

CONTENTS

[Abstract](#) [Introduction](#) [Architecture](#) [Memory Atoms](#) [Relevance Scorer](#) [Query Router](#) [Structure-Aware Inge](#)



Token-Native Agent Memory

Why the Next Generation of AI Memory Systems Should Think in Tokens, Not Vectors

ContextFit
Research •
May 2026

[← Back](#)
to github.com/ContextFit/cf
[Home](#)

X:
[@cponsart](#)

Download
Markdown
↓

*Modern AI agent memory systems share a common architectural assumption: that raw conversational context must be converted into embedding vectors before it can be retrieved. This paper challenges that assumption. We present **ContextFit**, a token-native memory retrieval system that operates directly on tokenized text without embedding APIs, LLM preprocessing, or vector databases.*

1. Introduction

The promise of AI agents is continuity: an assistant that remembers what you told it last week, understands your preferences, and applies prior context to new advice. Fulfilling this promise requires memory retrieval—the ability to surface the right prior conversation when it matters.

The dominant approach in 2026 is embedding-based retrieval: convert session text to dense vectors, store them in a vector database, and retrieve by cosine similarity. This architecture has three fundamental limitations:

- **Latency and Cost:** Every access requires API calls (embedding, LLM extraction, vector query), adding hundreds of milliseconds and ongoing costs.
- **Semantic Averaging:** Embedding models compress entire sessions into a single vector, often failing for vague queries like "what should I cook tonight?" where relevant context shares no vocabulary with the query.
- **Opacity:** A cosine distance of 0.73 does not provide an interpretable explanation of why a session was retrieved.

2. Token-Native Architecture

ContextFit's core insight is that the most valuable signals in conversational memory are **structural, not semantic**. We ask: *What kind of memory did the user express, and does this episode's memory type match what this query needs?*

The Token-Native Pipeline:

Ingest (2.7ms/session) → Tokenization → Inverted Index (BM25) → Deterministic Atom Extraction → LSH Signatures.

Query (0.4-9ms) → Deterministic Router → Mode Selection → Structural / Preference / Evidence-Coverage Reranking → Ranked Results.

3. Memory Atoms

Memory atoms are regex-pattern-based extractions of typed memory primitives from user turns. This is a deterministic alternative to LLM-based fact extraction.

TYPE	CAPTURES	EXAMPLE TRIGGER
user_preference	Likes, dislikes, favorites	"I love / I hate / my go-to"
user_interest	Current activities, exploration	"I'm getting into / working on"
user_goal	Stated intentions and plans	"I want to / I'm trying to"
user_constraint	Hard limits and requirements	"I can't / my budget / my allergy"
decision	Committed choices	"I decided / we went with"
temporal_update	State changes over time	"I switched / I now / no longer"
open_loop	Pending actions, reminders	"remind me / todo / follow up"
entity_fact	User-owned context facts	"I have / my X is / I bought"

4. Episode Relevance Scorer

For vague advice queries, the relevant session often shares almost no vocabulary with the query. The Episode Relevance Scorer solves this by identifying the *kind* of memory signal a query needs.

The scorer is deterministic, interpretable, and runs at **0.4ms average latency**, eliminating the need for embedding cosine similarity.

5. The Query Router

No single retrieval mode is optimal for all query types. The router selects the best mode at near-zero cost:

- **Vague Advice:** → Episode Score
- **Specific Fact:** → BM25
- **Temporal + Fact:** → Fusion
- **Personalized Preference Recommendation:** → Preference Rerank
- **Multi-session Synthesis:** → Evidence-Coverage Rerank

6. Preference Reranker

The newest token-native route targets personalized recommendations where prior explicit taste should beat generic topical overlap. It extracts user turns, detects preference markers, normalizes tokens lightly, and scores overlap inside preference windows — without embeddings or LLM calls.

On the 499-case agent-memory benchmark, preference recommendation Recall@1 improved from **56.5%** to **85.5%**, beating OpenAI embed-3-small (**77.4%**) and Cohere embed-v3 (**83.9%**) on that behavior.

7. Structure-Aware File Ingestion

ContextFit now chooses semantic file boundaries before final token encoding. Markdown files chunk by heading and block boundaries with `heading_path` metadata; plain text chunks by paragraphs and separators; TMD ledgers chunk by source rows while preserving schema/front-matter context.

This is a structural improvement, not a new benchmark claim: token-native storage and retrieval remain unchanged, but file chunks now map more closely to human document meaning and are easier to cite. The latest 499-case auto-router run remained stable/slightly improved at **62.3% Recall@1 / 93.0% Recall@3 / 99.8% Recall@5**.

8. Benchmark Results

Evaluated on a 499-case domain-agnostic agent-memory benchmark across 8 behaviors and 26 domains, plus a 79-case hard episodic subset.

SYSTEM	R@1	MRR	QUERY LATENCY	COST
OpenAI (embed-3-small)	55.7%	0.745	~150ms	API
Mem0 (GPT-4o-mini)	54.4%	0.716	~341ms	API
ContextFit (Token-Native)	69.6%	0.824	0.4ms	Free

On the 79-case hard episodic subset, ContextFit outperforms OpenAI embeddings by 14 points in Recall@1 while running 375x faster. On the expanded 499-case benchmark, the auto router with preference and evidence-coverage reranking reaches 62.3% overall R@1 / 93.0% R@3, 85.5% preference R@1, and 82.1% multi-session synthesis R@1 at zero API cost.

On fresh LongMemEval-S reruns, pure token-native ContextFit with conversation-aware parent/child chunks reaches 95.1% Any@5, improving the previous token baseline through turn-aware session boundaries plus full-session parent context with no embeddings. A token-only companion-evidence coverage reranker then improves complete evidence coverage: overall All@5 rises from 77.9% to 80.4% while preserving the 95.1% Any@5 headline. Optional OpenAI fusion reaches 96.0% Any@5; fusion is an optional boost, not the core claim.

By question type, parent/child token-only closes the top-5 preference gap: single-session preference Any@5 reaches 83.3%, matching OpenAI fusion, while preference Any@1 is higher than OpenAI fusion (43.3% vs 40.0%). On multi-session questions, token-only Any@5 improves to 95.0%, within roughly 0.8 points of OpenAI fusion, and coverage reranking lifts multi-session All@5 from 55.4% to 65.3%. The remaining OpenAI advantage is now narrower: 65.3% token-only versus 72.7% with OpenAI fusion.

8. Impact & Implications

By removing the embedding dependency tier, ContextFit eliminates vendor lock-in, reduces network failure modes in the critical retrieval path, and eliminates per-query costs at agent scale.

9. Conclusion

The assumption that conversational memory must be converted to vectors is a convention inherited from document retrieval, not a technical necessity. Token-native retrieval is not just faster and cheaper—for the hardest memory problems, it is more accurate.

ContextFit Research • May 2026 • github.com/ContextFit/cf • X: [@cponsart](https://twitter.com/cponsart)