PAST KNOWLEDGE PEOPLE

ISSN: 2834-4928

Research Article

Journal of Electrical Electronics Engineering

Sparse Context Augmentation (SCA): A Dual-Output Indexing Architecture for Mitigation of $O(N^2)$ Scaling and Auditable LLM Deployment

Chaiya Tantisukarom*

Independent Researcher, Thailand

***Corresponding Author**
Chaiya Tantisukarom, Independent Researcher, Thailand.

Submitted: 2026, May 01; **Accepted:** 2026, Jun 22; **Published:** 2026, Jul 03

Citation: Thantisukarom, C. (2026). Sparse Context Augmentation (SCA): A Dual-Output Indexing Architecture for Mitigation of O(N^2) Scaling and Auditable LLM Deployment. *J Electrical Electron Eng, 5*(4),01-05.

Abstract

*Current Large Language Model (LLM) architectures face a fundamental **quadratic scaling bottleneck** ($O(N^2)$) inherent to the self-attention mechanism, leading to prohibitive costs and latency for long-context tasks. This paper proposes a framework, **Sparse Context Augmentation (SCA)**, which implements a **DualOutput Indexing Architecture** utilizing a **Just-in-Time Retrieval-Augmented Generation (JIT-RAG)** strategy. This approach, intended as a strategic, creator-side mitigation, decouples long-context processing from the expensive internal context window by migrating history to external, low-cost storage. Crucially, the model is architected to simultaneously generate the standard user response (**var_2**) and a compact, fixed-maximum-size **semantic index** (**var_1**). This index (var1) acts as a highly condensed, low-fluidity key message log, providing **System State Transparency** and a predictable cost floor. This strategy maintains conversational coherence with a complexity of **$O(k^2)$** per turn, where k is a fixed, small maximum context size, effectively decoupling the cost from the total history length T. The framework provides a **mutual value proposition**, transforming the LLM's auditable state logs into user-facing transparency features, resulting in massive cost savings for the creator and unparalleled reliability for the user.*

1. Introduction and Architectural Mandate

The development of Large Language Models (LLMs) represents a significant technological leap, but progress is constrained by architectural limits, particularly the cost and computational complexity associated with scaling context windows. This analysis posits that continued adherence to the monolithic, dense-attention paradigm (where cost scales quadratically, $O(N^2)$) is economically and technically unsustainable. This work introduces an alternative paradigm rooted in the principle of **trust reversal**: shifting focus from architectural brute force to intelligent, sparse, and externalized memory management, making the system state **auditable** rather than opaque.

The primary goal of this paper is to establish the **meta-structural and economic necessity** of the SCA framework. The technical implementation details—such as designing the Index Head as a constrained summarization task or optimizing the low-latency key-value store—are considered **standard engineering viability challenges** based on mature NLP and data engineering primitives (e.g., standard RAG components, existing low-latency storage). Therefore, this article focuses solely on the necessary architectural and economic shift, demonstrating how the system state variables, when correctly managed, fundamentally change the complexity relative to the total conversation length T from O(T^2) to a near-constant **O(k^2)**.

1.1. The $O(N^2)$ Quadratic Scaling Bottleneck and Externalized Memory

The prevailing Transformer architecture relies on the **Self -Attention** mechanism, which necessitates computational complexity proportional to the square of the input sequence length, N (Vaswani et al., 2017). While mathematically elegant for correlating token dependencies, this $O(N^2)$ relationship imposes severe economic and performance limitations. The core computational complexity remains, defining the eventual limits of the hardware. The proposed solution is a strategic, temporary measure designed to mitigate the immediate, exponential cost growth imposed on end-users by bypassing this constraint entirely

through external memory management.

- **Cost Ceiling:** Increasing the context length N by a small factor requires a disproportionately large increase in computational resources (memory and time).
- **Computational Redundancy:** The dense attention calculation is often overkill, dedicating expensive computation to token interactions that lack material significance in the final output.
- **The Wall:** Attempting to surpass these limits solely through architectural modifications is akin to" hitting the wall." A systemic shift is required to bypass this constraint.

1.2. Sparse Context Augmentation (SCA): Externalized Memory

The solution lies in moving from **dense attention** over a long internal context to **sparse, targeted retrieval** augmented with a tightly managed working context.

- **Decoupling Context:** Instead of incurring the $\mathbf{O(N^2)}$ cost internally to hold conversational history, we **move the history cost to external low-cost storage.**
- **Capacity and Cost Efficiency:** Externalized memory utilizes simple, JSON formatted text logs (low-latency key-value storage) offering orders of magnitude greater capacity and lower cost than active context memory.
- **Retrieval-Augmented Generation (RAG):** This approach leverages RAG principles (Lewis et al., 2020), replacing expensive general-purpose attention over an entire corpus with a cheap, efficient **O(1)** semantic query against the memory bank.

This is enabled by the LLM's self-generated, high-fidelity index ($\mathbf{var_1}$). The subsequent computation remains $\mathbf{O(k^2)}$ relative to the fixed context window size k.

2. The Dual-Output Index Engine: System State Transparency

The **Sparse Context Augmentation (SCA)** framework is implemented via a **Compact Context Injection** mechanism powered by a **Dual-Output Index Engine**, Figure:(1). This implementation is an **engineering mandate** for LLM creators using mature, established components.

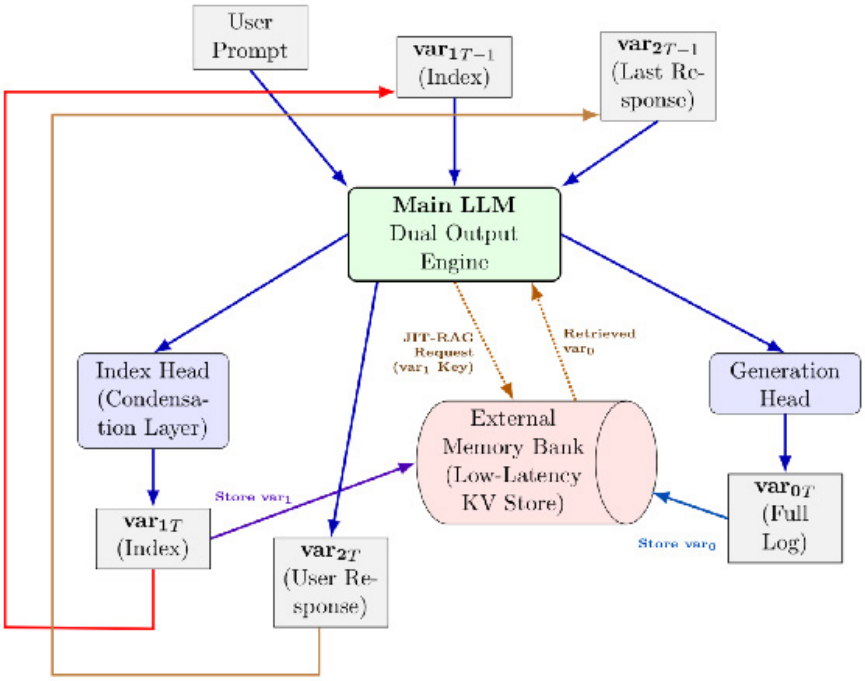

Figure 1: Conceptual Architecture of the Sparse Context Augmentation (SCA) Dual-Output Index Engine.

2.1. Concurrent Generation and State Variables

The core innovation is the simultaneous, concurrent preparation of two outputs for every turn by the main LLM, ensuring semantic consistency and maximizing efficiency. This establishes the **immutable, auditable system state.** All variables are stored in a simple JSON format.

We formally define the three state variables per turn:

- $\mathbf{var_0}$: The **Immutable External Full Text (Raw Log)**. Produced by the **Generation Head**. It is the verbose, complete output (output_0), serving as the auditable source of truth, stored in the external memory bank.

$$\mathbf{var_0} = \{\text{turn} : t, \text{prompt} : \text{output}_0(\text{raw log})\} \quad (1)$$

- $\mathbf{var_1}$: The **Immutable Compact Index (Retrieval Set Point)**. This is a **fixed-maximum-size semantic index** produced by the **Index Head (Condensation Layer)**. It is a high-condensed, low-fluidity key message log (output_1), where

length($output_1$) is considerably shorter than length($output_0$), ensuring a compression ratio of at least 2× minimum.

$\mathbf{var}_1$ = {turn : t,prompt : $output_1$(key message log)} (2)

- **$\mathbf{var}_2$:** The **Immediate Final User Response**. The current user-facing output, which is structurally identical to the raw log ($\mathbf{var}_0$) for that turn.

  $\mathbf{var}_2$ = {turn : current turn,prompt : $output_0$} (3)

2.1.1. The Condensation Layer: Constrained Semantic Summarization

The Condensation Layer (Index Head) is a constrained transformation mechanism, functioning as a fine-tuned **Semantic Summarization Layer**. It ensures the compact index ($\mathbf{var}_1$) achieves high information density and serves as a **self-indexed grounding truth**. This layer must be trained to perform **critical state extraction** within a fixed token budget ($N_{max}(\mathbf{var}_1)$), focusing only on facts and core instructions essential for long-term coherence. The term **low-fluidity** implies training that penalizes unnecessary changes in $\mathbf{var}_1$ across turns, maintaining a stable representation of the core conversational state. This token efficiency is paramount for reducing context window consumption.

2.2. Context Injection and Just-in-Time Retrieval (JIT-RAG)

- **Compact Injection (Optimized Fidelity):** The input for Turn T is defined as,

  $\mathbf{Input}_T$ = [CURRENT PROMPT] + $\mathbf{var}_{1T-1}$ + $\mathbf{var}_{2T-1}$. (4)

- This strategy achieves optimum utilization by including the compact semantic index ($\mathbf{var}_1$) for long-term state maintenance and the full text of the previous turn's response ($\mathbf{var}_2$) for immediate, high-fidelity coherence. The maximum size of N = length ($\mathbf{Input}_T$) is bounded by a constant, k, independent of the total conversation length T.

- **JIT-RAG:** The LLM is equipped with a **Retrieval Gate** mechanism (a mature component) which triggers focused retrieval if the compact context is insufficient. The prompt and $\mathbf{var}_1$ are used as a key reference for the low-latency storage to retrieve necessary full text ($\mathbf{var}_0$) from a targeted history turn, inserting it temporarily into the working context. The infrastructure requires high-throughput, **sub-50ms latency** indexing to ensure retrieval time does not negate the computational savings.

3. Visualization of Economic Impact: $O(T^2)$ to $O(k^2)$ Decoupling

The central benefit of the SCA framework is the transformation of the quadratic cost dependency into a near-constant linear cost per turn. This shift is secured by making the per-turn computational cost $O(k^2)$, where k is the fixed, bounded maximum active context window. **Relative to the total conversation history T, the complexity is effectively $O(k^2)$ per turn, independent of T.**

3.1. Scenario A: Long-Context Use (Technical/Verbose)

Assuming n = 1000 tokens per turn (Max $\mathbf{var}_2$) and a fixed index size of n_{index} = 50 tokens (Max $\mathbf{var}_1$). The active context $k \approx 1050$. The cost is $C_{Std} \propto (T \cdot 1000)^2$ vs. $C_{SCA} \propto 1050^2$, Figure:(2).

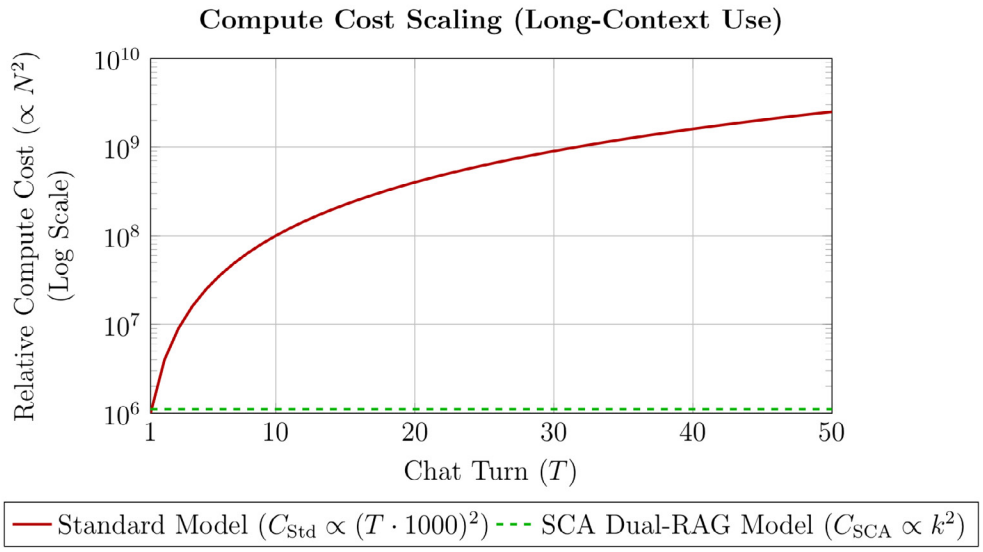

Figure 2: The exponential cost divergence for long, verbose chat sessions.

3.2. Scenario B: General User Context (Standard Chat)

Assuming a more realistic n = 150 tokens per turn (Max var2) and n_{index} = 50 tokens (Max $\mathbf{var}_1$). The active context $k \approx 200$. The cost is $C_{Std} \propto (T \cdot 150)^2$ vs. $C_{SCA} \propto 2002$, Figure:(3).

4. Mutual Value Proposition: Creator Economics and User Transparency

The SCA framework provides a **double-fold** benefit, ensuring that the economic savings for the creator directly translate into transparency and reliability benefits for the user, **Figure 2: Compute Cost Scaling (General User Context)** fulfilling the **Trust Reversal** mandate. The architecture introduces **System State Transparency** as a core feature.

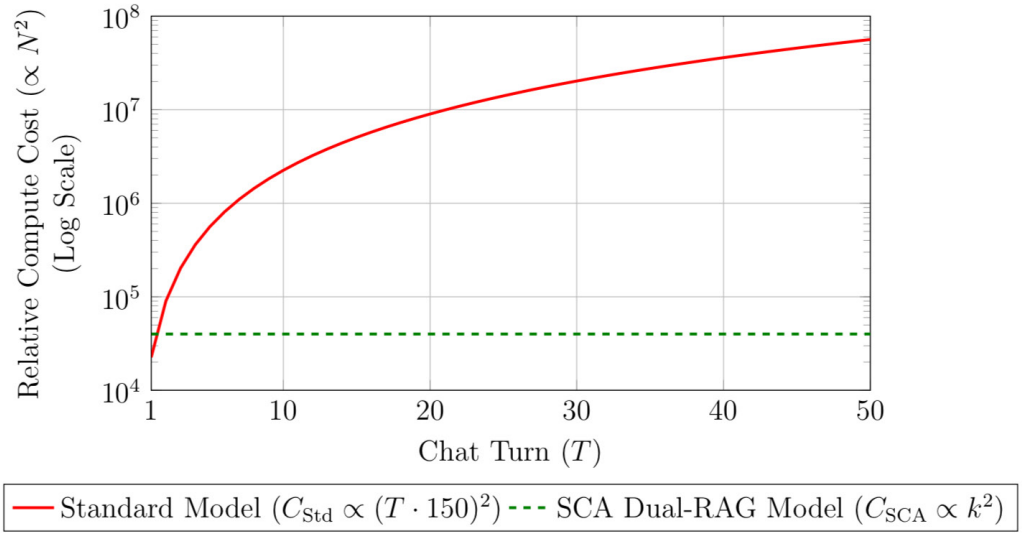

Figure 3: The quadratic cost divergence remains highly significant even for typical, shorter chat sessions. At turn 20, the Standard Model requires 225 times more compute than the SCA Model

4.1. Creator Economics: Auditable Savings

The economic benefits are derived from the elimination of quadratic scaling relative to history length T:

- $O(T^2)$ **to** $O(k^2)$ **Savings:** As visualized, the quadratic cost savings are massive (e.g., 225× factor at $T = 20$ in Figure 3). This capital, previously wasted on redundant computation, is reclaimed.
- **Auditable System State:** The existence of immutable logs ($\mathbf{var_0}$, $\mathbf{var_1}$) means that any failure, including confident hallucination, is transformed into an **auditable event**. This allows for precise post-mortem analysis and targeted model rectification, making the training loop more efficient by defining the system's exact grounding truth at the point of failure.

4.2. User Transparency: Enabling Self-Audit

The internal audit logs are exposed as user-facing features, empowering the user and establishing confidence:

- **Memory Dashboard:** A scrollable, on-screen interface that allows the user to inspect the system's memory ($\mathbf{var_0}$ and $\mathbf{var_1}$ history). This directly addresses the limitations of human memory by showing the immutable, complete context the LLM is leveraging.
- **Copyable References:** Providing a **Copy Icon** on each turn allows the user to instantly copy the immutable logs ($\mathbf{var_0}$ or $\mathbf{var_1}$). This enables users to self-audit, verify references, and provide indisputable documentation for external review, fulfilling compliance and accountability needs.

5. Conclusion and Recommendation

The Sparse Context Augmentation framework, implemented via a Dual-Output Indexing architecture, provides a robust, deployable solution to circumvent the immediate commercial and performance pressures imposed by $O(N^2)$ scaling. This approach is recommended as a mandatory, creator-side engineering solution to manage current resource constraints. While the $O(N^2)$ complexity remains the ultimate" ghost" in the machine, this framework offers LLM creators a critical window of 2−3 years to innovate without the immediate pressure of exponential scaling costs, while delivering significantly enhanced auditability and reliability to end-users via transparent, auditable features [1-3]. The shift from $O(T^2)$ to $O(k^2)$ scaling relative to conversation length T is an economic and architectural imperative.

References

1. Dao, T., Fu, D., Ermon, S., Rudra, A., & Ré, C. (2022). Flashattention: Fast and memory-efficient exact attention with io-awareness. Advances in neural information processing systems, 35, 16344-16359.
2. Lewis, P., Perez, E., Piktus, A., Petroni, F., Karpukhin, V., Goyal, N., Groß, S., Ma, H., Koˇcisky´, T., T., Gullapalli, P., Nogueira, L., and Kiela, D. (2020). RetrievalAugmented Generation for Knowledge-Intensive NLP Tasks. *Advances in Neural Information Processing Systems (NeurIPS) 33*.
3. Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, L., Jones, L., Gomez, A. N., Kaiser, L., and Polosukhin, I. (2017). Attention Is All You Need. *Advances in Neural Information Processing Systems (NeurIPS) 30*.

Appendix

Executive Summary for Leadership

Shifting from" I Don't Know" to" I Told You So" — A Strategy for Accountable AI

The current trajectory of Large Language Model (LLM) scaling is approaching a hard-economic wall due to the $O(N^2)$ **Quadratic Scaling Bottleneck**. Continuing to manage context through brute force is not only financially unsustainable but creates significant enterprise risk due to" black box" uncontrollability. The **Sparse Context Augmentation (SCA)** framework offers a strategic pivot:

- **The" I Told You So" Standard (Auditable State):** Traditional LLMs operate in a state of" I **don't know"**—when they hallucinate or fail, the cause is often untraceable. SCA introduces immutable state variables ($\mathbf{var}_0$ for text, $\mathbf{var}_1$ for compact context) that allow the system to definitively state," I **told you so."** Every output is tied to a specific, auditable set point. This transform hallucinations from mysterious errors into **auditable events**, critical for enterprise compliance and liability management.
- **Strategic ROI (Quadratic Savings):** By decoupled storage from compute, SCA reclaims wasted resources. In a standard user scenario (20 turns), the SCA architecture operates with a **225x cost efficiency advantage** over traditional models. This creates" idle capital" that can be reinvested into product features rather than redundant computation.
- **Recommendation:** Implement the **Dual-Output Index Engine** immediately. This is not merely an optimization; it is a fundamental architecture shift that secures a 2-3-year competitive advantage in scaling, reliability, and cost-structure, by fundamentally changing the complexity from $O(T^2)$ to $O(k^2)$ relative to total chat history . The reliance on mature, existing RAG and summarization primitives ensures rapid engineering viability.