An Algorithmic Architecture for a Synthetic Brain Inspired by Human Memory

Patrick Ribeiro Negri
Sao Paulo

May 2025

Abstract

The quest for artificial intelligence systems exhibiting human-like cognitive flexibility requires architectures capable of managing knowledge and experience in a structured, persistent, and context-aware manner. Many current AI models lack robust memory systems comparable to the multifaceted nature of human cognition. This paper details the algorithmic architecture of the `SyntheticBrain`, a system designed to emulate key aspects of human memory organization and processing. It features a multi-component memory system including Short-Term Memory (STM), and distinct Long-Term Memory (LTM) stores for Semantic, Episodic, and Implicit knowledge, drawing inspiration directly from cognitive science models. The architecture incorporates algorithms for encoding incoming information, consolidating memories based on importance and usage, retrieving contextually relevant information, and adaptively pruning less significant data. By focusing on these cognitively inspired algorithmic principles—such as capacity-limited working memory, distinct LTM pathways, importance-based retention, and context-dependent retrieval—the `SyntheticBrain` architecture aims to provide a foundation for AI systems with enhanced consistency, reasoning, and adaptability, grounded purely in its implementation and established memory concepts.

1 Introduction

Artificial Intelligence (AI) has made significant strides, yet replicating the nuanced cognitive capabilities of the human mind, particularly its sophisticated memory functions, remains a formidable challenge. While models excel at specific tasks, they often fall short in maintaining long-term consistency, reasoning deeply from past experiences, and dynamically managing context over extended interactions. This limitation stems, in part, from the absence of persistent, structured memory systems akin to those observed in human cognition. Addressing this gap requires the development of cognitive architectures that are not only powerful but also grounded in principles allowing for robust knowledge organization, retrieval, and adaptation.

This paper introduces the `SyntheticBrain`, an algorithmic architecture designed explicitly to model key functional aspects of human memory systems. Inspired by cognitive science research [1, 2], the `SyntheticBrain` implements a multi-component memory structure, including a capacity-limited Short-Term (Working) Memory and distinct Long-Term Memory stores for semantic (factual), episodic (experiential), and implicit (procedural/associative) knowledge. This structure facilitates a more organized and contextually relevant handling of information compared to simpler memory buffers.

The core contribution of this work lies in detailing the *algorithmic principles* that govern the `SyntheticBrain` architecture. We describe the algorithms responsible for encoding new information into appropriate memory stores, consolidating memories based on calculated importance and usage patterns, retrieving relevant knowledge based on current context, and adaptively pruning or forgetting less salient information. These processes are designed to mimic, at a functional level, the dynamics observed in human memory, such as the limited capacity of working memory, the distinct characteristics of different LTM types, the role of consolidation in learning, and the importance of context in retrieval [3, 4].

Unlike approaches that draw analogies from unrelated complex systems, the `SyntheticBrain` architecture is presented based solely on its internal design and its grounding in established models of human memory. By focusing on these cognitively inspired algorithms, we aim to provide a clear blueprint for developing AI systems with enhanced capabilities for sustained reasoning, learning from experience, and maintaining coherence over time. The following sections will review the relevant background on human memory (Section 2), elaborate on the `SyntheticBrain` architecture and its algorithms (Section 3), discuss its implications and limitations (Section 4), and offer concluding remarks (Section 5).

2 Background: Human Memory Systems

To provide a foundation for the `SyntheticBrain` architecture, we briefly review established models of human memory, focusing on the structures and processes that inspire its design.

2.1 Core Memory Systems

Human memory is understood as a collection of interacting systems rather than a single entity. The journey of information often begins in **Sensory Memory**, a very brief (milliseconds to seconds), high-capacity buffer holding raw sensory input. Information selected by attention proceeds to **Short-Term Memory (STM)**, characterized by its limited capacity (around 7 ± 2 items) and duration (seconds to minutes without active maintenance). The concept of **Working Memory (WM)** expands on STM, describing a more active system responsible for temporarily holding *and manipulating* information essential for ongoing

cognitive tasks like reasoning, comprehension, and learning. Influential models like Baddeley
's include components such as the Central Executive (attentional control) and specialized buffers for verbal (Phonological Loop) and visuospatial information [1, 2].

2.2 Long-Term Memory (LTM)

Information successfully processed in WM can be encoded into **Long-Term Memory (LTM)**, the brain
's vast repository for durable knowledge and experiences. LTM is typically categorized into:

- **Explicit (Declarative) Memory:** This involves conscious recollection. It further subdivides into:
  - *Episodic Memory:* Stores personal experiences and events, tagged with specific times and places (autobiographical memory).
  - *Semantic Memory:* Comprises general knowledge about the world, including facts, concepts, language, and schemas, detached from the specific learning context [1].
- **Implicit (Non-Declarative) Memory:** This influences behavior unconsciously, without deliberate recall. It includes:
  - *Procedural Memory:* Encodes skills, habits, and motor patterns acquired through repetition (e.g., riding a bicycle).
  - *Priming:* Exposure to a stimulus influences the response to a subsequent stimulus.
  - *Associative Learning:* Classical and operant conditioning [3].

The `SyntheticBrain` architecture attempts to functionally model the distinctions between STM/WM and the explicit LTM stores (Semantic and Episodic), while also providing a placeholder for Implicit memory.

2.3 Fundamental Memory Processes

Several dynamic processes govern how memories are formed, maintained, and used:

- **Encoding:** The initial process of transforming incoming information into a format that can be stored in memory. This can range from simple sensory registration to complex semantic elaboration.
- **Consolidation:** A time-dependent process that stabilizes a memory trace after its initial acquisition. It involves changes at synaptic and systems levels, making the memory more resistant to disruption. Sleep is thought to play a significant role in consolidation [3].

- **Retrieval:** The process of accessing stored information. Retrieval is often context-dependent, meaning cues present during encoding can facilitate later recall. Retrieval can be conscious (explicit) or unconscious (implicit).

- **Forgetting:** The inability to retrieve previously stored information. Far from being solely a failure, forgetting is considered an adaptive process that helps prioritize relevant information and prevent cognitive overload. Mechanisms include decay over time, interference from other memories, retrieval failure due to inadequate cues, or potentially active suppression processes [4].

The `SyntheticBrain` incorporates algorithmic analogues for encoding, consolidation (including importance-based retention), retrieval (context-aware), and forgetting (pruning based on capacity and importance).

3 The SyntheticBrain Architecture: Algorithmic Principles

The `SyntheticBrain` architecture provides a modular framework designed to equip AI systems with more structured and persistent memory, drawing functional inspiration from human cognitive models. It aims to manage information flow, storage, and retrieval in a way that supports more coherent and context-aware interactions. Figure 1 illustrates the overall architecture and data flow.

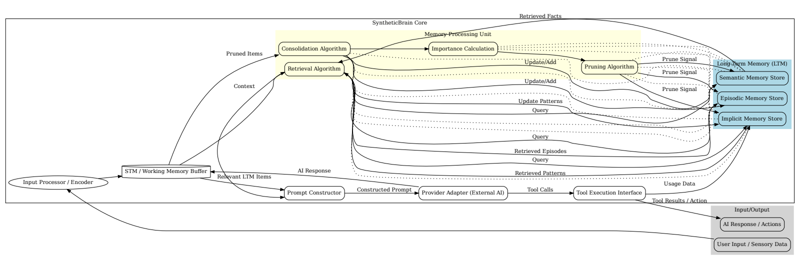

Figure 1: SyntheticBrain Architecture Diagram

3.1 Architectural Components

The architecture, as implemented, comprises several distinct components working in concert:

1. **Input Processor/Encoder:** This initial stage receives external inputs (e.g., user messages containing text, images, or files, represented by `UserMessage` and `ContentItem` classes) and structures them for internal processing.

While the provided code focuses on structuring input messages, a complete cognitive architecture implies a more sophisticated encoding process, potentially involving feature extraction and semantic tagging before information enters working memory.

2. **Short-Term Memory (STM) / Working Memory Buffer:** Implemented as the `ShortTermMemory` class, this component serves as a temporary, capacity-limited buffer for the immediate context of an interaction. It holds recent messages (user inputs and AI responses). The core algorithm involves adding new messages and pruning older ones when the defined `capacity` is exceeded. Notably, the implementation prioritizes retaining system messages during pruning, ensuring core instructions persist.

3. **Long-Term Memory (LTM) Stores:** The architecture implements distinct stores for different types of long-term knowledge, mirroring the declarative/implicit distinction:

   - *Semantic Memory Store (`SemanticMemory` class):* Designed to hold factual and conceptual knowledge, implemented as a key-value store where each entry is a `MemoryItem`. The algorithm allows adding new facts or updating existing ones based on a unique key. It includes basic tracking of `importance`, `access_count`, and `last_accessed` time for each item.
   - *Episodic Memory Store (`EpisodicMemory` class):* Stores sequences of events or specific interaction snippets as `MemoryItem` objects in a list. It has a defined `capacity`. The addition algorithm appends new items, and a pruning algorithm removes items when capacity is exceeded, prioritizing retention based on a combination of `importance`, `last_accessed` time, and `access_count`.
   - *Implicit Memory Store (`ImplicitMemory` class):* Intended to capture procedural knowledge, biases, or learned patterns. The implementation provides a structure for storing such data (e.g., tool usage statistics, interaction patterns) and includes methods for updating these patterns based on interaction history and outcomes.

4. **Memory Processing Unit:** This logical unit encompasses the algorithms that manage memory dynamics:

   - *Consolidation Algorithm:* The `consolidate` methods within each LTM store represent placeholders for more complex processes. In the current implementation (`SyntheticBrain.step`), consolidation is triggered after an interaction. It involves transferring information deemed important from the pruned STM items into the appropriate LTM stores (Semantic or Episodic) and potentially running the internal `consolidate` methods of the LTM stores themselves (which currently mainly update metadata like `last_consolidated`). A more advanced algorithm would involve recalculating importance scores and potentially restructuring memory links.

- *Importance Calculation Algorithm:* Each `MemoryItem` has an `importance` score. This score is assigned upon creation and can be updated. The pruning mechanism in `EpisodicMemory` explicitly uses `importance`, `last_accessed`, and `access_count` for retention decisions, implying an algorithm where access and recency contribute to perceived importance, alongside any initially assigned value.
- *Retrieval Algorithm:* The `_construct_prompt_with_memory` method embodies the retrieval process. It selects relevant memories from Semantic and Episodic stores based on the current STM context (recent messages). While the exact selection heuristic isn
't fully detailed in the provided snippets, it implies an algorithm that matches context keywords or semantic content against LTM items, potentially using metadata and importance scores to rank and select the most relevant memories for prompt construction.
- *Forgetting/Pruning Algorithm:* Explicitly implemented in `ShortTermMemory` (capacity-based FIFO, preserving system messages) and `EpisodicMemory` (capacity-based, prioritizing by importance/recency/access). Implicitly, items not accessed or reinforced might have their importance decay (though decay isn
't explicitly coded), making them candidates for pruning during consolidation.

5. **Prompt Constructor:** The `_construct_prompt_with_memory` method acts as the prompt constructor, dynamically assembling the input for the underlying AI model by combining system instructions, current STM content (conversation history), and relevant retrieved LTM items (semantic facts, past episodes).

6. **Tool Execution Interface:** The architecture includes mechanisms (`tools` parameter, `_handle_tool_calls`) to define and manage interactions with external tools. The algorithms differentiate between tool calls that allow immediate continuation (`continue`) and those requiring external processing (`wait`), managing the interaction flow accordingly.

7. **State Management:** The `save_state` and `load_state` methods provide algorithms for serializing and deserializing the entire cognitive state, including the contents and configurations of all memory components, enabling persistence and resumption of interactions.

3.2 Algorithmic Flow Example

A typical interaction (`SyntheticBrain.step`) follows this algorithmic flow:

1. Receive and encode the user message (`UserMessage`).

2. Add the message to `ShortTermMemory`.

3. Construct a prompt using system instructions, `ShortTermMemory` content, and relevant memories retrieved from LTM (`SemanticMemory`, `EpisodicMemory`) based on context.

4. Send the prompt to the underlying AI model via the `provider_adapter`.

5. Receive the AI model
’s response.

6. Add the AI response to `ShortTermMemory`.

7. Handle any tool calls requested by the AI model, potentially pausing (`wait`) or looping back (`continue`) after tool execution.

8. Perform memory consolidation: Prune `ShortTermMemory`, identify important pruned items, and add/update them in the appropriate LTM stores (`SemanticMemory`, `EpisodicMemory`). Update `ImplicitMemory` based on the interaction.

9. Return the final response, execution state (`wait` or `continue`), required tool calls, and token usage.

This architecture provides a structured approach to managing information within an AI system, grounded in functional parallels with human memory.

4 Discussion

The `SyntheticBrain` architecture, grounded in functional analogies to human memory systems, presents a structured approach to managing information within AI. Its design offers potential benefits for creating more consistent and context-aware agents, while also highlighting areas for further development.

Cognitive Plausibility: The architecture deliberately mirrors several key aspects of human memory. The distinction between a capacity-limited STM/WM buffer and distinct LTM stores (Semantic, Episodic, Implicit) aligns with dominant cognitive models [1, 2]. Processes like capacity-based pruning in STM, importance/recency-based retention in LTM (`EpisodicMemory` pruning), context-dependent retrieval (`_construct_prompt_with_memory`), and the concept of consolidation (transferring information from STM to LTM) are functional analogues of biological memory processes. However, these are high-level abstractions. The implementation uses explicit data structures and algorithms (like sorted lists for pruning) rather than simulating underlying neural mechanisms. The `importance` metric, while useful, is a simplification of the complex factors governing memory salience in humans.

Advantages of the Architecture: Implementing distinct memory components offers several advantages over simpler approaches like extending context windows or using undifferentiated vector stores. (1) **Improved Consistency:** Storing core facts (Semantic) and past interactions (Episodic) allows the AI to maintain consistency over longer dialogues. (2) **Enhanced Context Management:** The interplay between the limited STM buffer and context-aware retrieval from LTM allows the system to focus on relevant information without being overwhelmed by the entire history. (3) **Structured Knowledge:** Separating semantic, episodic, and implicit knowledge allows for different processing and retrieval strategies tailored to each type. (4) **Adaptation:** The mechanisms for updating memory importance and pruning less relevant items provide a basis for adaptation and learning from experience, albeit in a simplified form.

Algorithmic Considerations: The efficiency of the architecture depends on the underlying algorithms. Retrieval from LTM, especially as memory stores grow, could become a bottleneck if based on simple linear scans or basic metadata matching. More sophisticated indexing and search algorithms (e.g., semantic vector search within stores) would be necessary for true scalability. Consolidation frequency and the complexity of importance recalculation also impact performance. Parameter tuning—such as STM/LTM capacities, importance decay rates (if implemented), and pruning thresholds—is crucial and likely application-dependent.

Limitations: The current architecture, as inferred from the code, has limitations. The consolidation process is relatively simple, primarily involving transfer from STM to LTM and basic pruning, rather than complex reorganization or abstraction. The `importance` metric relies heavily on access counts and recency, potentially missing other dimensions of salience. The semantic understanding required for effective encoding, linking, and retrieval largely depends on the capabilities of the external AI model integrated via the `provider_adapter`, rather than being an intrinsic part of the memory architecture itself. The modeling of implicit/procedural memory is also less developed compared to the explicit stores.

Future Work: Several avenues exist for extending this architecture. Implementing more sophisticated consolidation algorithms that perform abstraction (e.g., deriving semantic rules from multiple episodes) or strengthen associative links would enhance learning. Developing richer models for calculating memory `importance`, potentially incorporating simulated emotional valence or feedback from task success (reinforcement learning), could improve relevance assessment. Exploring advanced data structures and indexing techniques (e.g., graph databases for semantic memory, vector indices for similarity search) within the LTM stores is crucial for scalability. Refining the implicit memory component to better model skill acquisition and automaticity would also be valuable.

5 Conclusion

This paper detailed the algorithmic architecture of the `SyntheticBrain`, a system designed to provide AI with a more structured and human-like memory framework. Inspired by cognitive models, it features distinct Short-Term and Long-Term (Semantic, Episodic, Implicit) memory components, governed by algorithms for encoding, consolidation, context-aware retrieval, and importance-based pruning. By implementing these cognitively grounded principles—such as limited working memory capacity, distinct LTM pathways, and dynamic memory updates—the `SyntheticBrain` offers a step towards AI systems capable of greater consistency, context retention, and adaptation over time. While representing a functional abstraction of complex biological processes, its modular design and focus on algorithmic memory management provide a valuable blueprint for developing more sophisticated and capable artificial cognitive agents.

References

[1] NeuronUP (2022). *A memória: definição, tipos, exercícios e avaliação [Memory: definition, types, exercises, and evaluation]*. https://neuronup.com/br/noticias-de-estimulacao-cognitiva/funcoes-cognitivas/memoria/a-memoria-definicao-tipos-exercicios-e-avaliacao/ (Accessed: 2025-05-04).

[2] Wikipedia (n.d.). *Memória humana [Human memory]*. https://pt.wikipedia.org/wiki/Mem%C3%B3ria_humana (Accessed: 2025-05-04).

[3] Cecierj (n.d.). *A aquisição de memórias [The acquisition of memories]*. Neurocognição. https://extensao.cecierj.edu.br/material_didatico/edc903/html/verde/ma03.htm (Accessed: 2025-05-04).

[4] Meu Cérebro (2024). *Memória e esquecimento: duas faces necessárias da mesma função cerebral? [Memory and forgetting: two necessary faces of the same brain function?]*. https://meucerebro.com/memoria-e-esquecimento/ (Accessed: 2025-05-04).

[5] Anderson, John R., Bothell, Daniel, Byrne, Michael D., Douglass, Scott, Lebiere, Christian, & Qin, Yulin (2004). *An Integrated Theory of the Mind.* Psychological Review, 111(4), 1036-1060.