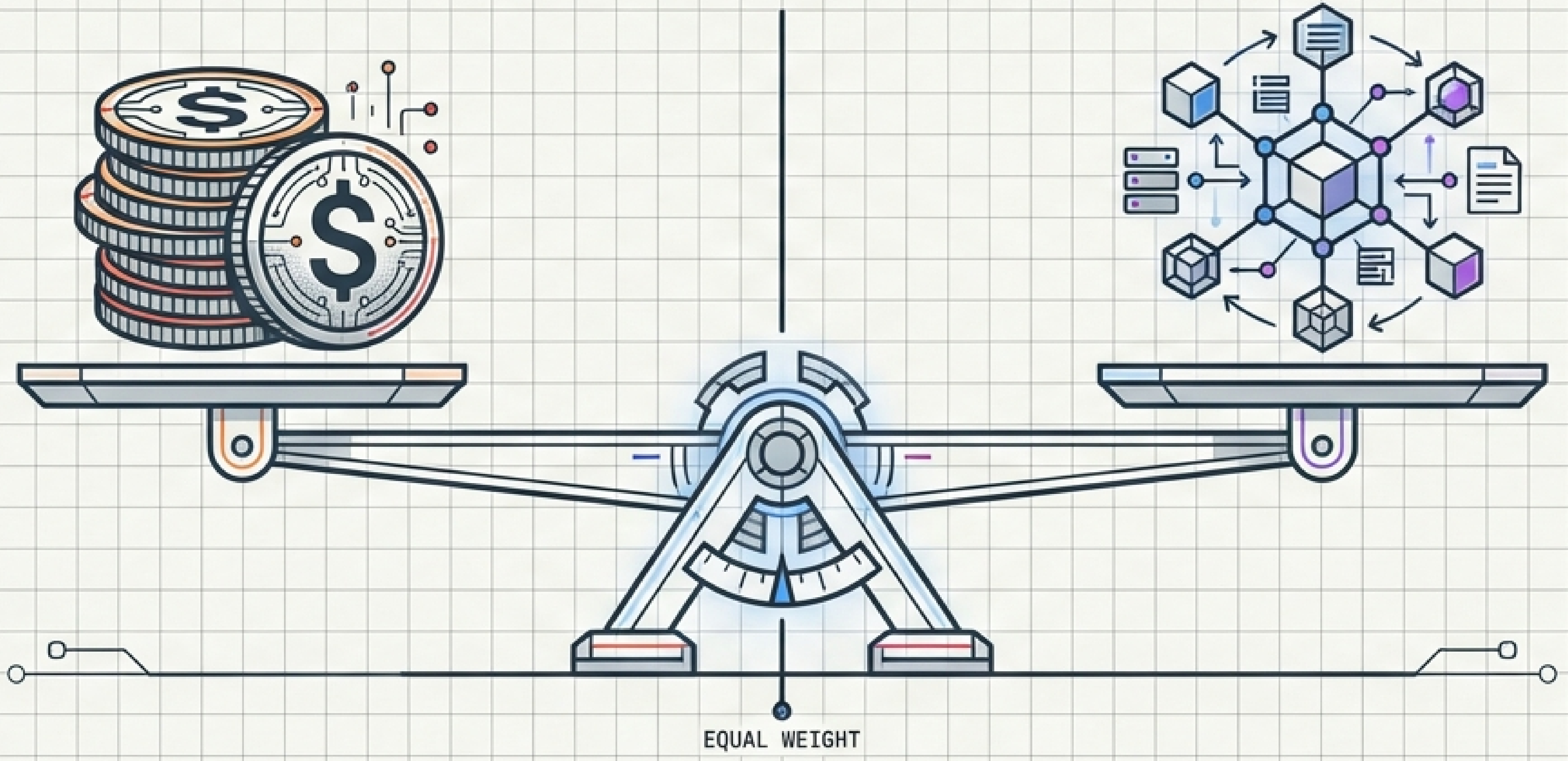
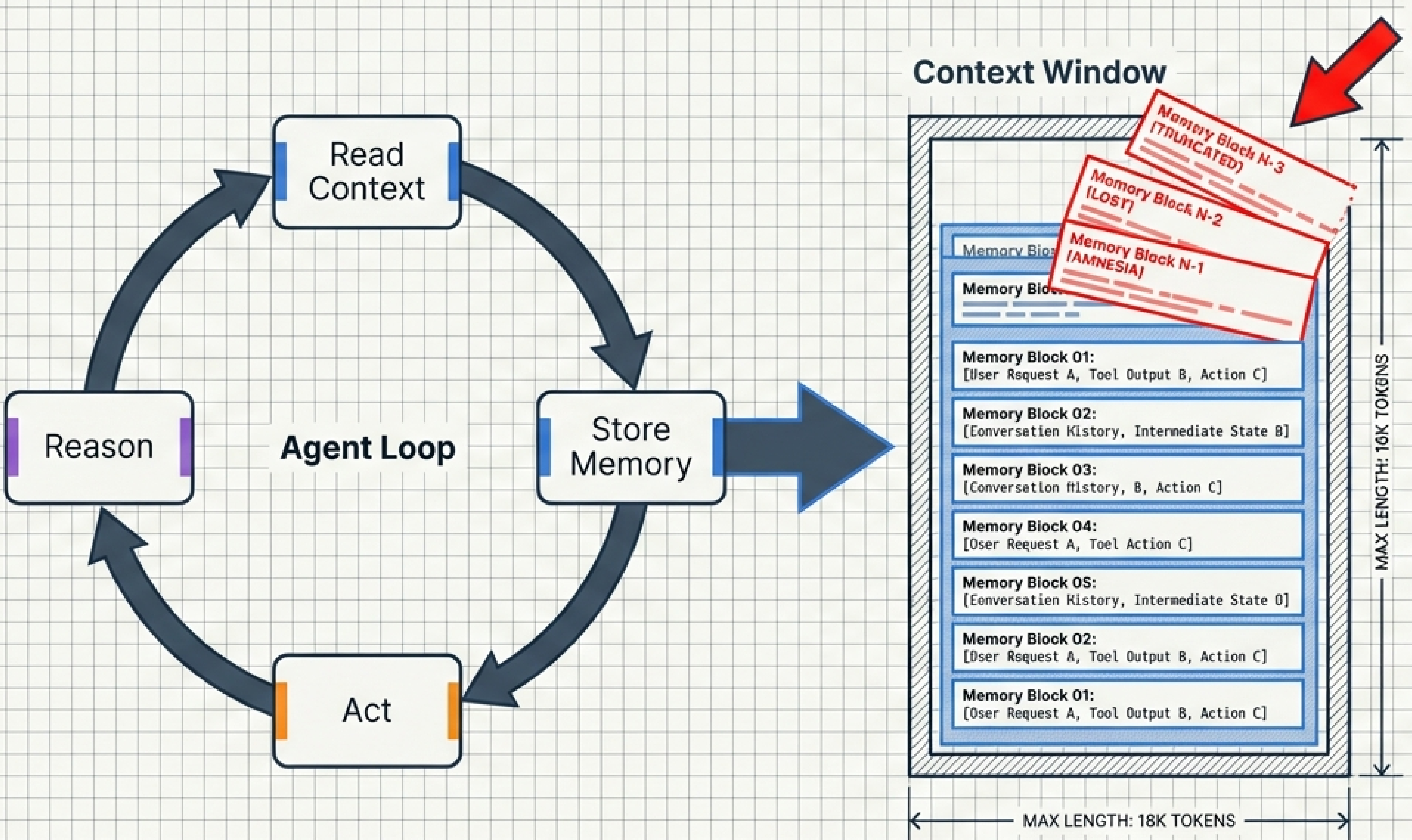
Optimizing Agent Memory: An Information-Theoretic Approach

Balancing Token Budget and Information Retention with ITAMC



Based on research by Anonymous Author(s) | Conference ’17

The Infinite Loop vs. The Finite Window



- **The Agent's Dilemma:** Agents accumulate history indefinitely, but context windows are fixed.
- **The Current Reality:** Overflow leads to crude truncation (forgetting) or blind compression (loss of detail).
- **The Consequence:** Loss of task-critical information required for long-horizon reasoning.

Memory as a Rate-Distortion Optimization Problem

Objective: Maximize Information Retention

$$\max \sum (w_i \cdot \rho_i)$$

Subject to Constraint: Token Budget

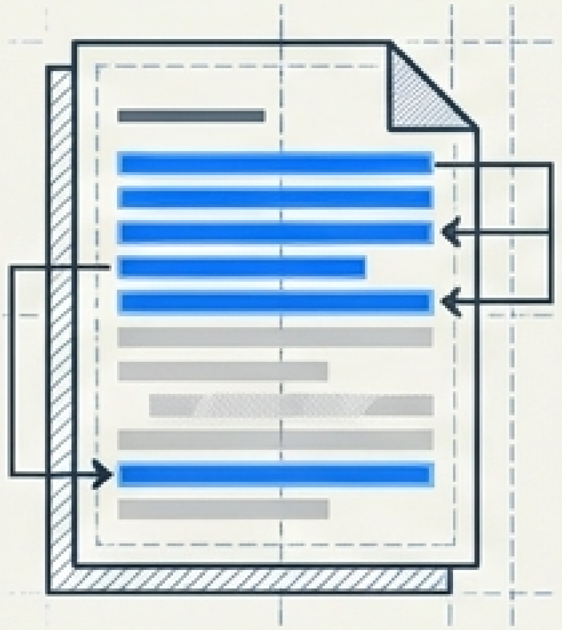
$$\sum |C(m_i)| \leq B$$

ρ_i (Retention): Fraction of salient facts preserved.		**w_i (Weight):** Importance of the memory episode.	
B (Budget): Global token limit (The Economic Constraint).		**r_i (Ratio):** Compression level applied.	

Core Insight: We are not just shortening text; we are maximizing fact preservation within a strict economic budget.

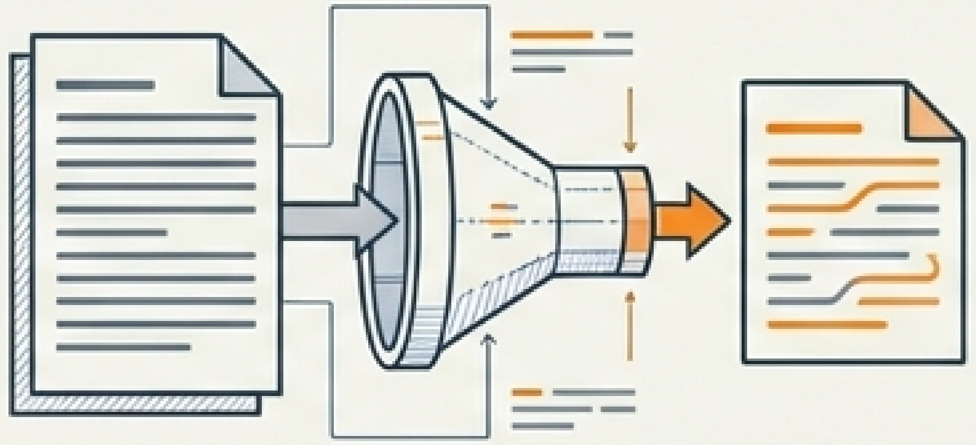
Three Operators for Memory Compression

Extractive



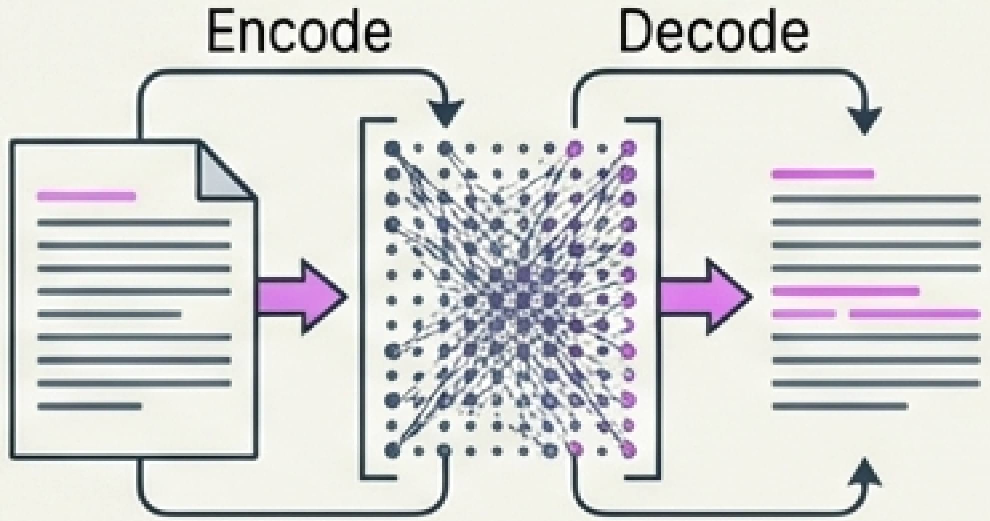
- **Mechanism:** Selects top-k sentences based on density.
- **Behavior:** Binary retention. Facts are either kept or lost entirely.
- **Analogue:** LexRank / TextRank.

Abstractive



- **Mechanism:** LLM-based rewriting and summarization.
- **Behavior:** Smooth degradation. Facts retained probabilistically.
- **Analogue:** GPT-4 Summarizer.

Latent



- **Mechanism:** Dense vector embeddings decoded to text.
- **Behavior:** Graceful degradation. Captures broad semantics.
- **Analogue:** Embedding Storage.

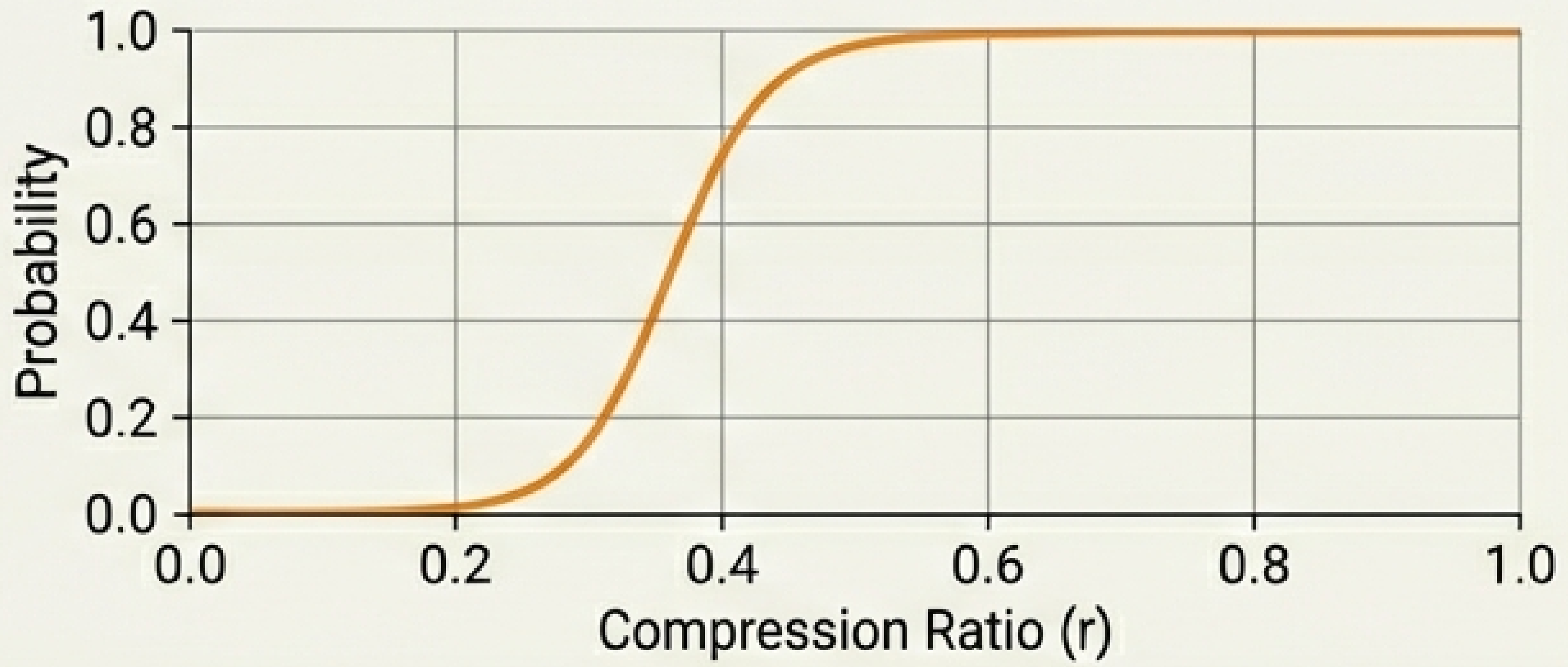
Modeling Information Loss

Defining the probability of retention (P) given compression ratio (r).

Abstractive Model

$$P(\text{retain}) = \text{sigmoid}(k \cdot (r_i - \tau))$$

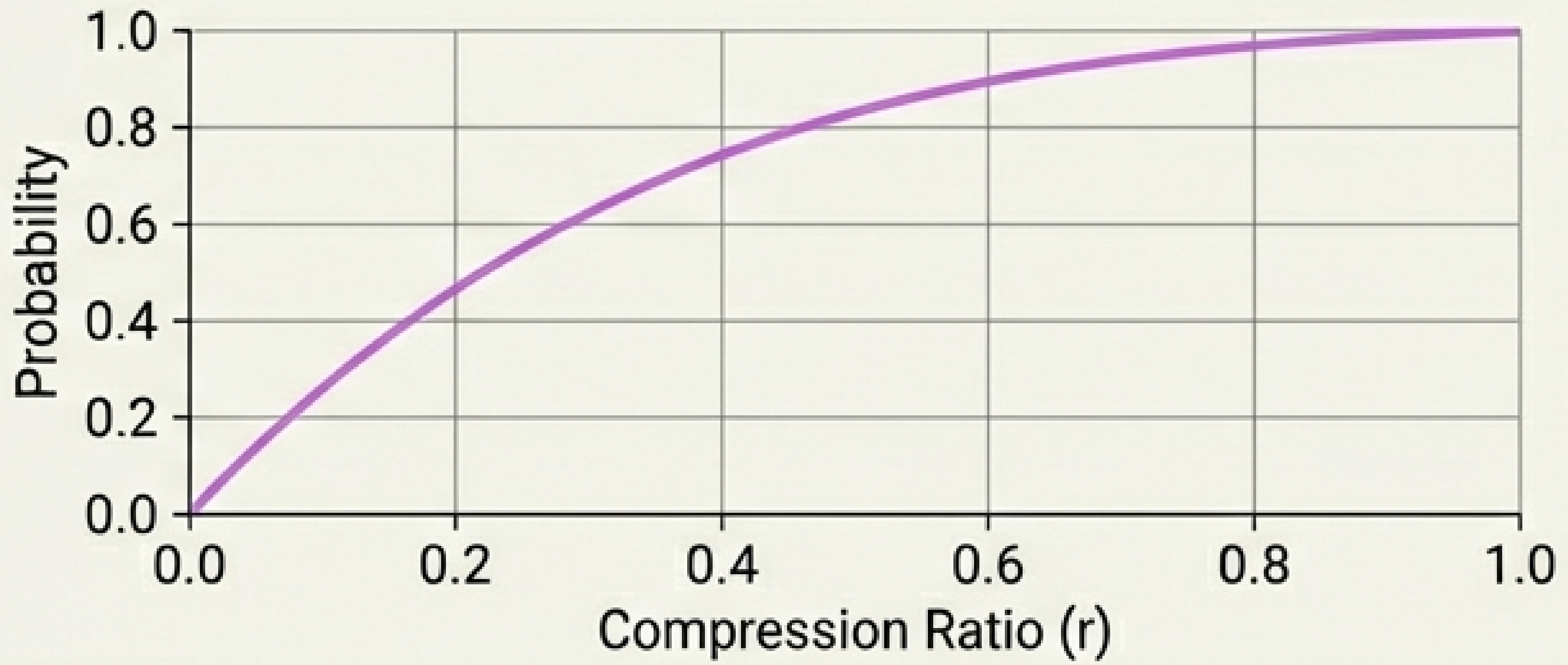
Models the S-curve loss of LLM summarizers.



Latent Model

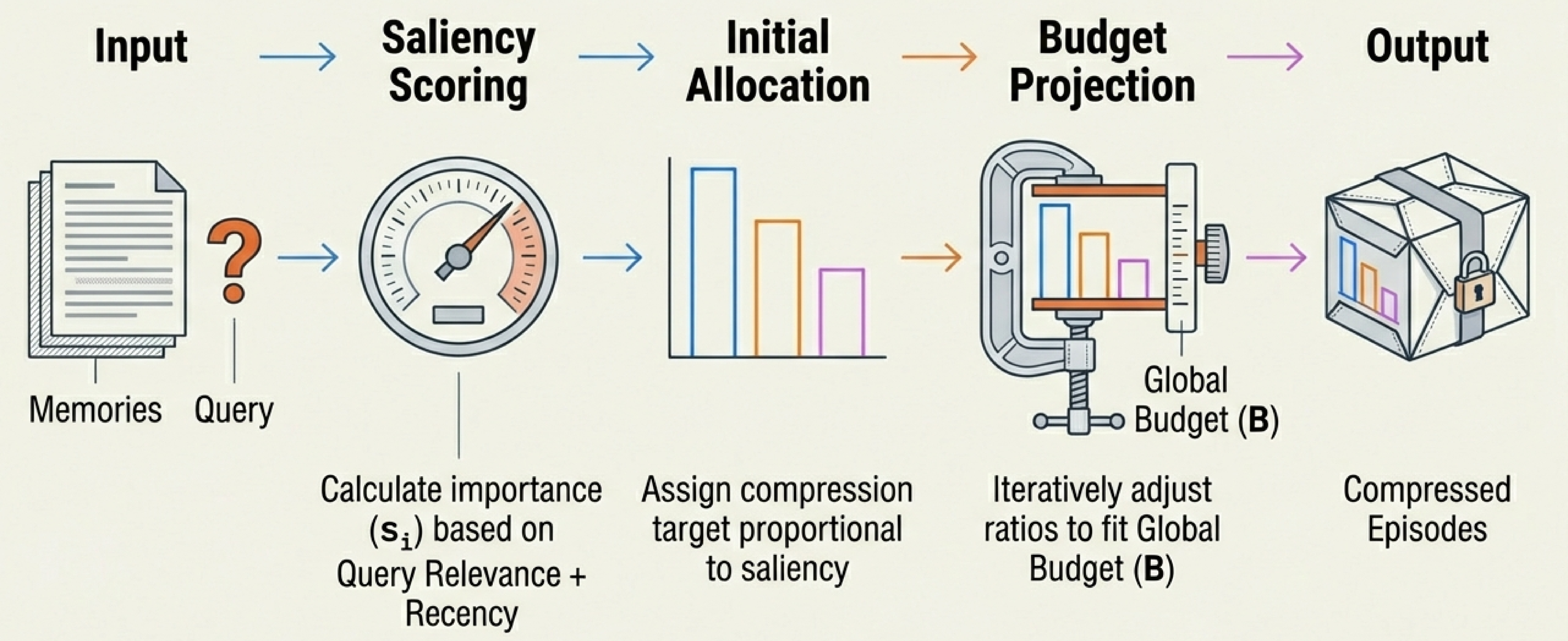
$$P(\text{retain}) \sim \text{Beta}(r^{0.6} \cdot \kappa, (1 - r^{0.6}) \cdot \kappa)$$

Sub-linear exponent models graceful degradation of embeddings.



The ITAMC Controller

Saliency-Guided Adaptive Allocation



Computing Saliency

$$s_i = 0.6 \cdot \text{LexicalOverlap} + 0.4 \cdot \text{TimeDecay}$$

Relevance: $|\text{tokens}(q) \cap \text{tokens}(m)| \,/\, |\text{tokens}(q)|$

Measures how much the memory overlaps with the current task query.

Recency: $e^{(-\lambda(T - t))}$

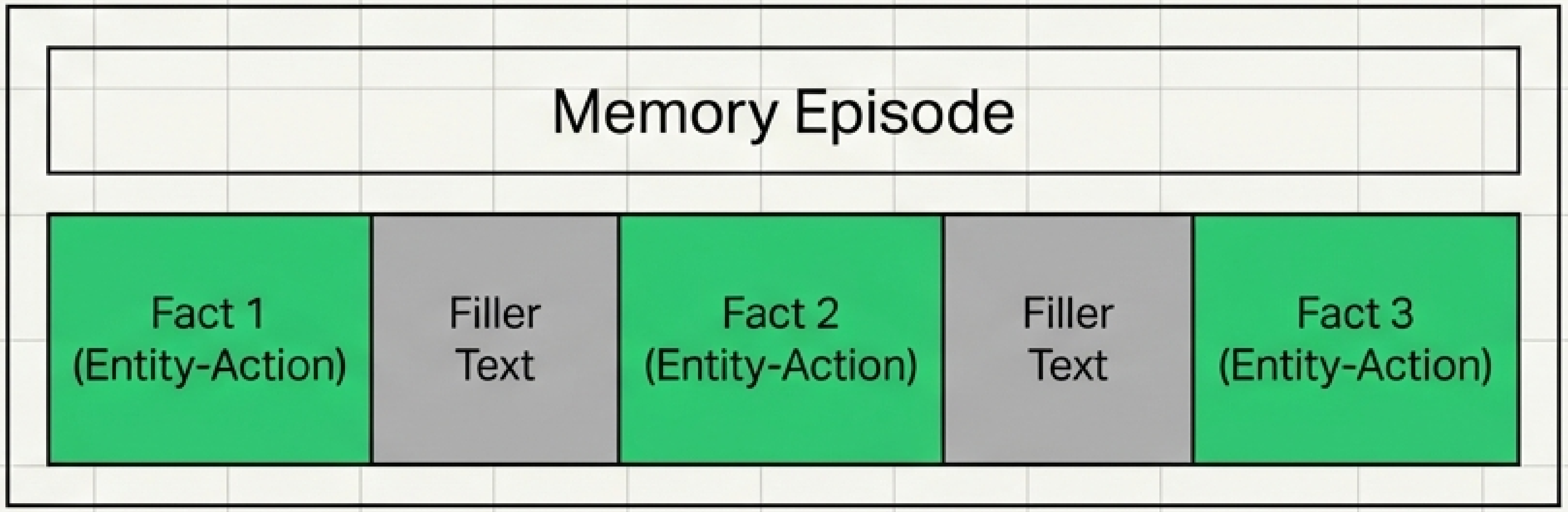
Favors recent memories (λ=0.02) to simulate human short-term bias.

Note: Combines Search (Relevance) with Chronology (Recency).

The Experimental Rig

The Challenge:
Natural language traces are ambiguous. It is hard to prove strictly if an agent “forgot” a fact.

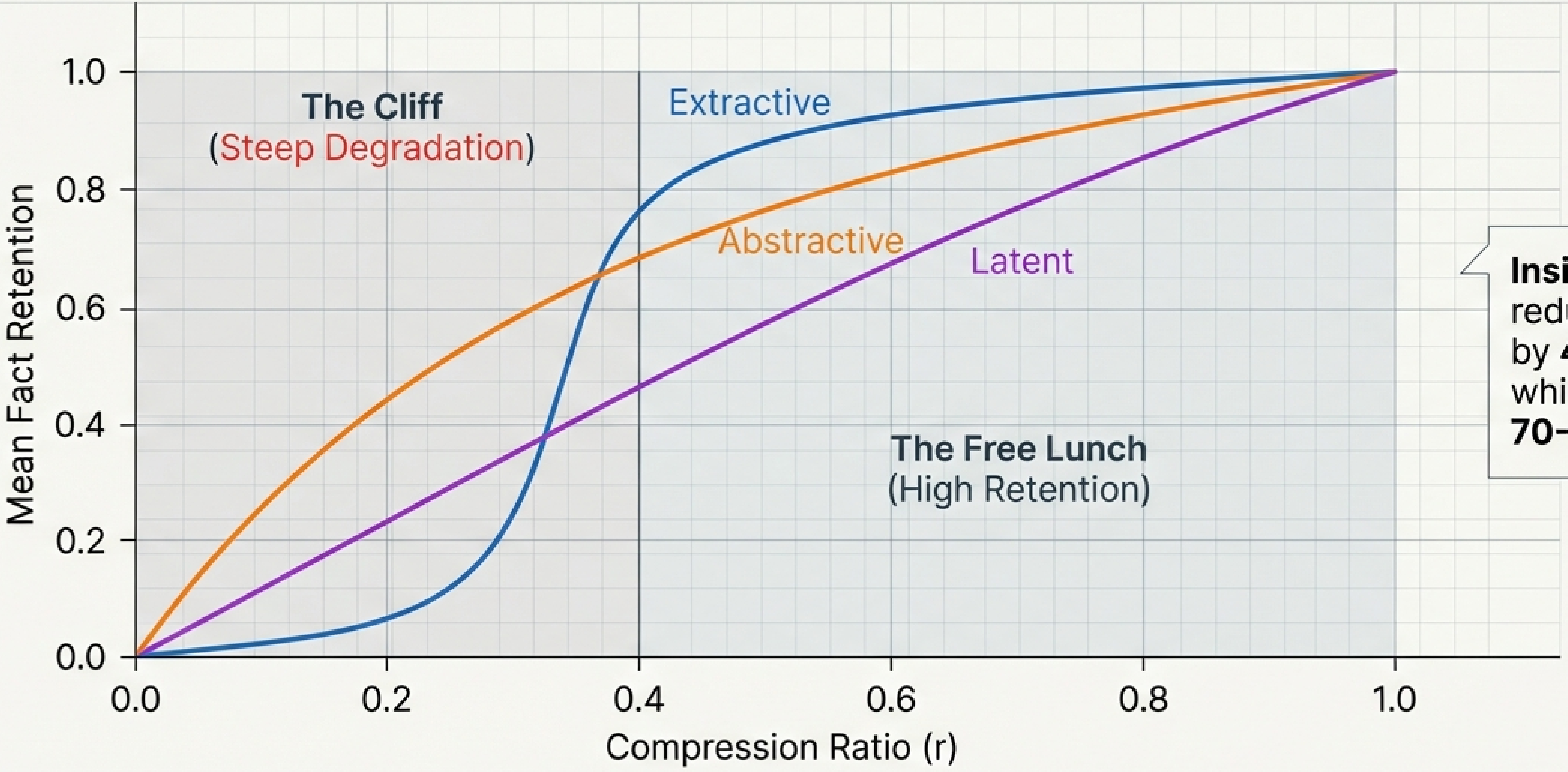
The Solution:
Synthetic Data Evaluation.




```
JetBrains Mono
Total Volume: 100 Episodes,
300 Ground-Truth Facts.

JetBrains Mono
Metric: Exact Retention Ratio (Are the
green blocks recoverable?)
```

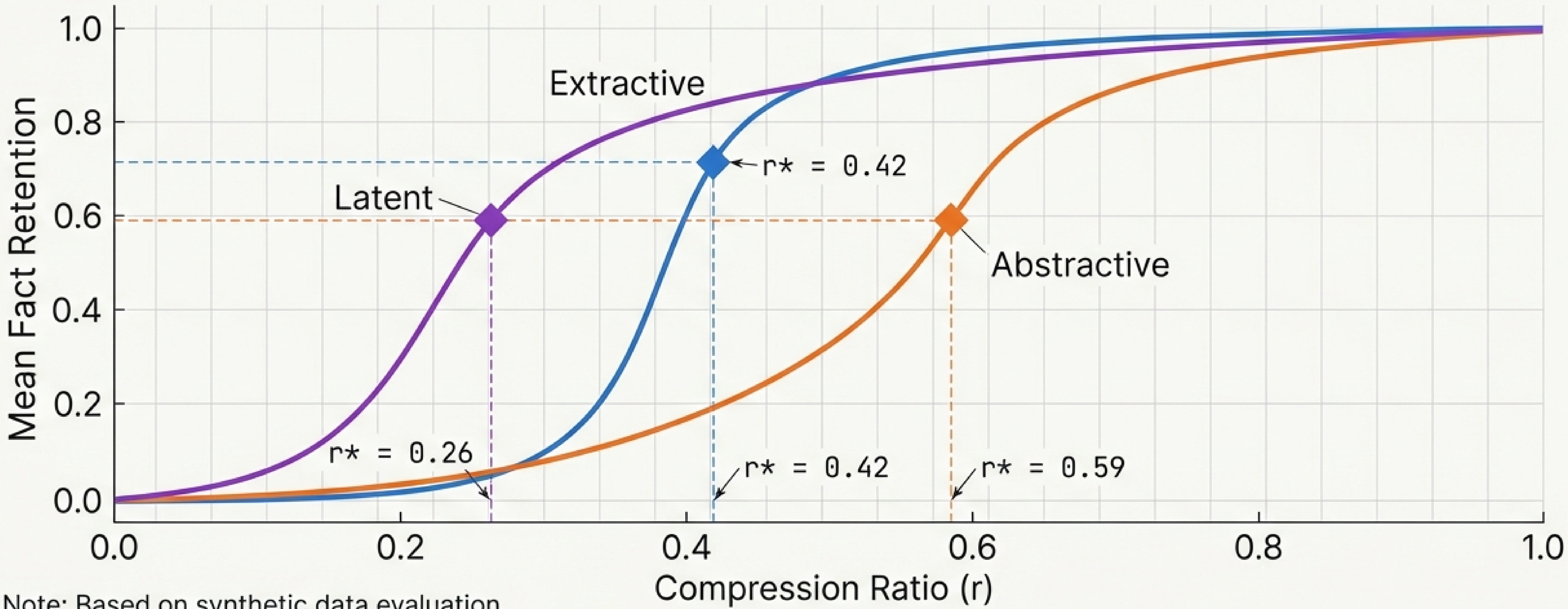
Law 1: The Concave Frontier



Insight: You can reduce tokens by **40-60%** while retaining **70-87%** of facts.

Note: Based on synthetic data evaluation.

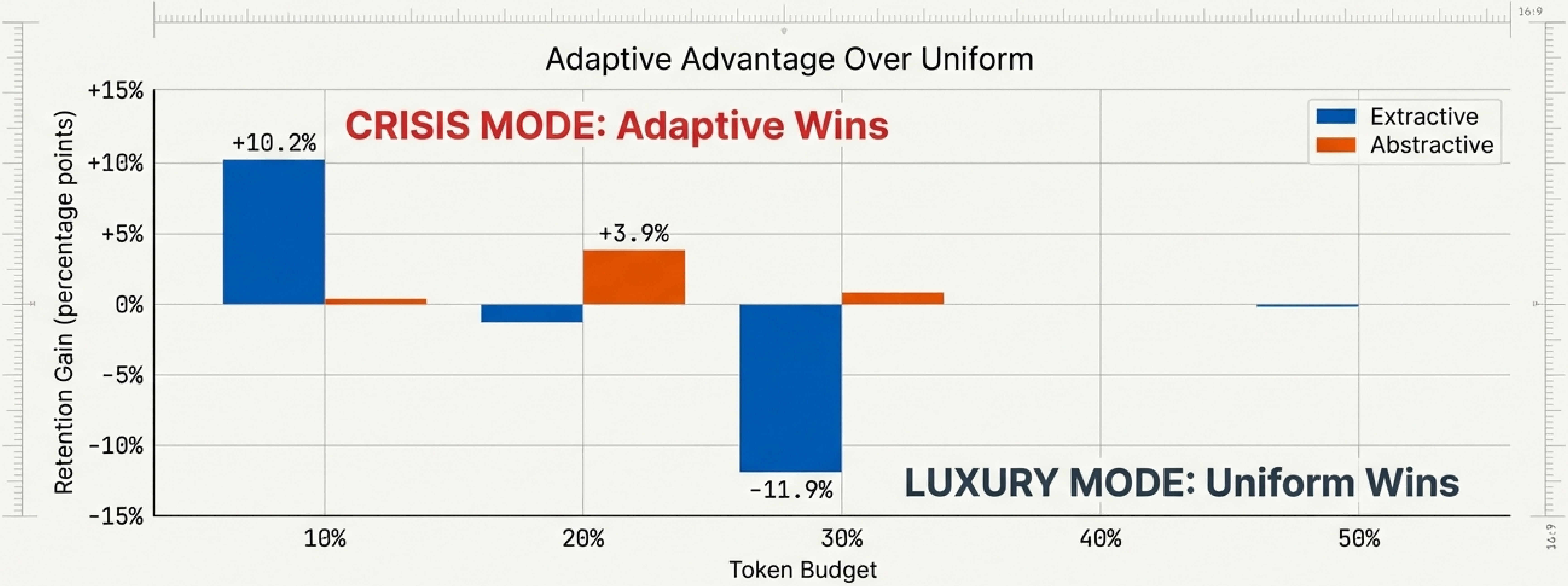
Law 2: The Knee-Points (Optimal Ratios)



Note: Based on synthetic data evaluation.

Takeaway: Optimal settings are operator-dependent. Latent is most efficient; Abstractive needs more room.

Law 3: Adaptive is for Crisis Mode

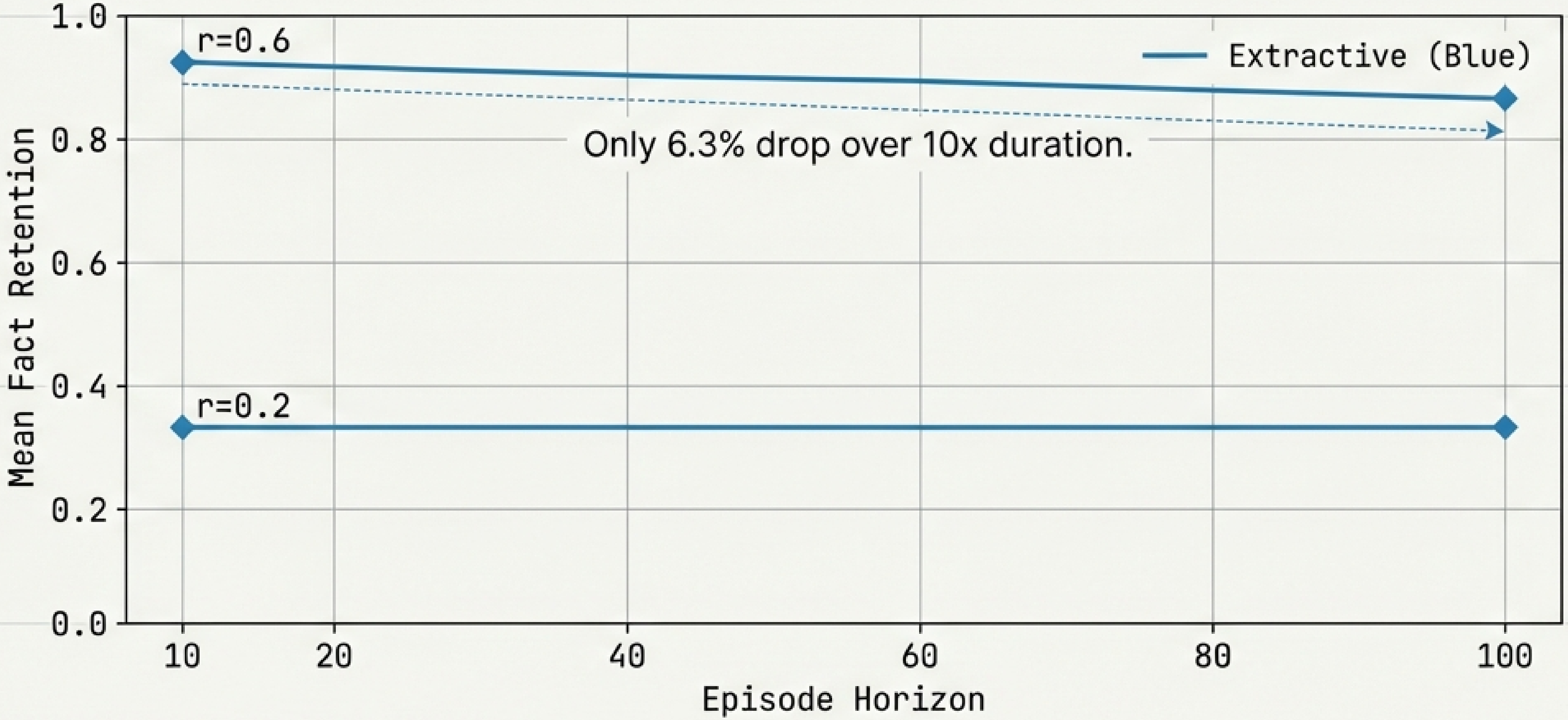


Adaptive allocation is critical when resources are scarce.
At high budgets, uniform compression is sufficient.

Note: Based on synthetic data evaluation. Refer to the data pattern in Figure 4 and Table 2.

Law 4: Stability Over Horizons

Compression errors do not compound catastrophically. If you compress well once, the memory stays valid for the long haul.



Note: Based on synthetic data evaluation. Refer to Figure 5.

16:9

Law 5: Saliency vs. Compressibility

	Low Saliency	Medium Saliency	High Saliency
Extractive	0.74	0.72	0.70
Abstractive	0.61	0.63	0.64
Latent	0.63	0.58	0.57

Low Saliency Medium Saliency High Saliency

- **Counter-Intuitive:** High importance facts are not “harder” to compress.
- **Insight:** Saliency dictates *allocation* (budget), not *compressibility* (difficulty).
- **Takeaway:** Operator choice matters more than episode content.

Note: Reference Figure 6 for the data values.

The Engineer's Cheat Sheet

Scenario A: High Budget (>50%)

Use Uniform Abstractive.

Ratio r ≈ 0.6. Uniform retention is high; adaptive overhead isn't worth it.

Scenario B: Survival Mode (<20%)

Use Adaptive Extractive.

Ratio r ≈ 0.42. You need the sharp efficiency of extraction to save critical facts.

Scenario C: Long-Term Storage

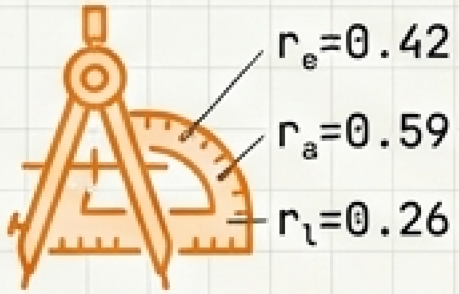
Use Latent Compression.

Ratio r ≈ 0.26. Lowest storage cost with graceful degradation for retrieval.

Summary of Findings

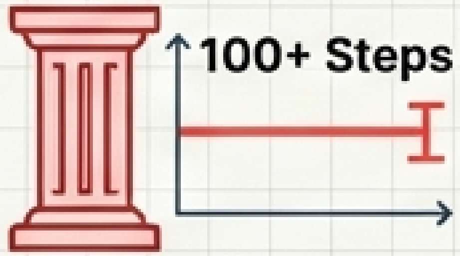
1 **Concavity:** The first 40% of token reduction is ‘free’ (high retention). The curve is concave.

2 **Specificity:** Optimal ratios are fixed constants.
Extractive=0.42, Abstractive=0.59, Latent=0.26.



3 **Adaptation:** Use Saliency-Guided Adaptive allocation **ONLY** for extreme constraints (Crisis Mode).

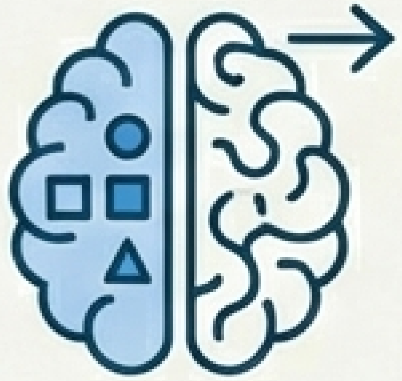
4 **Stability:** Compression errors do not compound catastrophically over 100+ steps.



Note: Aggregated results from experimental trials.

Limitations & Future Directions

Synthetic vs. Natural



Synthetic vs. Natural

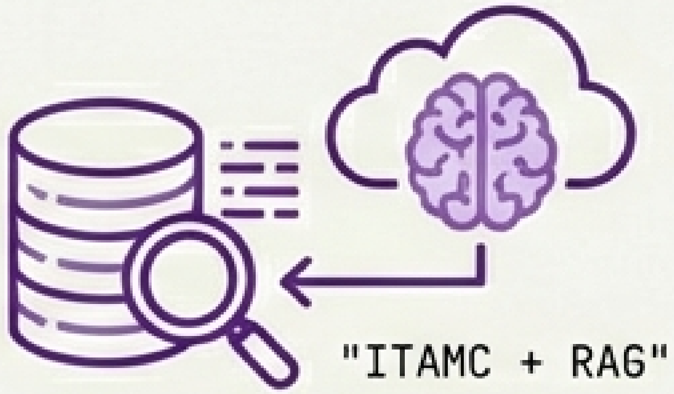
Study used synthetic data for precision. Future work for precision. Future work must validate with noisy, natural language traces.

Real-Time Saliency

Real-Time Saliency

Currently, saliency is static. Future systems need ‘Online Saliency’ that shifts as agent goals change.

RAG Integration



RAG Integration

ITAMC acts as ‘soft retrieval’. Integrating this hard retrieval (RAG) is the next logical step.

References & Resources

Primary Source:
Information-Theoretic Adaptive Memory Compression for LLM-Based Agents (Anonymous Author(s), Conference ‘17)

- Berger (1971): Rate Distortion Theory
- Yang et al. (2026): Survey on Efficient Agents
- MemGPT / Reflexion: Memory Architectures

Code and simulation framework available for reproducibility.