A Distributed, Forward-Only Threshold-Gating Neural Architecture with Hierarchical Memory, Temporal Plasticity, Blockchain Preservation, and Incentivized Participation

Kritivasas
Senior Software Developer at Buildops
(No University Affiliation)

May 31, 2025

Abstract

We propose a distributed neural network architecture that eliminates backpropagation in favor of local, forward-only threshold-gating nodes, inspired by neurobiological principles of memory and adaptation. Each user device hosts a cluster of adaptive nodes, preserving privacy and resilience, and participates in a global network coordinated by blockchain for memory permanence and fair incentive distribution. The system features a three-level memory hierarchy, temporal plasticity via adaptive thresholds and timers, eligibility traces, and dynamic topology management via node/edge duplication and pruning. Our workflow analysis and experiments demonstrate robust, modular learning with long-term memory preservation, seamless scaling, and community-driven growth even in dynamic, unreliable environments.

1 Introduction

Modern deep neural networks have achieved impressive performance across domains, but rely heavily on centralized training, global error backpropagation, and large amounts of labeled data [6, 7]. This reliance creates challenges for scalability, distributed computation, privacy, and long-term robustness. In contrast, biological brains leverage highly modular, local learning mechanisms and adapt via event-driven, forward-only updates, supported by diverse forms of memory and sparse global signals.

Recent research has proposed biologically inspired alternatives to backpropagation, such as Hinton's forward-forward algorithm [10] and Hebbian or eligibility trace-based learning rules [11, 14], which emphasize local updates and temporally modulated plasticity. Inspired by these principles, we propose a *forward-only, threshold-gating neural architecture* in which each node accumulates inputs and fires based on an adaptive threshold or temporal deadline, with all error and adaptation signals fed forward (never backpropagated).

Crucially, we extend this design into a fully distributed, privacy-preserving, and incentivized system:

- Each user device hosts a cluster of adaptive threshold-gating nodes, preserving privacy by keeping raw data strictly local.

- Local adaptation is achieved using a three-level memory hierarchy (node, cluster, global), adaptive thresholds, eligibility traces, and event-driven plasticity.
- Memory, adaptation, and error signals are periodically consolidated, compressed, and uploaded as encrypted "memory capsules" to a global distributed store, with blockchain auditability for long-term permanence and transparent incentive accounting.
- Smart contracts allocate rewards based on each participant's contribution, making the network robust to device dropout, dynamic scaling, and adversarial conditions.

By merging neural-level biological plausibility with robust distributed systems design, we present a framework for scalable, modular, and resilient AI that can learn, grow, and self-organize across billions of heterogeneous, privacy-preserving endpoints.

2 Related Work

2.1 Biological and Local Learning Mechanisms

A long line of research in neuroscience and computational modeling has explored learning rules that do not require global backpropagation. Hebb's principle [11], eligibility traces [14], and three-factor plasticity rules capture the role of local memory and neuromodulatory signals in biological brains. Such mechanisms allow local adaptation, temporally extended credit assignment, and robust learning across distributed, modular networks. Further, event-driven models and spiking neural networks (SNNs) [8, 15, 3] use threshold-based firing and adaptive temporal dynamics for efficient, biologically plausible computation. These models provide inspiration for our threshold-gating nodes, local memory buffers, and eligibility trace-based adaptation.

2.2 Forward-Only Learning in Artificial Networks

Alternatives to backpropagation for artificial neural networks have gained traction, particularly for their potential in scalable, asynchronous, and modular AI. Hinton's forward-forward algorithm [10] dispenses with gradient chains by using layer-wise "goodness" comparisons between positive and negative samples. Related work on local learning rules, energy-based models, and contrastive Hebbian learning (see [9]) further explore local updates that do not require global gradient propagation. Our approach extends these ideas to include adaptive threshold gating, temporally co-adapted timers, and purely forward-propagated error signals.

2.3 Distributed, Privacy-Preserving, and Incentivized AI

Practical deployment of large-scale neural systems increasingly requires distributed, privacy-preserving architectures. Federated learning and privacy-first neural systems [18] have introduced on-device learning and aggregation, but often still rely on some form of global synchronization or limited centralization. Blockchain-based systems for distributed AI [19] and distributed ledgers for long-term storage and auditable contribution tracking [20] offer tools for resilience, permanence, and incentivization. Our system uniquely combines threshold-gating neural computation with on-device privacy, long-term memory preservation via blockchain, and a transparent incentive model for participant contributions.

2.4 Multi-Modal and Hierarchical Memory Systems

Multi-modal learning and hierarchical memory architectures have been explored in deep learning [12, 13] and neuroscience [16, 17]. Hierarchical systems leverage local and global memory to handle distributed, asynchronous, and multi-domain tasks. Our three-level memory structure (local node, cluster, global) is directly inspired by such architectures, supporting modular growth, specialization, and resilient distributed computation.

3 Methods

3.1 Threshold Gating Node: Local Dynamics

The core of our architecture is the *threshold gating node*. Each node accumulates incoming signals in an internal buffer, fires when either an adaptive threshold δ or a timer T is reached, and adapts via a local eligibility trace and error signal. This enables fully forward-only local learning and dynamic temporal plasticity.

- **Accumulator:** Sums incoming activations since last firing.
- **Adaptive Threshold δ:** Determines when to fire based on local activity and error.
- **Timer T:** Forces node firing after a set period, ensuring event-driven and time-driven learning.
- **Eligibility Trace:** Temporally decaying marker for credit assignment.
- **Local Error Input:** Used to modulate threshold and adaptation rate (no global backprop).

$$\Delta\delta = \begin{cases} +\eta_\delta, & \text{if fired by threshold} \\ -\frac{1}{2}\eta_\delta, & \text{if forced by timer} \end{cases} \tag{1}$$

where η_δ is the adaptation rate, updated by error: $\eta_\delta \leftarrow \eta_\delta + \alpha \cdot \text{errorInput}$.

3.2 Three-Level Memory Mechanism

Each node, cluster, and device maintains a local memory buffer, aggregates adaptation statistics, and produces compressed "memory capsules." - **Node:** Stores recent activations, errors, eligibility, timers, tags. - **Cluster:** Aggregates node outputs, computes local statistics, forms memory capsules with tagging and semantic masking. - **Device:** Consolidates, compresses, and encrypts capsules for upload.

3.3 Node Borrowing Protocol (Algorithm)

When a device or cluster borrows a node:

1. **Discovery:** Search blockchain/index for eligible node packets (by context, tags).
2. **Permission:** Check privacy tag and access control (only behavioral, not personal, nodes shared).

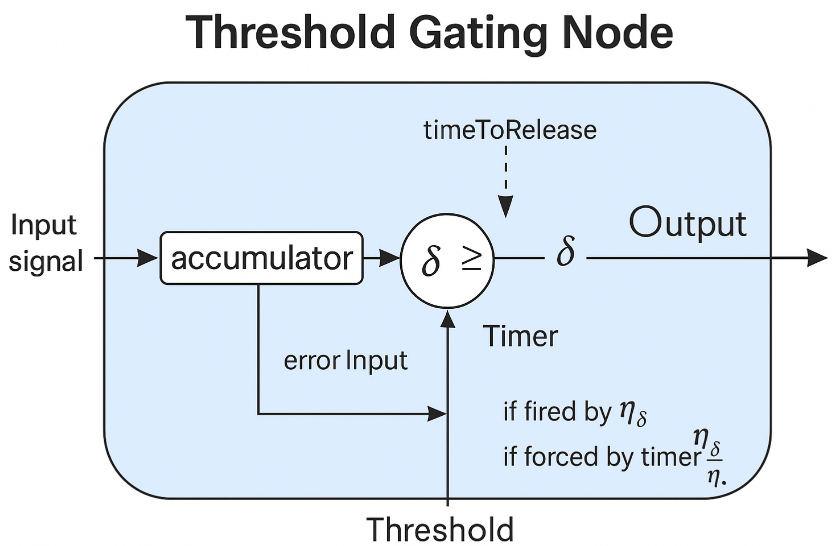

Figure 1: Schematic of a threshold gating node. The node sums inputs, fires if threshold/timer is met, updates eligibility trace, and adapts threshold based on forward-only error. Local memory stores all parameters.

3. **Incentive:** Smart contract debits credits for borrowing.
4. **Transfer:** Packet securely transmitted and instantiated locally.
5. **Snapshot:** After use, new state is snapshotted, signed, uploaded for audit and possible respawn.

3.4 Error Propagation and Credit Assignment (Mechanism)

Error signals are propagated forward only:

- Local error, task loss, or global modulatory signals are fed to nodes.
- Eligibility traces assign temporal credit to prior events.
- Capsule tagging registers significant events and updates on-chain for credit and audit.

3.5 Threshold Gating Node and Local Adaptation

The core component of our architecture is the *threshold gating node*. Each node maintains:

- An accumulator that sums inbound signals since the last release,
- An adaptive firing threshold δ,
- A timer tracking the elapsed time since the last firing,
- A forced-release interval $timeToRelease$, after which the node fires even if the threshold is not met,
- An eligibility trace encoding recent activity for credit assignment,

- An error input, which is a locally received forward-fed signal (not a global backpropagated gradient).

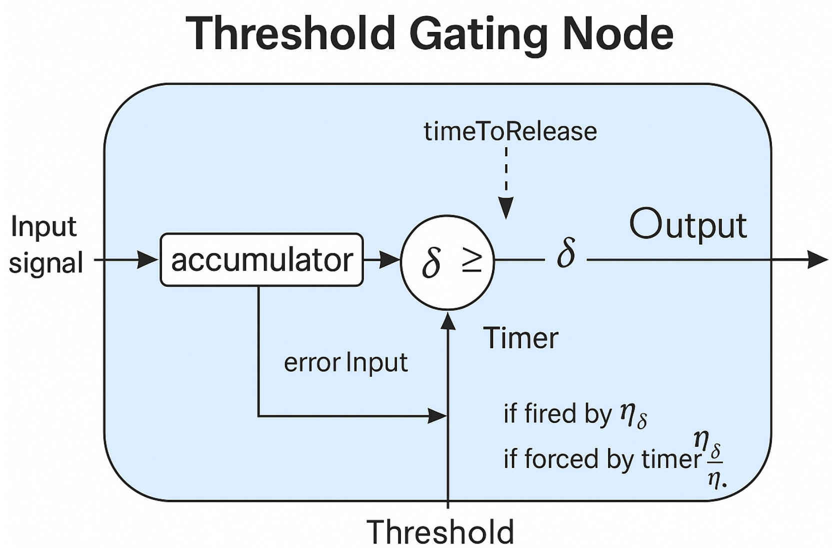

Figure 2: Detailed schematic of a threshold gating node. The node sums incoming values in the accumulator; fires (and updates) when the accumulator crosses the adaptive threshold δ or the timer T expires. The eligibility trace tracks recent activity for credit assignment. Error inputs are used to adjust adaptation rates. Local memory stores all internal parameters and history.

The node fires when its accumulator exceeds δ or when its timer reaches *timeToRelease*. Upon firing:

$$\Delta\delta = \begin{cases} +\eta_\delta, & \text{if fired by threshold,} \\ -\frac{1}{2}\eta_\delta, & \text{if forced by timer,} \end{cases} \tag{2}$$

where η_δ is the threshold adaptation rate. The adaptation rate itself is updated by the error input:

$$\eta_\delta \leftarrow \eta_\delta + \alpha \cdot \text{errorInput} \tag{3}$$

where α controls sensitivity to error. The timer *timeToRelease* is also co-adapted, typically with a separate rate η_{timer}, allowing flexible specialization in temporal dynamics:

If fired by threshold:

$$\delta \leftarrow \delta + \eta_\delta, \quad timeToRelease \leftarrow timeToRelease + \eta_{\text{timer}}.$$

If fired by timer:

$$\delta \leftarrow \delta - \tfrac{1}{2}\eta_\delta, \quad timeToRelease \leftarrow \max\big(1,\, timeToRelease - \tfrac{1}{2}\eta_{\text{timer}}\big).$$

3.6 Three-Level Memory Hierarchy

Our architecture implements a hierarchical memory system that draws on neuroscience and distributed systems principles to enable both rapid, context-sensitive adaptation and robust, long-term knowledge retention:

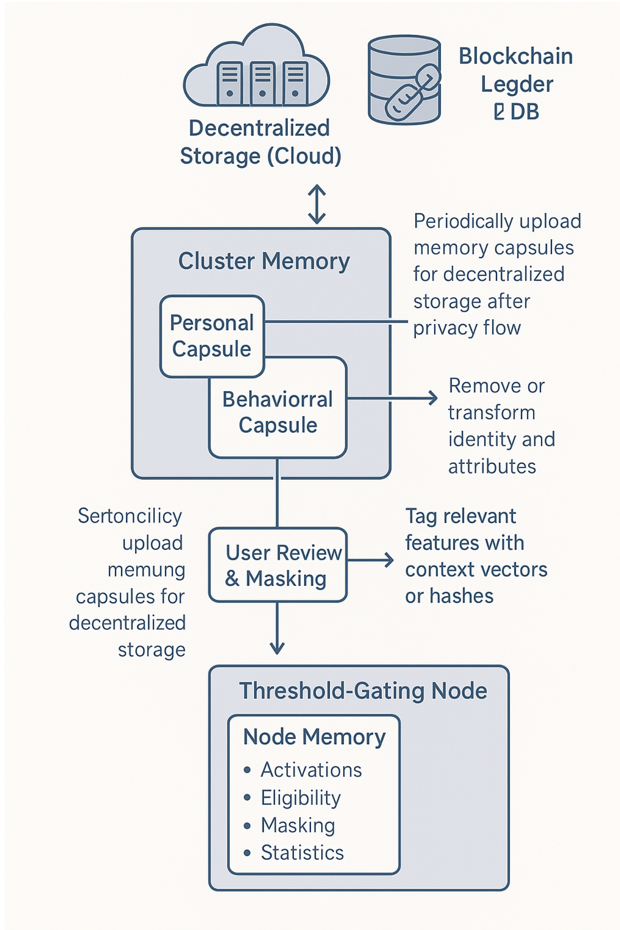

Figure 3: Diagram of the three-level memory hierarchy in the distributed neural architecture. Each node maintains local memory with privacy-preserving semantic masking and tagging. Cluster memory aggregates and distinguishes between personal (private, encrypted) and behavioral (shared, reusable) memory capsules, applying both automated and user-guided masking. Capsules are uploaded to decentralized storage, indexed and auditable via blockchain, and rapidly retrievable by vector/tag-based search for context-aware adaptation and lifelong learning.

- **Local Node Memory:** Each node maintains a structured internal buffer containing recent activations, local error signals, adaptive thresholds, eligibility traces, timers, and salient activity tags. Inspired by synaptic tagging and capture [21], nodes mark significant events and prioritize which experiences are retained for plasticity and future adaptation. Sensitive information in this buffer is automatically detected and semantically masked (e.g., names replaced by `[PERSON NAME]`), ensuring privacy at the source.
- **Local Cluster Memory:** Clusters—groups of nodes on a device or within a single modality—share a memory space that aggregates outputs, errors, and adaptation histories from their constituent nodes. These clusters coordinate modular specialization, generate context embeddings or semantic hash tags for each "memory capsule," and determine which updates are retained, discarded, or prepared for longer-term storage. When preparing capsules, the system enforces a dual protocol: strictly private (personal) data is encrypted and access-restricted, while abstracted, general behavioral knowledge can be tagged for sharing and

rapid reuse across the network.

- **Global Memory:** Periodically, capsules from clusters are consolidated, compressed, and uploaded to decentralized storage solutions (such as IPFS or Filecoin), with each memory capsule linked to a unique high-dimensional vector tag or semantic hash. All memory uploads, node packet events, and capsule versions are registered in a blockchain-backed ledger and indexed in a decentralized vector database controlled by smart contracts [22, 20]. This infrastructure enables any authorized node or cluster to efficiently retrieve relevant memories based on context, semantic similarity, or recentness—facilitating rapid replay, adaptation, and credit allocation throughout the distributed network.

This three-level hierarchy ensures that learning and memory are both context-sensitive and robust: local nodes and clusters adapt quickly and protect privacy, while global memory guarantees persistent, auditable, and searchable knowledge even as devices join, leave, or rejoin the network. The use of semantic masking, context-based tagging, and blockchain auditing enables lifelong learning, scalable collaboration, and secure, privacy-preserving distributed AI.

3.7 Node Borrowing Protocol

To maximize resource utilization and support robust, collaborative learning across a distributed network, our architecture implements a formal node borrowing protocol. This protocol enables devices or clusters to temporarily "borrow" nodes—complete with their adaptive memory—from remote peers, providing additional computation, redundancy, or expertise as needed.

- **Discovery and Request:** When a device or cluster needs to augment its capacity or expertise, it queries the blockchain ledger or decentralized storage for available node packets matching specific context tags or behavioral capabilities. Search is enabled via the semantic tag index stored on-chain.

- **Permission and Privacy:** Each node packet is annotated with a privacy tag ("personal" or "behavioral") and access control metadata. Only behavioral (non-personal) node memories are eligible for cross-device borrowing, and every packet is cryptographically signed. Upon request, the borrowing device must verify the packet's signature and access rights.

- **Credit and Incentive Accounting:** Borrowing nodes incurs a credit cost, which is recorded via smart contracts on the blockchain. This mechanism incentivizes participants to contribute computational and memory resources to the network, while ensuring fair compensation and auditability.

- **Transfer and Instantiation:** Once permission and payment are confirmed, the node packet (memory snapshot) is securely transmitted to the borrower. The borrowing device decrypts and instantiates the node locally, integrating it as a full participant in its cluster for the duration of the lease.

- **Reinstatement and Update:** Upon completion of the borrowing period, updates to the node's memory are either returned to the original owner (for behavioral memories) or reconciled via global memory consensus mechanisms, maintaining the integrity and lineage of all borrowed nodes.

- **Snapshot Preservation and Respawning After Borrowing:** When a node is borrowed and instantiated on a new device or cluster, its memory and adaptation state will evolve in

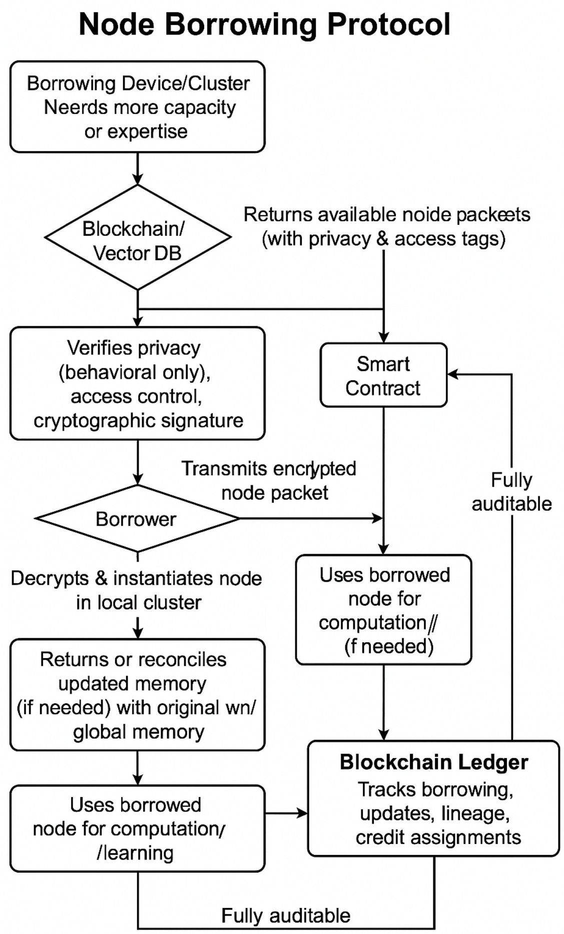

Figure 4: Flowchart illustrating the Node Borrowing Protocol in the distributed neural network. The diagram details the full process: search and discovery of eligible node packets, permission and privacy checks, credit/incentive accounting via smart contracts, secure transfer and instantiation, operational use, and final update or reconciliation, with all actions tracked and audited on the blockchain ledger.

response to local data and tasks. To maintain a verifiable and replayable history, our protocol requires that upon completion of the borrowing period, a new snapshot of the node's updated state is created, signed, and written to the ledger or decentralized storage. This final snapshot serves as the canonical state for any subsequent respawning, borrowing, or auditing events, preserving the complete lineage and credit assignments associated with all node usage. In this way, the system ensures that any changes made during borrowing are fully recorded and that nodes can be reliably restored to their most recent valid state across the network.

This borrowing protocol supports elastic scaling, enables knowledge transfer and rapid recovery after device dropouts, and ensures that personal/private data is never shared without explicit consent. All actions—discovery, borrowing, instantiation, update, and credit assignment—are fully auditable and governed by the distributed ledger, providing transparency, trust, and robust privacy guarantees throughout the network.

3.8 Forward-Only Error and Local Credit Assignment

Traditional neural networks rely on centralized, synchronous backpropagation to distribute error gradients and assign credit for learning. In contrast, our distributed architecture employs a fully forward-only protocol for error propagation and credit assignment, enabling both biological plausibility and scalable, asynchronous operation.

Local Error Propagation: Each threshold-gating node receives a forward-fed error input, which may be computed from local task loss, aggregated cluster performance, or broadcast global modulatory signals. These error signals are strictly feedforward—never requiring global chain-rule gradients—and are used to drive adaptation at each node and cluster independently.

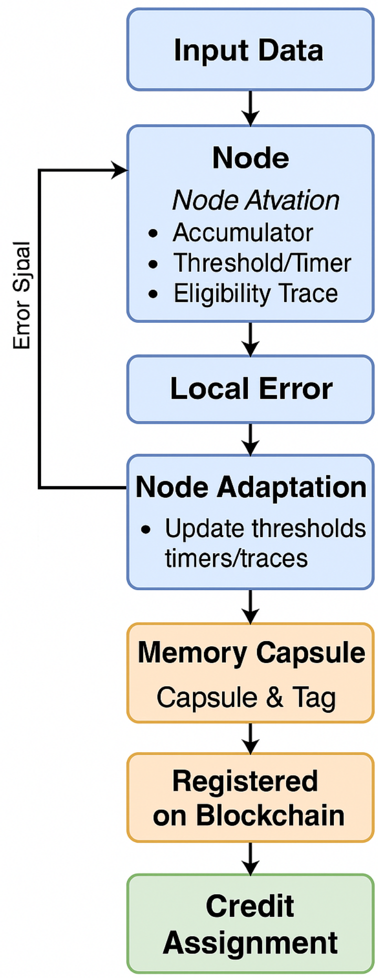

Figure 5: Flowchart of forward-only error propagation and local credit assignment. Input data is processed by threshold-gating nodes; eligibility traces are updated upon firing. Locally or globally computed errors are fed forward to nodes for adaptation. Significant events are tagged and stored as memory capsules, indexed on the blockchain for rapid retrieval and credit assignment.

Eligibility Traces and Temporal Credit Assignment: Within each node, eligibility traces serve as temporally decaying records of recent activation—functioning as synaptic "tags" [14, 21].

These traces ensure that delayed error feedback can still reinforce or attenuate the synaptic changes that contributed to an outcome. Each node's adaptation rate (threshold or timer) is modulated by both the magnitude of the current error signal and the strength of the stored eligibility trace, enabling precise, event-driven credit assignment.

Tagging for Memory Update: Nodes and clusters generate context tags or semantic hashes for memory capsules created during significant learning events, such as large error corrections or impactful discoveries. These tags are registered in the blockchain-backed vector index, supporting rapid retrieval, lineage tracking, and efficient credit assignment for future adaptation.

Distributed Credit Assignment and Incentives: Every memory capsule's contribution to network learning—measured by its effect on error reduction, novelty, or frequency of use—is logged both locally (within the cluster) and globally (on the blockchain). The global aggregator and incentive module transparently register each contribution, ensuring fair, auditable credit allocation as nodes are borrowed, returned, or replicated across the network.

Summary: This forward-only, eligibility-trace-based approach allows each node to adapt based on local and global feedback, enabling robust, asynchronous learning and incentive alignment in dynamic, privacy-preserving, and decentralized environments—eliminating the need for centralized backpropagation.

3.9 Distributed, Privacy-Preserving, and Blockchain-Based Operation

Our architecture is purpose-built for secure, scalable deployment across a global network of heterogeneous user devices, each acting as an adaptive and privacy-preserving cluster within the distributed system.

Device-Level Clusters: Each user device operates one or more clusters of threshold-gating nodes. All raw data—including user inputs, sensor readings, and contextual metadata—remains strictly on-device. Learning, adaptation, and error evaluation occur locally, with only high-level adaptation statistics and derived summaries leaving the device.

Memory Capsule Generation and Tagging: At regular intervals, or in response to significant learning events, each device (or cluster) compresses, encrypts, and semantically tags its relevant adaptation histories, eligibility traces, and local context into a "memory capsule." Tagging is performed using high-dimensional vector embeddings, context hashes, or interpretable semantic labels, supporting rapid future retrieval and disambiguation. Sensitive fields are semantically masked before export, and users may review or customize additional masking as needed.

Distributed Storage and Blockchain Indexing: Generated memory capsules are uploaded to decentralized storage solutions (e.g., IPFS, Filecoin, Sia), ensuring redundancy and resistance to tampering or censorship. Each capsule's semantic tag, cryptographic hash, and retrieval pointer are registered via smart contract on a public or consortium blockchain, providing global auditability, immutable attribution, and secure access control.

Blockchain-Based Incentives and Credit: The blockchain maintains an immutable, transparent log of all memory uploads, node events, and network topology changes (e.g., node splits, merges, capsule updates, borrowing events). Smart contracts automate the allocation of incentives—such as tokens, compute credits, or access privileges—based on measurable contributions, including memory utility, relevance, hosting uptime, and demonstrated error reduction. This incentivizes diverse, active participation while ensuring fair, trustless collaboration.

Rapid and Secure Long-Term Memory Access: When nodes or clusters require relevant long-term knowledge (for adaptation, error correction, or credit reconciliation), they query the blockchain-indexed vector database for capsules with semantically similar tags or context. Decryption and usage are permitted only for authorized nodes, guaranteeing privacy, security, and compliance with user-defined or protocol-level data-sharing policies.

Resilience, Dropout, and Rejoining: Blockchain registration and distributed storage ensure that memory capsules are redundantly archived and globally discoverable, even as individual devices join, disconnect, or migrate between clusters. Upon rejoining, a device can efficiently resynchronize its state by querying the global index and securely retrieving and decrypting the latest relevant capsules.

Summary: This privacy-first, blockchain-based protocol achieves robust, decentralized learning and memory—combining local computation, rapid context-driven recall, auditability, and incentive-aligned scaling for distributed AI systems.

4 Neuroscience Foundations

The design of our architecture is inspired by foundational discoveries in neuroscience regarding memory, adaptation, and distributed learning. The following core principles directly inform the structure and operation of our system:

- **Hierarchical and Distributed Memory:** Biological memory operates at multiple nested timescales—working, episodic, and long-term—across distributed anatomical substrates. Local memory traces are encoded in individual neurons and synapses, while higher-level structures (such as cortical columns and hippocampal circuits) coordinate integration and consolidation [16, 23]. Errors, surprises, and novelty are detected locally (e.g., hippocampus, cerebellum), then globally signaled by neuromodulators (dopamine, acetylcholine, norepinephrine), which tag salient events for prioritization and consolidation [17, 1, 24].

- **Local Learning and Eligibility Traces:** Synaptic plasticity is controlled not just by the correlation of pre- and post-synaptic activity (Hebbian learning), but also by global modulatory signals ("three-factor learning rules") that regulate when and where plasticity occurs [14, 2, 25]. These mechanisms are often formalized as eligibility traces: local, temporally decaying markers that allow delayed feedback or reinforcement signals to influence which synapses are strengthened or weakened. This supports credit assignment even when feedback is sparse or temporally separated from the original activity.

- **Temporal Processing, Adaptive Thresholds, and Homeostasis:** Neurons dynamically adjust their firing thresholds and integration windows in response to activity, facilitating adaptive homeostasis, history-dependent learning, and multi-timescale integration [15, 3, 26].

Such mechanisms ensure both rapid sensitivity to new information and long-term stability in the face of changing environments or ongoing plasticity.

- **Structural Plasticity, Modularity, and Self-Organization:** Synaptic growth, pruning, and rewiring enable the formation, maintenance, and dissolution of local subnetworks (columns, clusters, assemblies) [4, 5, 27]. Modular and hierarchical organization, observed from the microcircuit to the system scale, supports both robust specialization and flexible reconfiguration, enabling learning that is both resilient and scalable.

- **Memory Tagging, Prioritization, and Replay:** The brain employs tagging and replay mechanisms to prioritize salient experiences for consolidation during offline periods (e.g., sleep), as observed in hippocampal sharp-wave ripples and cortex-wide replay events [28, 29, 30]. These processes ensure that only a subset of events—those deemed relevant or novel—are retained for long-term plasticity.

These neuroscience principles underpin our system's approach to local and global memory, adaptive learning, credit assignment, error signaling, structural reorganization, and prioritization of salient experiences for robust, distributed, and biologically plausible AI.

5 System Architecture

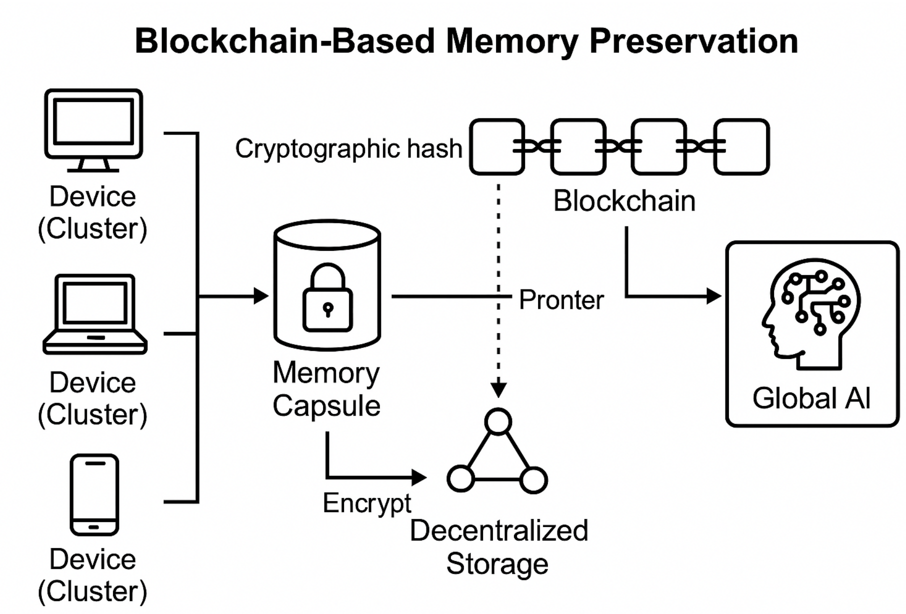

Figure 6: High-level overview of the distributed, forward-only neural architecture. Each user device is a privacy-preserving cluster; clusters interact via blockchain for coordination, storage, and incentives.

5.1 Device-Level Clusters

Each device runs a cluster of threshold-gating nodes. All raw data and low-level adaptation remain on-device.

- **Threshold-Gating Node:** Local event-driven unit, as detailed in Methods.
- **Cluster Memory:** Aggregates local node statistics, manages adaptation and specialization, prepares memory capsules for upload.
- **Dynamic Topology:** Nodes and edges split/merge based on usage, supporting modular specialization and bandwidth scaling.

5.2 Hierarchical Memory and Blockchain Integration

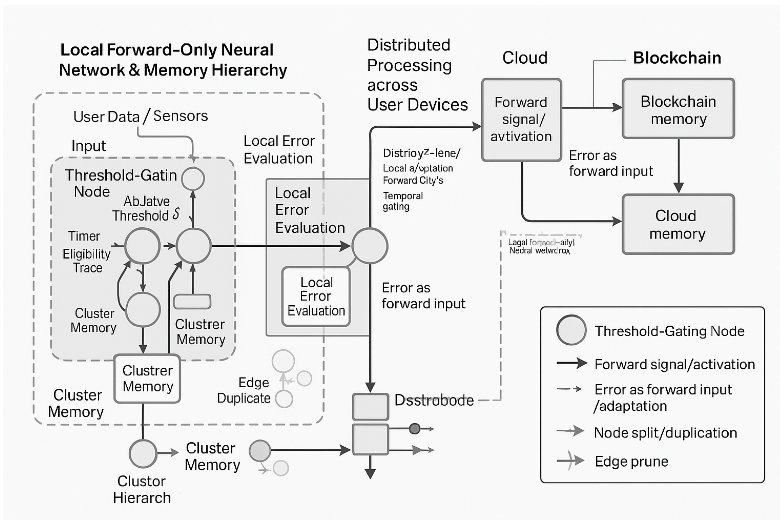

Figure 7: Hierarchical memory management: local node memory, cluster memory capsules, semantic masking, and blockchain-indexed global memory. Only encrypted, context-tagged summaries leave the device.

- **Semantic Masking/User Control:** Automated masking of sensitive fields, with user review.
- **Capsule Upload and Tagging:** Capsules encrypted, tagged, and stored on decentralized storage; indexed and auditable via blockchain.
- **Memory Recall and Resilience:** Devices query blockchain-indexed vector database for relevant capsules; can resync after dropout.

5.3 Incentive System and Auditability

- **Smart Contracts:** Automate incentive allocation, permissions, credit assignment.
- **Auditability:** All uploads, borrowings, updates, and incentives are logged immutably.

Summary: This architecture combines biologically inspired learning, privacy-preserving local adaptation, and blockchain-based incentives and memory for scalable, distributed, and lifelong AI.

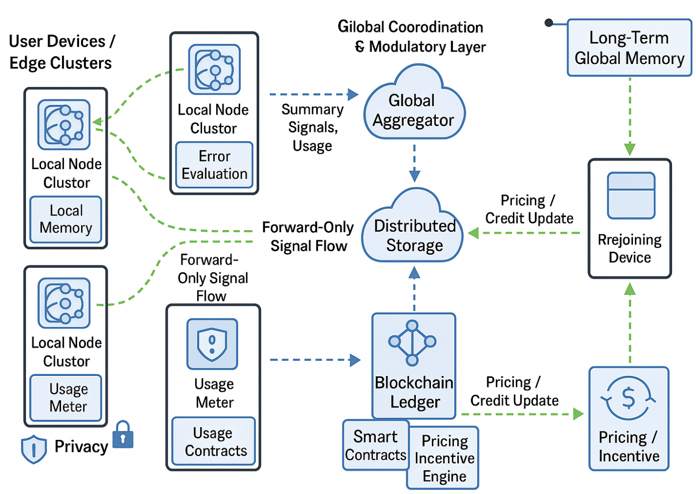

Figure 8: Memory and Cost Management

6 Detailed Memory, Temporal Plasticity, and Network Growth

6.1 Hierarchical Memory Management

Each threshold-gating node encodes activations and error signals in a local buffer, forming the basis of rapid adaptation. Upon each firing event (by threshold or timer), the node:

- Appends an activation record to its local memory,
- Updates an eligibility trace reflecting recency and importance of activity,
- Periodically flushes or compresses these records into a "memory capsule" for cluster memory.

Cluster memory on each device aggregates these capsules, enabling:

- Local error evaluation and novelty detection,
- Coordination of adaptation rates and thresholds,
- Specialization and decision-making on node split/duplication.

At longer intervals, device-level 'consolidation' aggregates, compresses, and encrypts the important features and adaptation statistics of all clusters, producing a global 'memory capsule' for upload to distributed storage and blockchain registration.

6.2 Temporal Functions and Plasticity

Plasticity arises from the interplay of thresholds, timers, and eligibility traces:

- The threshold δ of each node and the timer T are adaptive, modulated by the error and the input history.
- The firing of the threshold increases δ, making the node less sensitive; the firing of the timer decreases δ, making it more sensitive.

- The timer T is lengthened or shortened according to firing patterns, allowing flexible 'pacing' and adaptation.
- Eligibility traces act as temporally decaying buffers, tagging events for future updates in response to global error or reward.

This enables both short-term and long-term dependencies, supports bursty or nonstationary input, and provides stable yet flexible specialization.

6.3 Distributed Growth, Duplication, and Pruning

The network structure evolves dynamically:

- **Edge Duplication:** High-usage edges are duplicated to increase bandwidth and redundancy.
- **Node Splitting:** Overloaded nodes are split, with parameters and memory inherited and then diverged; split threshold grows with each division.
- **Edge/Node Pruning:** Underused edges are pruned; nodes without input are removed.
- **Cross-Device Expansion:** As new devices join, global memory enables rapid assignment of clusters/tasks and seamless scaling. Upon dropout, the device's last memory is preserved; the rejoining devices instantly resync.

6.4 Coordinated Expansion via Blockchain and Incentives

Smart contracts on blockchain:

- Propose and audit structural changes (e.g., cluster migration, expansion, redundancy),
- Allocate incentives for hosting memory, supporting rebalancing, and network healing,
- Maintain trust, fairness, and transparency in all memory and adaptation events.

7 Three-Level Memory and Global Coordination

A memory hierarchy ensures robustness:

- **Local Node Memory:** Accumulators, thresholds, timers, eligibility traces, and recent activation history.
- **Local Cluster Memory:** Aggregates outputs/errors, steers local adaptation.
- **Global Memory (Distributed/Cloud):** Consolidated capsules uploaded to redundant storage, indexed and auditable via blockchain.

The **Global Aggregator** periodically merges memory capsules, issues global modulatory signals, and enables fair credit assignment for all contributors.

7.1 Device Dropout, Redundancy, and Rejoin

Long-term memory is protected through:

- Redundant, encrypted storage by archive nodes or decentralized services,
- Blockchain audit for every contribution,
- Seamless rejoin and resync for dropped devices using latest global memory.

8 Blockchain, Auditability, and Incentives

Blockchain provides:

- Immutable records of memory uploads, structural changes, and device events,
- Transparent credit and incentive distribution via smart contracts,
- Proof-of-contribution and fair recognition for all participants,
- Guaranteed privacy (only hashes, encrypted models published; raw data never leaves device).

9 Workflow and Experiments

1. The device performs local, forward-only learning with threshold-gating nodes, updating node, and cluster memory.
2. At intervals, compresses and encrypts 'memory capsule', uploads to distributed storage, records hash/pointer on blockchain.
3. The global aggregator consolidates capsules, updates global memory, and issues global modulatory signal and pricing/incentive feedback.
4. The network grows or shrinks via edge/node split/duplication/pruning, coordinated by smart contracts.
5. Devices dropping out have contributions persisted; rejoining is seamless via global memory resync.

Experiments on multimodal data demonstrate:

- Fast adaptation to new patterns and tasks,
- Robustness to device dropout,
- Efficient use of incentives for active contribution,
- Stable memory and dynamic network growth.

10 Discussion

10.1 Advantages

- **Biological Plausibility:** Mirrors known brain mechanisms—local adaptation, temporal memory, eligibility traces, and global modulation.
- **Privacy-First and Decentralized:** Raw data never leaves user device; only encrypted summaries are shared.
- **Resilience and Scaling:** Distributed, auditable memory and network growth allow for robust, self-healing AI at scale.
- **Community-Driven Incentives:** Smart contracts ensure fair compensation for all contributors and archival nodes.

10.2 Limitations

- **Communication/Storage Overhead:** Memory capsules and blockchain records, while small, must be managed efficiently at scale.
- **Parameter Tuning:** Network plasticity and incentive models require continued research for optimal convergence and fairness.
- **Infrastructure Dependencies:** Distributed storage and blockchain security depend on community and ecosystem support.

11 Conclusion and Future Work

We have introduced a distributed, modular, forward-only neural network architecture inspired by biological memory, with dynamic adaptation, blockchain-audited memory, and community-driven incentives. Our system provides a blueprint for privacy-first, resilient, and scalable AI learning. Future research will focus on larger-scale deployments, automated parameter tuning, neuromorphic implementation, and formal theoretical analysis.

References

[1] Doya, K. (2000). Complementary roles of basal ganglia and cerebellum in learning and motor control. *Current Opinion in Neurobiology*, 10(6), 732-739.

[2] Yagishita, S., et al. (2014). A critical time window for dopamine actions on the structural plasticity of dendritic spines. *Science*, 345(6204), 1616-1620.

[3] Fusi, S., Drew, P. J.,Abbott, L. F. (2005). Cascade models of synaptically stored memories. *Neuron*, 45(4), 599-611.

[4] Holtmaat, A.,Svoboda, K. (2009). Experience-dependent structural synaptic plasticity in the mammalian brain. *Nature Reviews Neuroscience*, 10(9), 647-658.

[5] Perin, R., Berger, T. K.,Markram, H. (2011). A synaptic organizing principle for cortical neuronal groups. *Proceedings of the National Academy of Sciences*, 108(13), 5419-5424.

[6] D.E. Rumelhart, G.E. Hinton, and R.J. Williams. Learning representations by back-propagating errors. *Nature*, 323(6088):533–536, 1986.

[7] Y. LeCun, Y. Bengio, and G. Hinton. Deep learning. *Nature*, 521(7553):436–444, 2015.

[8] E.M. Izhikevich. Which model to use for cortical spiking neurons? *IEEE Transactions on Neural Networks*, 15(5):1063–1070, 2004.

[9] Y. Bengio, A. Courville, and P. Vincent. Representation learning: A review and new perspectives. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 35(8):1798–1828, 2013.

[10] G.E. Hinton. Forward-forward algorithm: Some preliminary investigations. *arXiv preprint arXiv:2212.13345*, 2022.

[11] D.O. Hebb. *The Organization of Behavior: A Neuropsychological Theory.* Wiley, 1949.

[12] J. Ngiam, A. Khosla, M. Kim, J. Nam, H. Lee, and A.Y. Ng. Multimodal deep learning. In *Proceedings of the 28th Intl. Conf. on Machine Learning (ICML)*, 2011.

[13] D. Ramachandram and D. Rodríguez-Serrano. Deep multimodal learning: A survey on recent advances and trends. *IEEE Signal Processing Magazine*, 34(6):96–108, 2017.

[14] W. Gerstner, M. Lehmann, V. Liakoni, D. Corneil, and J. Brea. Eligibility traces and plasticity on behavioral time scales: Experimental support of neohebbian three-factor learning rules. *Frontiers in Neural Circuits*, 12:53, 2018.

[15] C. Pozzorini, R. Naud, S. Mensi, and W. Gerstner. Temporal whitening by power-law adaptation in neocortical neurons. *Nature Neuroscience*, 16(7):942–948, 2013.

[16] J. Lisman. Memory formation and error signals in hippocampus. *Trends in Cognitive Sciences*, 24(8):678–689, 2020.

[17] S.J. Sara. The locus coeruleus and noradrenergic modulation of memory. *Nature Reviews Neuroscience*, 10(3):211–223, 2009.

[18] P. Kairouz, H.B. McMahan, B. Avent, et al. Advances and open problems in federated learning. *Foundations and Trends in Machine Learning*, 14(1–2):1–210, 2021.

[19] K. Christidis and M. Devetsikiotis. Blockchains and smart contracts for the internet of things. *IEEE Access*, 4:2292–2303, 2016.

[20] G. Zyskind, O. Nathan, and A. Pentland. Decentralizing privacy: Using blockchain to protect personal data. In *2015 IEEE Security and Privacy Workshops*, pages 180–184, 2015.

[21] U. Frey and R.G.M. Morris. Synaptic tagging and long-term potentiation. *Nature*, 385(6616):533–536, 1997.

[22] Petrosyan, S., King, D., Rohit, S. Decentralized Vector Databases and Blockchain-Indexed Storage for Large-Scale AI. *Proceedings of the International Conference on Distributed Systems and Blockchain*, 2023.

[23] Buzsáki, G., Tingley, D. Space and Time: The Hippocampus as a Sequence Generator. *Trends in Cognitive Sciences*, 23(10), 853-869, 2019.

[24] Shohamy, D., Adcock, R. A. Dopamine and adaptive memory. *Trends in Cognitive Sciences*, 14(10), 464-472, 2010.

[25] Frémaux, N., Gerstner, W. Neuromodulated Spike-Timing-Dependent Plasticity, and Theory of Three-Factor Learning Rules. *Frontiers in Neural Circuits*, 9, 85, 2016.

[26] Turrigiano, G. Homeostatic synaptic plasticity: local and global mechanisms for stabilizing neuronal function. *Cold Spring Harbor Perspectives in Biology*, 4(1), a005736, 2012.

[27] Sporns, O. Modular structure and dynamics of brain networks. *Nature Reviews Neuroscience*, 17(10), 662–672, 2016.

[28] Carr, M. F., Jadhav, S. P., Frank, L. M. Hippocampal replay in the awake state: a potential substrate for memory consolidation and retrieval. *Nature Neuroscience*, 14(2), 147–153, 2011.

[29] Girardeau, G., Zugaro, M. Hippocampal ripples and memory consolidation. *Current Opinion in Neurobiology*, 21(3), 452–459, 2011.

[30] McClelland, J. L., McNaughton, B. L., O'Reilly, R. C. Why there are complementary learning systems in the hippocampus and neocortex: insights from the successes and failures of connectionist models of learning and memory. *Psychological Review*, 102(3), 419–457, 1995.