Journal of Machine Learning **ISSN**: 2790-2048(e), 2790-203X(p)

Memory^3: Language Modeling with Explicit Memory

Hongkang Yang *[1], Zehao Lin[1], Wenjin Wang[1], Hao Wu[1], Zhiyu Li[1], Bo Tang[1], Wenqiang Wei[1], Jinbo Wang[1,4], Zeyun Tang[1], Shichao Song[1], Chenyang Xi[1], Yu Yu[1], Kai Chen[1], Feiyu Xiong[1], Linpeng Tang[2], and Weinan E[1,3,4,5]

[1]Center for LLM, Institute for Advanced Algorithms Research, Shanghai 200233, China.
[2]Moqi Inc, Beijing 100080, China.
[3]Center for Machine Learning Research, Peking University, Beijing 100871, China.
[4]School of Mathematical Sciences, Peking University, Beijing 100871, China.
[5]AI for Science Institute, Beijing 100083, China.

Abstract. The training and inference of large language models (LLMs) are together a costly process that transports knowledge from raw data to meaningful computation. Inspired by the memory hierarchy of the human brain, we reduce this cost by equipping LLMs with explicit memory, a memory format cheaper than model parameters and text retrieval-augmented generation (RAG). Conceptually, with most of its knowledge externalized to explicit memories, the LLM can enjoy a smaller parameter size, training cost, and inference cost, all proportional to the amount of remaining "abstract knowledge". As a preliminary proof of concept, we train from scratch a 2.4 B LLM, which achieves better performance than much larger LLMs as well as RAG models, and maintains higher decoding speed than RAG. The model is named Memory^3, since explicit memory is the third form of memory in LLMs after implicit memory (model parameters) and working memory (context key-values). We introduce a memory circuitry theory to support the externalization of knowledge, and present novel techniques including a memory sparsification mechanism that makes storage tractable and a two-stage pretraining scheme that facilitates memory formation.

Keywords:
Large language model,
Explicit memory,
Large-scale pretraining,
Efficient inference,
AI database.

Article Info.:
Volume: 3
Number: 3
Pages: 300 - 346
Date: September/2024
doi.org/10.4208/jml.240708

Article History:
Received: 08/07/2024
Accepted: 20/08/2024

Communicated by:
Zhi-Qin Xu

1 Introduction

Large language models have enjoyed unprecedented popularity in recent years thanks to their extraordinary performance [1,2,6,9,51,53,108,125]. The prospect of scaling laws [50, 57, 95] and emergent abilities [101, 117] constantly drives for substantially larger models, resulting in the rapid increase in the cost of LLM training and inference. People have been trying to reduce this cost through optimizations in various aspects, including architecture [3,27,37,71,86,107], data quality [45,55,63,100], operator [29,60], parallelization [59,88,92, 98], optimizer [67,115,123], scaling laws [50,126], generalization theory [52,130], hardware [30], etc.

We introduce the novel approach of optimizing knowledge storage. The combined cost of LLM training and inference can be seen as the cost of encoding the knowledge from text

*Corresponding author: hongkang@alumni.princeton.edu

data into various memory formats, plus the cost of reading from these memories during inference

$$\sum_{\text{knowledge } k} \min_{\text{format } m} \text{cost}_{\text{write}}(k, m) + n_k \cdot \text{cost}_{\text{read}}(k, m), \tag{1.1}$$

where $\text{cost}_{\text{write}}$ is the cost of encoding a piece of knowledge k into memory format m, $\text{cost}_{\text{read}}$ is the cost of integrating k from format m into inference, and n_k is the expected usage count of this knowledge during the lifespan of this LLM (e.g. a few months for each version of ChatGPT [8,83]). The definitions of knowledge and memory in the context of LLMs are provided in Section 2, and this paper uses knowledge as a countable noun. Typical memory formats include model parameters and plain text for retrieval-augmented generative models, their write functions and read functions are listed in Table 1.1, and their $\text{cost}_{\text{write}}$ and $\text{cost}_{\text{read}}$ are provided in Fig. 1.1.

We introduce a new memory format, explicit memory, characterized by moderately low write cost and read cost. As depicted in Fig. 1.2, our model first converts a knowledge base (or any text dataset) into explicit memories, implemented as sparse attention key-values, and then during inference, recalls these memories and integrates them into the self-

Table 1.1: Analogy of the memory hierarchies of humans and LLMs.

Memory format of humans	Example	Memory format of LLMs	Write	Read
Implicit memory	Common expressions	Model parameters	Training	Matrix multiplication
Explicit memory	Books read	This work	Memory encoding	Self-attention
External information	Open-book exam	Plain text (RAG)	None	Encode from scratch

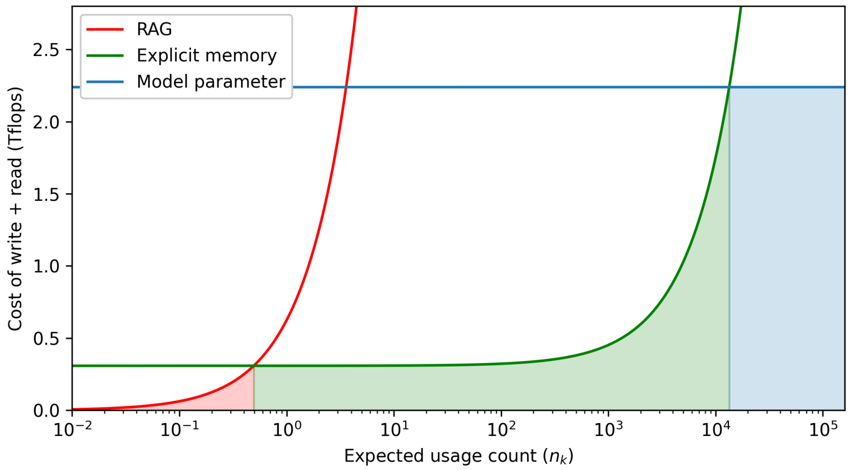

Figure 1.1: The total cost (TFlops) of writing and reading a piece of knowledge by our 2.4 B model with respect to its expected usage count. The curves represent the cost of different memory formats, and the shaded area represents the minimum cost given the optimal format. The plot indicates that $(0.494, 13400)$ is the advantage interval for explicit memory. The calculations are provided in Appendix A. (The blue curve is only a coarse lower bound on the cost of model parameters.)