Multi-Needle Retrieval F1 Degradation on LongBench with CAKE and Static Eviction in 7B Models

Assignee Research

June 8, 2026

Abstract

This report synthesises findings from 12 peer-reviewed papers addressing the following research question: What is the degradation in Multi-needle retrieval F1 scores on LongBench for Python-heavy contexts when applying CAKE versus static eviction strategies in 7B parameter models. 8 claims were extracted from source literature; 0 were independently verified against retrieved documents. An automated multi-reviewer quality assessment produced a score of 4.2/10. This report is a machine-generated literature synthesis and does not constitute original research.

1 Introduction

This paper examines: CAKE: Cascading and Adaptive KV Cache Eviction with Layer Preferences. Research question: What is the degradation in Multi-needle retrieval F1 scores on LongBench for Python-heavy contexts when applying CAKE versus static eviction strategies in 7B parameter models?.

2 Methodology

Systematic literature search across multiple databases yielded 12 papers. Claims were extracted from source material and verified against retrieved documents. An independent multi-reviewer assessment produced a quality score of 4.2/10.

3 Results

12 papers retrieved. 8 claims extracted; 0 independently verified. Quality review score: 4.2/10.

4 Limitations

This report is a machine-generated literature synthesis and does not constitute original research. Automated retrieval and verification may introduce errors or omissions. Review scores reflect automated assessment, not human peer review. Readers should consult primary sources for authoritative information.

5 Extracted Claims

Claim	Verified	Confidence
CAKE achieves an approximate 48.63% reduction in peak memory usage compared to the full cache implementation with a 128K	×	0.06
CAKE demonstrates over 10$\times$ speedup in decoding latency compared to the full cache approach when processing sequences wit	×	0.14
CAKE maintains a relatively stable decoding speed by preserving a fixed amount of KV cache, resulting in significantly l	×	0.11
Methods equipped with CAKE's allocation strategy consistently improve performance across nearly all tasks compared to va	×	0.07
CAKE achieves significant overall performance gains across different eviction methods and tasks.	×	0.07
The preference score P for an attention layer's cache size is defined as P = H^(1/τ1) $\cdot$ V^(1/τ2), where H and V are meas	×	0.07
Layers with high preference score P benefit more from larger KV cache to maintain performance.	×	0.06
The preference-prioritized adaptive allocation strategy considers each layer's unique characteristics and adaptively all	×	0.11

References

- http://arxiv.org/abs/2503.14800v3
- http://arxiv.org/abs/2503.12491v2
- http://arxiv.org/abs/2602.10238v1