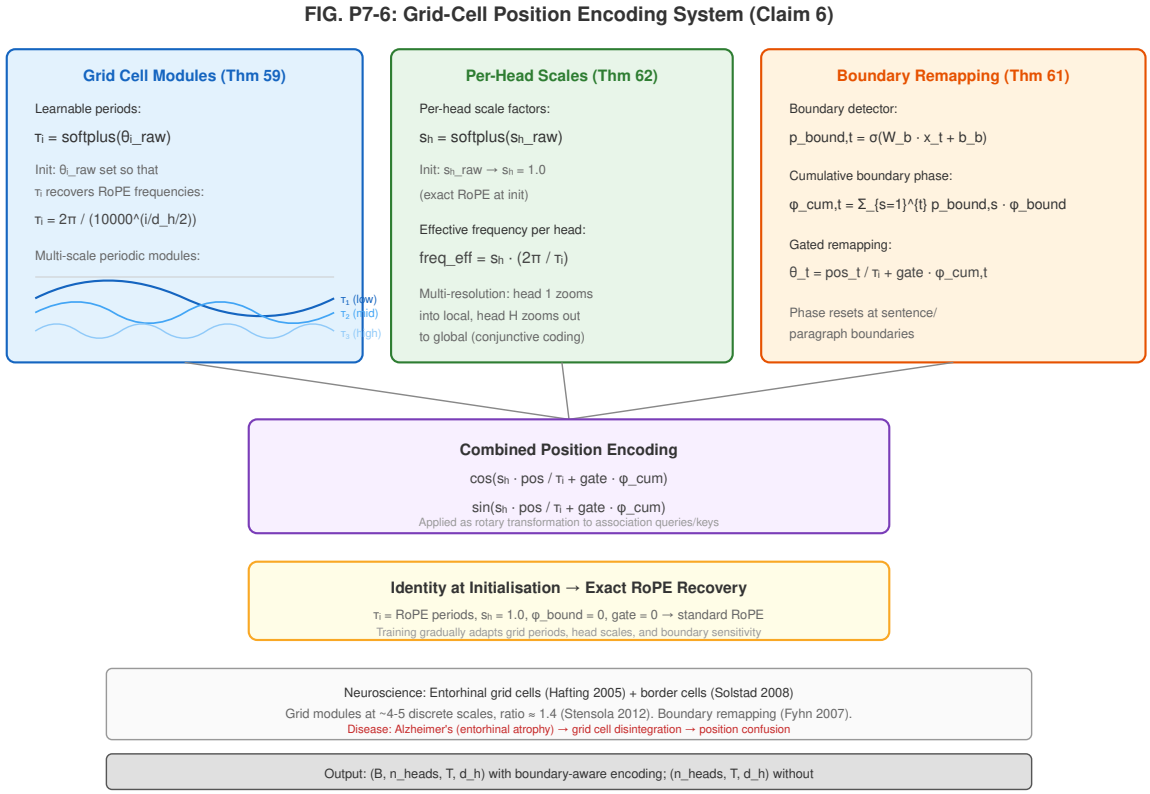


Figure 10: Claim 6 — Per-Head Grid Cell Scale Factors

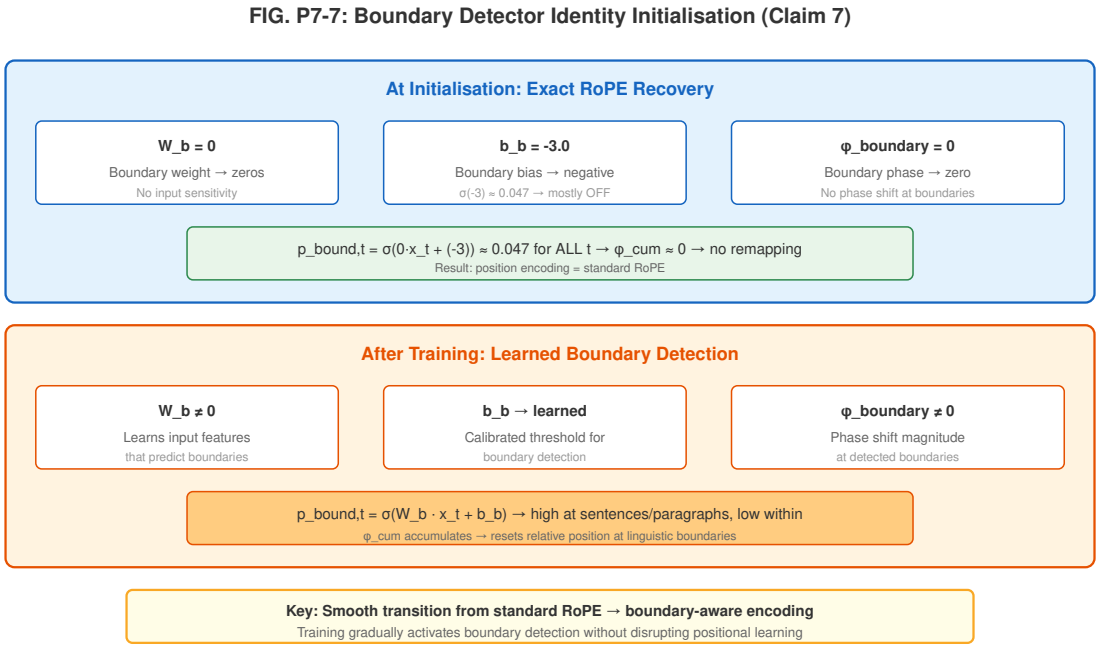


Figure 11: Claim 7 — Boundary Detector Identity Initialisation

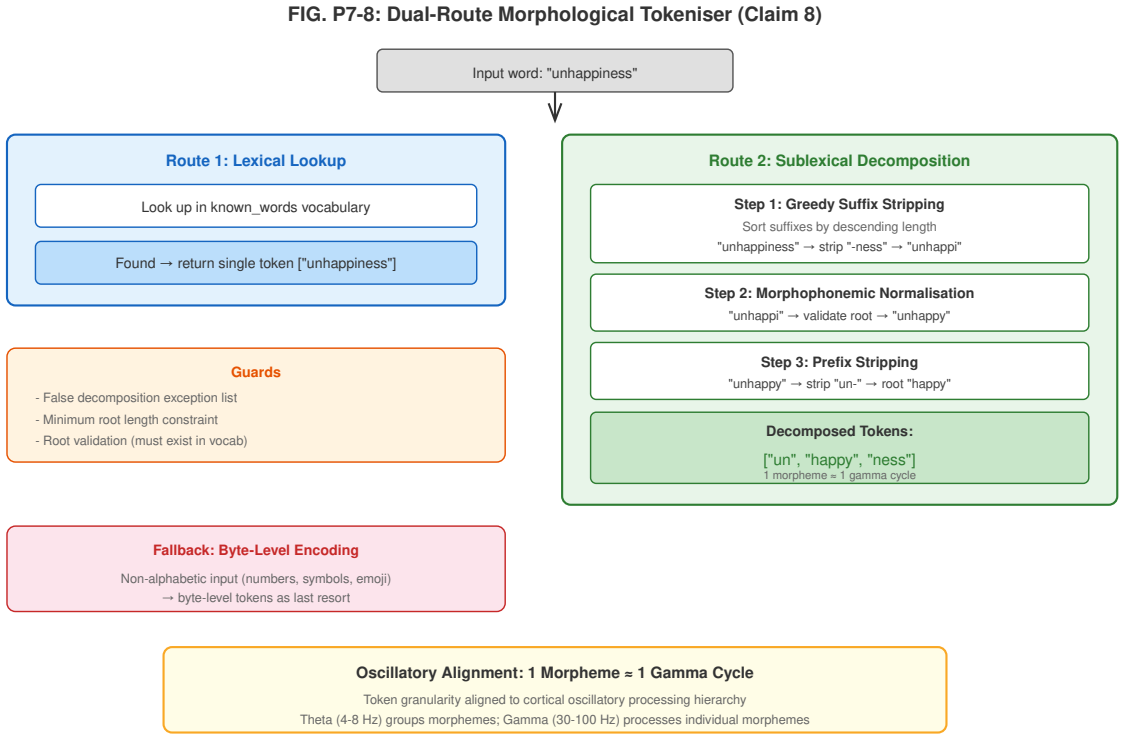


Figure 12: Claim 8 — Dual-Route Morphological Tokeniser

FIG. P7-9: Dynamic Column Scaling Method — AutoScale (Claim 9)

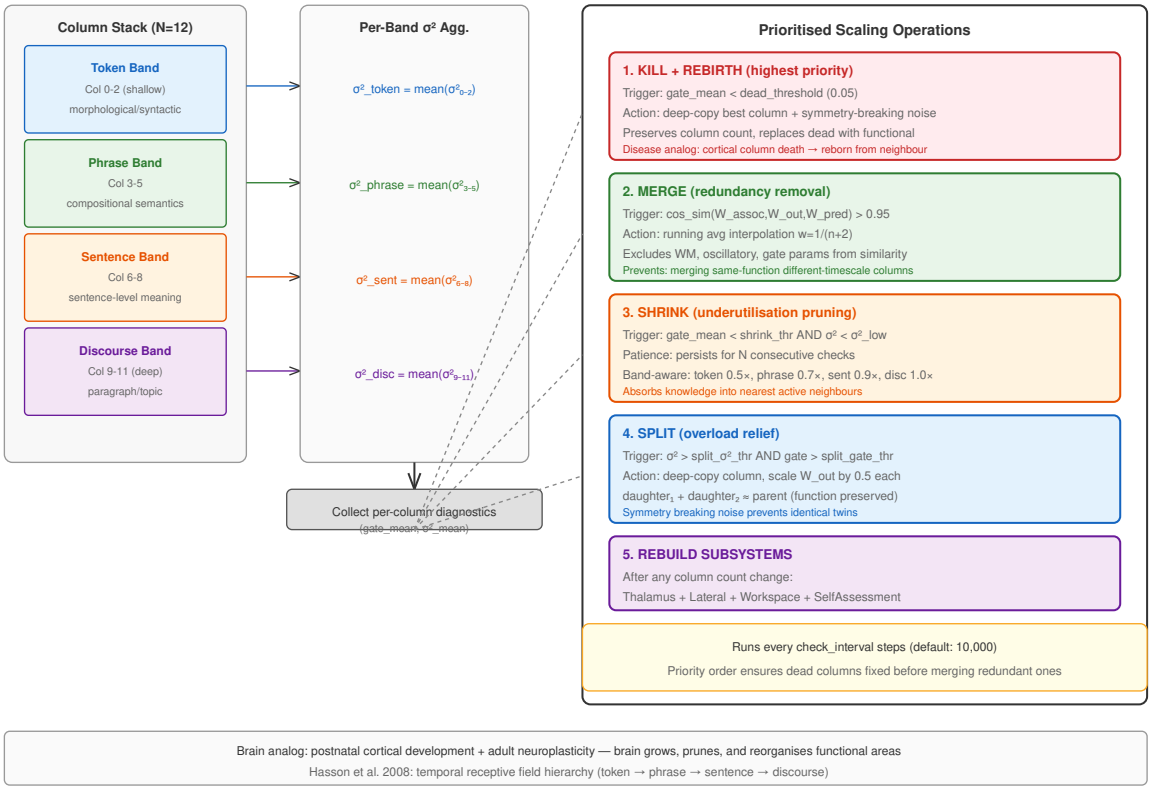


Figure 13: Claim 9 — AutoScale Dynamic Column Scaling

FIG. P7-10: Four Depth Bands (Claim 10)

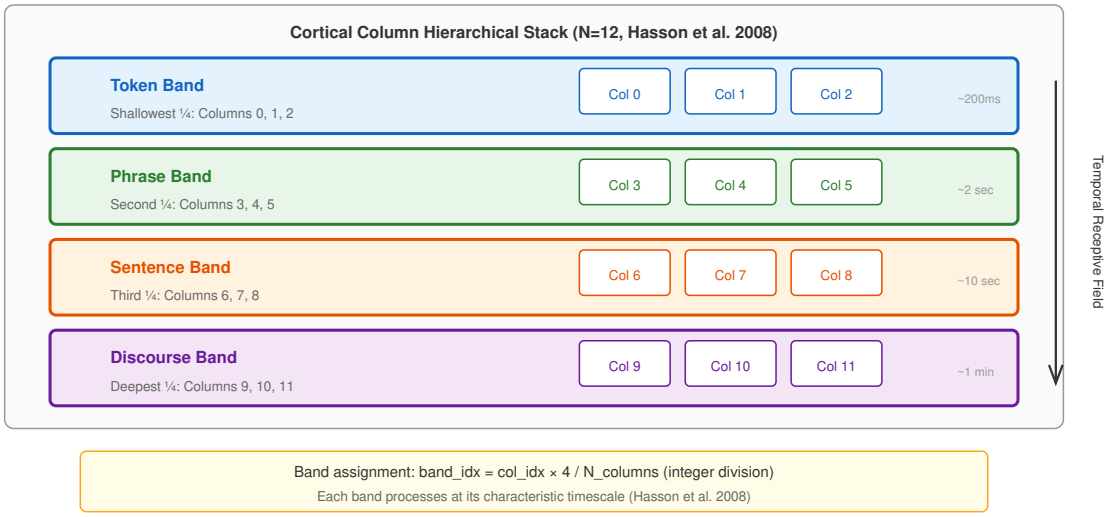


Figure 14: Claim 10 — Four Depth Bands

FIG. P7-11: Band-Aware Shrink Thresholds (Claim 11)

Shallow columns (primary sensory) require lower gate activity to be classified as underutilised

Depth Band	Scale Factor	Effective Threshold	Rationale
Token Band (shallowest ¼)	0.5	0.5 × 0.15 = 0.075 hardest to classify dead	Primary sensory (V1/A1) Always needed, never prune
Phrase Band (second ¼)	0.7	0.7 × 0.15 = 0.105 moderate protection	Compositional processing
Sentence Band (third ¼)	0.9	0.9 × 0.15 = 0.135 near baseline	Higher-level integration
Discourse Band (deepest ¼)	1.0	1.0 × 0.15 = 0.150 easiest to prune	Associative / abstract May be redundant

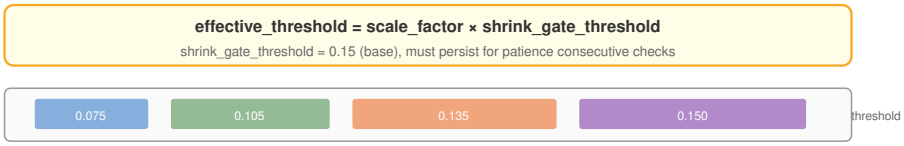


Figure 15: Claim 11 — Band-Aware Shrink Thresholds

FIG. P7-12: Merge — Functional Weight Cosine Similarity (Claim 12)

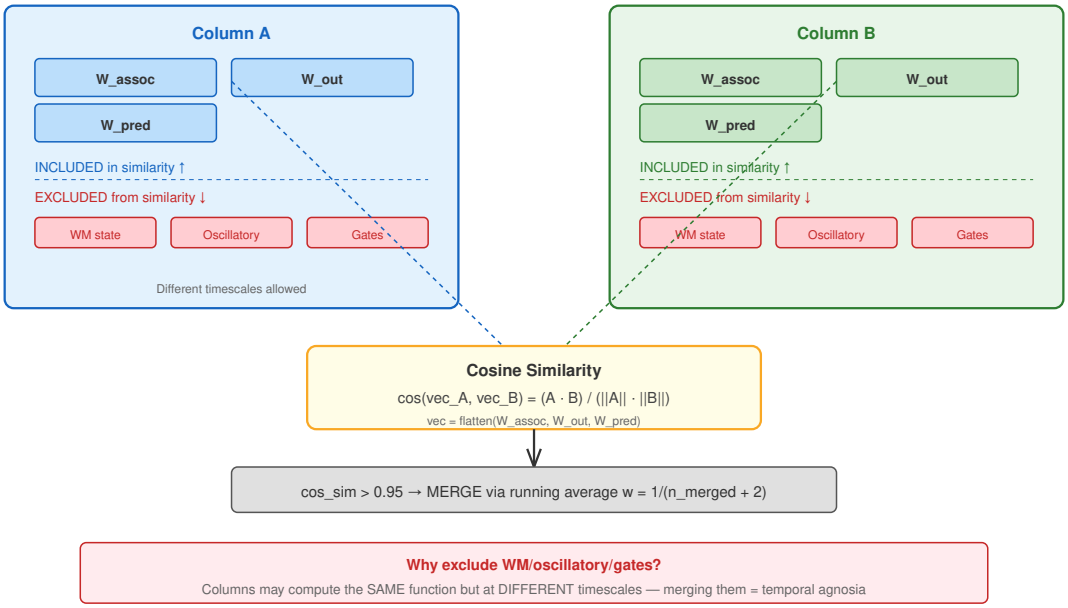


Figure 16: Claim 12 — Merge via Functional Weight Cosine Similarity

FIG. P7-13: Grow Operation — Depth-Band-Aware Insertion (Claim 13)

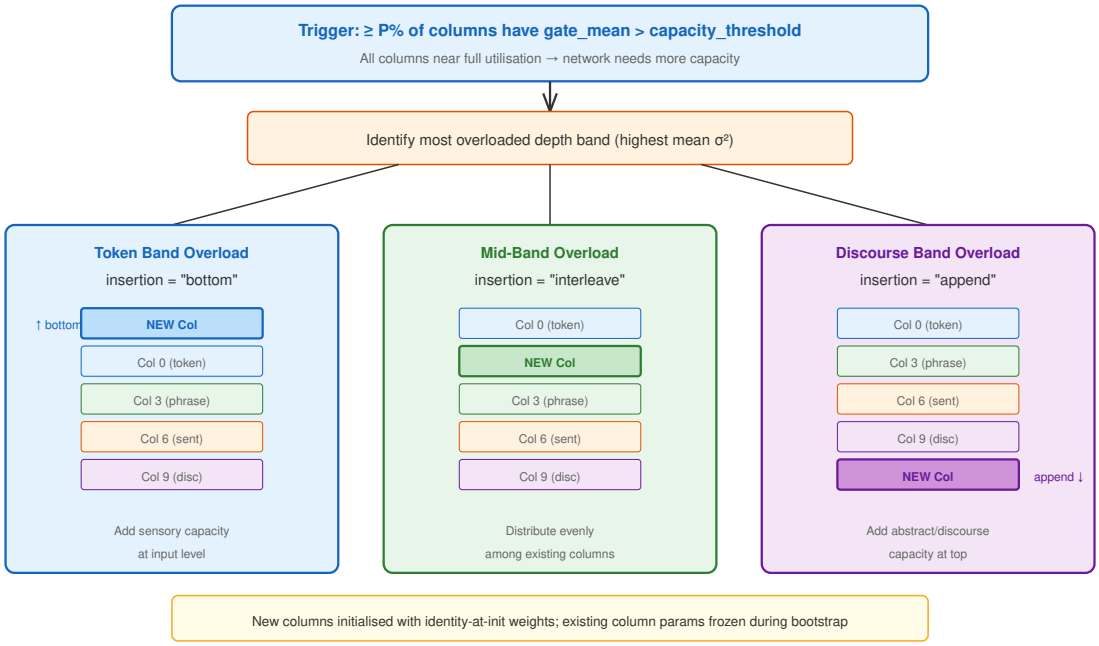


Figure 17: Claim 13 — Grow Operation with Depth-Band-Aware Insertion

FIG. P7-14: muP Hyperparameter Transfer (Claim 14)

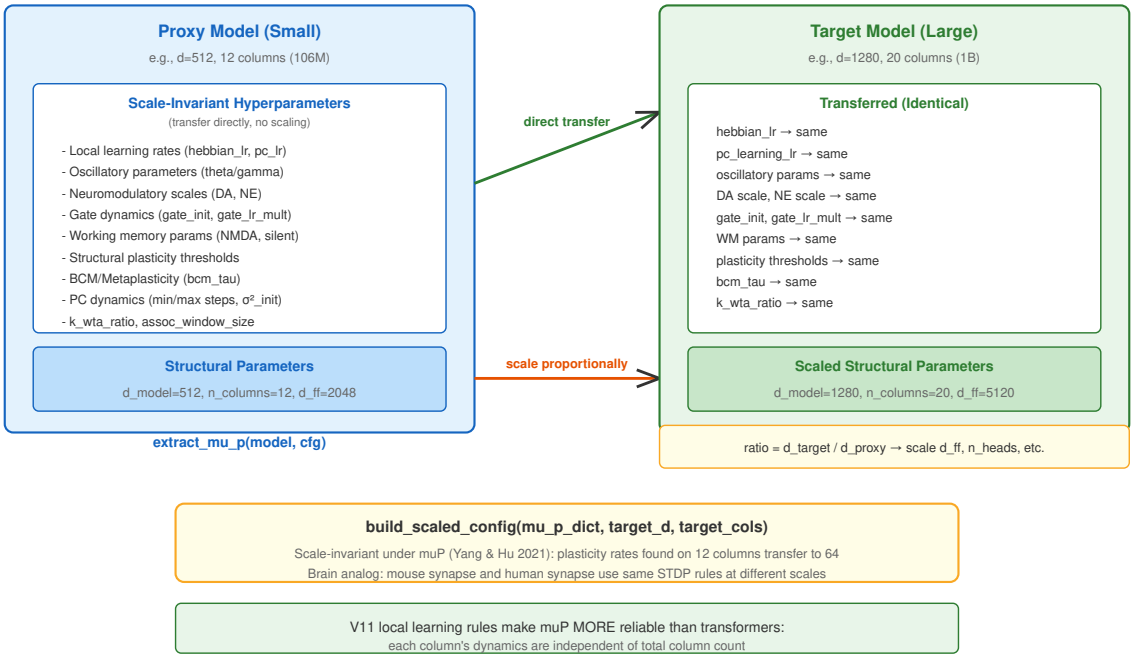


Figure 18: Claim 14 — muP Hyperparameter Transfer

FIG. P7-15: Neuromodulator Scale Transfer from Brainstem (Claim 15)

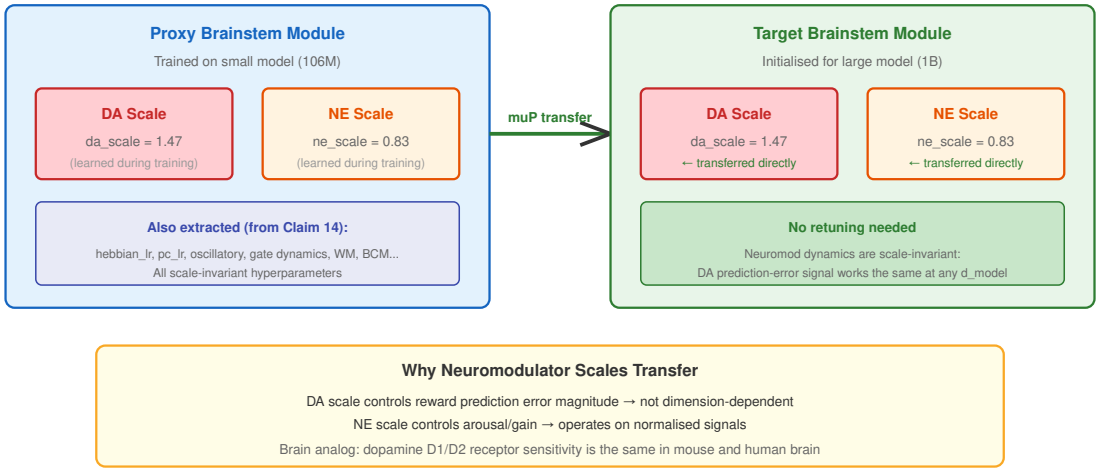


Figure 19: Claim 15 — Neuromodulator Scale Transfer from Brainstem

FIG. P7-16: σ²-Guided Deployment System (Claim 16)

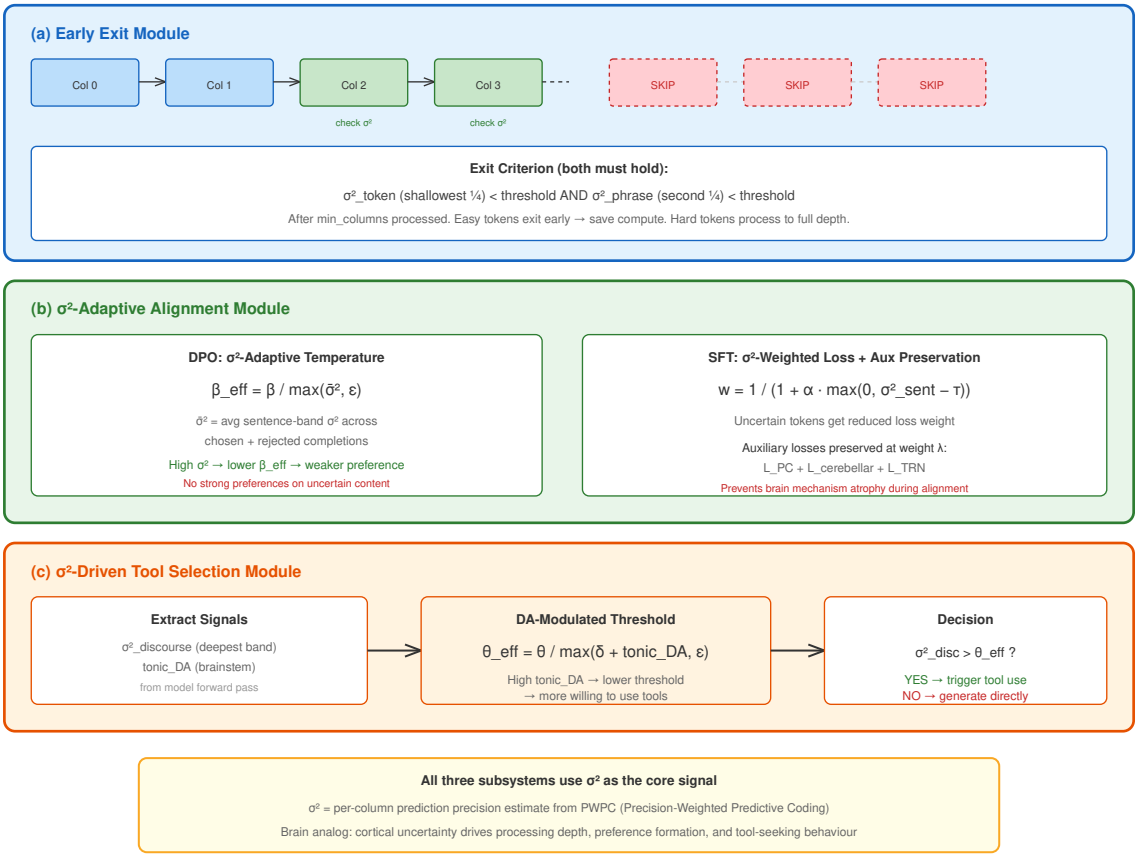


Figure 20: Claim 16 — σ^2-Guided Deployment System

FIG. P7-17: Early Exit WM State Reset (Claim 17)

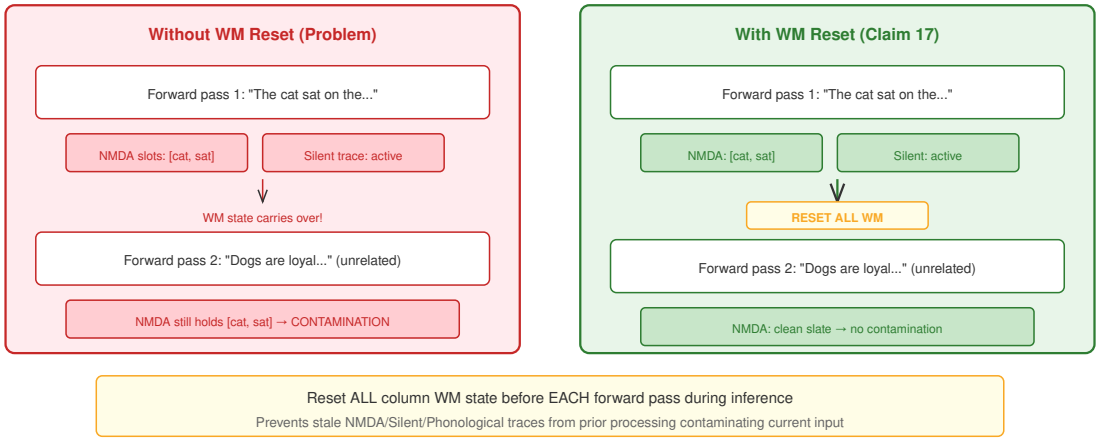


Figure 21: Claim 17 — Early Exit Working Memory State Reset

FIG. P7-18: Auxiliary Loss Preservation During SFT (Claim 18)

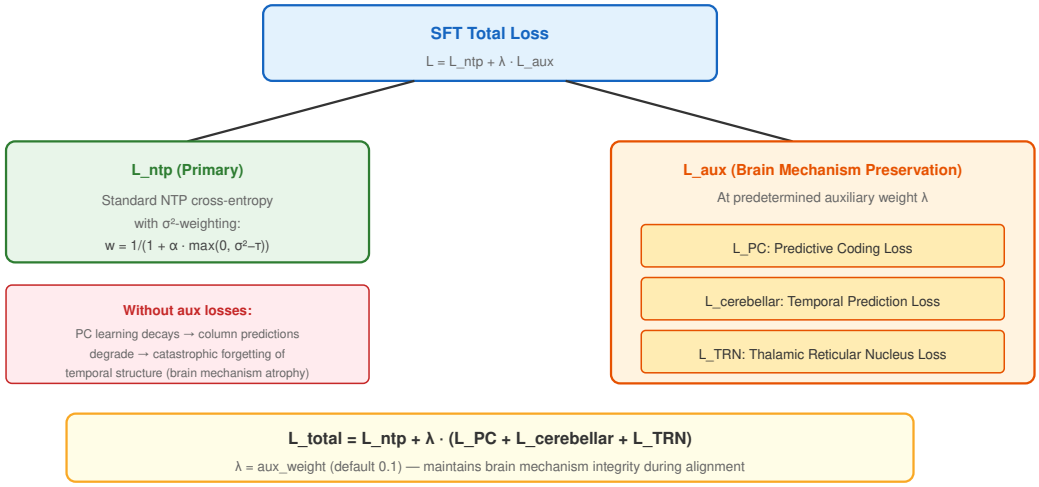


Figure 22: Claim 18 — Auxiliary Loss Preservation During SFT

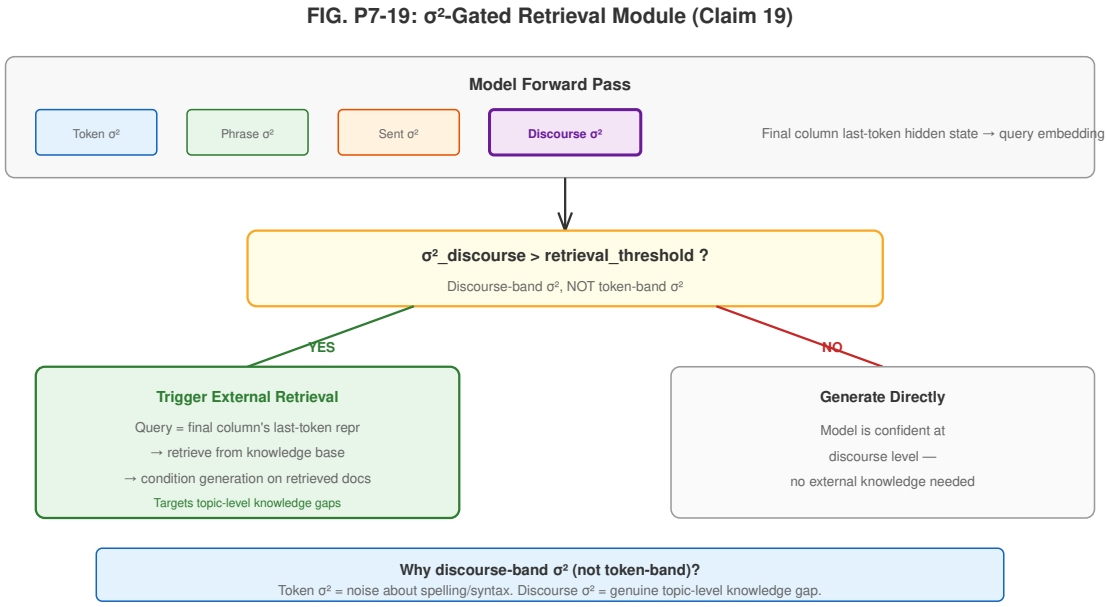


Figure 23: Claim 19 — σ^2-Gated Retrieval Module

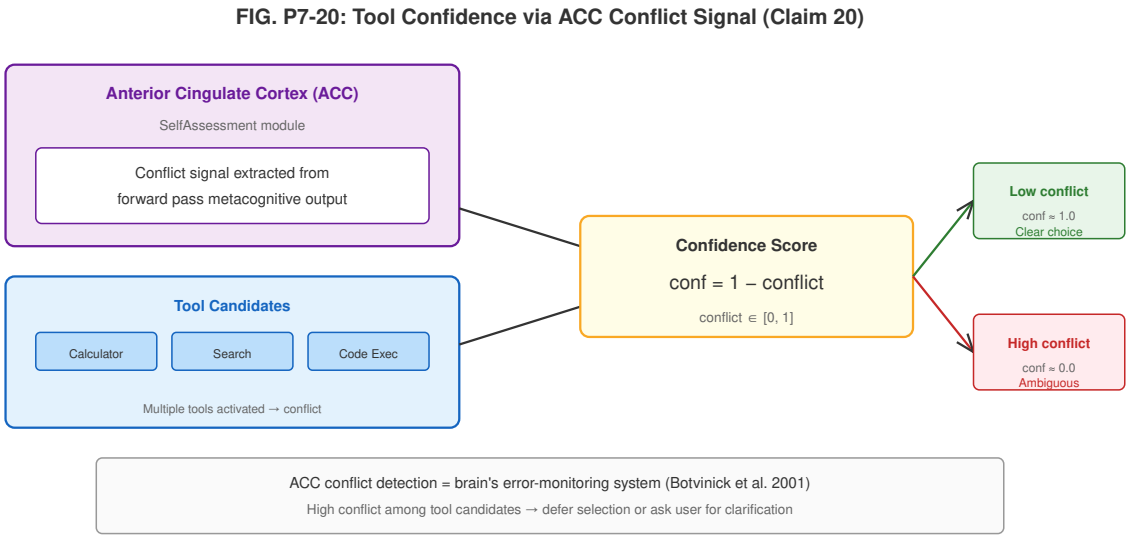


Figure 24: Claim 20 — Tool Confidence via ACC Conflict Signal

FIG. P7-21: Early Exit Band Assignment (Claim 21)

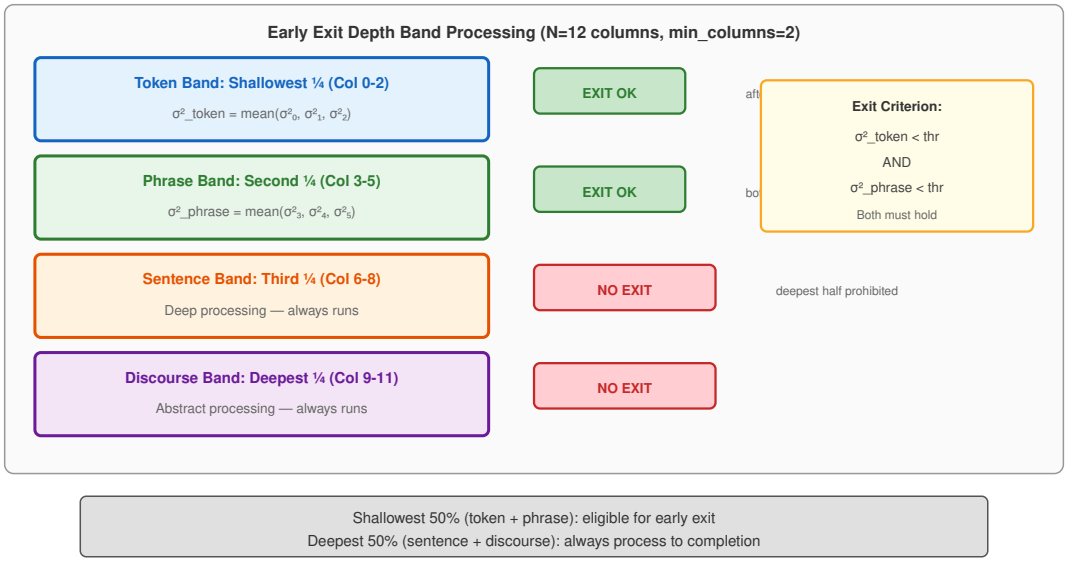


Figure 25: Claim 21 — Early Exit Band Assignment

FIG. P7-22: Parallel Prefix Scan Kernel (Claim 22)

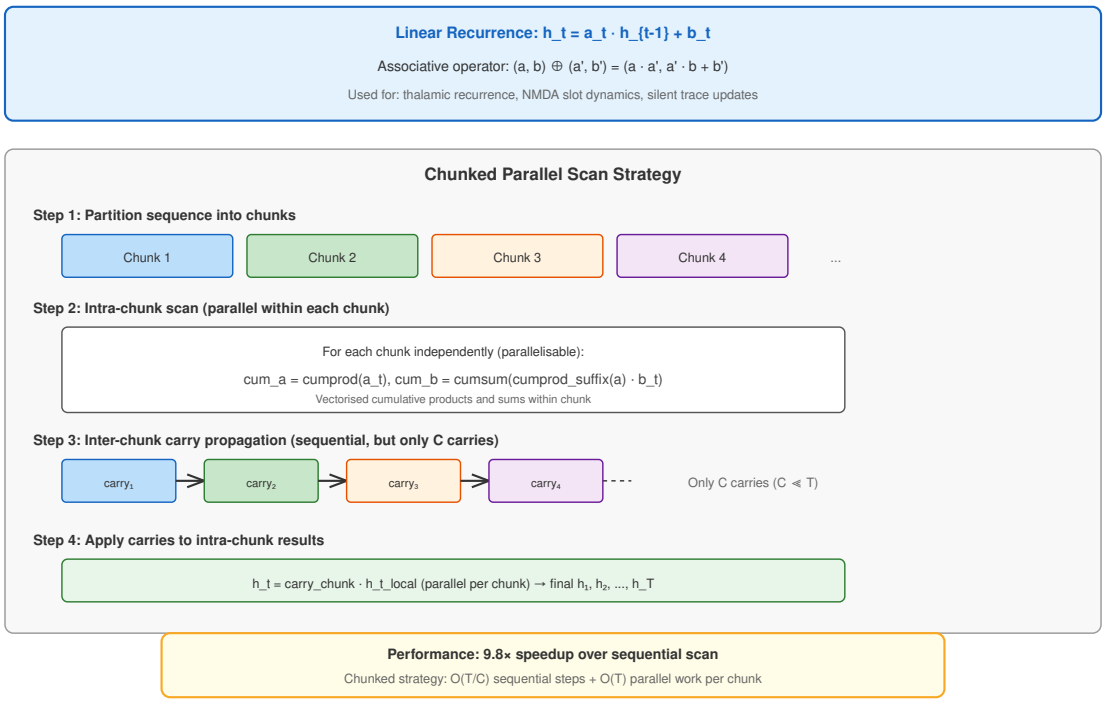


Figure 26: Claim 22 — Parallel Prefix Scan Kernel

FIG. P7-23: Windowed Hebbian Scalar Decomposition Kernel (Claim 23)

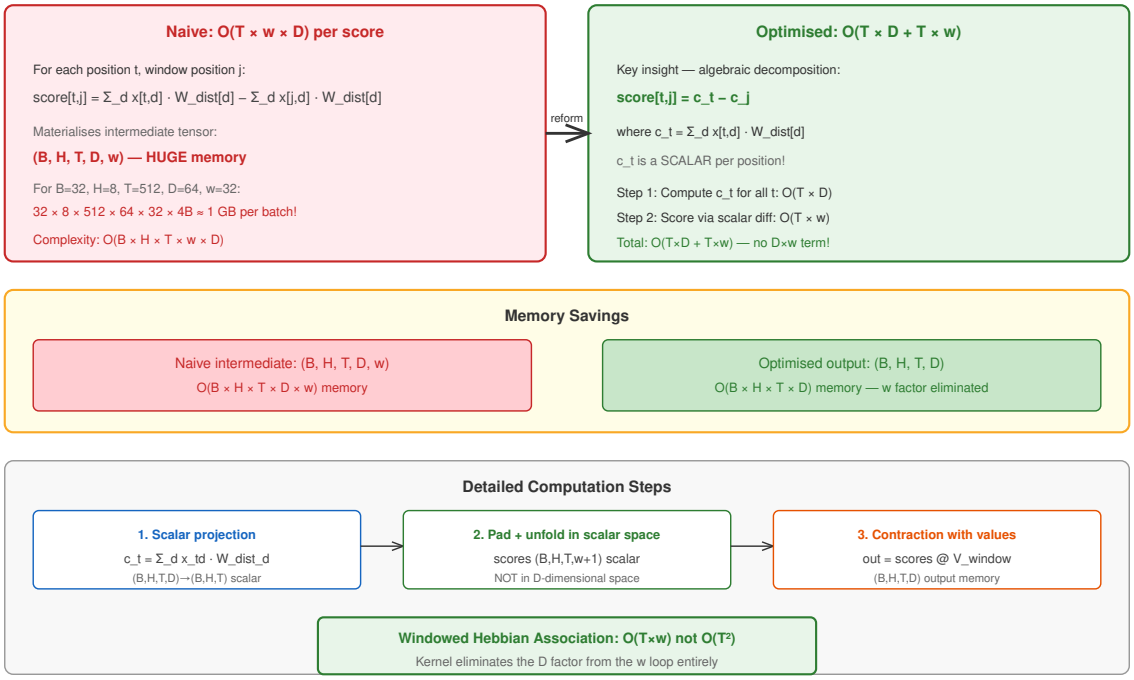


Figure 27: Claim 23 — Windowed Hebbian Scalar Decomposition Kernel

FIG. P7-24: Windowed Score via Scalar Pad-and-Unfold (Claim 24)

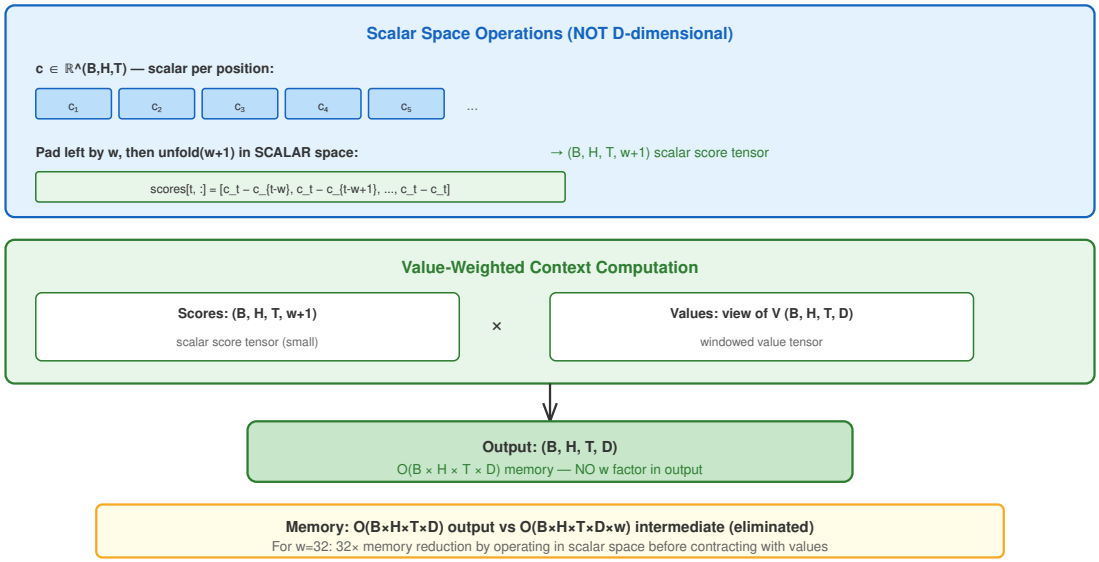


Figure 28: Claim 24 — Windowed Score via Scalar Pad-and-Unfold

FIG. P7-25: NMDA Parallel Scan Kernel (Claim 25)

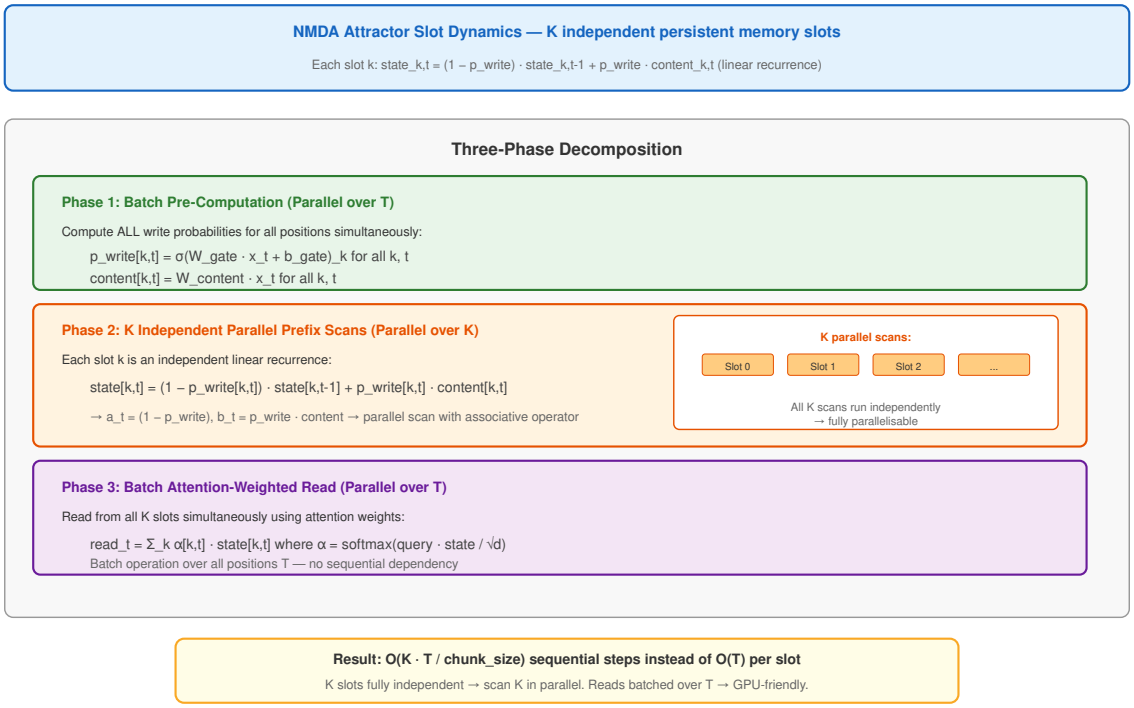


Figure 29: Claim 25 — NMDA Parallel Scan Kernel

FIG. P7-26: Non-Transitory Computer-Readable Medium (Claim 26)

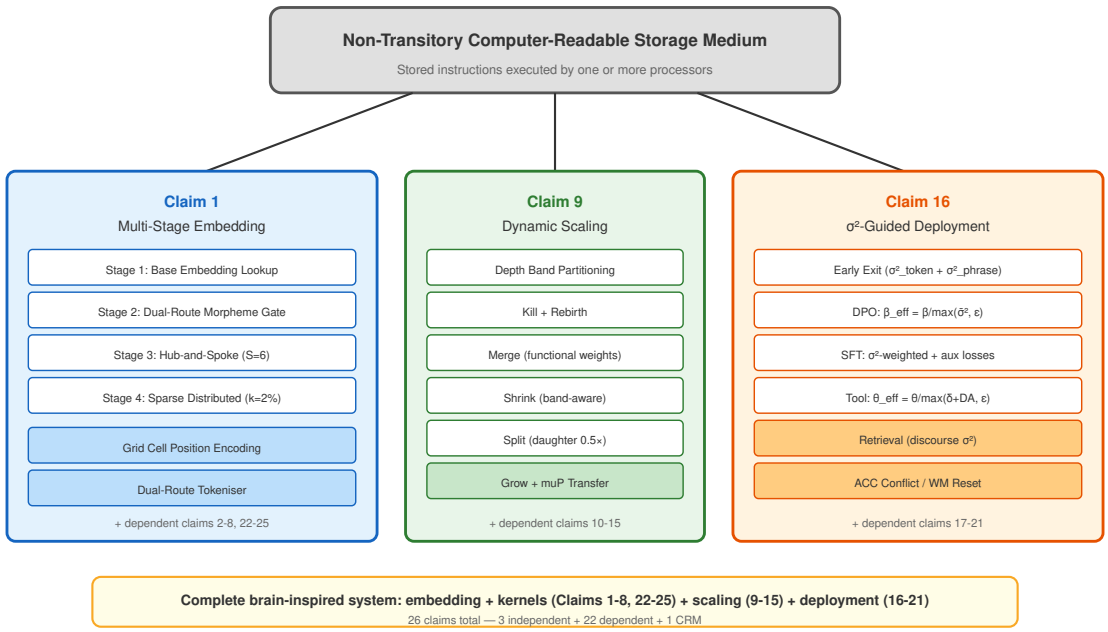


Figure 30: Claim 26 — Non-Transitory Computer-Readable Medium

5. DETAILED DESCRIPTION

The following description sets forth specific embodiments of the invention with sufficient detail to enable one skilled in the art to practise the invention. The embodiments are described in the context of a cortical column network (CCN) for natural language processing, but the principles disclosed herein are applicable to any neural network architecture that processes sequential data through a hierarchical stack of processing units.

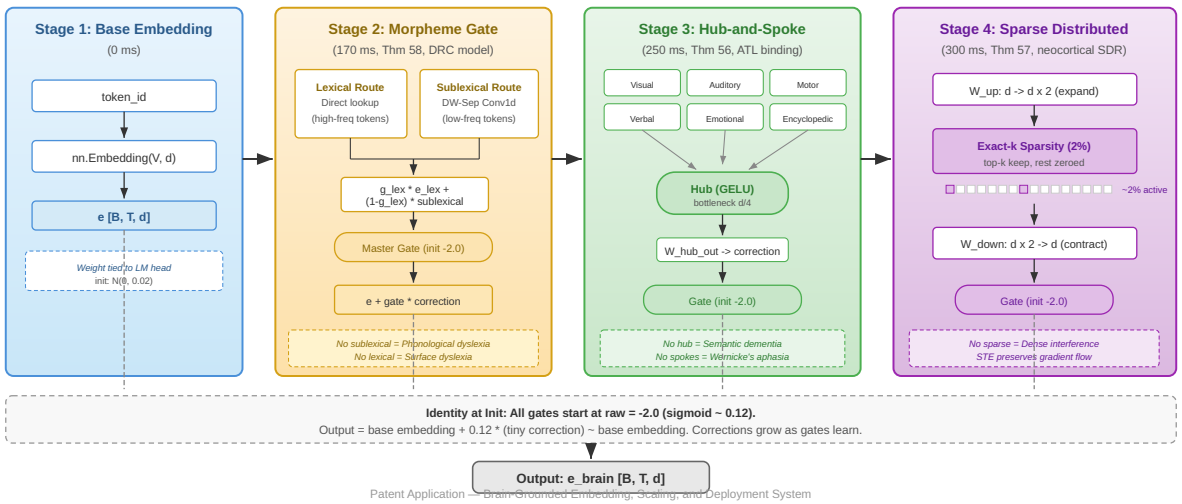
Throughout this description, the following notation is employed:

- B denotes the batch size, T the sequence length, d the model dimension, and V the vocabulary size.
- σ^2 denotes the per-token prediction precision estimate produced by the cortical column's Layer 5 output sublayer.
- $\text{sigmoid}(\cdot)$ denotes the logistic sigmoid function.
- $\text{softplus}(\cdot)$ denotes the softplus function $\text{softplus}(x) = \log(1 + e^x)$.
- $\text{ReLU}(\cdot)$ denotes the rectified linear unit.

5.1. Four-Stage Brain Embedding Pipeline

Referring to Figure 1, the brain embedding pipeline transforms integer token identifiers into dense continuous representations through four cascaded stages, each producing a gated residual correction to the evolving representation.

FIG. 21



Patent Application — Brain-Grounded Embedding, Scaling, and Deployment System

Figure 31: Four-stage brain embedding pipeline: base lookup → dual-route morphological composition → hub-and-spoke semantic integration → sparse distributed projection.

5.1.1. *Stage 1: Base Embedding Lookup*

A standard embedding table $\mathbf{E} \in \mathbb{R}^{V \times d}$ maps each token identifier t_i to a d-dimensional vector:

$$\mathbf{e}_i = \mathbf{E}[t_i] \in \mathbb{R}^d \tag{1}$$

The embedding weights are initialised with $\mathcal{N}(0, 0.02^2)$ and are tied with the language model output head for weight sharing.

5.1.2. *Stage 2: Dual-Route Morphological Composition (Theorem 58)*

Referring to Figure 2, this stage implements the Dual-Route Cascaded (DRC) reading model of Coltheart et al. (2001), which posits that the brain accesses words through two parallel routes: a lexical route for familiar words (direct whole-word lookup) and a sublexical route for novel or low-frequency words (compositional analysis of subword structure).

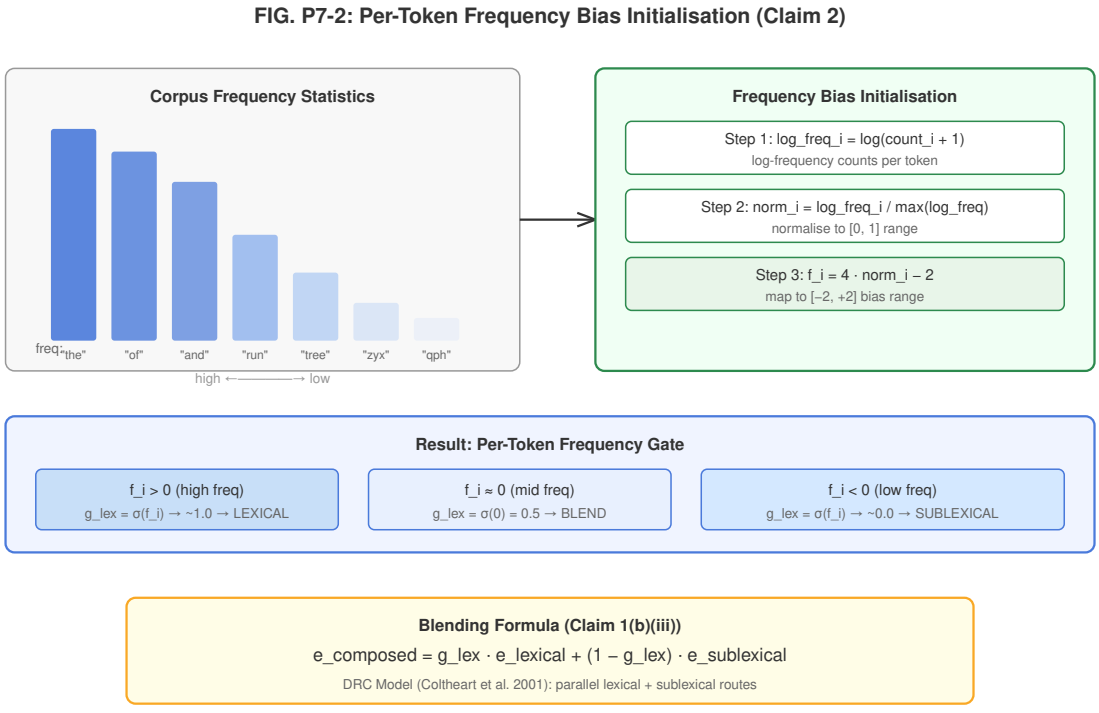


Figure 32: Per-token frequency bias initialisation for the dual-route morphological composition gate, showing corpus frequency statistics mapped to the $[-2, +2]$ bias range.

Sublexical route. The sublexical route computes a compositional representation from local context using a causal depthwise-separable convolution:

$$\mathbf{e}_{\text{sub}} = \text{PW}(\text{DW}_{\text{causal}}(\mathbf{e})) \tag{2}$$

where $\text{DW}_{\text{causal}}$ is a causal depthwise convolution (Conv1d with kernel size $K = 3$, groups $= d$, left-padding of $K - 1$ positions, no right-padding) and PW is a pointwise linear projection ($d \to d$, no bias). The causal constraint ensures that the sublexical composition at position t depends only on positions $t - K + 1, \ldots, t$, preventing future information leakage. The depthwise convolution preserves per-channel identity (each output channel depends only on its own input channel across the kernel window), while the pointwise projection enables cross-channel mixing.

The depthwise convolution is initialised near-identity: the rightmost kernel position (corresponding to the current token) is initialised to 1.0, and all other positions are initialised with small Gaussian noise ($\sigma = 0.01$). The pointwise projection is initialised as the identity matrix plus small noise ($\sigma = 0.01$). This ensures that at initialisation, $\mathbf{e}_{\text{sub}} \approx \mathbf{e}$ (the sublexical route outputs approximately the same embedding as the lexical route).

Per-token frequency gate. A learnable frequency bias $f_i \in \mathbb{R}$ for each token $i \in \{0, \ldots, V-1\}$ controls the route preference:

$$g_{\text{lex},i} = \text{sigmoid}(f_i) \tag{3}$$

High-frequency tokens have positive f_i (favouring the lexical route), while low-frequency tokens have negative f_i (favouring the sublexical route). The frequency biases may be initialised from corpus statistics by normalising log-frequency counts to the range $[-2, 2]$:

$$f_i \leftarrow 4.0 \cdot \frac{\log(1 + c_i)}{\max_j \log(1 + c_j)} - 2.0 \tag{4}$$

where c_i is the count of token i in the training corpus. Without corpus-based initialisation, all frequency biases are initialised to zero ($g_{\text{lex}} = 0.5$, equal blend).

The frequency bias parameter is named `freq_logit` (not `gate_raw`) to distinguish it from internal gate parameters and to enable separate learning rate grouping in optimiser configurations.

Dual-route blend. The two routes are blended per-token:

$$\mathbf{e}_{\text{composed}} = g_{\text{lex},i} \cdot \mathbf{e}_i + (1 - g_{\text{lex},i}) \cdot \mathbf{e}_{\text{sub},i} \tag{5}$$

Master identity-at-init(Stale from transformer era check with /nct-research) gate. A scalar gate $g_{\text{master}} = \text{sigmoid}(\gamma)$ with γ initialised to -2.0 ($g_{\text{master}} \approx 0.12$) controls the overall correction magnitude:

$$\mathbf{e}' = (1 - g_{\text{master}}) \cdot \mathbf{e} + g_{\text{master}} \cdot \mathbf{e}_{\text{composed}} \tag{6}$$

At initialisation, because $g_{\text{master}} \approx 0.12$ and $\mathbf{e}_{\text{composed}} \approx \mathbf{e}$ (from near-identity convolution initialisation), the output is $\mathbf{e}' \approx \mathbf{e}$.

Disease mapping validation. Ablation of the sublexical route produces a deficit analogous to phonological dyslexia (inability to read novel words). Ablation of the lexical route produces a deficit analogous to surface dyslexia (inability to read irregular high-frequency words). These predictable failure modes validate the biological grounding of the architecture.

5.1.3. Stage 3: Hub-and-Spoke Semantic Integration (Theorem 56)

Referring to Figure 3, this stage implements the hub-and-spoke model of semantic cognition (Patterson et al., 2007; Lambon Ralph et al., 2017), in which modality-specific cortical regions (spokes) contribute partial representations that are integrated through a shared amodal hub in the anterior temporal lobe (ATL).

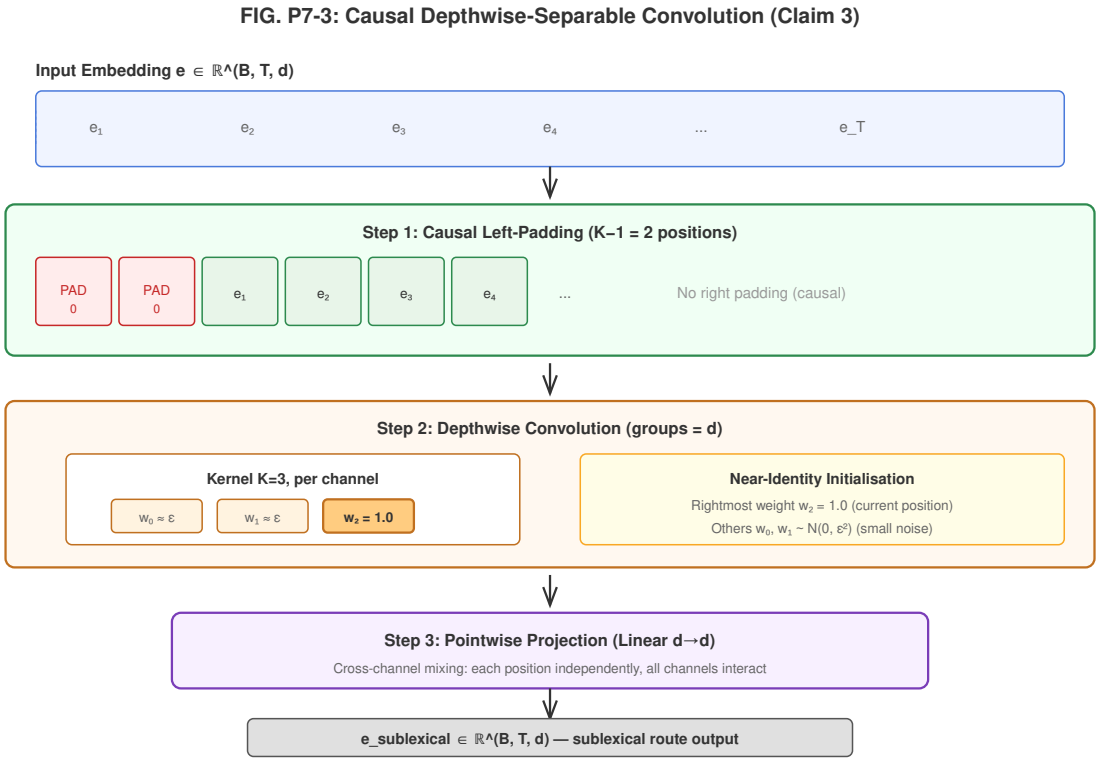


Figure 33: Hub-and-spoke semantic integration: $S = 6$ modality-specific spokes projected independently, concatenated into a nonlinear hub bottleneck, with gated residual correction.

Spoke decomposition. The embedding vector $\mathbf{e}' \in \mathbb{R}^d$ is partitioned along the feature dimension into $S = 6$ contiguous subvectors (spokes), corresponding to visual, auditory, motor, verbal, emotional, and encyclopedic modalities:

$$\mathbf{e}' = [\mathbf{e}'_1 \| \mathbf{e}'_2 \| \cdots \| \mathbf{e}'_S], \quad \mathbf{e}'_s \in \mathbb{R}^{d_s}, \quad \sum_{s=1}^{S} d_s = d \tag{7}$$

Each spoke has dimension $d_s = \lfloor d/S \rfloor$, with the last spoke absorbing any remainder. Each spoke is independently projected to a uniform output dimension $d_{\text{out}} = \lfloor d/(2S) \rfloor$:

$$\mathbf{h}_s = \mathbf{W}_s \cdot \mathbf{e}'_s \in \mathbb{R}^{d_{\text{out}}}, \quad s = 1, \ldots, S \tag{8}$$

Hub bottleneck. The projected spokes are concatenated and fed through a nonlinear bottleneck hub:

$$\text{correction} = \mathbf{W}_{\text{out}} \cdot \text{GELU}(\mathbf{W}_{\text{in}} \cdot [\mathbf{h}_1 \| \cdots \| \mathbf{h}_S] + \mathbf{b}_{\text{in}}) + \mathbf{b}_{\text{out}} \tag{9}$$

where $\mathbf{W}_{\text{in}} \in \mathbb{R}^{d_{\text{mid}} \times (S \cdot d_{\text{out}})}$ and $\mathbf{W}_{\text{out}} \in \mathbb{R}^{d \times d_{\text{mid}}}$ with $d_{\text{mid}} = \lfloor S \cdot d_{\text{out}}/2 \rfloor$. The bottleneck forces cross-modal integration: the hub must compress multimodal information through a narrower representation, preventing individual modalities from dominating.

Gated residual correction. A scalar gate $g_{\text{hub}} = \text{sigmoid}(\gamma_{\text{hub}})$ with γ_{hub} initialised to -2.0 controls the correction:

$$\mathbf{e}'' = \mathbf{e}' + g_{\text{hub}} \cdot \text{correction} \tag{10}$$

At initialisation, $g_{\text{hub}} \approx 0.12$ and $\mathbf{W}_{\text{out}}$ is initialised with $\mathcal{N}(0, 0.01^2)$, producing a near-zero correction.

Disease mapping validation. Ablation of the hub bottleneck produces a deficit analogous to semantic dementia (pan-modal degradation of semantic knowledge from ATL atrophy). Ablation of individual spokes produces modality-specific semantic deficits analogous to Wernicke's aphasia.

5.1.4. Stage 4: Sparse Distributed Projection (Theorem 57)

Referring to Figure 4, this stage implements the neocortical sparse population coding principle (Ahmad and Hawkins, 2016; Shoham et al., 2006), in which at most 2% of neurons are active at any time, producing high-capacity noise-robust representations.

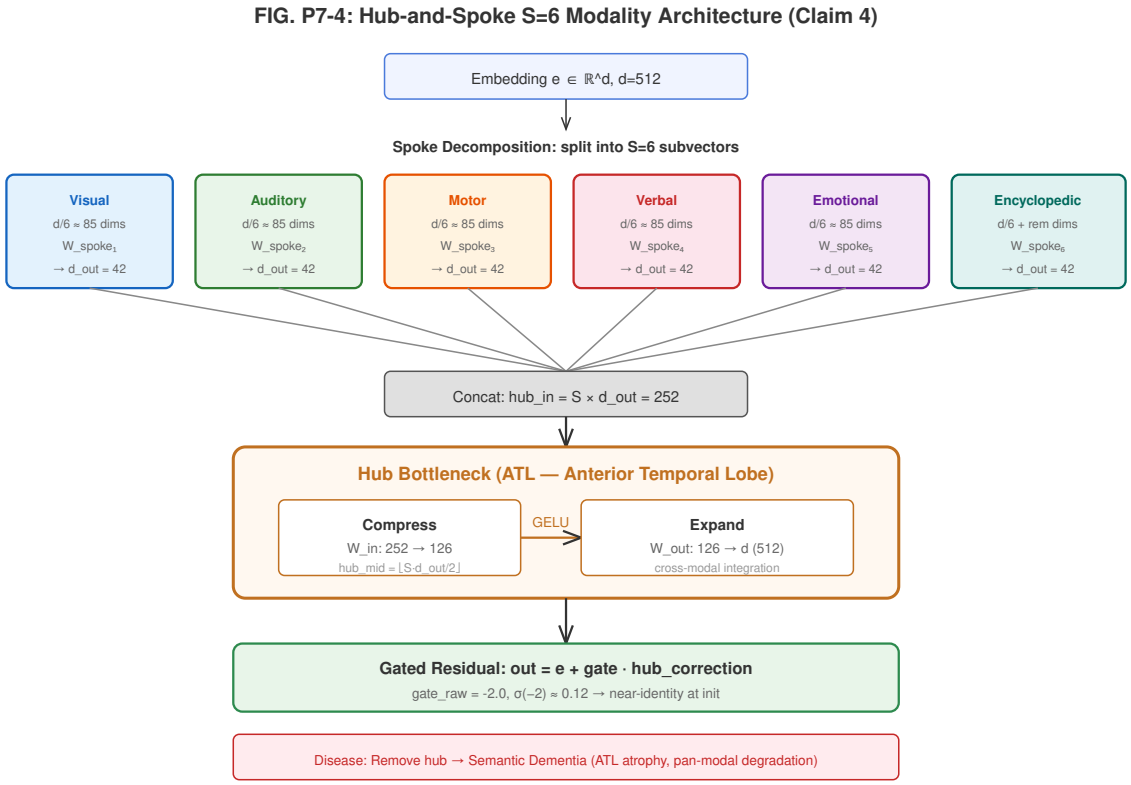


Figure 34: Sparse distributed projection: expansion via W_{up}, exact-k sparsity ($k = \lceil d_{\text{sdr}} \times 0.02 \rceil = 21$), straight-through estimator, and contraction via W_{down}.

Expansion and sparsification. The embedding is projected to an expanded space, sparsified, and projected back:

$$\mathbf{h} = \mathbf{W}_{\text{up}} \cdot \mathbf{e}'' \in \mathbb{R}^{d_{\text{sdr}}}, \quad d_{\text{sdr}} = d \cdot r \tag{11}$$

$$\mathbf{h}_{\text{relu}} = \text{ReLU}(\mathbf{h}) \tag{12}$$

$$\mathbf{h}_{\text{sparse}} = \text{top-}k(\mathbf{h}_{\text{relu}}, k), \quad k = \lceil d_{\text{sdr}} \cdot 0.02 \rceil \tag{13}$$

where top-k retains only the k largest values and zeros all others. The expansion factor $r = 2$ and sparsity fraction 0.02 yield, for $d = 512$: $d_{\text{sdr}} = 1024$, $k = \lceil 1024 \times 0.02 \rceil = 21$. The combinatorial pattern capacity is $\binom{1024}{21} > 10^{45}$.

Straight-through estimator. Because top-k is non-differentiable, a straight-through estimator (STE) is used for gradient computation:

$$\mathbf{h}_{\text{sparse}} = \mathbf{h}_{\text{relu}} + (\mathbf{h}_{\text{sparse}} - \mathbf{h}_{\text{relu}}).\text{detach}() \tag{14}$$

This allows gradients to flow through $\mathbf{h}_{\text{relu}}$ as if sparsity were not applied, while the forward pass uses the exact sparse representation.

Projection and gated correction. The sparse representation is projected back to d dimensions and added as a gated correction:

$$\mathbf{e}_{\text{out}} = \mathbf{e}'' + g_{\text{sdr}} \cdot \mathbf{W}_{\text{down}} \cdot \mathbf{h}_{\text{sparse}} \tag{15}$$

with $g_{\text{sdr}} = \text{sigmoid}(\gamma_{\text{sdr}})$, γ_{sdr} initialised to -2.0.

5.2. Grid Cell Position Encoding

Referring to Figure 5, the grid cell position encoding replaces fixed RoPE with a biologically-grounded system inspired by the entorhinal cortex grid cell system (Hafting et al., 2005).

FIG. 22

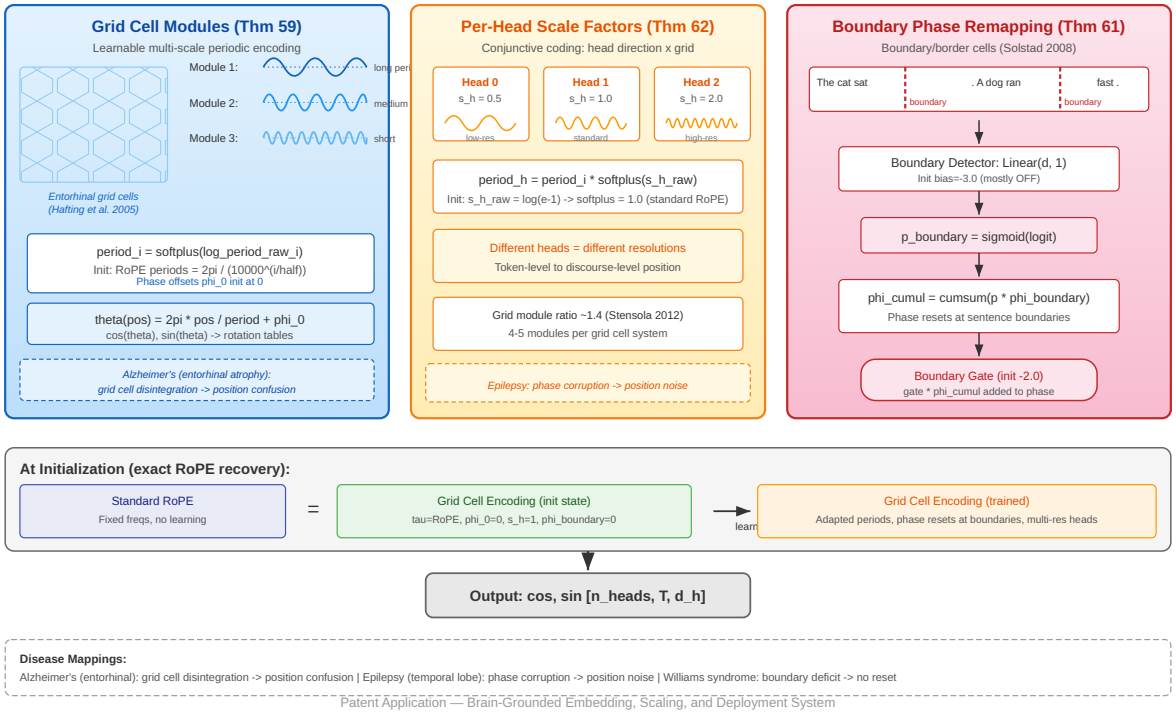


Figure 35: Grid-cell-inspired position encoding: learnable periods (Theorem 59), per-head scale factors (Theorem 62), and boundary-aware phase remapping (Theorem 61).

5.2.1. Learnable Grid Cell Module Periods (Theorem 59)

Standard RoPE uses fixed periods $\tau_i = 2\pi/(10000^{i/d_h})$ for each pair of dimensions in the per-head space $\mathbb{R}^{d_h}$. The present invention replaces these with learnable periods:

$$\tau_i = \text{softplus}(\theta_i^{\text{raw}}), \quad i = 0, \ldots, d_h/2 - 1 \tag{16}$$

where θ_i^{raw} is initialised to the inverse softplus of the standard RoPE period via:

$$\theta_i^{\text{raw}} = \tau_i^{\text{RoPE}} + \log(1 - e^{-\tau_i^{\text{RoPE}}}) \tag{17}$$

This ensures exact recovery of standard RoPE at initialisation while allowing the periods to adapt during training. Additionally, learnable phase offsets $\phi_i^{(0)}$ (initialised to zero) provide per-dimension phase control.

5.2.2. Per-Head Scale Factors (Theorem 62)

Inspired by conjunctive coding in the entorhinal cortex (where grid cell firing fields are modulated by head direction), each attention/association head h receives its own period scale factor:

$$\tau_{h,i} = \tau_i \cdot s_h, \quad s_h = \text{softplus}(s_h^{\text{raw}}) \tag{18}$$

where s_h^{raw} is initialised to $\log(e-1) \approx 0.5414$ (the inverse softplus of 1.0), ensuring $s_h = 1.0$ at initialisation and therefore exact RoPE recovery. During training, different heads learn different scale factors, enabling multi-resolution position sensitivity.

5.2.3. Boundary-Aware Phase Remapping (Theorem 61)

Inspired by grid cell remapping at environmental boundaries (Fyhn et al., 2007), the position encoding detects sentence/paragraph boundaries in the input and triggers phase resets:

$$p_{\text{boundary},t} = \text{sigmoid}(\mathbf{W}_b \cdot \mathbf{x}_t + b_b) \in [0, 1] \tag{19}$$

where $\mathbf{W}_b \in \mathbb{R}^{1\times d}$ is initialised to zeros and $b_b = -3.0$ (boundary detection mostly OFF at initialisation). The cumulative boundary phase for each dimension i is:

$$\phi_{\text{cum},t,i} = \sum_{s=1}^{t} p_{\text{boundary},s} \cdot \phi_{\text{boundary},i} \tag{20}$$

where $\phi_{\text{boundary},i}$ (initialised to zero) is a learnable per-dimension boundary phase shift. A gate $g_{\text{bnd}} = \text{sigmoid}(\gamma_{\text{bnd}})$ with $\gamma_{\text{bnd}} = -2.0$ controls the strength of boundary remapping:

$$\alpha_{h,t,i} = \frac{2\pi \cdot t}{\tau_{h,i}} + \phi_i^{(0)} + g_{\text{bnd}} \cdot \phi_{\text{cum},t,i} \tag{21}$$

The final position encoding is RoPE-compatible: $(\cos\alpha, \sin\alpha)$ pairs are duplicated across both halves of the per-head dimension.

When boundary detection is active, the output tensors have shape $(B, n_{\text{heads}}, T, d_h)$ (batch-specific due to content-dependent boundaries). When inactive, the shape is $(n_{\text{heads}}, T, d_h)$.

5.3. Dual-Route Morphological Tokeniser

Referring to Figure 6, the NCT (Neural Cortical Tokeniser) implements the Coltheart DRC model at the tokenisation level, complementing the dual-route embedding composition described in

Section 5.1.2.

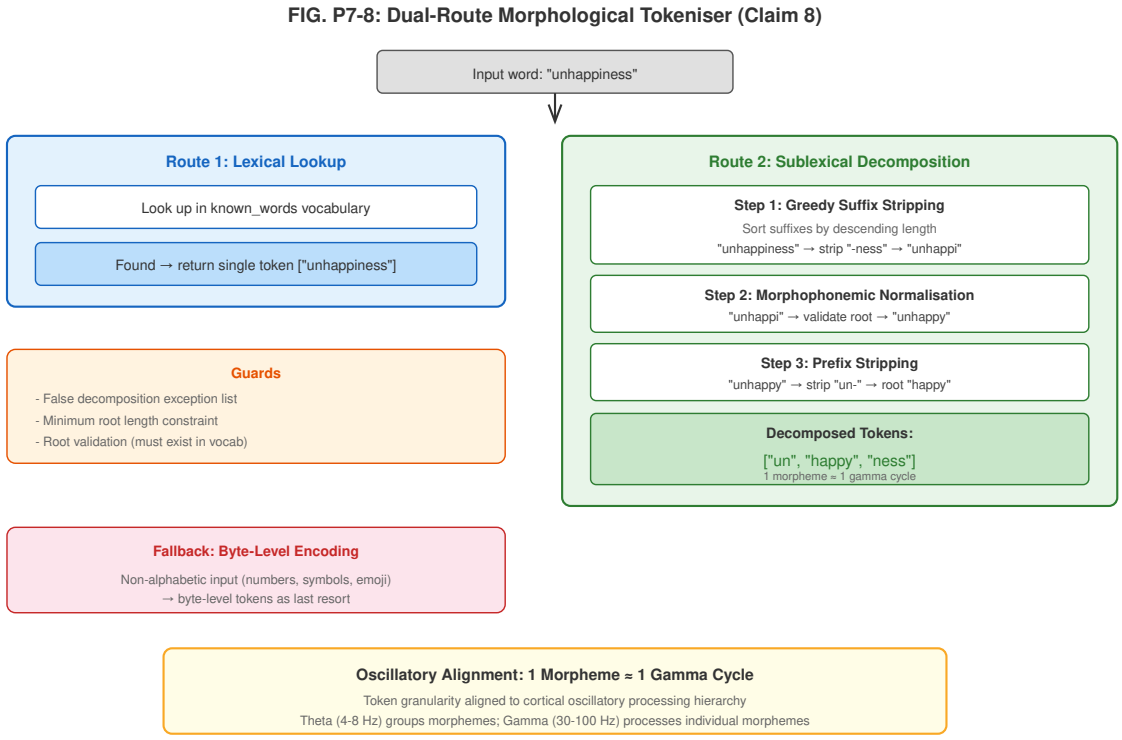


Figure 36: Dual-route morphological tokeniser: lexical route (whole-word lookup), sublexical route (greedy affix stripping), byte-level fallback, and oscillatory alignment.

5.3.1. Lexical Route

Known high-frequency words are stored in a vocabulary lookup table. When the input text contains a word present in the known-word dictionary, the lexical route returns the word as a single token. This mirrors the brain's fast lexical access for familiar words ($\sim$100ms in the Visual Word Form Area).

5.3.2. Sublexical Route

For words not in the known-word dictionary, greedy affix stripping decomposes the word into morphemes:

1. **Suffix stripping**: iteratively remove the longest matching suffix from a sorted suffix inventory (sorted by descending length for longest-match-first), checking at each step whether the remainder is a known root (with morphophonemic normalisation handling consonant doubling, e-dropping, y-to-i alternation, and f-to-v alternation). At most 3 suffixes are stripped.

2. **Prefix stripping**: iteratively remove the longest matching prefix from a sorted prefix inventory (also sorted by descending length). At most 2 prefixes are stripped.

3. **Root validation**: the final stem is validated against the known-word dictionary, with morphophonemic normalisation generating candidate canonical forms.

4. **False decomposition guard**: a curated exception list prevents incorrect decomposition of words whose orthographic form superficially matches affixes but whose morphological structure is monomorphemic (e.g., "uncle" does not decompose as "un" + "cle").

5. **Minimum root constraint**: a minimum root length of 3 characters prevents pathological over-decomposition.

5.3.3. Byte-Level Fallback

For non-alphabetic input (numbers, symbols, code), a byte-level encoding maps each byte to a dedicated token in the range [0, 255], ensuring lossless encoding of any input.

5.3.4. Oscillatory Alignment

Each token corresponds to approximately one morpheme, which in turn corresponds to approximately one gamma cycle ($\sim$1–2 temporal steps) in the cortical oscillatory hierarchy. Words span theta cycles ($\sim$8–16 temporal steps, 3–7 morphemes), and phrases span alpha cycles ($\sim$20 temporal steps). This alignment enables the oscillatory dynamics of the cortical column network to naturally segment and group tokens at biologically appropriate timescales.

5.4. Dynamic Column Scaling System (AutoScale)

Referring to Figure 7, the AutoScale system automatically modifies the number and configuration of cortical columns during training, driven by multi-scale prediction uncertainty signals.

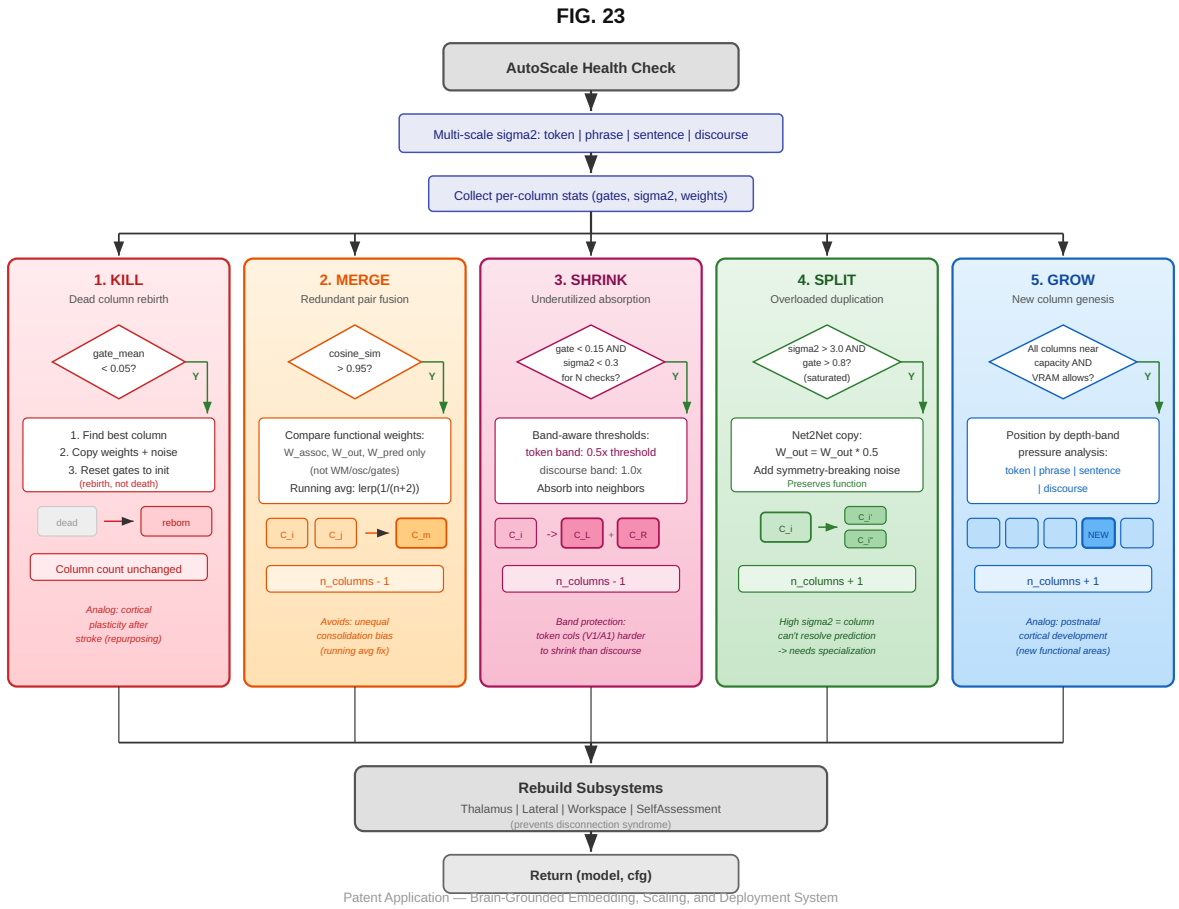


Figure 37: AutoScale dynamic column scaling: five operations (kill, merge, shrink, split, grow) with priority ordering and trigger conditions based on multi-scale σ^2 and gate statistics.

5.4.1. Multi-Scale σ^2 Depth Bands

The N cortical columns are partitioned into four depth bands following the temporal receptive field hierarchy of Hasson et al. (2008):

- **Token band** (columns 0 to $\lfloor N/4 \rfloor - 1$): morphological and lexical uncertainty.
- **Phrase band** (columns $\lfloor N/4 \rfloor$ to $\lfloor N/2 \rfloor - 1$): syntactic uncertainty.
- **Sentence band** (columns $\lfloor N/2 \rfloor$ to $\lfloor 3N/4 \rfloor - 1$): semantic uncertainty.
- **Discourse band** (columns $\lfloor 3N/4 \rfloor$ to $N - 1$): topic and narrative uncertainty.

Multi-scale σ^2 aggregates are computed by averaging the σ^2 values within each band. These aggregates drive the scaling decisions described below.

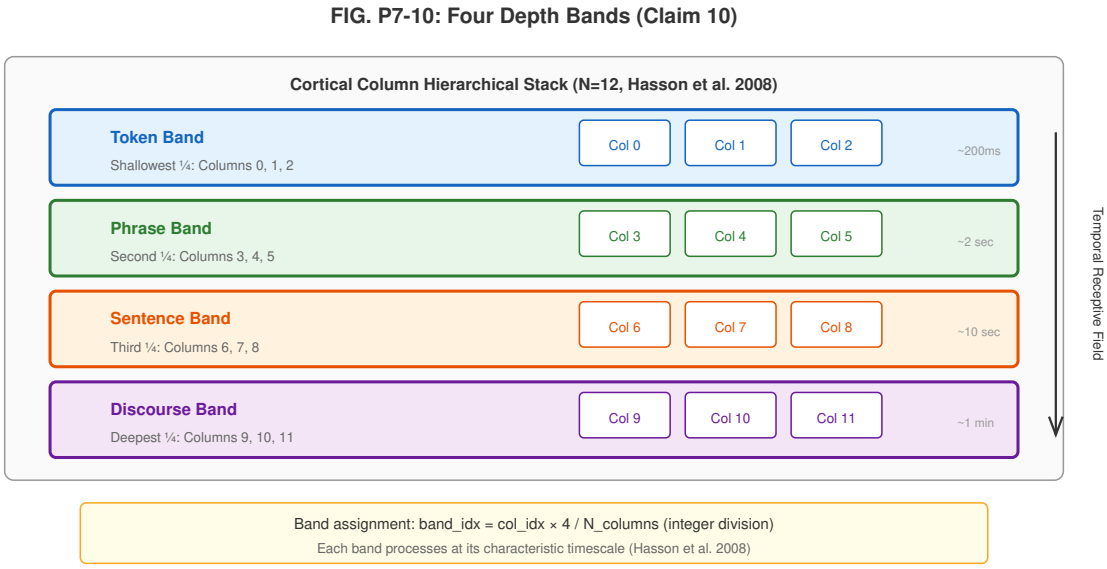


Figure 38: Four depth bands: token (shallowest $\frac{1}{4}$), phrase, sentence, and discourse (deepest $\frac{1}{4}$), with characteristic temporal receptive fields from $\sim$200 ms to $\sim$1 min.

5.4.2. Health Check Protocol

At every `check_interval` training steps (default: 10,000), AutoScale collects per-column diagnostics:

- Mean gate value across all gating mechanisms (proxy for mechanism utilisation).
- Mean σ^2 (proxy for remaining learning opportunity).
- Depth band assignment.
- Parameter norm (proxy for contribution magnitude).

The five scaling operations are applied in a fixed priority order: **kill** $\rightarrow$ **merge** $\rightarrow$ **shrink** $\rightarrow$ **split** $\rightarrow$ **grow**. This ordering ensures that dead and redundant columns are cleaned up before new capacity is added.

5.4.3. Operation 1: Kill and Rebirth

Referring to the kill path in Figure 7, columns with mean gate value below the dead gate threshold ($\tau_{\text{dead}} = 0.05$) are identified as dead. Each dead column is replaced by a deep copy of the best-performing column (highest mean gate) plus small Gaussian noise ($\sigma = 0.01$) for symmetry breaking. The column count does not change; only the dead column's parameters are replaced.

Biological analog: cortical areas that undergo complete deafferentation (loss of all input) are repurposed through cross-modal plasticity (e.g., visual cortex processing tactile input in blind individuals).

5.4.4. *Operation 2: Merge Redundant Columns*

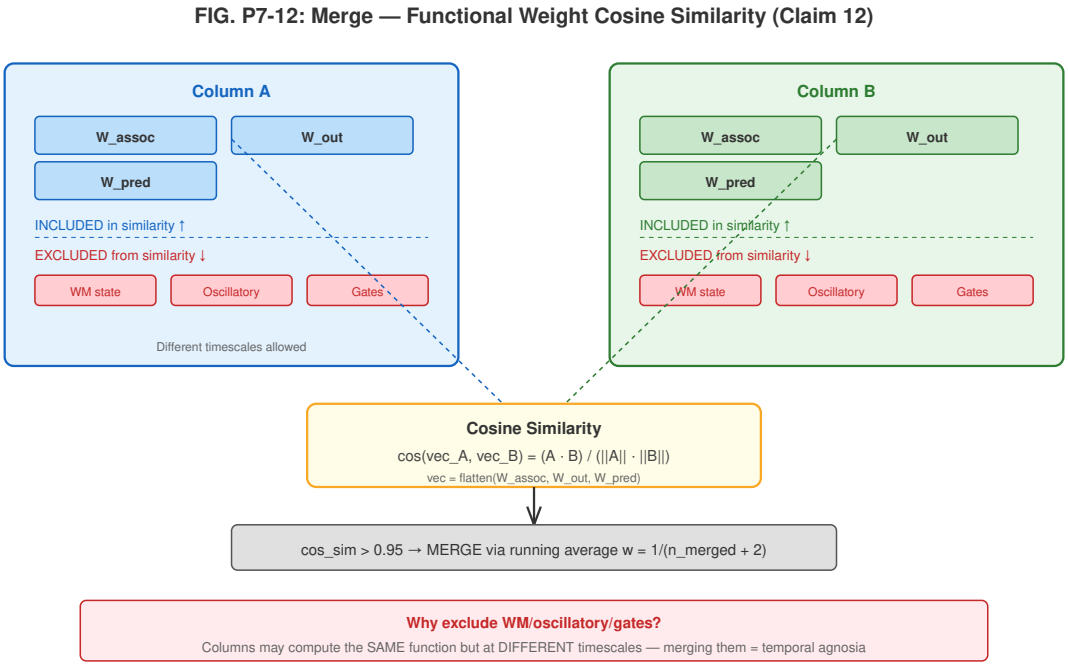


Figure 39: Merge operation: cosine similarity on functional weights only (W_{assoc}, W_{out}, W_{pred}), excluding WM/oscillatory/gate state to prevent temporal agnosia.

Referring to Figure 9, columns are deemed redundant when their *functional* weight vectors have cosine similarity above the merge threshold ($\tau_{\text{merge}} = 0.95$). Critically, the cosine similarity is computed only over functional weights — the association weights (W_{assoc}), output weights (W_{out}), and prediction weights (W_{pred}) — and not over working memory state, oscillatory parameters, or gate values. This distinction prevents the erroneous merging of columns that compute the same function but operate at different timescales (which would constitute temporal agnosia — the inability to distinguish events at different timescales).

When columns i and j are merged (j absorbed into i), the parameters of the surviving column are updated via running average:

$$\mathbf{p}_i \leftarrow (1 - w) \cdot \mathbf{p}_i + w \cdot \mathbf{p}_j, \quad w = \frac{1}{n_{\text{merged}} + 2} \tag{22}$$

where n_{merged} is the number of columns previously absorbed by column i. This running average ensures that all absorbed columns contribute equally to the survivor's parameters, preventing the exponential decay of early contributions that would result from repeated $\text{lerp}(0.5)$.

After all merges, the absorbed columns are removed, column indices are reassigned, and all subsystems (thalamus, lateral connections, workspace, self-assessment) are rebuilt for the new column count.

5.4.5. *Operation 3: Shrink Underutilized Columns*

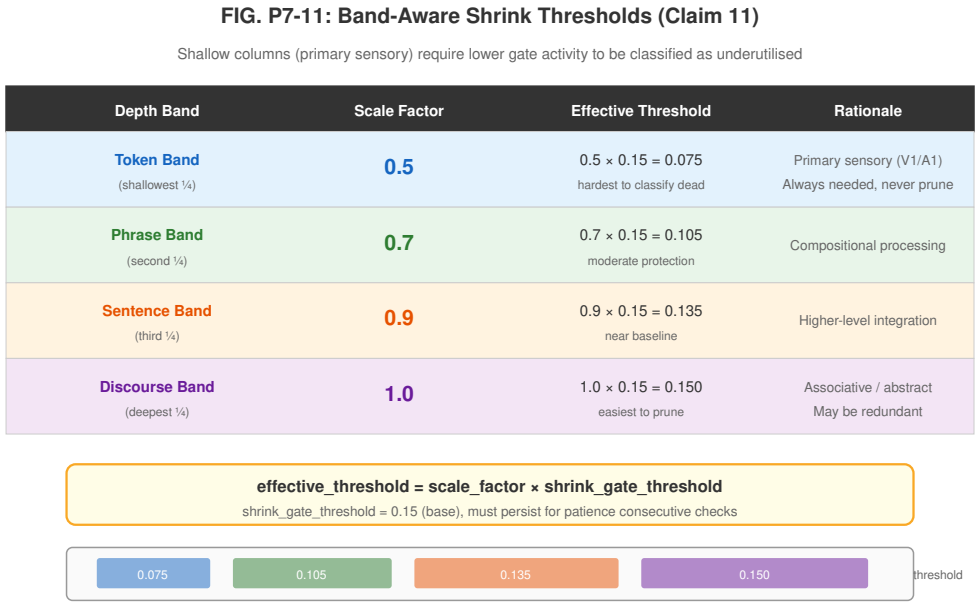


Figure 40: Band-aware shrink thresholds: token-band columns (primary sensory) receive 0.5× threshold scaling, discourse-band columns receive 1.0×.

Referring to Figure 10, a column is classified as underutilized when its mean gate value falls below the shrink gate threshold *and* its σ^2 falls below the shrink σ^2 threshold, persisting for at least `shrink_patience` consecutive health checks (default: 3).

A key innovation is depth-band-aware thresholds. Shallow columns (token band) process primary sensory information analogous to V1/A1 cortical areas, which the brain prunes less aggressively than associative areas. The effective thresholds are scaled:

Depth Band	Gate Scale	Effective Threshold
Token	0.5	$0.15 \times 0.5 = 0.075$
Phrase	0.7	$0.15 \times 0.7 = 0.105$
Sentence	0.9	$0.15 \times 0.9 = 0.135$
Discourse	1.0	$0.15 \times 1.0 = 0.150$

Before removal, each underutilised column's learned knowledge is absorbed into its nearest active neighbour by weighted averaging: the neighbour retains 80% of its own parameters and incorporates 20% from the dying column. After all removals, subsystems are rebuilt.

Biological analog: synaptic pruning during adolescence, in which the brain eliminates approximately 40% of synapses between ages 2–16, retaining only the most utilised circuits.

5.4.6. *Operation 4: Split Overloaded Columns (Net2Net)*

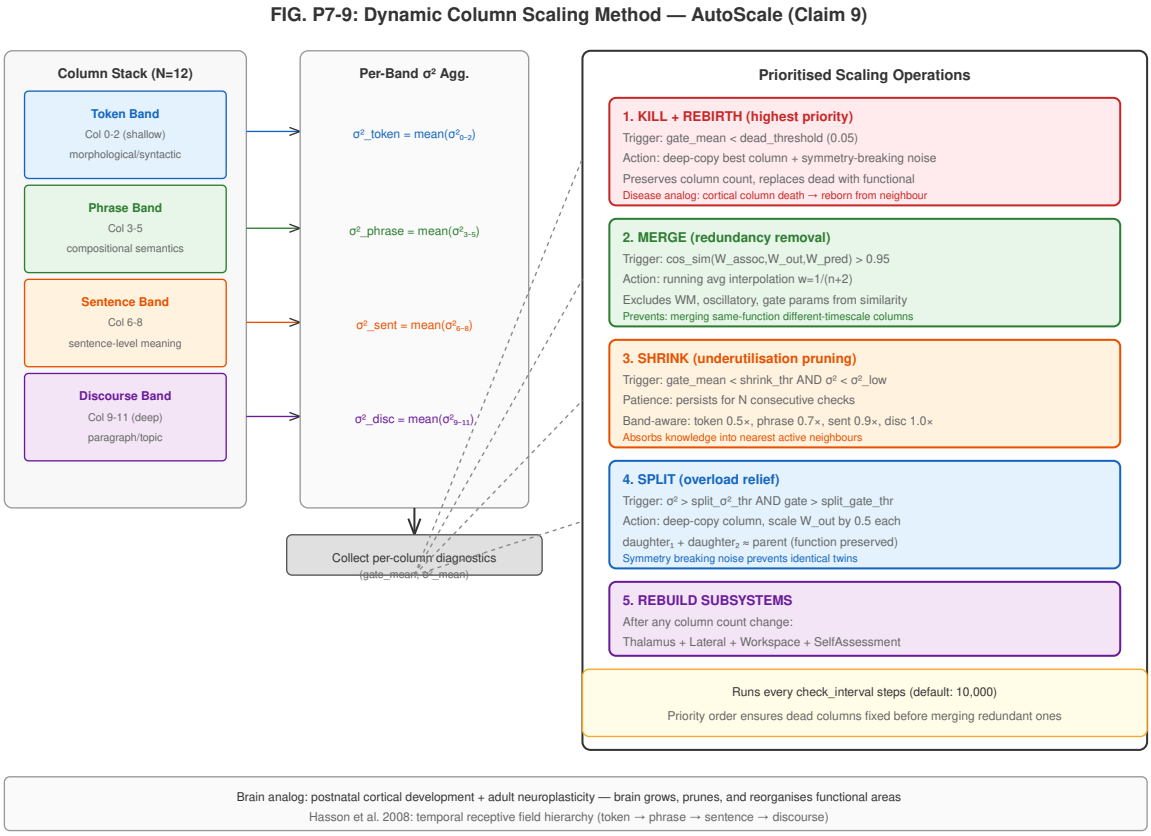


Figure 41: Net2Net column split: parent column deep-copied to two daughters with halved output weights ($0.5 \times W_{\text{out}}$) and symmetry-breaking noise for function preservation.

Referring to Figure 8, a column is classified as overloaded when its σ^2 exceeds the split threshold *and* its mean gate exceeds the gate saturation threshold. The most overloaded column is selected using a priority score that combines the column's own σ^2 with the aggregate σ^2 of its depth band, directing splits to the depth band under greatest pressure.

The split operation proceeds as follows:

1. Deep-copy the parent column to produce two daughter columns (child A and child B).
2. Scale all output-facing weights in both daughters by 0.5: $W_{\text{out}} \leftarrow 0.5 \cdot W_{\text{out}}$ for the L5 output projection, L5 prediction head, L6 thalamic feedback, and L2/3 output projection. This ensures $\text{output}_A + \text{output}_B \approx \text{output}_{\text{parent}}$ at the moment of split (function-preserving).
3. Add symmetry-breaking noise ($\sigma = 0.01$) to child B's parameters so that competitive Hebbian learning can drive differentiation.
4. Insert child B immediately after child A in the column stack.

5. Expand the lateral connections: the new column inherits a copy of the parent's lateral weights, maintaining the neighbourhood structure.

6. Expand the thalamic routing, workspace, and self-assessment for the new column count.

At most one split is performed per health check (because column indices change after a split, invalidating the remaining candidates).

5.4.7. Operation 5: Grow New Columns

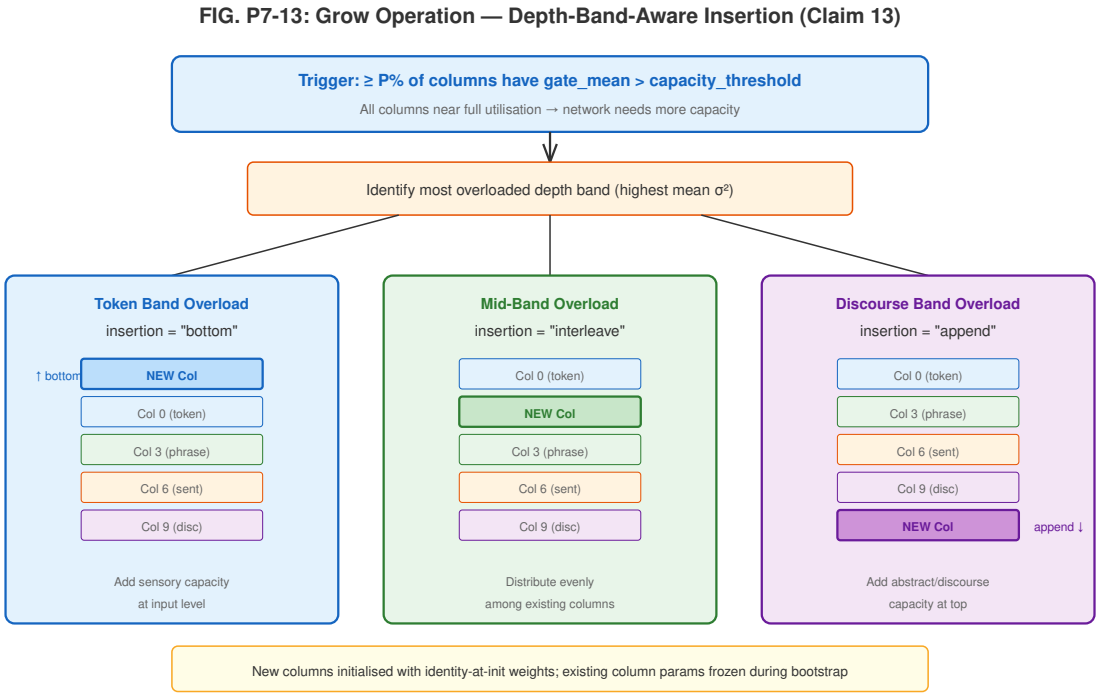


Figure 42: Grow operation with depth-band-aware insertion: token-band overload inserts at bottom, mid-band interleaves, discourse-band appends at top.

Referring to the grow path in Figure 7, growth is triggered when at least 80% of existing columns have mean gate values exceeding $0.8 \times \tau_{\text{split_gate}}$ (indicating high capacity utilisation) and the maximum column count has not been reached. The insertion position is determined by the most overloaded depth band:

- High discourse $\sigma^2 \rightarrow$ append (add deeper abstract columns at the top).
- High token $\sigma^2 \rightarrow$ insert at bottom (add shallow lexical columns).
- High phrase/sentence $\sigma^2 \rightarrow$ interleave (distribute evenly).

The grow operation:

1. Creates n_{new} new cortical columns with standard initialisation plus small symmetry-breaking noise ($\sigma = 0.01$).

2. Inserts them at the determined positions with appropriate column index reassignment.

3. Expands the thalamic routing (new relay groups created, existing relay weights copied), lateral connections (new columns receive zero-initialized send weights for identity-at-init(Stale from transformer era check with /nct-research)), workspace (new read/write projections), and self-assessment.

4. Optionally freezes all existing column parameters during a bootstrap period to prevent catastrophic interference.

Biological analog: postnatal cortical development, in which new functional areas (e.g., language areas around age 2) come online alongside existing areas without disrupting established processing.

5.5. Maximal Update Parameterisation (muP) Transfer

Referring to Figure 11, the muP transfer method extracts scale-invariant hyperparameters from a trained small proxy model and constructs a scaled target configuration.

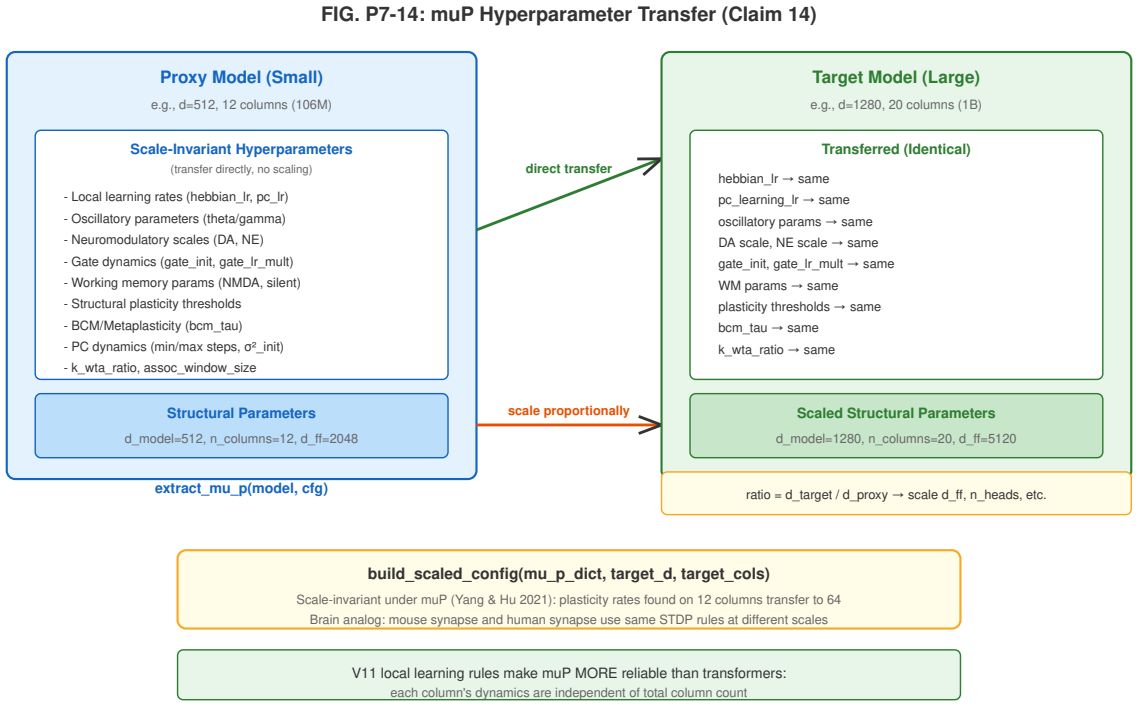


Figure 43: muP hyperparameter transfer: scale-invariant parameters (local learning rates, oscillatory, neuromod scales) transfer directly; structural parameters (d_model, columns, d_ff) scale proportionally.

5.5.1. Scale-Invariant Parameters

The following hyperparameters are extracted from the proxy model and transferred directly to any target scale, because they are independent of model dimension and column count:

- Local learning rates: Hebbian learning rate, predictive coding learning rate, base learning rate.
- Predictive coding dynamics: min/max inference steps, convergence threshold, σ^2 initialisation.
- Hebbian parameters: k-WTA ratio, settle steps, association window size.
- Gate dynamics: initial gate value, gate learning rate multiplier.
- BCM homeostasis: BCM time constant τ_{BCM}.
- Oscillatory parameters: gamma, theta, alpha, delta periods.
- Working memory parameters: NMDA slot count, phonological loop capacity.
- Wake-sleep parameters: sleep interval, replay steps, synaptic downscale factor.
- Structural plasticity: growth threshold, prune threshold, prune interval.
- Regularisation: dropout, weight decay, gradient clip norm.
- Trained neuromodulator scales: DA scale, NE scale (extracted from the brainstem module).

5.5.2. Dimension-Proportional Parameters

The following structural parameters are scaled proportionally to the d_{model} ratio $\rho = d_{\text{target}}/d_{\text{proxy}}$:

- Number of association heads: $\lfloor n_{\text{heads}}^{\text{proxy}} \cdot \rho \rfloor$.
- Inhibitory dimension, NMDA slot dimension, silent trace dimension, phonological slot dimension, relay dimension, TRN dimension, cerebellar dimension, basal ganglia dimension: each scaled as $\lfloor d^{\text{proxy}} \cdot \rho \rfloor$ with appropriate minimums.
- Number of relay groups: proportional to column count ratio.
- Hippocampal sparse dimension and memory size: proportional to ρ.

This two-class decomposition (scale-invariant vs. dimension-proportional) is justified by the local nature of cortical column learning rules: each column's Hebbian learning rate, predictive coding dynamics, and oscillatory parameters are determined by local synaptic properties (analogous to how a mouse synapse and a human synapse use the same STDP rules despite operating in cortices of vastly different size).

5.6. σ^2-Guided Deployment Methods

5.6.1. σ^2-Guided Early Exit

FIG. 24

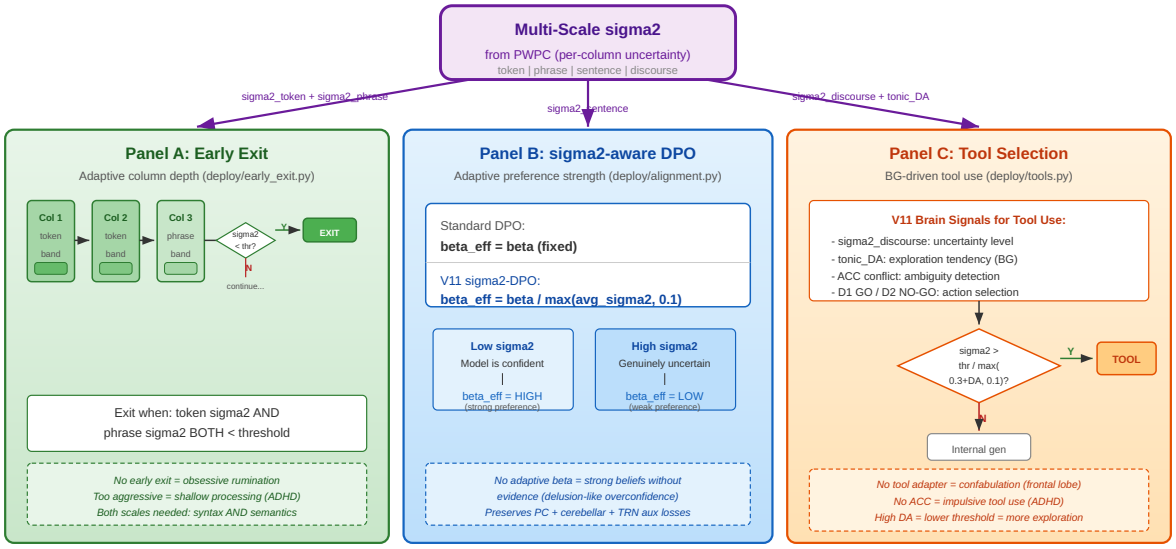


Figure 44: σ^2-guided deployment system overview: early exit (a), σ^2-adaptive alignment (b), and σ^2-driven tool selection (c).

Referring to Figure 12, the early exit mechanism terminates column processing when the model is sufficiently confident in its predictions, saving compute on easy inputs.

The key innovation is a *multi-scale* exit criterion that prevents premature termination. After processing column i (where $i \geq i_{\text{min}}$, the minimum column count), the system computes:

$$\bar{\sigma}^2_{\text{token}} = \frac{1}{q}\sum_{j=0}^{q-1}\sigma_j^2, \quad q = \max\left(1, \left\lfloor \frac{i+1}{4} \right\rfloor\right) \tag{23}$$

$$\bar{\sigma}^2_{\text{phrase}} = \frac{1}{|P|}\sum_{j\in P}\sigma_j^2, \quad P = \{q, \ldots, \min(2q-1, i)\} \tag{24}$$

Early exit occurs only when *both* $\bar{\sigma}^2_{\text{token}} < \theta_{\text{exit}}$ *and* $\bar{\sigma}^2_{\text{phrase}} < \theta_{\text{exit}}$. This dual condition ensures that both syntactic (shallow) and semantic (mid-depth) processing are complete before termination. Without this, a model confident about individual tokens (low token σ^2) but uncertain about their compositional meaning (high phrase σ^2) would exit prematurely — analogous to reading one word and thinking you understand the sentence.

After early exit, the brainstem neuromodulatory system, basal ganglia, cerebellum, and hippocampus process the representation from the last completed column, and the language model head produces output logits. Working memory state is reset before each early-exit forward pass to prevent stale state from prior processing (preventing perseveration).

5.6.2. σ^2-Adaptive Direct Preference Optimisation (DPO)

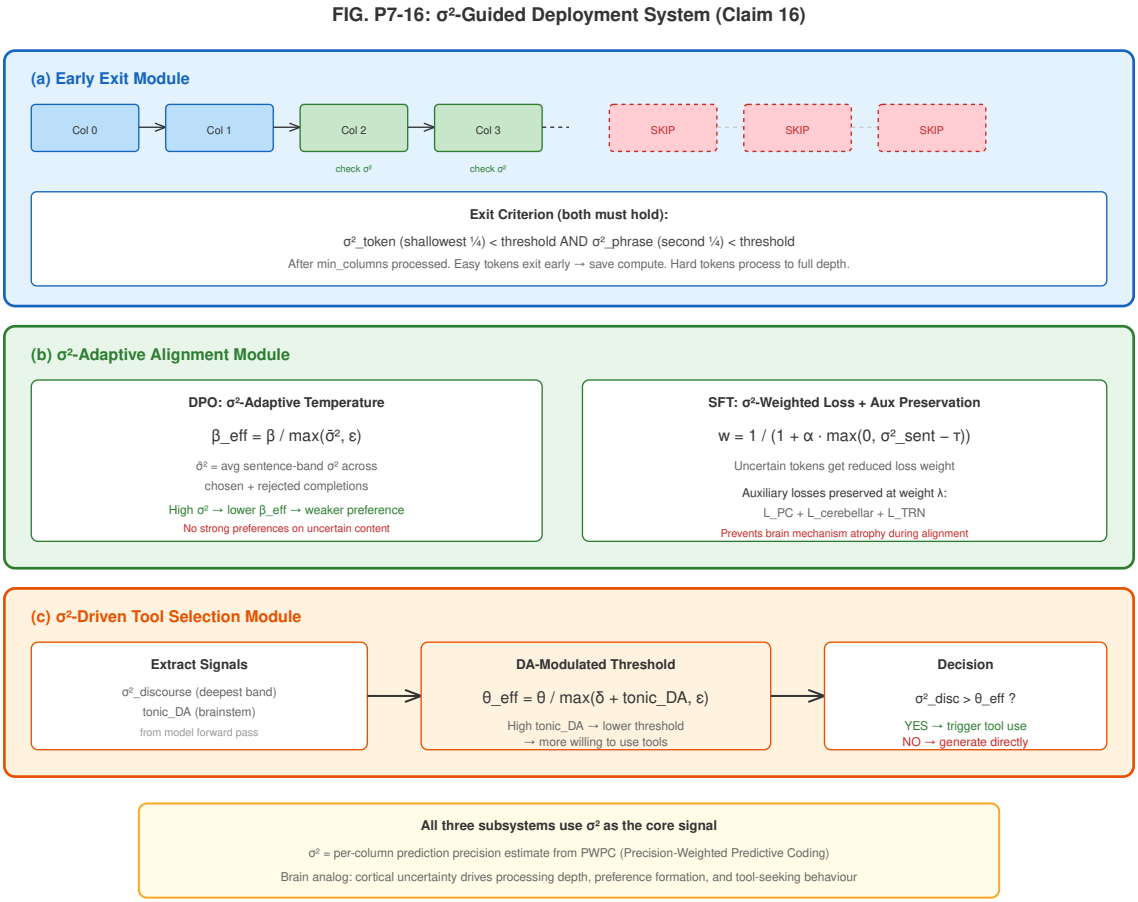


Figure 45: σ^2-guided deployment: early exit with multi-band σ^2 criterion, σ^2-adaptive DPO temperature ($\beta_{\text{eff}} = \beta / \max(\bar{\sigma}^2, \varepsilon)$), SFT with auxiliary loss preservation, and DA-modulated tool selection.

Referring to Figure 13, the DPO alignment method modulates the preference temperature based on the model's prediction uncertainty:

$$\beta_{\text{eff}} = \frac{\beta}{\max(\bar{\sigma}^2_{\text{sentence}}, 0.1)} \tag{25}$$

where $\bar{\sigma}^2_{\text{sentence}}$ is the average sentence-band σ^2 across the chosen and rejected completions, and β is the base DPO temperature. When the model is uncertain ($\bar{\sigma}^2_{\text{sentence}}$ high), β_{eff} is reduced, weakening the preference enforcement. When the model is confident ($\bar{\sigma}^2_{\text{sentence}}$ low), β_{eff} is increased, strengthening the preference.

The DPO loss is then:

$$\mathcal{L}_{\text{DPO}} = -\log \text{sigmoid}\big(\beta_{\text{eff}} \cdot \big[\Delta_{\text{chosen}} - \Delta_{\text{rejected}}\big]\big) \tag{26}$$

where $\Delta = \log \pi - \log \pi_{\text{ref}}$ is the log-probability ratio between the policy and reference models.

Brain analog: the basal ganglia D1/D2 pathways implement action selection with dopamine-modulated strength. σ^2 maps to the strength of the preference signal: high uncertainty weakens the D1 (GO for chosen) vs. D2 (NO-GO for rejected) distinction, favouring exploration over exploitation.

5.6.3. σ^2-Adaptive Supervised Fine-Tuning (SFT)

The SFT method applies response-only loss (masking prompt tokens) with σ^2-aware weighting:

$$w_{\sigma^2} = \frac{1}{1 + 0.1 \cdot \max(0, \sigma^2_{\text{sentence}} - 2.0)} \tag{27}$$

This down-weights the loss on tokens where the model is genuinely uncertain, preventing forced confidence on ambiguous content. Additionally, auxiliary losses (predictive coding loss, cerebellar temporal prediction loss, thalamic reticular nucleus loss) are preserved at weight $\alpha_{\text{aux}} = 0.1$ to prevent brain mechanism atrophy during alignment:

$$\mathcal{L}_{\text{SFT}} = w_{\sigma^2} \cdot \mathcal{L}_{\text{NTP}} + \alpha_{\text{aux}} \cdot (\mathcal{L}_{\text{PC}} + \mathcal{L}_{\text{CB}} + \mathcal{L}_{\text{TRN}}) \tag{28}$$

5.6.4. σ^2-Driven Tool Selection

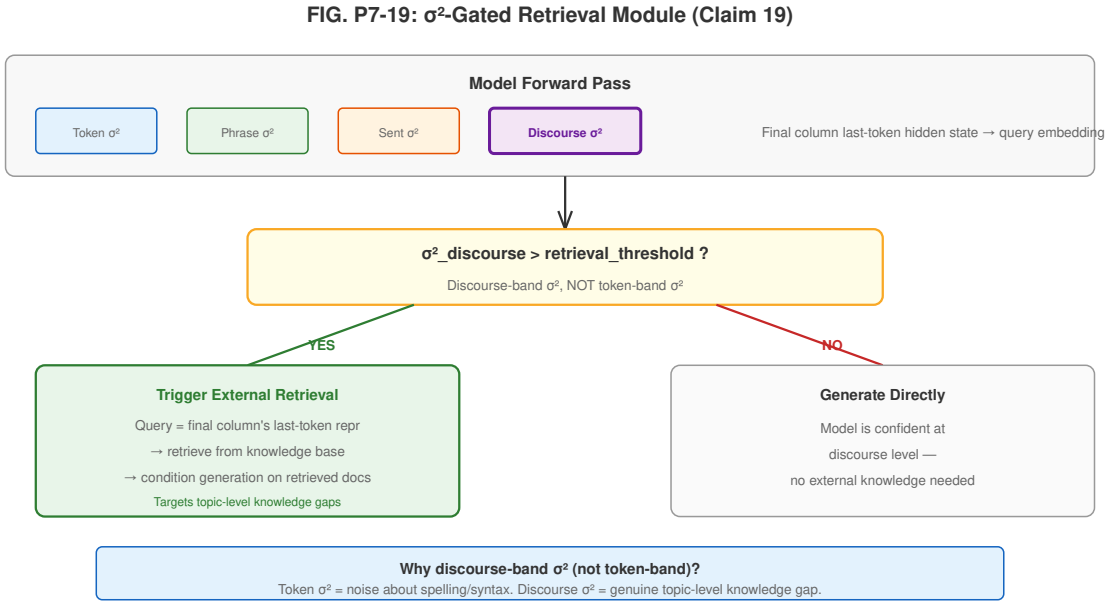


Figure 46: σ^2-gated retrieval module: discourse-band σ^2 (not token-band) triggers external knowledge retrieval when topic-level uncertainty exceeds threshold.

Referring to Figure 14, the tool-use adapter extracts intrinsic brain signals from the model's forward pass and makes a binary tool-use decision:

$$\text{use_tool} = \left(\sigma^2_{\text{discourse}} > \frac{\theta_{\text{tool}}}{\max(0.3 + \text{tonic_DA}, 0.1)} \right) \tag{29}$$

where $\sigma^2_{\text{discourse}}$ is the discourse-band uncertainty (deepest columns), θ_{tool} is the base threshold, and tonic_DA is the tonic dopamine level from the brainstem. High dopamine (exploration tendency) lowers the effective threshold, making tool use more likely. Low σ^2 (confidence) raises the effective threshold, favouring continued text generation.

A retrieval adapter complements the tool adapter by gating external knowledge retrieval on discourse-level σ^2, triggering retrieval only when the model has topic-level uncertainty (not token-level noise). A confidence score derived from the anterior cingulate cortex (ACC) conflict signal provides a secondary quality measure for tool selection.

5.7. Computational Kernel Optimisations

5.7.1. Parallel Prefix Scan with Log-Space Arithmetic

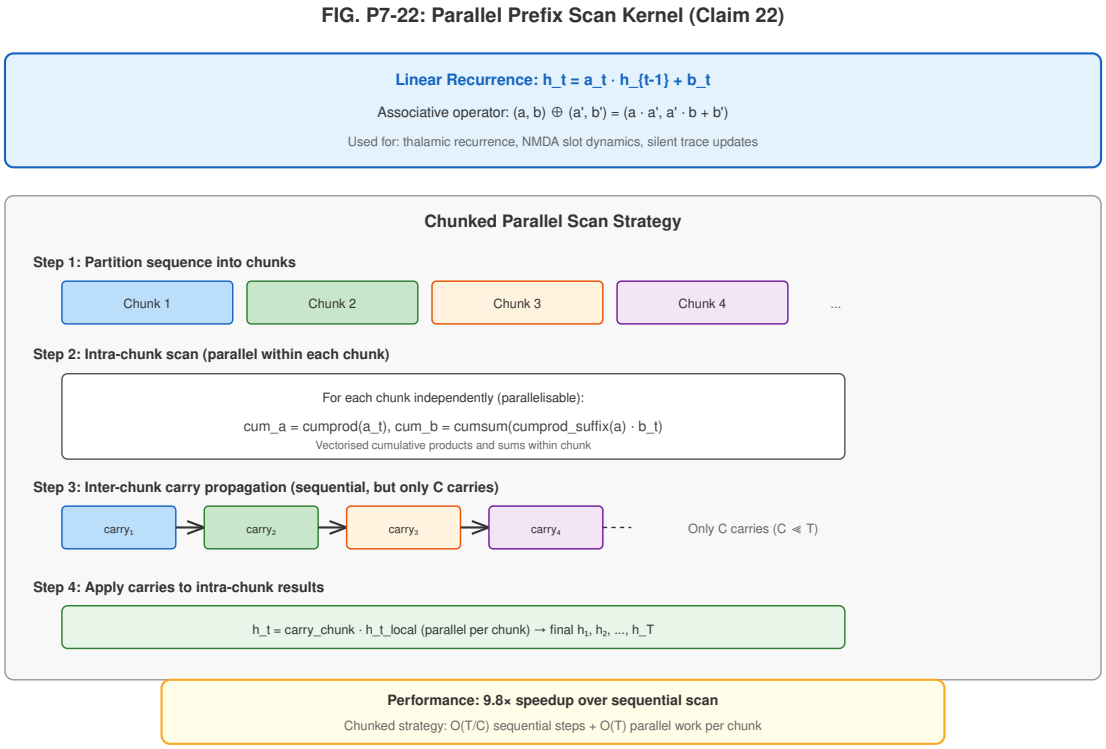


Figure 47: Parallel prefix scan kernel: chunked strategy with vectorised intra-chunk scans and sequential inter-chunk carry propagation, achieving 9.8× speedup.

Referring to Figure 15, the linear recurrence $h_t = a_t \cdot h_{t-1} + b_t$ that governs all working memory dynamics (NMDA attractors, silent trace, phonological loop, thalamic relay) is computed via a

parallel prefix scan using the associative operator:

$$(a, b) \oplus (a', b') = (a \cdot a', a' \cdot b + b') \tag{30}$$

The implementation uses a chunked strategy: the sequence of length T is divided into chunks of size C. Within each chunk, the scan is fully vectorised using cumulative products and cumulative sums. Across chunks, the inter-chunk carry is propagated sequentially. This yields $O(T/C)$ sequential steps (across chunks) and $O(C)$ parallel operations (within chunks), compared to $O(T)$ sequential steps in a naïve loop. On NVIDIA V100 GPUs, this achieves a measured 9.8× speedup by eliminating Python-level sequential loops.

5.7.2. Algebraic Windowed Context Decomposition

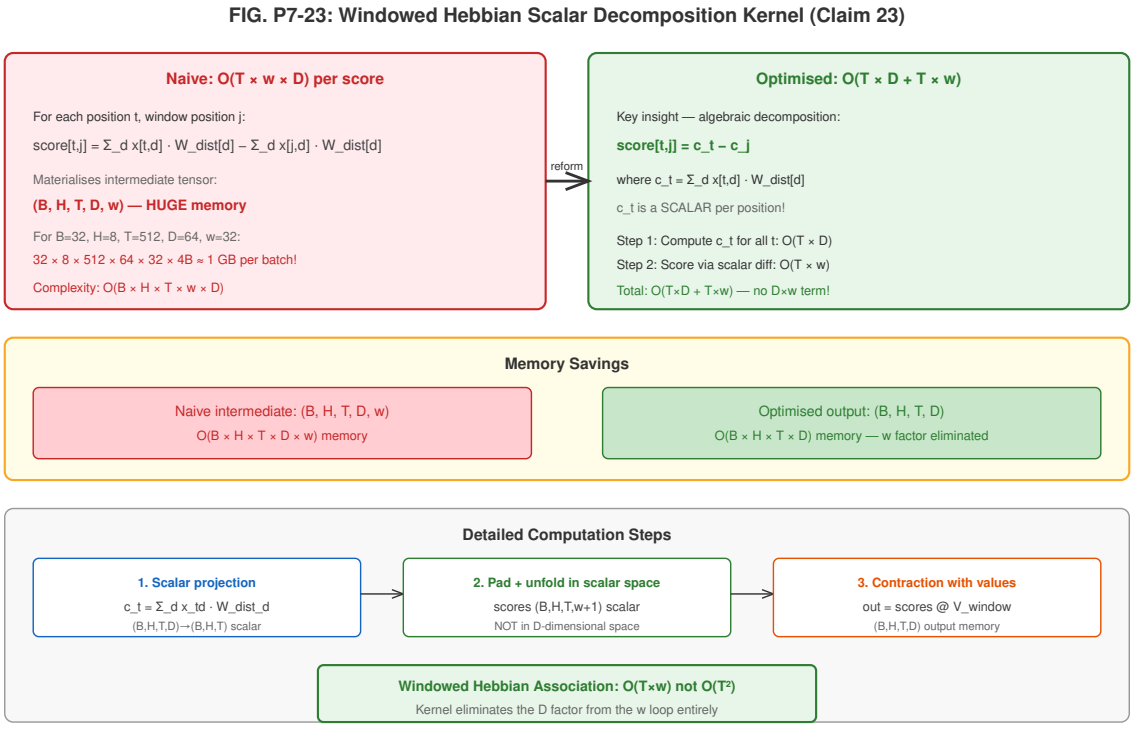


Figure 48: Windowed Hebbian scalar decomposition: algebraic insight $\text{score}[t, j] = c_t - c_j$ eliminates the D factor from the window loop, reducing from $O(T \times w \times D)$ to $O(T \times D + T \times w)$.

Referring to Figure 16, the windowed Hebbian association in Layer 2/3 computes distance-weighted context scores:

$$\text{score}[t, j] = \sum_{d=1}^{D_h} \left(x_{t,d} - x_{j,d}\right) \cdot W_{\text{dist},d} \tag{31}$$

Naïvely, this requires $O(T \times w \times D_h)$ operations and materialises a (B, H, T, D_h, w) intermediate tensor consuming up to 26 GB for typical training configurations.

The key algebraic insight is that the score decomposes as a difference of scalars:

$$\text{score}[t, j] = c_t - c_j, \quad \text{where } c_t = \sum_{d=1}^{D_h} x_{t,d} \cdot W_{\text{dist},d} \tag{32}$$

This reduces score computation from $O(T \times w \times D_h)$ to $O(T \times D_h + T \times w)$ and eliminates the (B, H, T, D_h, w) intermediate tensor entirely. The scalar c_t is a single dot product per position, and the windowed scores are computed by subtracting a padded and unfolded version of c from the current position's c — operating entirely in scalar space.

5.7.3. NMDA Parallel Scan Decomposition

FIG. P7-25: NMDA Parallel Scan Kernel (Claim 25)

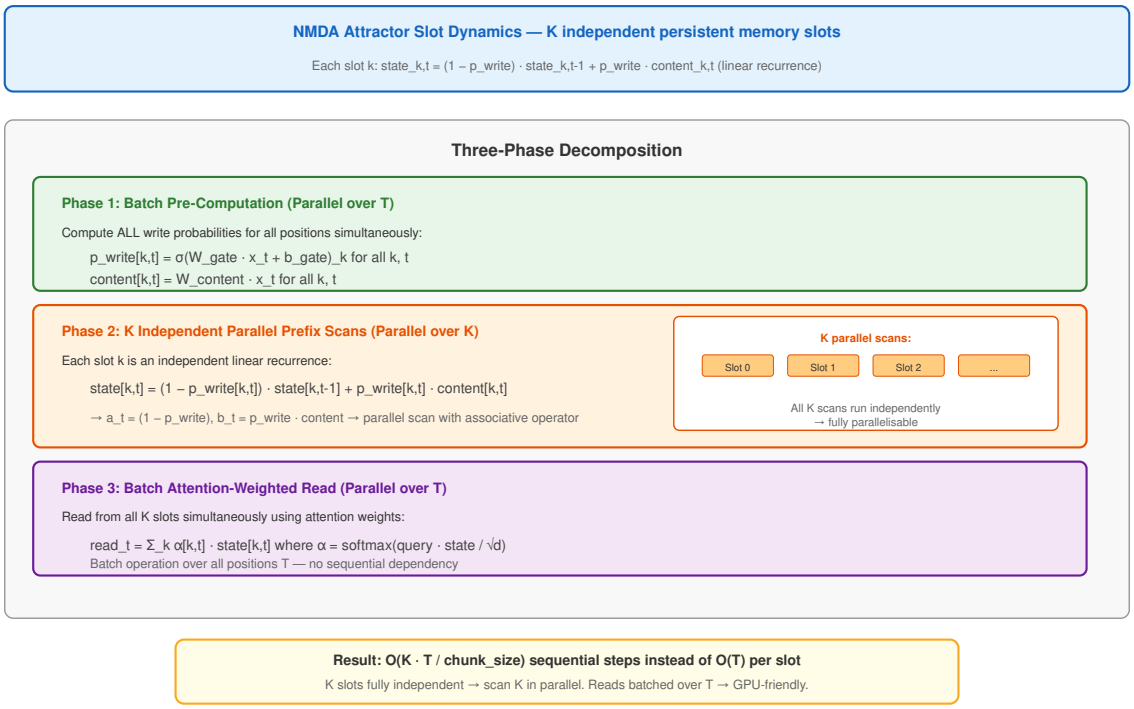


Figure 49: NMDA parallel scan: three-phase decomposition — batch pre-computation of write probabilities, K independent per-slot parallel prefix scans, and batch attention-weighted reads.

Referring to Figure 17, the NMDA attractor slot update $\mathbf{s}_t^k = \delta \cdot \mathbf{s}_{t-1}^k + p_t^k \cdot \mathbf{c}_t$ is decomposed into three batch operations:

1. **Pre-compute**: all write probabilities p_t^k and contents $\mathbf{c}_t$ for all T positions via batch matrix multiplications.

2. K **parallel scans**: each of the K slots runs an independent parallel prefix scan (since slots evolve independently).

3. **Batch reads**: attention-weighted read over the scan output, computed as a single batch operation.

This eliminates the T-step Python loop in the original NMDA implementation while producing numerically identical results (parity tolerance $< 10^{-4}$).

5.7.4. Tsodyks-Markram Linearisation as Dual Sequential Scans

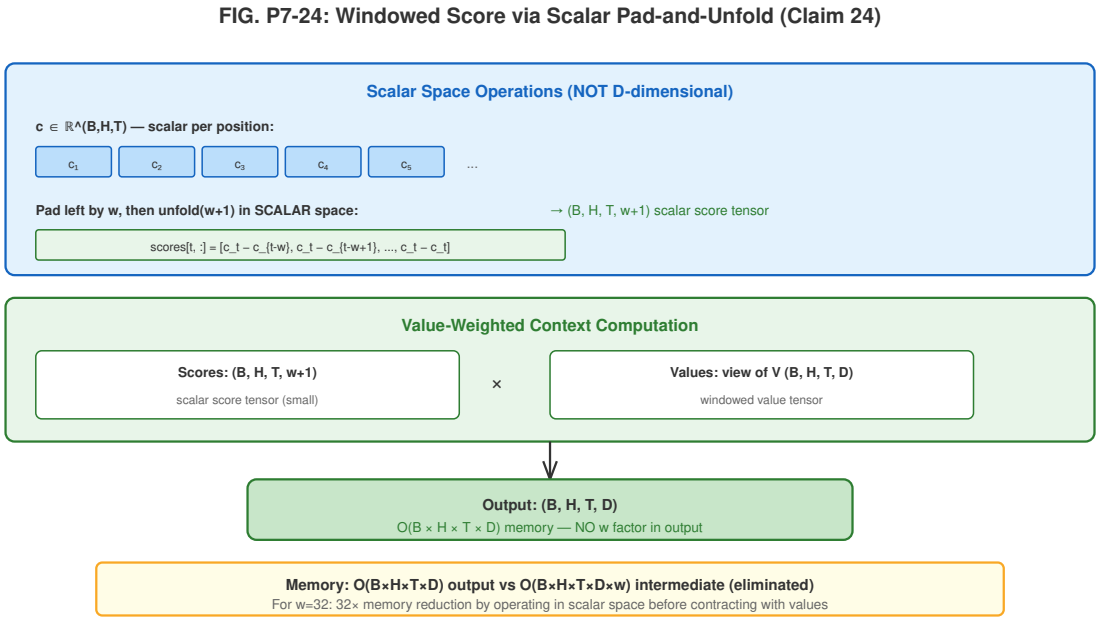


Figure 50: Windowed score computation via scalar pad-and-unfold: operating in scalar space (B, H, T) before contracting with values eliminates the w factor from memory, achieving $32\times$ reduction for $w = 32$.

Referring to Figure 18, the activity-silent trace uses Tsodyks-Markram synaptic dynamics with facilitation (u) and depression (x) variables. The facilitation update:

$$u_t = \alpha_f \cdot u_{t-1} + (1 - u_{t-1}) \cdot g_t \cdot |c_t| \tag{33}$$

is linearised as:

$$u_t = u_{t-1} \cdot (\alpha_f - g_t|c_t|) + g_t|c_t| \tag{34}$$

This is a standard linear recurrence with $a_t = \alpha_f - g_t|c_t|$ and $b_t = g_t|c_t|$, solvable by parallel scan.

The depression update $x_t = f(x_{t-1}, u_t)$ depends on u_t, so it must be computed *after* the facilitation scan. It is similarly linearised:

$$x_t = x_{t-1} \cdot (2\alpha_d - 1 - \delta_f \cdot u_t) + (1 - \alpha_d) \tag{35}$$

yielding a second parallel scan with $a_t = 2\alpha_d - 1 - \delta_f \cdot u_t$ and $b_t = 1 - \alpha_d$.

The two sequential scans (facilitation first, then depression using facilitation output) replace the original T-step Python loop, with the effective synaptic release computed as $j_{\text{eff}} = u \odot x$.

6. CLAIMS

We claim:

Claim 1. A computer-implemented multi-stage embedding system for neural language models, comprising:

(a) a base embedding lookup stage mapping integer token identifiers to dense continuous vectors of dimension d using a learned embedding table;

(b) a dual-route morphological composition stage comprising:

(i) a sublexical route computing a compositional representation from local context via a causal depthwise-separable convolution over the embedding vectors, wherein the causal constraint is enforced by left-padding of $K-1$ positions with no right-padding;

(ii) a lexical route comprising the base embedding vector itself;

(iii) a per-token frequency gate $g_{\text{lex},i} = \text{sigmoid}(f_i)$, where f_i is a learnable per-token frequency bias parameter, configured to blend the lexical route and the sublexical route such that high-frequency tokens favour the lexical route and low-frequency tokens favour the sublexical route; and

(iv) a master gate $g_{\text{master}} = \text{sigmoid}(\gamma)$ controlling the overall correction magnitude, with γ initialised to produce $g_{\text{master}} \approx 0.12$ at initialisation;

(c) a hub-and-spoke semantic integration stage comprising:

(i) a spoke decomposition partitioning each embedding vector into S contiguous subvectors along the feature dimension;

(ii) S independent spoke projection layers, each projecting a respective spoke subvector to a uniform output dimension;

(iii) a hub bottleneck comprising a nonlinear transformation of the concatenated spoke projections through a narrower intermediate dimension, forcing cross-modal information integration; and

(iv) a gated residual connection adding the hub output to the embedding, controlled by a learnable gate initialised to produce a near-zero correction;

(d) a sparse distributed projection stage comprising:

(i) an expansion projection from dimension d to an expanded dimension $d_{\text{sdr}} = d \times r$;

(ii) an exact-k sparsity enforcement selecting the top-k activated units and zeroing all others, with $k = \lceil d_{\text{sdr}} \times s \rceil$ for a sparsity fraction s;

(iii) a straight-through estimator allowing gradient flow through the non-differentiable top-k operation;

(iv) a contraction projection from dimension d_{sdr} back to dimension d; and

(v) a gated residual connection adding the sparse correction to the embedding;

wherein each of stages (b), (c), and (d) satisfies an identity-at-init(Stale from transformer era check with /nct-research)ialisation property such that the complete pipeline outputs the base embedding at initialisation and gradually learns corrections during training.

Claim 2. The system of Claim 1, wherein the per-token frequency bias parameters f_i are initialised from corpus frequency statistics by normalising log-frequency counts to a predetermined range and mapping high-frequency tokens to positive values favouring the lexical route and low-frequency tokens to negative values favouring the sublexical route.

Claim 3. The system of Claim 1, wherein the causal depthwise-separable convolution of the sublexical route comprises a depthwise convolution with groups equal to the embedding dimension d followed by a pointwise linear projection, and wherein the depthwise convolution is initialised near-identity with the rightmost kernel weight set to 1.0 and all other kernel positions initialised with small Gaussian noise.

Claim 4. The system of Claim 1, wherein the S spokes of the hub-and-spoke stage correspond to $S = 6$ modality types comprising visual, auditory, motor, verbal, emotional, and encyclopedic modalities, and wherein the hub bottleneck intermediate dimension is $\lfloor S \cdot d_{\text{out}}/2 \rfloor$ where d_{out} is the per-spoke output dimension.

Claim 5. The system of Claim 1, wherein the sparsity fraction $s = 0.02$ and the expansion factor $r = 2$, yielding a combinatorial pattern capacity of $\binom{d \times r}{\lceil d \times r \times 0.02 \rceil}$ distinct active patterns per token.

Claim 6. The system of Claim 1, further comprising a grid-cell-inspired position encoding system comprising:

(i) learnable grid cell module periods $\tau_i = \text{softplus}(\theta_i^{\text{raw}})$ initialised to recover standard Rotary Position Embedding (RoPE) frequencies;

(ii) per-head scale factors $s_h = \text{softplus}(s_h^{\text{raw}})$ enabling multi-resolution position sensitivity across different association heads; and

(iii) a boundary-aware phase remapping mechanism comprising a boundary detector producing a boundary probability $p_{\text{boundary},t} = \text{sigmoid}(\mathbf{W}_b \cdot \mathbf{x}_t + b_b)$, a cumulative boundary phase $\phi_{\text{cum},t} = \sum_{s=1}^{t} p_{\text{boundary},s} \cdot \phi_{\text{boundary}}$, and a learnable gate controlling the strength of boundary remapping.

Claim 7. The system of Claim 6, wherein the boundary detector weight $\mathbf{W}_b$ is initialised to zeros and the boundary detector bias b_b is initialised to a negative value such that boundary detection is predominantly inactive at initialisation, and wherein the boundary phase parameters ϕ_{boundary} are initialised to zero, ensuring exact recovery of standard RoPE at initialisation.

Claim 8. The system of Claim 1, further comprising a dual-route morphological tokeniser configured to:

(i) look up input words in a known-word vocabulary for a lexical route;

(ii) for words not in the known-word vocabulary, apply greedy affix stripping comprising iterative suffix removal (sorted by descending length, with morphophonemic normalisation and root validation) followed by iterative prefix removal, with a false decomposition exception list and a minimum root length constraint; and

(iii) apply byte-level encoding as a fallback for non-alphabetic input;

wherein each token corresponds to approximately one morpheme aligned to one gamma cycle in an oscillatory processing hierarchy.

Claim 9. A computer-implemented method for dynamically scaling a cortical column neural network during training, the neural network comprising a plurality of cortical column processing units arranged in a hierarchical stack, the method comprising:

(a) partitioning the cortical columns into depth bands based on their position in the hierarchical stack, each depth band corresponding to a distinct temporal receptive field scale;

(b) computing multi-scale prediction uncertainty aggregates by averaging the per-column prediction precision estimates (σ^2) within each depth band;

(c) periodically collecting per-column diagnostics comprising mean gate activation values and mean σ^2 values;

(d) applying a prioritised sequence of scaling operations comprising:

  (i) a kill-and-rebirth operation replacing columns identified as dead based on mean gate values falling below a dead gate threshold, by deep-copying the best-performing column with symmetry-breaking noise;

  (ii) a merge operation identifying redundant column pairs based on cosine similarity of functional weight vectors computed exclusively over association, output, and prediction weights and not over working memory, oscillatory, or gate parameters, and merging redundant columns via running average parameter interpolation with weight $w = 1/(n_{\text{merged}} + 2)$;

(iii) a shrink operation identifying underutilized columns based on low gate activation and low σ^2 persisting for a patience count of consecutive health checks, with depth-band-aware thresholds applying lower effective thresholds to shallow columns than to deep columns, and absorbing the knowledge of removed columns into nearest active neighbours; and

(iv) a split operation identifying overloaded columns based on high σ^2 and high gate saturation, deep-copying the overloaded column, scaling output-facing weights by 0.5 in each copy with symmetry-breaking noise, such that the sum of the two daughter columns preserves the parent's function; and

(e) rebuilding thalamic routing, lateral connections, global workspace, and self-assessment subsystems after any operation that changes the column count.

Claim 10. The method of Claim 9, wherein the depth bands comprise four bands: a token band comprising the shallowest quarter of columns, a phrase band comprising the second quarter, a sentence band comprising the third quarter, and a discourse band comprising the deepest quarter, corresponding to the temporal receptive field hierarchy of Hasson et al. (2008).

Claim 11. The method of Claim 9, wherein the depth-band-aware thresholds for the shrink operation apply scaling factors of 0.5 for the token band, 0.7 for the phrase band, 0.9 for the sentence band, and 1.0 for the discourse band, such that shallow columns processing primary sensory information require lower gate activity to be classified as underutilised than deep columns processing abstract information.

Claim 12. The method of Claim 9, wherein the merge operation computes cosine similarity exclusively over functional weights comprising association weights, output weights, and prediction weights, and excludes working memory state parameters, oscillatory parameters, and gate parameters from the similarity computation, thereby preventing erroneous merging of columns that compute the same function but operate at different timescales.

Claim 13. The method of Claim 9, further comprising a grow operation triggered when at least a predetermined proportion of existing columns have mean gate values exceeding a capacity threshold, wherein the insertion position for new columns is determined by the most overloaded depth band: appending for discourse-band overload, inserting at the bottom for token-band overload, and interleaving for mid-band overload.

Claim 14. The method of Claim 9, further comprising a maximal update parameterisation (muP) transfer step comprising:

(i) extracting scale-invariant hyperparameters from a trained proxy model, the scale-invariant hyperparameters comprising local learning rates, oscillatory parameters, neuromodulatory scales, gate dynamics parameters, working memory parameters, and structural plasticity thresholds;

(ii) scaling dimension-dependent structural parameters proportionally to the ratio of target model dimension to proxy model dimension; and

(iii) constructing a target model configuration combining the transferred scale-invariant hyperparameters with the proportionally scaled structural parameters.

Claim 15. The method of Claim 14, wherein the scale-invariant hyperparameters further comprise trained neuromodulator scale values extracted from a brainstem neuromodulatory module of the proxy model, comprising dopamine scale and norepinephrine scale values learned during proxy model training.

Claim 16. A computer-implemented system for deploying a cortical column neural network, the neural network comprising a plurality of cortical columns each producing a per-token prediction precision estimate (σ^2), the system comprising:

(a) an early exit module configured to:

(i) process cortical columns sequentially from the shallowest to the deepest;

(ii) after processing each column beyond a minimum column count, compute a token-band σ^2 aggregate from the shallowest processed columns and a phrase-band σ^2 aggregate from early-mid processed columns;

(iii) terminate column processing when both the token-band σ^2 aggregate and the phrase-band σ^2 aggregate fall below a threshold; and

(iv) complete the forward pass using only the outputs from the processed columns;

(b) a σ^2-adaptive alignment module configured to:

(i) for direct preference optimisation, compute an effective temperature $\beta_{\text{eff}} = \beta / \max(\bar{\sigma}^2, \epsilon)$ where $\bar{\sigma}^2$ is the average sentence-band σ^2 across chosen and rejected completions, thereby reducing preference enforcement when the model is uncertain; and

(ii) for supervised fine-tuning, weight the loss by $w = 1/(1 + \alpha \cdot \max(0, \sigma^2_{\text{sentence}} - \tau))$ and preserve auxiliary losses comprising predictive coding loss, cerebellar loss, and thalamic reticular nucleus loss at a predetermined weight to prevent brain mechanism atrophy during alignment;

(c) a σ^2-driven tool selection module configured to:

(i) extract discourse-band σ^2 and tonic dopamine level from the neural network's forward pass;

(ii) compute a dopamine-modulated effective threshold $\theta_{\text{eff}} = \theta / \max(\delta + \text{tonic_DA}, \epsilon)$; and

(iii) signal tool use when the discourse-band σ^2 exceeds the effective threshold.

Claim 17. The system of Claim 16, wherein the early exit module further resets working memory state of all cortical columns before each forward pass, preventing stale working memory from prior processing from contaminating the current inference.

Claim 18. The system of Claim 16, wherein the σ^2-adaptive alignment module further preserves predictive coding loss, cerebellar temporal prediction loss, and thalamic reticular nucleus loss as auxiliary losses during supervised fine-tuning at a predetermined auxiliary weight, maintaining brain mechanism integrity during alignment.

Claim 19. The system of Claim 16, further comprising a σ^2-gated retrieval module configured to trigger external knowledge retrieval when discourse-band σ^2 exceeds a retrieval threshold, using the final column's last-token representation as a retrieval query embedding, wherein discourse-band σ^2 rather than token-band σ^2 is used to ensure retrieval targets topic-level knowledge gaps rather than token-level noise.

Claim 20. The system of Claim 16, further comprising a tool confidence module that extracts an anterior cingulate cortex conflict signal from the neural network and computes a tool selection confidence score as $1 -$ conflict, wherein high conflict indicates ambiguity among candidate tools.

Claim 21. The system of Claim 16, wherein the token-band σ^2 aggregate is computed from the shallowest quarter of processed columns and the phrase-band σ^2 aggregate is computed from the second quarter of processed columns, and wherein early exit is prohibited for the deepest half of columns corresponding to sentence-band and discourse-band processing.

Claim 22. The system of Claim 1, further comprising a parallel prefix scan kernel for computing the linear recurrence $h_t = a_t \cdot h_{t-1} + b_t$ using the associative operator $(a, b) \oplus (a', b') = (a \cdot a', a' \cdot b + b')$, implemented via a chunked strategy wherein each chunk is computed using vectorised cumulative products and cumulative sums, and inter-chunk carries are propagated sequentially.

Claim 23. The system of Claim 1, further comprising a windowed context kernel that computes distance-weighted scores for windowed Hebbian association via an algebraic decomposition $\text{score}[t, j] = c_t - c_j$ where $c_t = \sum_d x_{t,d} \cdot W_{\text{dist},d}$ is a scalar per position, reducing score computation from $O(T \times w \times D)$ to $O(T \times D + T \times w)$ and eliminating materialisation of intermediate tensors of size (B, H, T, D, w).

Claim 24. The system of Claim 22, wherein the windowed context kernel operates on the scalar c_t by padding and unfolding in scalar space rather than in the full D-dimensional embedding space, such that the value-weighted context is computed as a contraction over a $(B, H, T, w+1)$ scalar score tensor and a view of the value tensor, consuming $O(B \times H \times T \times D)$ output memory rather than $O(B \times H \times T \times D \times w)$ intermediate memory.

Claim 25. The system of Claim 1, further comprising an NMDA parallel scan kernel decomposing NMDA attractor slot dynamics into: a batch pre-computation of all write probabilities

and contents for all sequence positions; K independent parallel prefix scans for the K slot state updates; and a batch attention-weighted read operation over the scan output.

Claim 26. A non-transitory computer-readable storage medium storing instructions that, when executed by one or more processors, cause the processors to implement the multi-stage embedding system of Claim 1, the dynamic scaling method of Claim 9, and the deployment system of Claim 16.

ABSTRACT

A computer-implemented system and method for brain-inspired embedding, dynamic scaling, and deployment of cortical column neural language models. The embedding system comprises a four-stage cascade: (i) base embedding lookup; (ii) dual-route morphological composition implementing the Coltheart DRC model, wherein a per-token frequency gate blends a lexical route (direct lookup) with a sublexical route (causal depthwise-separable convolution); (iii) hub-and-spoke semantic integration modelling anterior temporal lobe cross-modal binding through modality-specific spoke projections integrated via a bottleneck hub; and (iv) sparse distributed projection enforcing exact-k sparsity with a straight-through estimator, producing high-capacity population codes. A grid-cell-inspired position encoding provides learnable periods, per-head scale factors, and boundary-aware phase remapping. A dual-route morphological tokeniser segments text along morpheme boundaries with oscillatory alignment. A dynamic scaling system (AutoScale) grows, splits, merges, kills, and shrinks cortical columns during training, driven by multi-scale prediction uncertainty (σ^2) aggregated across depth bands, with depth-band-aware thresholds and function-preserving split operations. A maximal update parameterisation transfer method extracts scale-invariant hyperparameters from small proxy models. Deployment methods exploit multi-scale σ^2 for early exit (dual token-band and phrase-band threshold), uncertainty-adaptive DPO and SFT alignment, and dopamine-modulated tool selection. Computational kernel optimisations include parallel prefix scan (9.8$\times$ speedup), algebraic windowed context decomposition, and Tsodyks-Markram recurrence linearisation as dual sequential parallel scans.

(150 words — within IPO 150-word abstract limit)