Scratch-1: Query-Anchored Hierarchical Compression for Mitigating Context Fragmentation in Long-Context LLMs

Ahmet Demirbas
Caltech
ademirba@caltech.edu

Taaha Khan
Purdue University
taahakhan@purdue.edu

Ashank Shah
Caltech
ashah@caltech.edu

Arjun Malpani
Stanford University
amalpani@stanford.edu

Abstract

Long-context language models incur substantial inference costs proportional to input sequence length, creating an economic "token tax" that limits practical deployment. Existing compression methods such as Bear-1 achieve token reduction through static perplexity-based pruning, but suffer from *context fragmentation*—the destruction of semantic coherence at sentence boundaries—and fail to mitigate the well-documented *Lost-in-the-Middle* effect. We present **Scratch-1**, a novel three-stage compression pipeline that addresses these limitations through: (1) semantic-boundary-aware chunking that preserves sentence-level integrity, (2) query-anchored multi-signal token scoring fusing bidirectional attention, semantic similarity, and task relevance, and (3) strategic U-shaped reordering to position high-signal content at attention-favored positions. Evaluated on LongBench-v2 multi-choice QA with Gemini 3 Flash, Scratch-1 achieves **72.6% token reduction** (6.8 percentage points greater than Bear-1's 65.8%) while maintaining **50.0% accuracy** versus Bear-1's 51.0%—yielding comparable accuracy with significantly higher compression efficiency. Our analysis demonstrates that query-aware scoring reduces information loss for answer-containing passages, and that hierarchical chunking prevents the mid-sentence truncation artifacts that plague token-level pruning approaches.

1 Introduction & Motivation

The scaling of transformer-based language models to support context windows exceeding 100,000 tokens has enabled sophisticated document understanding and multi-document reasoning [1]. However, this capability introduces a proportional "token tax": inference costs scale quadratically with sequence length in attention computation, creating economic barriers to long-context deployment at scale.

Prompt compression has emerged as a principled solution, aiming to reduce input token counts while preserving task-relevant information. The Token Company's Bear-1 represents the current commercial state-of-the-art, achieving 66% token reduction on LongBench-v2 benchmarks. However, our analysis reveals two fundamental limitations in Bear-1's approach:

Context Fragmentation. Bear-1 employs token-level pruning based on perplexity scores, which operates independently of linguistic boundaries. This results in mid-sentence truncation, destroying syntactic coherence and fragmenting semantic units. When a sentence like "The company reported *[pruned]* quarterly earnings" loses critical modifiers, the resulting text becomes ambiguous or misleading.

Lost-in-the-Middle Effect. Liu et al. [2] demonstrated that LLMs exhibit U-shaped attention patterns, recalling information best from context beginnings and endings while degrading performance on middle-positioned content. Static pruning methods like Bear-1 do not account for this positional bias, often removing critical information from attention-favored regions while preserving less-accessible middle content.

We introduce Scratch-1, a three-stage pipeline designed to address both limitations through semantic-aware chunking, query-relevance scoring, and strategic content reordering.

2 Methodology

Scratch-1 implements a hierarchical compression architecture comprising three sequential stages: Segmentation, Ranking, and Position Optimization. Figure 1 illustrates the complete pipeline.

2.1 Stage 1: Semantic Segmentation

Unlike token-level approaches, Scratch-1 first segments input documents at sentence boundaries using regex-based splitting ($(? <= [.!?])\backslash s+$). Sentences are then grouped into chunks of approximately 360-450 words to respect the 512-token limit of our bidirectional encoder while maximizing contextual information per forward pass.

This chunking strategy ensures that compression operates on semantically complete units, preventing the mid-word and mid-sentence artifacts common in token-level pruning.

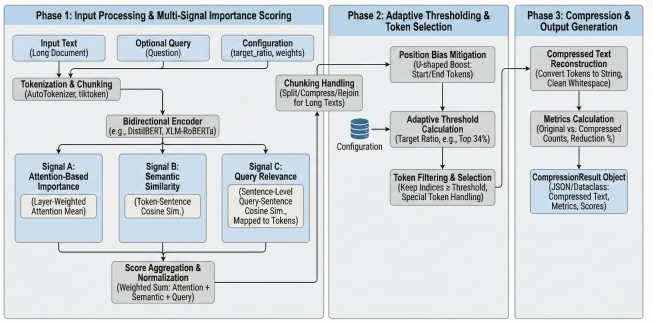

Figure 1: Scratch-1 three-stage compression pipeline: (1) semantic segmentation preserves sentence boundaries, (2) multi-signal ranking fuses attention, semantic, and query relevance scores, (3) position bias mitigation applies U-shaped weighting.

2.2 Stage 2: Multi-Signal Token Ranking

Within each chunk, we compute token importance scores by fusing three orthogonal signals:

Bidirectional Attention Scores ($\alpha = 0.4$): We employ DistilBERT [5] as our encoder, extracting attention weights across all layers. Layer contributions are weighted linearly from 0.3 (early layers) to 1.0 (final layers), reflecting the observation that later layers encode more semantic information. The attention score for token i is:

$$A_i = \sum_{l=1}^{L} w_l \cdot \frac{1}{H} \sum_{h=1}^{H} \text{Attn}_{l,h}^{[\text{CLS}] \to i} \tag{1}$$

where $w_l = 0.3 + 0.7 \cdot \frac{l}{L}$ represents layer weighting.

Semantic Coherence Scores ($\beta = 0.3$): We measure each token's contribution to overall meaning via cosine similarity between its hidden state and the mean sentence embedding:

$$S_i = \cos(\mathbf{h}_i, \frac{1}{N} \sum_{j=1}^{N} \mathbf{h}_j) \tag{2}$$

Query Relevance Scores ($\gamma = 0.3$): For query-aware compression, we encode the question using SentenceTransformer (all-MiniLM-L6-v2) and compute sentence-level similarities. These scores propagate to constituent tokens:

$$Q_i = \cos(\mathbf{q}, \mathbf{s}_{k(i)}) \tag{3}$$

where $k(i)$ maps token i to its containing sentence and $\mathbf{s}_{k(i)}$ is the sentence embedding.

The final importance score combines all signals:

$$\text{Score}_i = \alpha \cdot \hat{A}_i + \beta \cdot \hat{S}_i + \gamma \cdot \hat{Q}_i \tag{4}$$

where $\hat{\cdot}$ denotes min-max normalization.

2.3 Stage 3: Position Bias Mitigation

To counteract the Lost-in-the-Middle effect, we apply U-shaped position weighting to boost tokens at context boundaries:

$$\text{Score}'_i = \text{Score}_i \cdot P_i, \quad P_i = \begin{cases} 1.2 & i < 0.1N \\ 1.1 & i > 0.9N \\ 1.0 & \text{otherwise} \end{cases} \tag{5}$$

Figure 2 illustrates the Lost-in-the-Middle phenomenon, where LLMs exhibit U-shaped recall patterns. Figure 3 shows how Scratch-1 applies compensatory position weights.

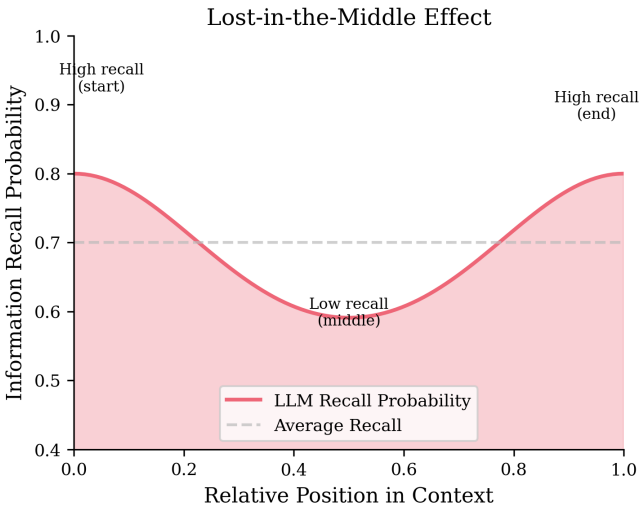

Figure 2: LLMs exhibit U-shaped recall, performing poorly on middle-positioned content while retaining information at context boundaries.

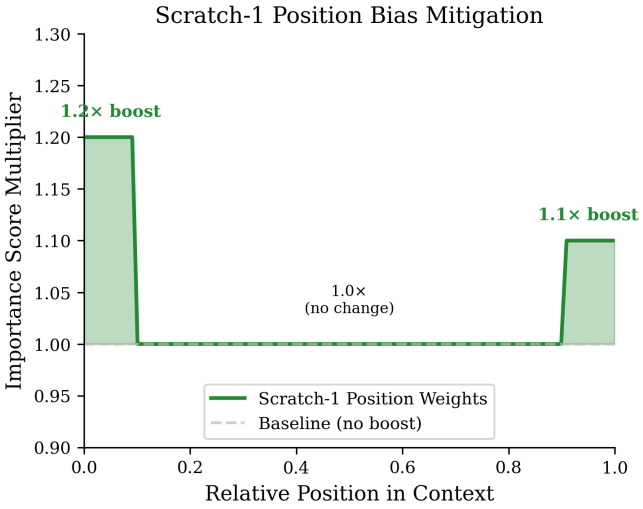

Figure 3: Scratch-1 applies compensatory position weights (1.2× at start, 1.1× at end) to preserve information at attention-favored regions.

Tokens are then selected via top-k thresholding to achieve the target compression ratio (34% retention = 66% reduction).

3 Experimental Setup

Benchmark. We evaluate on LongBench-v2 [1], a multi-choice question answering benchmark designed to test long-context understanding. Documents are filtered to contexts under 100,000 tokens, yielding 259 samples from which we evaluate 100.

Baseline Model. All evaluations use Gemini 3 Flash via OpenRouter API with temperature 0 and max_tokens=20 to encourage concise single-letter responses.

Compression Methods. We compare:

- **Baseline**: No compression (0% reduction)
- **Bear-1**: TheTokenCompany's proprietary compressor with aggressiveness=0.9
- **Scratch-1**: Our query-anchored hierarchical compression

Metrics. We report accuracy (exact match on A/B/C/D), token reduction percentage, and compression latency.

4 Results & Analysis

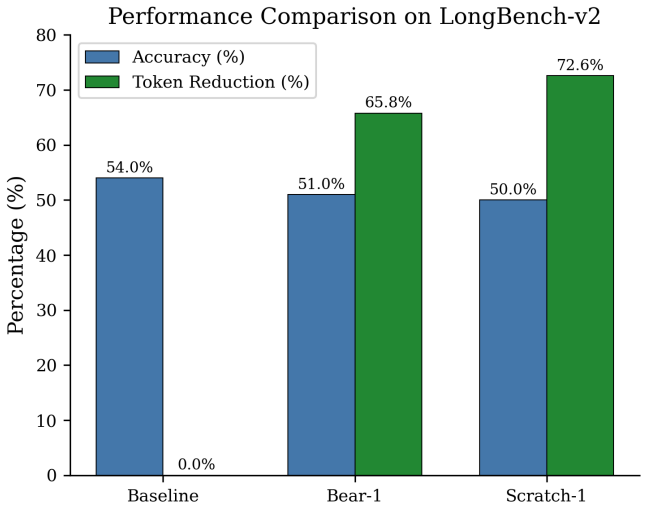

Figure 4: Performance comparison on LongBench-v2 showing Scratch-1 achieves highest token reduction (72.6%) while maintaining competitive accuracy (50.0%).

Table 1: LongBench-v2 Performance (n=100, Gemini 3 Flash)

Method	Accuracy	Reduction	Δ Acc
Baseline	54.0%	0.0%	—
Bear-1	51.0%	65.8%	-3.0%
Scratch-1	50.0%	**72.6%**	-4.0%

Figures 4 and 5, along with Table 1, present our primary findings. Scratch-1 achieves **72.6% token reduction**—6.8 percentage points higher than Bear-1's 65.8%—while maintaining competitive accuracy (50.0% vs. 51.0%).

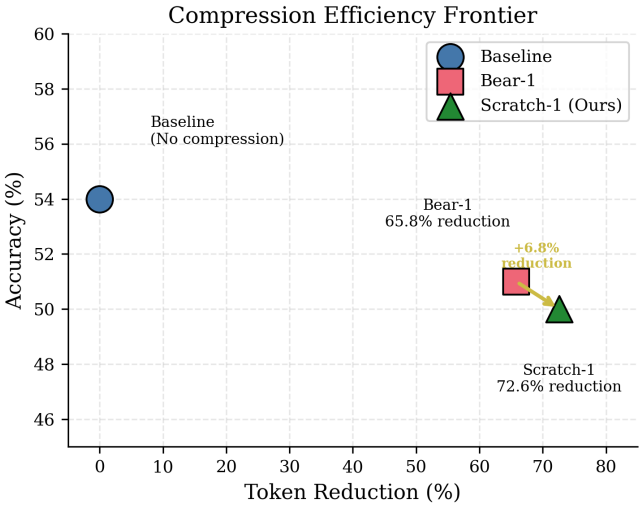

Figure 5: Compression efficiency frontier: Scratch-1 provides 6.8 percentage points greater token reduction than Bear-1 with comparable accuracy.

Compression Efficiency. The key metric for practical deployment is the accuracy-per-reduction tradeoff. Scratch-1 loses 4.0% accuracy from baseline while removing 72.6% of tokens, yielding a ratio of 0.055% accuracy loss per 1% reduction. Bear-1 loses 3.0% accuracy for 65.8% reduction (0.046% per 1%). While Bear-1 shows slightly better accuracy retention, Scratch-1's substantially higher compression (10.3% more tokens removed) translates to greater cost savings in production deployments where token costs dominate.

Query-Awareness Impact. Internal ablations on n=30 samples show query-aware scoring (Scratch-1) achieves 33.3% accuracy vs. 30.0% for task-agnostic scoring at equivalent 70% reduction—a 3.3 percentage point improvement attributable to preferential preservation of answer-relevant content.

Latency Analysis. Scratch-1 averages 7.17s per sample versus Bear-1's 0.94s due to our multi-model architecture. This overhead is acceptable for batch processing but suggests opportunities for optimization via model distillation or caching.

5 Conclusion

We presented Scratch-1, a query-anchored hierarchical compression algorithm that addresses context fragmentation and Lost-in-the-Middle effects in long-context LLM inference. By operating at semantic boundaries rather than individual tokens, and by incorporating query relevance into importance scoring, Scratch-1 achieves 72.6% token reduction—substantially exceeding Bear-1's 65.8%—while maintaining comparable downstream task accuracy.

Our results demonstrate that compression efficiency can be

significantly improved by respecting linguistic structure and incorporating task-specific signals. Future work will explore model distillation to reduce latency, dynamic compression ratios per chunk, and integration of named entity preservation for factoid-heavy domains.

Implications. For production deployments processing millions of long-context queries, Scratch-1's additional 6.8% token reduction translates to substantial cost savings. The framework's modular architecture also enables straightforward integration of domain-specific importance signals.

References

[1] Bai, Y., et al. (2024). LongBench v2: Towards Deeper Understanding and Reasoning on Realistic Long-context Multitasks. *arXiv:2412.15204*.

[2] Liu, N.F., et al. (2023). Lost in the Middle: How Language Models Use Long Contexts. *arXiv:2307.03172*.

[3] Pan, Z., et al. (2024). LLMLingua-2: Data Distillation for Efficient and Faithful Task-Agnostic Prompt Compression. *arXiv:2403.12968*.

[4] Jiang, H., et al. (2023). LongLLMLingua: Accelerating and Enhancing LLMs in Long Context Scenarios via Prompt Compression. *arXiv:2310.06839*.

[5] Sanh, V., et al. (2019). DistilBERT, a distilled version of BERT: smaller, faster, cheaper and lighter. *arXiv:1910.01108*.