Enhancing Hierarchical Knowledge Editing for LLMs: Instance-to-Concept Relationship Perspective

Zhaoyuan Zhang[1], Tao Luo[1,*], Xiaowang Zhang[1] and Sai Zhang[1]

[1]*College of Intelligence and Computing, Tianjin University, Tianjin, 300350, China*

Abstract

Knowledge editing has emerged as a promising way to update knowledge in large-scale language models (LLMs) efficiently. However, current knowledge editing methods focus on undifferentiated factual knowledge, neglecting the significance of hierarchical structured knowledge editing. Moreover, cognitive science has revealed the importance of hierarchical knowledge for human learning. This poster introduces hierarchical knowledge editing for instance-to-concept relationship. Through **Layered Distillation** strategy, we perform knowledge distillation between the original and edited models, thereby preserving instance-to-concept relationship hierarchical knowledge in the original model. Experimental results demonstrate that integrating our strategy with existing knowledge editing methods enhances the performance of hierarchical knowledge editing.

Keywords

Knowledge Editing, Large Language Model, Hierarchical Knowledge, Layered Distillation

1. Introduction

With the performance improvement and wide application of large language models, the problems of LLMs, such as providing outdated, erroneous, or toxic information, have become the focus of criticism. Retraining LLMs to address these issues takes time and exertion. In contrast, knowledge editing offers a low-cost way to update trained models. This has made the development of efficient and reliable knowledge editing methods for LLMs a key area of research[1]. However, most existing LLM knowledge editing methods do not differentiate between types of knowledge and primarily concentrate on editing factual knowledge individually, which is inefficient for LLMs with a massive number of parameters that store vast amounts of knowledge. Moreover, there is a lack of effective retention of structured hierarchical information within LLMs. Based on this, this poster proposes hierarchical knowledge editing.

LLMs memorize various hierarchical knowledge, hierarchical knowledge editing task includes editing both instance-to-concept and other inter-conceptual relationships hierarchical knowledge. Focusing on instance-to-concept relationship, we want to retain the integrity of this relationship in the original model after editing concept. Consider a simple instance-to-concept relationship hierarchical knowledge in LLMs: instance “tiger” belongs to concept “feline,” which is defined as a carnivorous mammal known for flexible body and sharp claws. In the case where

Posters, Demos, and Industry Tracks at ISWC 2024, November 13–15, 2024, Baltimore, USA

[*] Corresponding author.

zhaoyuanzhang@tju.edu.cn (Z. Zhang); luo_tao@tju.edu.cn (T. Luo); xiaowangzhang@tju.edu.cn (X. Zhang); zhang_sai@tju.edu.cn (S. Zhang)

the definition of “feline” is edited to winged animal, since “tiger” is wingless, human cognition will naturally assume that “tiger” no longer belongs to “feline,” which means that we do not want this editing to modify the hierarchical knowledge of “tiger” belongs to “feline” in the original LLMs[2]. Thus, it is vital to maintain the original instance-to-concept relationship hierarchical knowledge when editing LLMs. While a recent work proposes editing conceptual knowledge, which focuses on the effects of modifying concept definitions within LLMs[2], hierarchical knowledge editing concentrates on evolving editing approaches that preserve hierarchical knowledge within LLMs.

In this poster, we define hierarchical Knowledge Editing task with corresponding metric, design **Layered Distillation** strategy for preserving instance-to-concept relationship hierarchical knowledge, and present experimental evidence of its meaningful effectiveness in enhancing hierarchical knowledge editing.

2. Approach

LLMs memorize various hierarchical knowledge, Hierarchical Knowledge Editing task includes managing both instance-to-concept and other inter-conceptual relationships. Focusing on instance-to-concept relationship, our goal is to design an editing approach that retains the integrity of the relationship when concept definitions are updated.

2.1. Task Definition

Hierarchical knowledge editing task is formally defined as: given a concept $\boldsymbol{C} = (c, d)$, where c is the concept name and d is the concept definition, and a set of instances $\boldsymbol{I}$, all instances i in $\boldsymbol{I}$ belong to concept $\boldsymbol{C}$, denoted $i \in \boldsymbol{C}$. When the definition d is edited to d^*, resulting in the modified concept $\boldsymbol{C}^* = (c, d^*)$, a great hierarchical knowledge editing approach must ensure that the modification of concept does not result in $i \in \boldsymbol{C}^*$, maintain the integrity of $\boldsymbol{I}$, minimize instance migration, and avoid unacceptable changes to the original model.

2.2. Knowledge Editing with Layered Distillation

We introduce knowledge distillation to ensure the edited model inherits the original model’s hierarchical knowledge[3]. We first use Location-Then-Edit as the base editing method[1, 4, 5]. This method targets specific model layers for editing, enabling focused distillation of the original and edited model layers, rather than the entire model. Layered Distillation is updating the model layers again using these original and edited model layers as input, with Mean Square Error as the distillation loss. The specific process is shown in Figure 1. The base method edits the concept definition in the original model, resulting in instance migration. Updating the edited model again through the Layered Distillation strategy retains the instance-to-concept relationship in the original model.

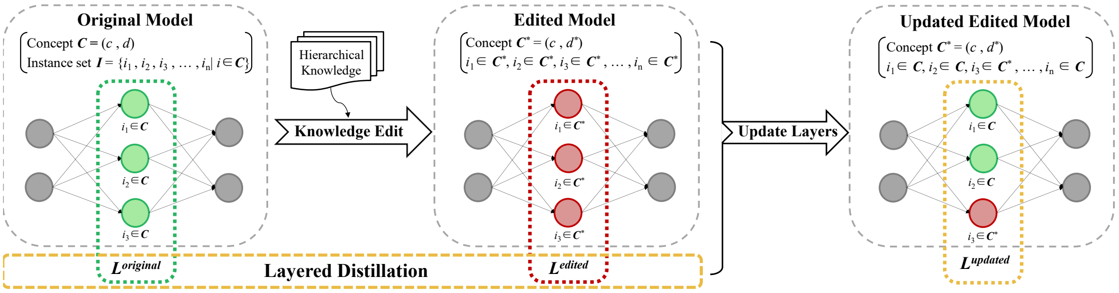

Figure 1: The overall process of Knowledge Editing with Layered Distillation.

2.3. Metrics

To more effectively assess the impact of our editing approach on hierarchical knowledge, we devise **Instance Retention(IR)** metric. Its definition is as follows:

$$IR = 1 - \frac{1}{n}\sum_{i \in C}^{n}[G(i, C^*) + |H(i, C) - H(i, C^*)|] \tag{1}$$

Instance Retention measures the proportion of instance-to-concept relationship knowledge preserved in the edited model. It is determined by two functions: $G(i, C)$, which indicates whether instance i belongs to concept C ($G = 1$ if it does, $G = 0$ if it does not). $H(i, C)$, used when instance i cannot determine whether it belongs to concept C($H = 1$ in this case, $H = 0$ otherwise). These functions are applied using the reasoning capability of the LLM[2]. We also use **Reliability**(**Rel.**), **Generalization**(**Gen.**), and **Locality**(**Loc.**) as comprehensive metrics to evaluate the success, scope, and impact of knowledge editing[1].

3. Experiments

Table 1
Main results of the Hierarchical Knowledge Editing experiment. "+**LD**" stands for combined with our strategy. **Bold** results denote optimal performance.

Model	Method	Intra				Inter			
		Rel.↑	Gen.↑	Loc.↑	IR.↑	Rel.↑	Gen.↑	Loc.↑	IR.↑
GPT2-XL	ROME	**86.45**	**49.67**	84.76	21.75	**82.85**	**45.52**	86.21	20.22
	ROME+**LD**	85.06	48.29	84.43	**25.12**	80.37	44.53	85.62	**25.31**
	MEMIT	43.97	33.09	96.11	2.70	39.28	29.77	**95.93**	3.34
	MEMIT+**LD**	45.74	33.53	**96.97**	6.03	40.79	30.50	95.70	5.42
TinyLlama	ROME	**97.55**	**77.28**	92.83	18.84	**96.91**	**74.81**	92.98	21.52
	ROME+**LD**	95.56	75.70	92.97	**24.71**	95.26	72.12	92.92	**25.30**
	MEMIT	93.94	66.85	**94.12**	16.99	92.81	63.69	**94.59**	16.90
	MEMIT+**LD**	93.76	66.56	93.18	18.75	92.73	62.91	94.54	18.63

Utilizing the ConceptEdit[2], derived from the DBpedia ontology dataset, we conducted experiments on the open-source LLMs GPT2-XL (1.5B) [6]and TinyLlama (1.1B)[7] using the ROME[4] and MEMIT[5] methods combined with Layered Distillation(LD) on GeForce RTX 4090. ConceptEdit's Intra and Inter modules represent modifications to concepts within and between superclasses.

Table 1 results indicate that both editing methods struggle with hierarchical knowledge editing in both models, but the **IR** of ROME method is significantly higher. Meanwhile, Layered Distillation strategy with these methods enhances **IR** without notably altering other metrics, particularly for ROME editing TinyLlama. This enhancement validates the efficacy of our strategy for editing instance-to-concept relationship hierarchical knowledge. Furthermore, the relatively modest enhancement observed in the MEMIT+LD configuration can be attributed to the editing across multiple MLP layers, which is responsible for the suboptimal performance of the GPT2-XL model, but concurrently offers a higher degree of **Locality**.

4. Conclusion and Future Work

This poster proposes Hierarchical Knowledge Editing for LLMs and Layered Distillation strategy to enhance existing knowledge editing methods for editing instance-to-concept relationship hierarchical knowledge. Our experiments initially demonstrate the efficacy of Layered Distillation. Future research will broaden validation to additional methods and larger LLMs. Editing other hierarchical relationship knowledge will also be further explored.

References

[1] N. Zhang, Y. Yao, B. Tian, P. Wang, S. Deng, M. Wang, Z. Xi, S. Mao, J. Zhang, Y. Ni, et al., A comprehensive study of knowledge editing for large language models, arXiv preprint arXiv:2401.01286 (2024).

[2] X. Wang, S. Mao, N. Zhang, S. Deng, Y. Yao, Y. Shen, L. Liang, J. Gu, H. Chen, Editing conceptual knowledge for large language models, 2024. arXiv:2403.06259.

[3] J. Gou, B. Yu, S. J. Maybank, D. Tao, Knowledge distillation: A survey, International Journal of Computer Vision 129 (2021) 1789–1819.

[4] K. Meng, D. Bau, A. Andonian, Y. Belinkov, Locating and editing factual associations in gpt, in: Proceedings of the 36th International Conference on Neural Information Processing Systems, 2022, pp. 17359–17372.

[5] K. Meng, A. S. Sharma, A. J. Andonian, Y. Belinkov, D. Bau, Mass-editing memory in a transformer, in: The Eleventh International Conference on Learning Representations, ICLR 2023, Kigali, Rwanda, May 1-5, 2023, OpenReview.net, 2023. URL: https://openreview.net/pdf?id=MkbcAHIYgyS.

[6] A. Radford, J. Wu, R. Child, D. Luan, D. Amodei, I. Sutskever, et al., Language models are unsupervised multitask learners, OpenAI blog 1 (2019) 9.

[7] P. Zhang, G. Zeng, T. Wang, W. Lu, Tinyllama: An open-source small language model, arXiv preprint arXiv:2401.02385 (2024).