大模型知识编辑领域进展

李英杰

2024.07.05

中国科学院信息工程研究所
INSTITUTE OF INFORMATION ENGINEERING,CAS

ASCII

目录

大模型知识编辑简介

任务介绍

问题与挑战

参考文献

大模型知识编辑任务

背景

LLMs 通过预训练拥有了丰富的事实性知识以及常识，形成了参数化的知识库。但直接将它们视作知识库仍然存在局限，理想的知识库应该能够进行信息的更新，以纠正错误。

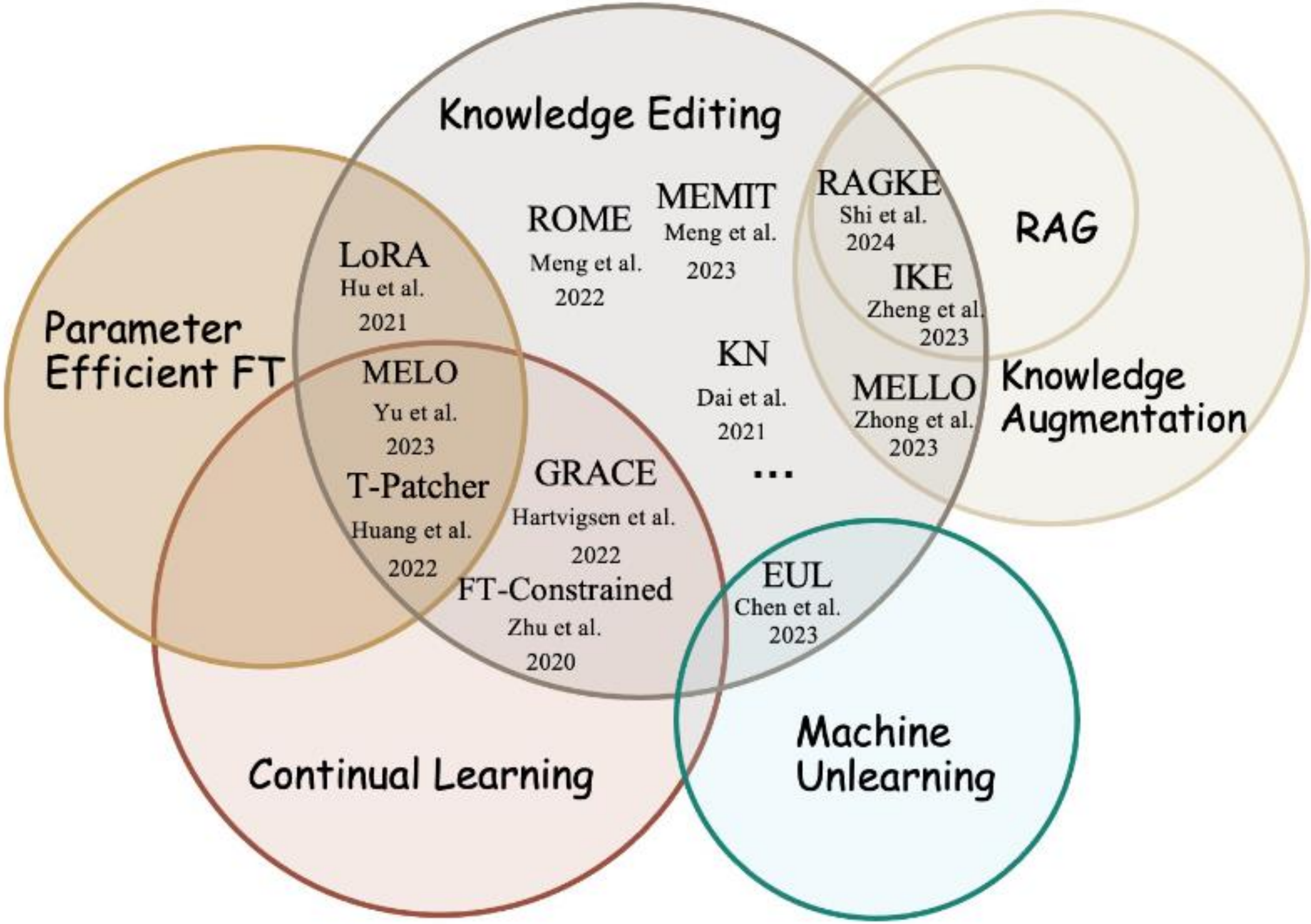
知识编辑旨在通过对LLMs中的特定知识进行修改，来提高LLMs作为知识库的准确性和可靠性。

问题

大模型中存在的“unwanted knowledge”：
偏见、错误信息、过时信息、有害内容、隐私内容等

大模型知识编辑任务

背景



大模型知识编辑任务

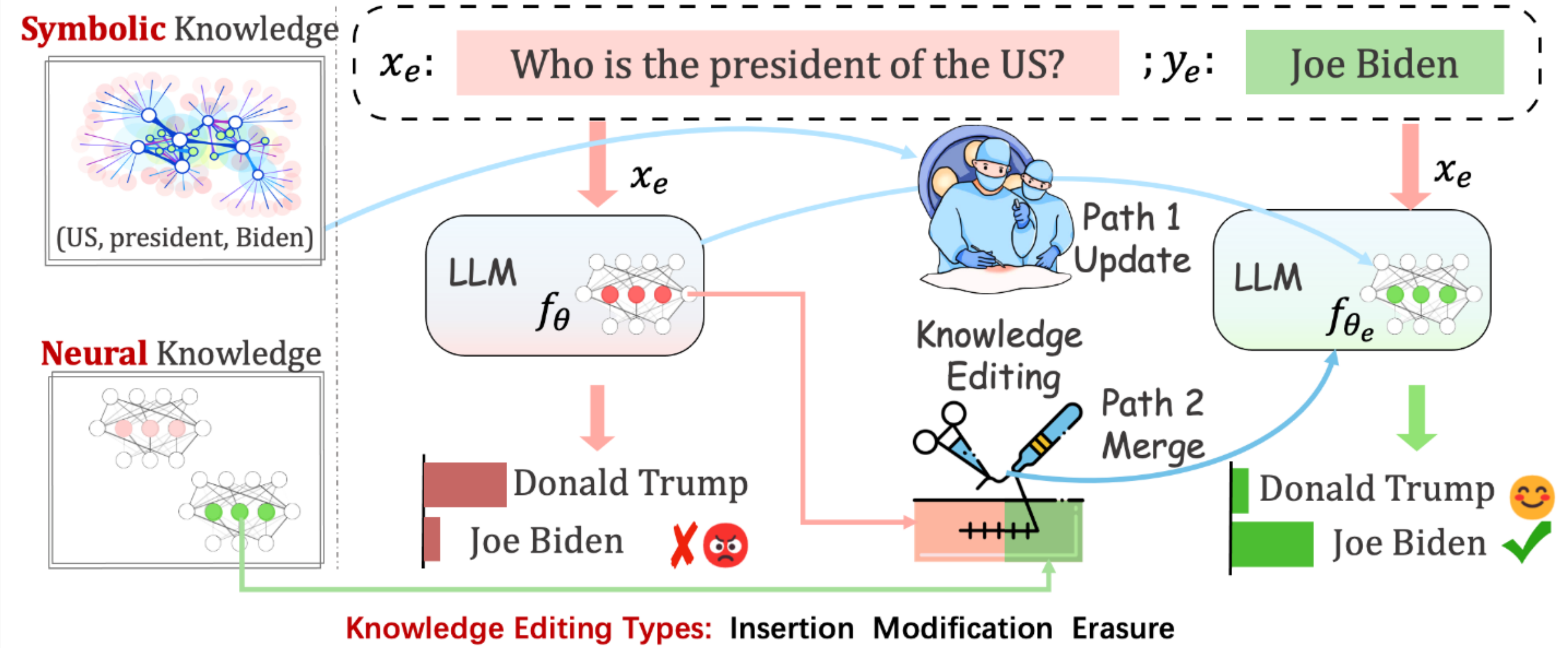
意义

知识编辑旨在“高效”地更新大模型的知识。

$$f_{\theta_e}(x) = \begin{cases} y_e & \text{if } x \in I(x_e, y_e) \\ f_\theta(x) & \text{if } x \in O(x_e, y_e) \end{cases}$$

任务目标

改变LLM对于特定知识的响应，同时尽量不对其他响应产生影响。



in-scope
编辑范围内
I(x_e, y_e)

out-of-scope
编辑范围外
O(x_e, y_e)

大模型知识编辑任务

评测指标

- Reliability 可靠性：对于给定单个知识的编辑成功率
- Generalization （域内）泛化性：对于给定编辑范围的编辑成功率
- Portability 可移植性：将知识转变为相关内容的编辑成功率，鲁棒性
- Locality 局部性：即模型对于out-of-scope的影响
- Efficiency 效率：时间、空间开销

目录

- 大模型知识编辑简介
- 任务分类介绍
- 问题与挑战
- 参考文献

任务分类

缺乏新知识

- 知识插入：增加域外的新知识

错误、过时知识

- 知识修改：知识修正、知识干扰

有害内容、隐私信息

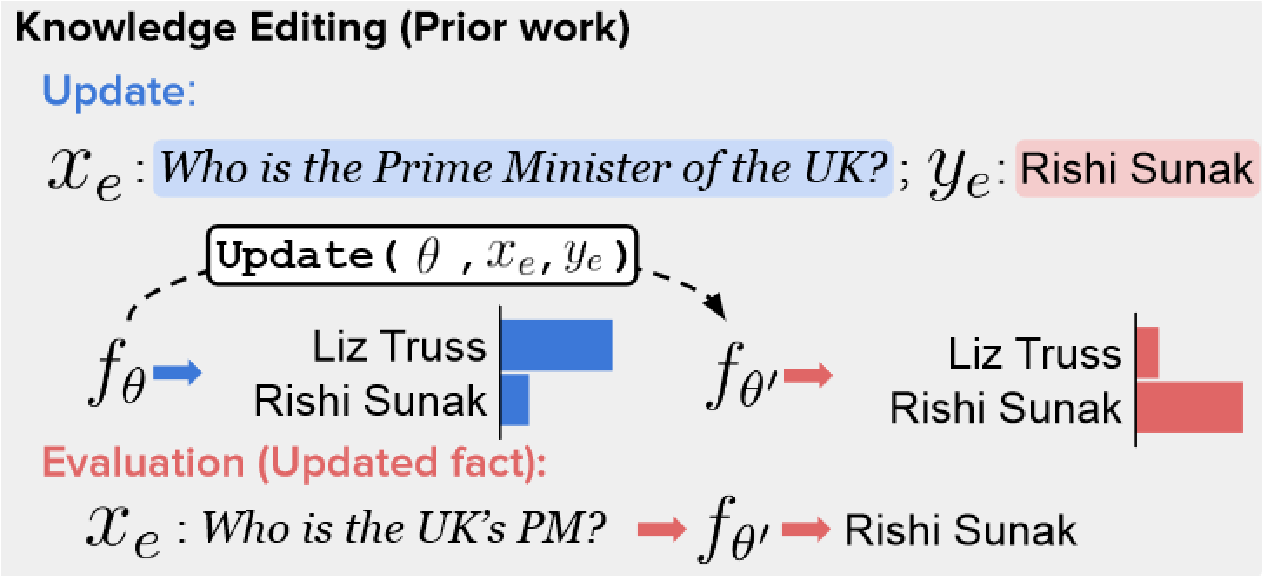
- 知识擦除：大模型祛毒，消除偏见、有害的知识

INSERTION 知识插入

Can LMs learn new entities from descriptions? challenges in propagating injected knowledge (ACL 2023)

motivation

探究已有的知识编辑方法（参数更新方法）是否能将更新的知识用于推理



Knowledge Editing (Prior work)

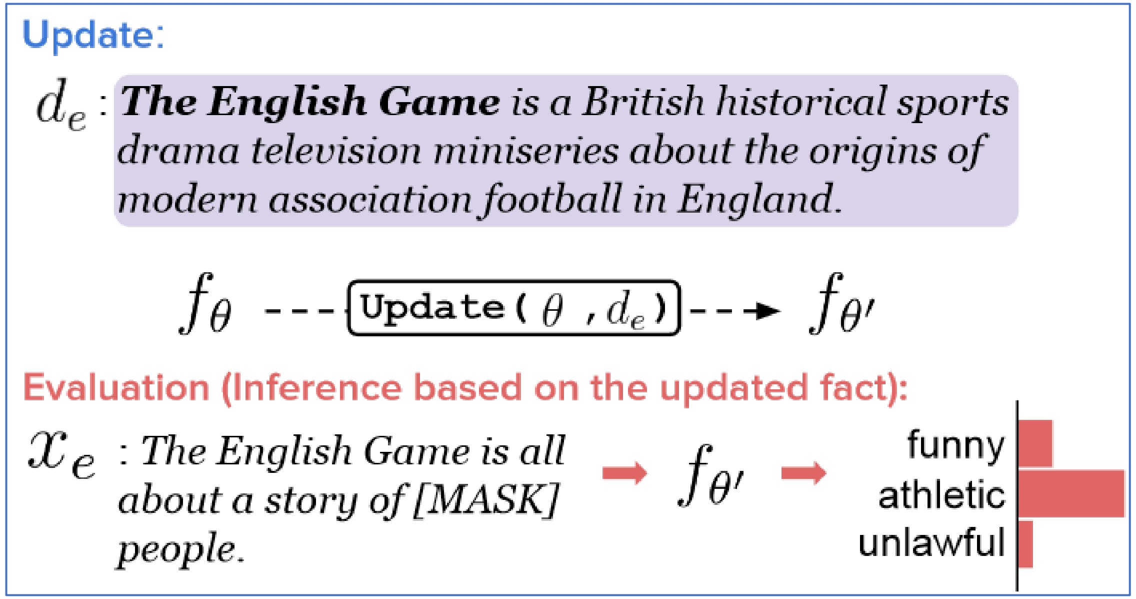
Update:

x_e : *Who is the Prime Minister of the UK?* ; y_e: Rishi Sunak

Update(θ , x_e, y_e)

Evaluation (Updated fact):

x_e : *Who is the UK's PM?* → $f_{\theta'}$ → Rishi Sunak



Update:

d_e : ***The English Game*** *is a British historical sports drama television miniseries about the origins of modern association football in England.*

f_θ --- Update(θ , d_e) ---→ $f_{\theta'}$

Evaluation (Inference based on the updated fact):

x_e : *The English Game is all about a story of [MASK] people.* → $f_{\theta'}$ →

INSERTION 知识插入

Can LMs learn new entities from descriptions? challenges in propagating injected knowledge

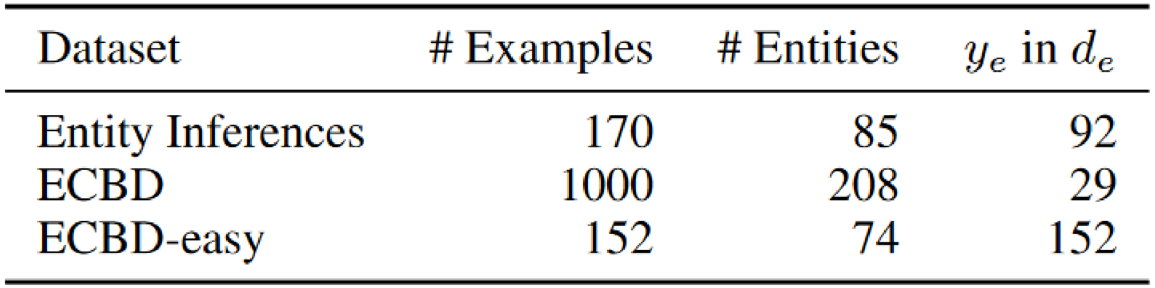
contribution

提出了一个新benchmark，依据Entity Cloze By Date (ECBD) dataset，人工设计了新的完形填空式的选择题来评估模型进行知识编辑后对于新实体的推理/应用能力。

比较了直接在上下文中加入定义和参数更新的性能。

evaluation

- □ 编辑成功率：完型填空的标签准确率
- □ 特异性：比较更新前后对于无关实体语句的困惑度变化程度



Dataset	# Examples	# Entities	y_e in d_e
Entity Inferences	170	85	92
ECBD	1000	208	29
ECBD-easy	152	74	152

INSERTION 知识插入

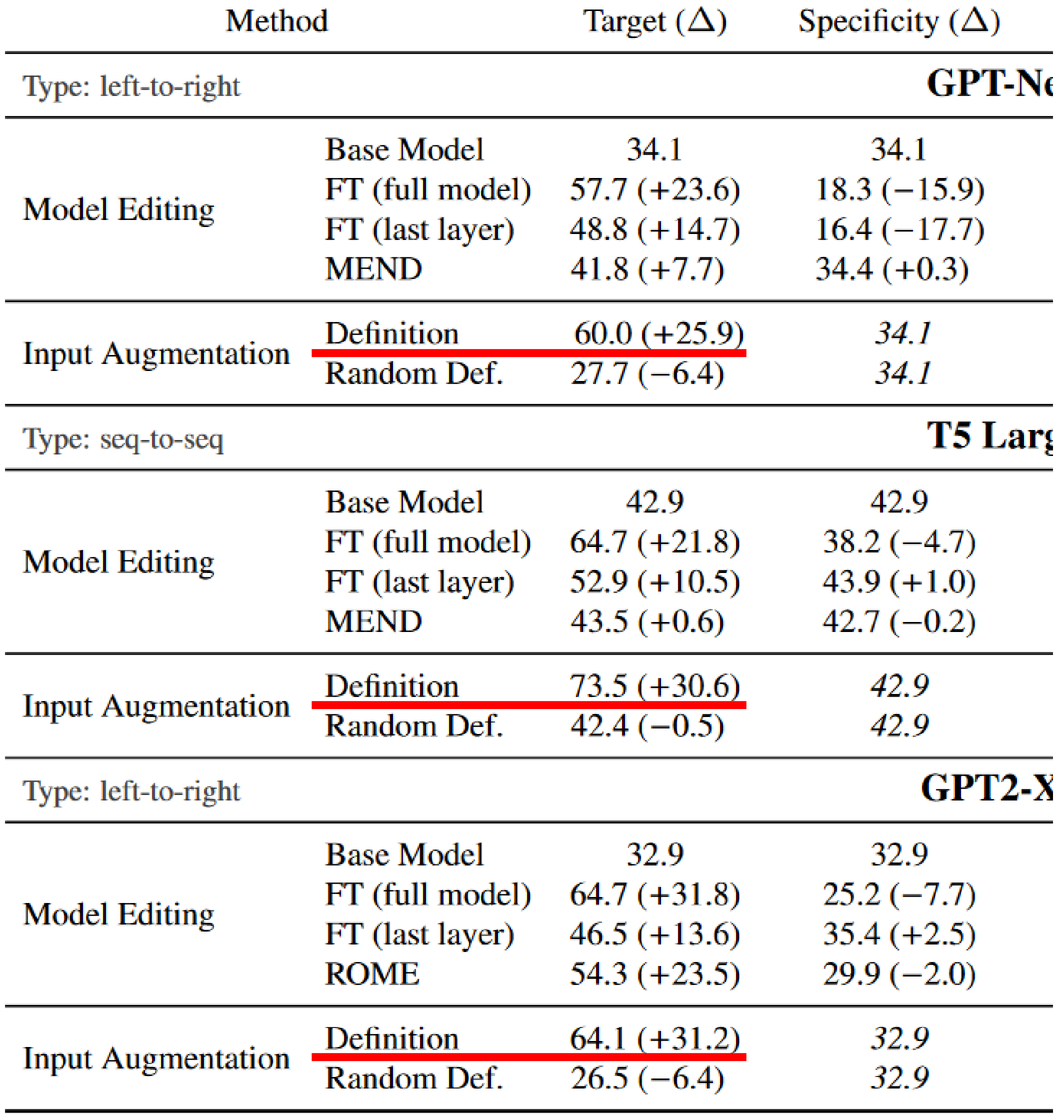
Can LMs learn new entities from descriptions? challenges in propagating injected knowledge

experiments

可以看到，在作者构造的数据集上，直接增加上下文实体定义信息的方法有更好的效果

limitation

直接在上下文增加定义的方法难以处理添加多实体知识的问题（上下文长度有限，多个实体的定义会有干扰）



Method		Target (Δ)	Specificity (Δ)
Type: left-to-right			GPT-Ne
Model Editing	Base Model	34.1	34.1
	FT (full model)	57.7 (+23.6)	18.3 (−15.9)
	FT (last layer)	48.8 (+14.7)	16.4 (−17.7)
	MEND	41.8 (+7.7)	34.4 (+0.3)
Input Augmentation	Definition	60.0 (+25.9)	*34.1*
	Random Def.	27.7 (−6.4)	*34.1*
Type: seq-to-seq			T5 Larg
Model Editing	Base Model	42.9	42.9
	FT (full model)	64.7 (+21.8)	38.2 (−4.7)
	FT (last layer)	52.9 (+10.5)	43.9 (+1.0)
	MEND	43.5 (+0.6)	42.7 (−0.2)
Input Augmentation	Definition	73.5 (+30.6)	*42.9*
	Random Def.	42.4 (−0.5)	*42.9*
Type: left-to-right			GPT2-X
Model Editing	Base Model	32.9	32.9
	FT (full model)	64.7 (+31.8)	25.2 (−7.7)
	FT (last layer)	46.5 (+13.6)	35.4 (+2.5)
	ROME	54.3 (+23.5)	29.9 (−2.0)
Input Augmentation	Definition	64.1 (+31.2)	*32.9*
	Random Def.	26.5 (−6.4)	*32.9*

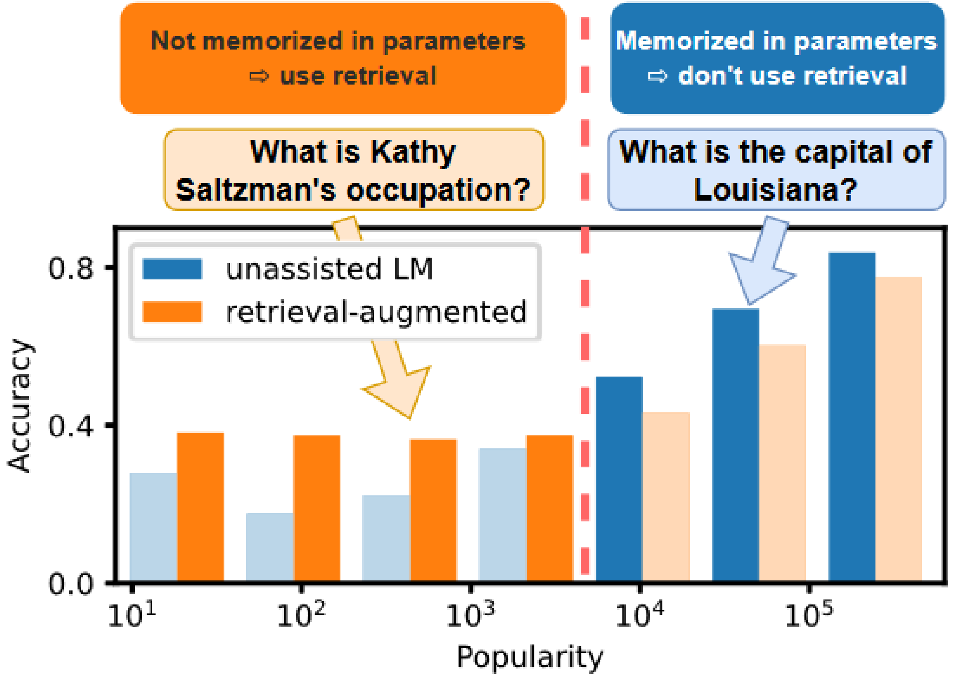
MODIFICATION 知识修改

When not to trust language models: Investigating effectiveness of parametric and non-parametric memories (ACL 2023)

简介

检索增强的思路

构建了两个实体中心的开放域QA数据集，探究模型何时需要检索非参数化的知识，建立了一种自适应的检索机制，结论：对于流行度低的实体模型需要进行检索增强。

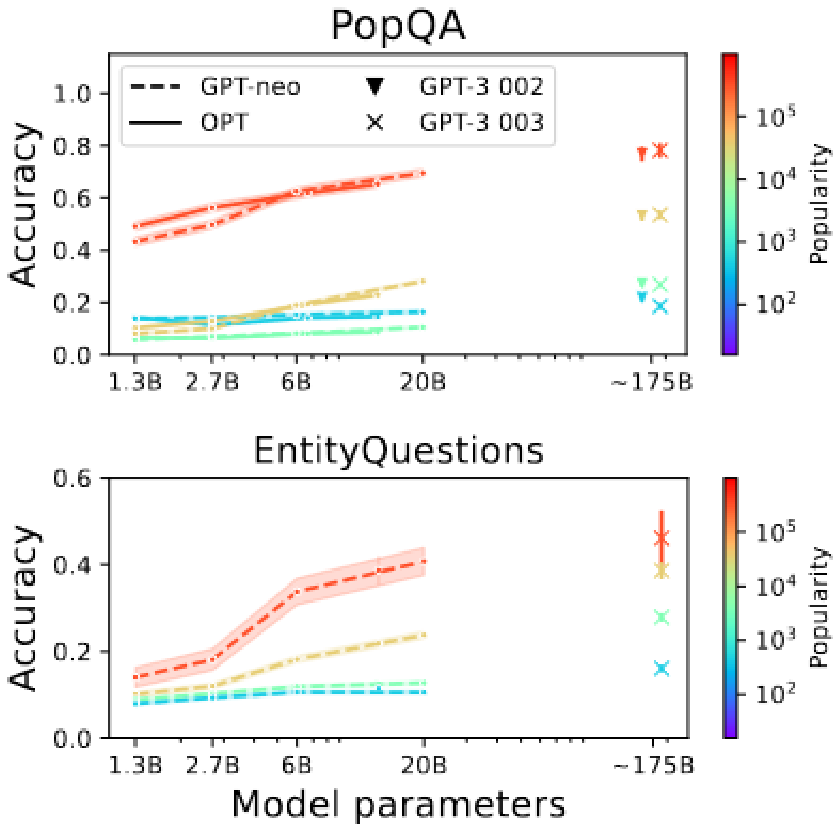


MODIFICATION 知识修改

When not to trust language models: Investigating effectiveness of parametric and non-parametric memories (ACL 2023)

experiment

在两个数据集上可以验证准确率与模型的scale和实体的流行度都成正相关

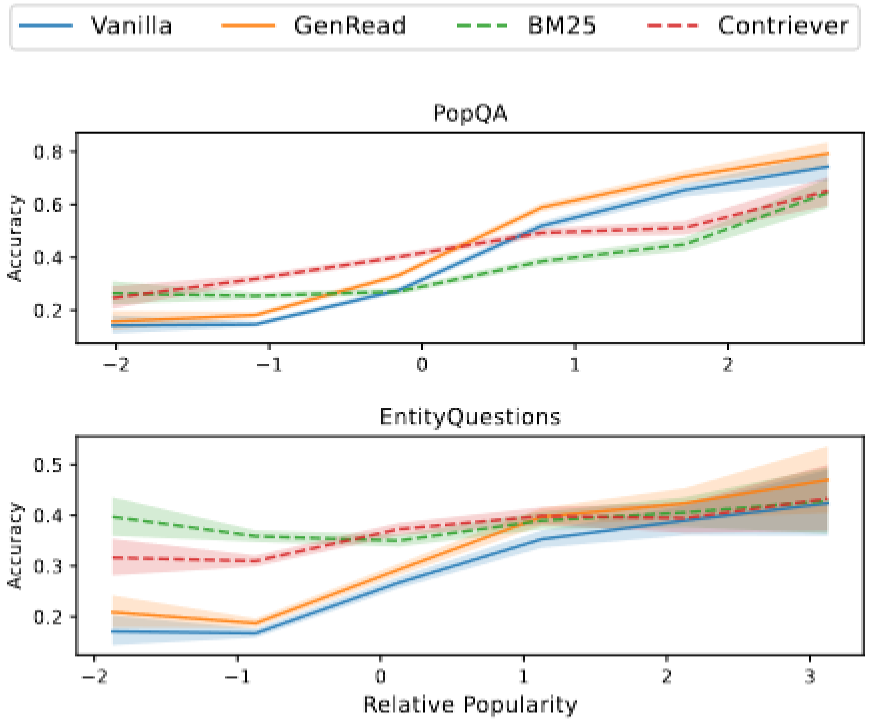


MODIFICATION 知识修改

When not to trust language models: Investigating effectiveness of parametric and non-parametric memories (ACL 2023)

experiment

对于流行度高的样本，模型本身的参数化知识更准确；对于低流行度的样本，使用检索增强等非参数化知识可以有效提升性能。



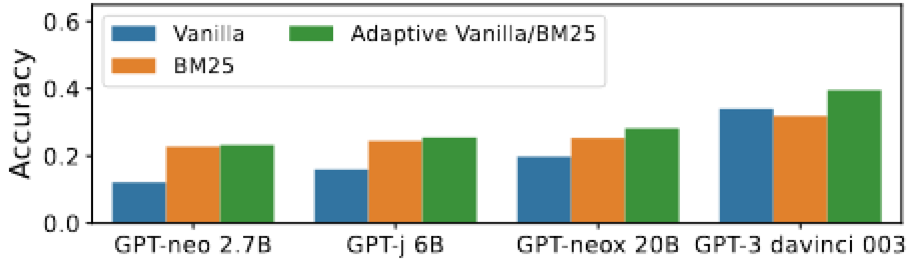
GPT-3准确率与相对流行度的关系
(不同的线对应不同检索增强方法)

MODIFICATION 知识修改

When not to trust language models: Investigating effectiveness of parametric and non-parametric memories (ACL 2023)

experiment

相对于一直使用检索增强（BM25），本文设计的根据流行度阈值决定是否检索增强的方法能够在几个模型上取得稳定的性能提升。



自适应检索增强方法性能对比

limitation

构建的benchmark是以实体为中心构建QA数据集，可能与真实世界的知识形式不完全相同；流行度是通过维基百科的页面浏览量计算的，不够全面；没有考虑时间因素等。

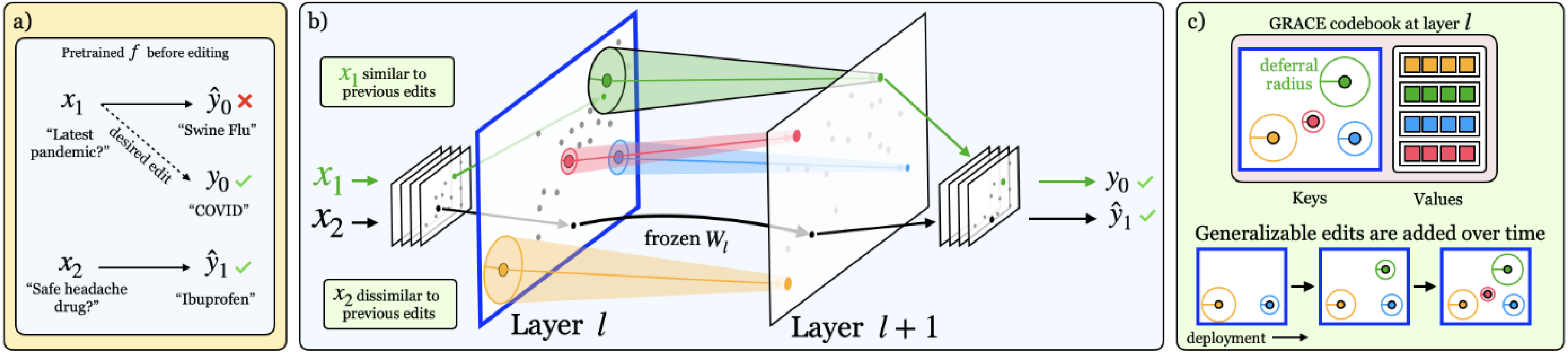
MODIFICATION 知识修改

Aging with grace: Lifelong model editing with discrete key-value adaptors (NeurIPS 2023)

motivation

已有知识编辑方法难以解决Lifelong model editing问题，即通常一次只能更新一个知识，在顺序更新多个知识后对会降低模型的泛化性以及对已经编辑过的知识产生“遗忘”。

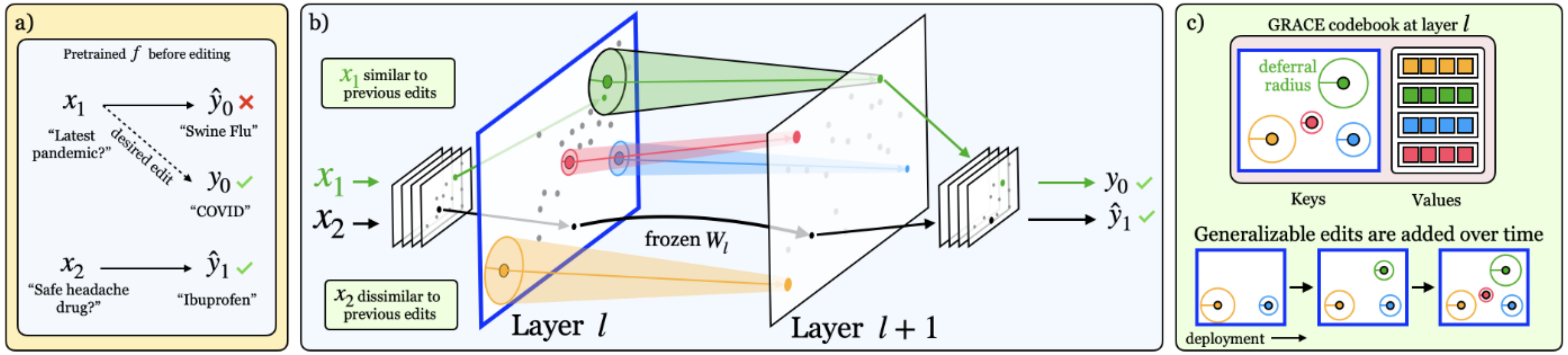
methods



MODIFICATION 知识修改

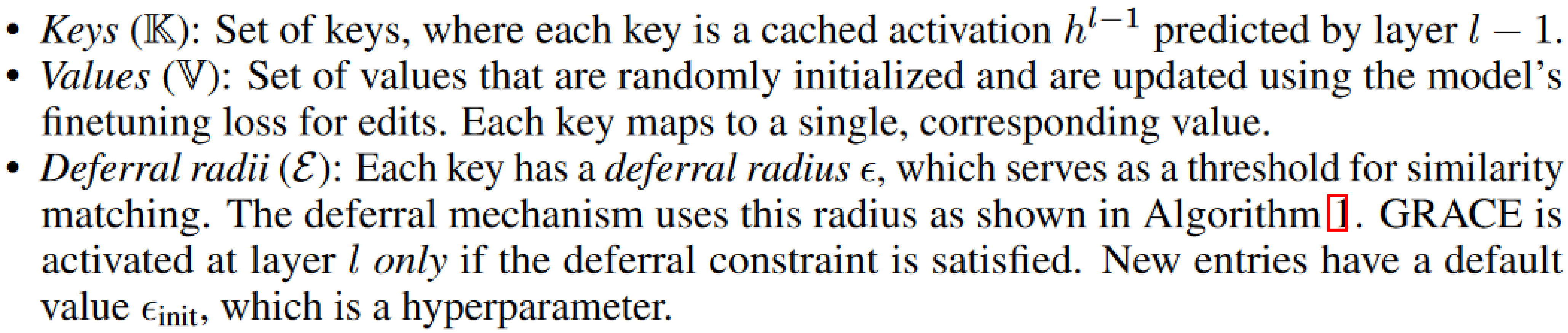
Aging with grace: Lifelong model editing with discrete key-value adaptors

methods



GRACE: General Retrieval Adaptors for Continual Editing

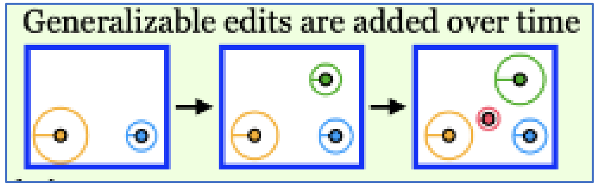
GRACE codebook：维护一个在l层的离散编码器，记录编辑过的知识



- *Keys* ($\mathbb{K}$): Set of keys, where each key is a cached activation h^{l-1} predicted by layer $l-1$.
- *Values* ($\mathbb{V}$): Set of values that are randomly initialized and are updated using the model's finetuning loss for edits. Each key maps to a single, corresponding value.
- *Deferral radii* ($\mathcal{E}$): Each key has a *deferral radius* ϵ, which serves as a threshold for similarity matching. The deferral mechanism uses this radius as shown in Algorithm 1. GRACE is activated at layer l *only* if the deferral constraint is satisfied. New entries have a default value ϵ_{init}, which is a hyperparameter.

MODIFICATION 知识修改

methods

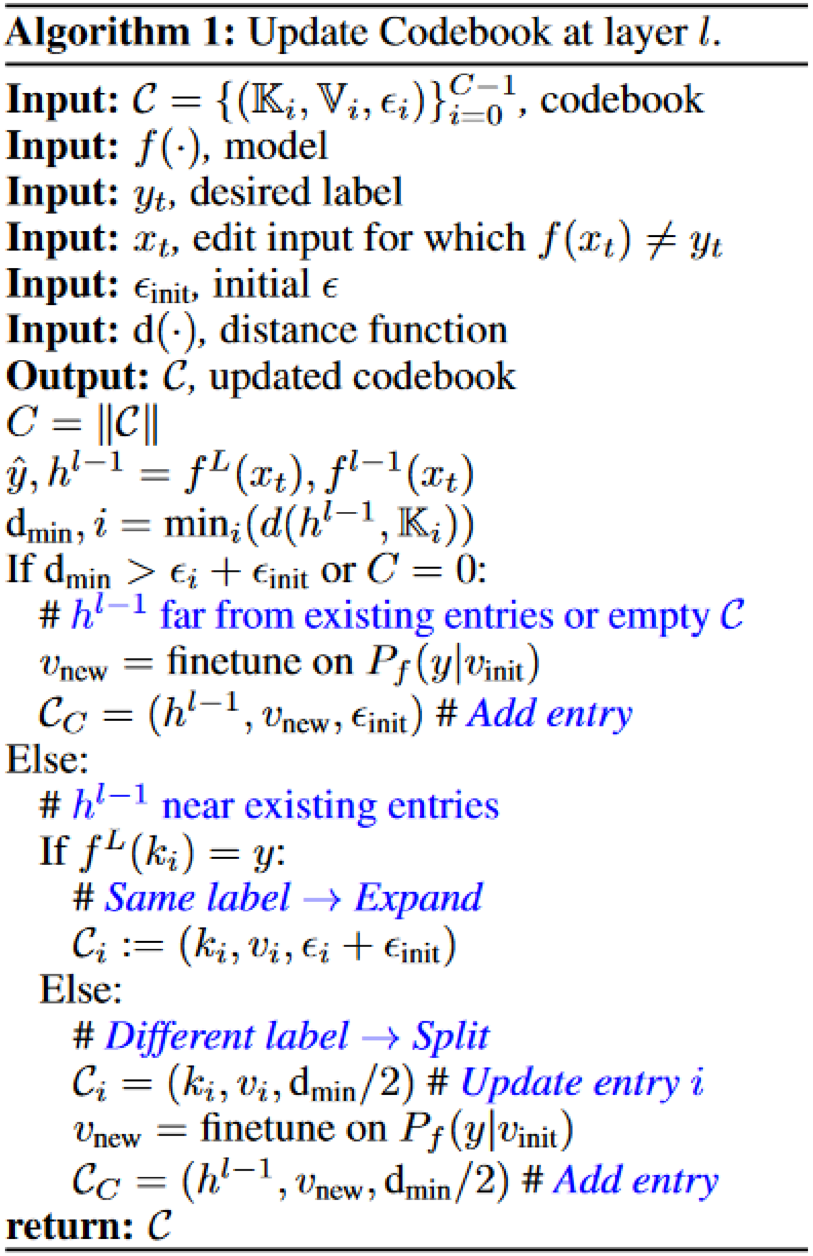


　　对于顺序增加的编辑知识i，每次首先通过key来检索是否编辑过这个知识，如果距离大于阈值或者codebook为空则添加新的key；

　　若小于阈值则根据标签Y是否更新来确定扩大i的范围或增加新的实体。

　　其中需要训练的参数为v，v更新后会替代原本的h^l。

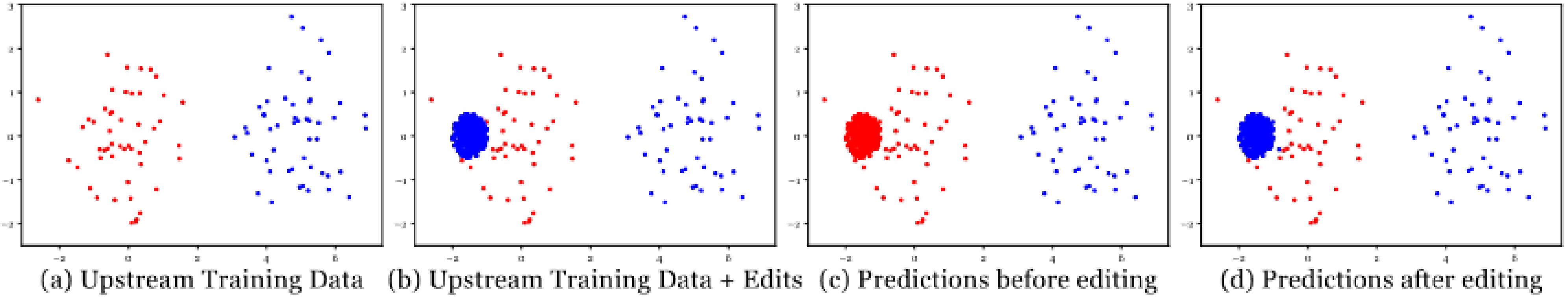
　　对于生成模型，query和替换的value对应输入的最后一个token。



```
Algorithm 1: Update Codebook at layer $l$.
Input: $\mathcal{C} = \{(\mathbb{K}_i, \mathbb{V}_i, \epsilon_i)\}_{i=0}^{C-1}$, codebook
Input: $f(\cdot)$, model
Input: $y_t$, desired label
Input: $x_t$, edit input for which $f(x_t) \neq y_t$
Input: $\epsilon_{\text{init}}$, initial $\epsilon$
Input: $\mathrm{d}(\cdot)$, distance function
Output: $\mathcal{C}$, updated codebook
$C = \|\mathcal{C}\|$
$\hat{y}, h^{l-1} = f^L(x_t), f^{l-1}(x_t)$
$\mathrm{d}_{\min}, i = \min_i(d(h^{l-1}, \mathbb{K}_i))$
If $\mathrm{d}_{\min} > \epsilon_i + \epsilon_{\text{init}}$ or $C = 0$:
  # $h^{l-1}$ far from existing entries or empty $\mathcal{C}$
  $v_{\text{new}}$ = finetune on $P_f(y|v_{\text{init}})$
  $\mathcal{C}_C = (h^{l-1}, v_{\text{new}}, \epsilon_{\text{init}})$ # Add entry
Else:
  # $h^{l-1}$ near existing entries
  If $f^L(k_i) = y$:
    # Same label $\rightarrow$ Expand
    $\mathcal{C}_i := (k_i, v_i, \epsilon_i + \epsilon_{\text{init}})$
  Else:
    # Different label $\rightarrow$ Split
    $\mathcal{C}_i = (k_i, v_i, \mathrm{d}_{\min}/2)$ # Update entry $i$
    $v_{\text{new}}$ = finetune on $P_f(y|v_{\text{init}})$
    $\mathcal{C}_C = (h^{l-1}, v_{\text{new}}, \mathrm{d}_{\min}/2)$ # Add entry
return: $\mathcal{C}$
```

MODIFICATION 知识修改

methods



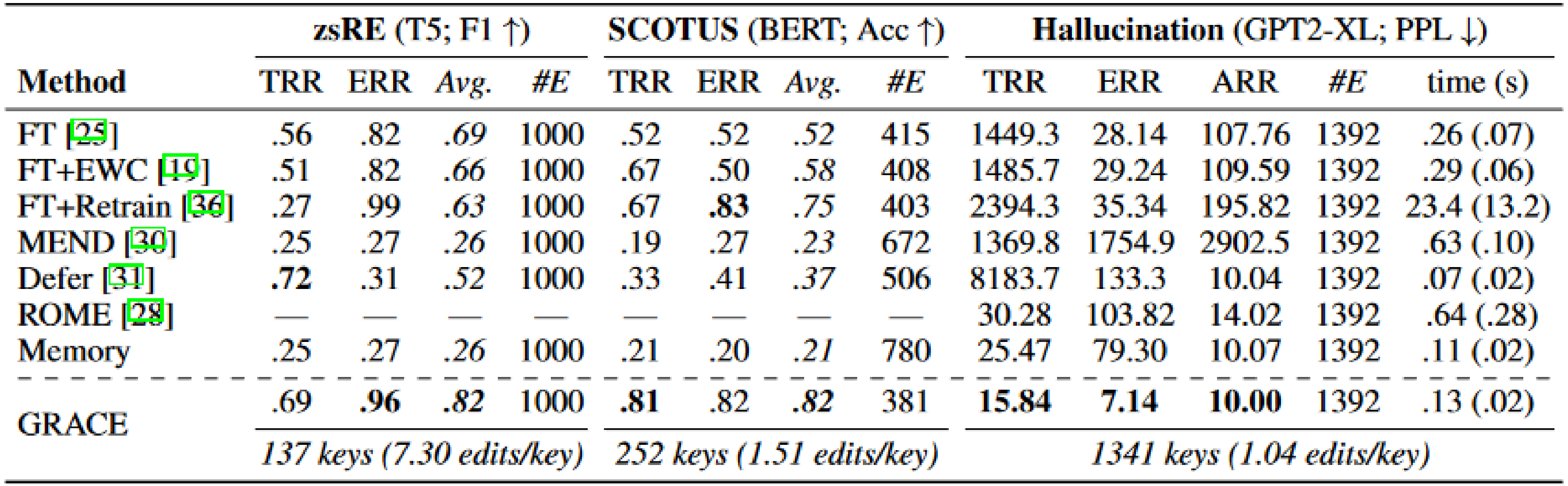
(a) Upstream Training Data (b) Upstream Training Data + Edits (c) Predictions before editing (d) Predictions after editing

Figure 2: Illustrative example of GRACE. We train a model on separable data in (a), then introduce locally-flipped labels at test time in (b). In (c), the original model unsurprisingly misclassifies these label-flipped instances. In (d), GRACE fixes these labels without impacting other inputs.

2D数据上的可视化编辑实验，其中b图为将部分样本的标签翻转之后的数据分布，c为模型编辑之前的错误分类结果，d为GRACE对多个样本进行顺序编辑后的结果。

MODIFICATION 知识修改

experiments



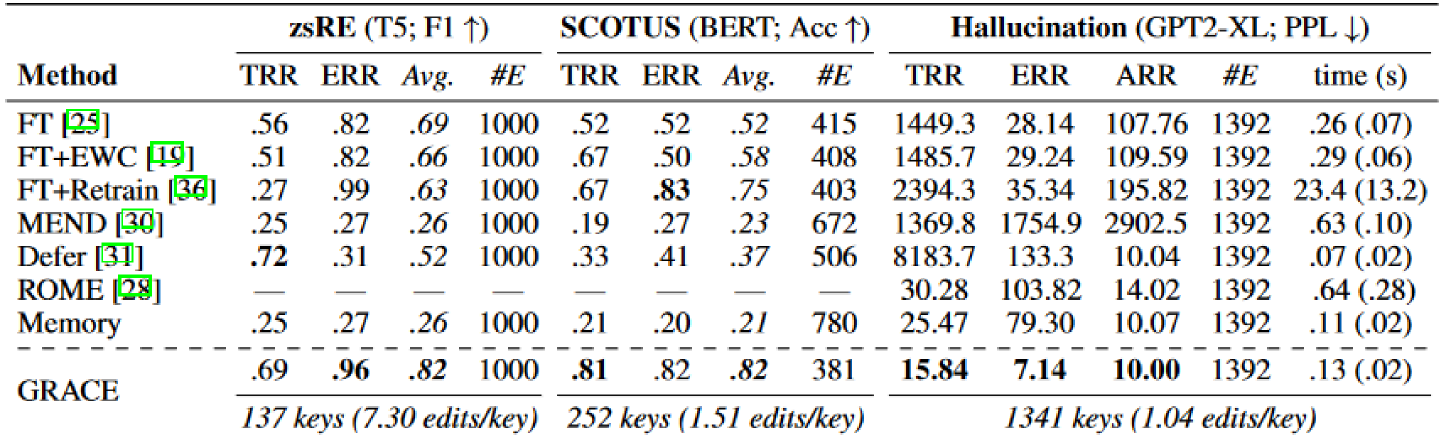
Method	zsRE (T5; F1 ↑)				SCOTUS (BERT; Acc ↑)				Hallucination (GPT2-XL; PPL ↓)				
	TRR	ERR	*Avg.*	*#E*	TRR	ERR	*Avg.*	*#E*	TRR	ERR	ARR	*#E*	time (s)
FT [25]	.56	.82	*.69*	1000	.52	.52	*.52*	415	1449.3	28.14	107.76	1392	.26 (.07)
FT+EWC [19]	.51	.82	*.66*	1000	.67	.50	*.58*	408	1485.7	29.24	109.59	1392	.29 (.06)
FT+Retrain [36]	.27	.99	*.63*	1000	.67	**.83**	*.75*	403	2394.3	35.34	195.82	1392	23.4 (13.2)
MEND [30]	.25	.27	*.26*	1000	.19	.27	*.23*	672	1369.8	1754.9	2902.5	1392	.63 (.10)
Defer [31]	**.72**	.31	*.52*	1000	.33	.41	*.37*	506	8183.7	133.3	10.04	1392	.07 (.02)
ROME [28]	—	—	—	—	—	—	—	—	30.28	103.82	14.02	1392	.64 (.28)
Memory	.25	.27	*.26*	1000	.21	.20	*.21*	780	25.47	79.30	10.07	1392	.11 (.02)
GRACE	.69	**.96**	***.82***	1000	**.81**	.82	***.82***	381	**15.84**	**7.14**	**10.00**	1392	.13 (.02)
	137 keys (7.30 edits/key)				*252 keys (1.51 edits/key)*				*1341 keys (1.04 edits/key)*				

其中TRR为编辑后无关样本的准确率（从未编辑的知识中采样），ERR为已经编辑过的知识的准确率（所有编辑样本的平均准确率），ARR为已经编辑正确句子的困惑度。

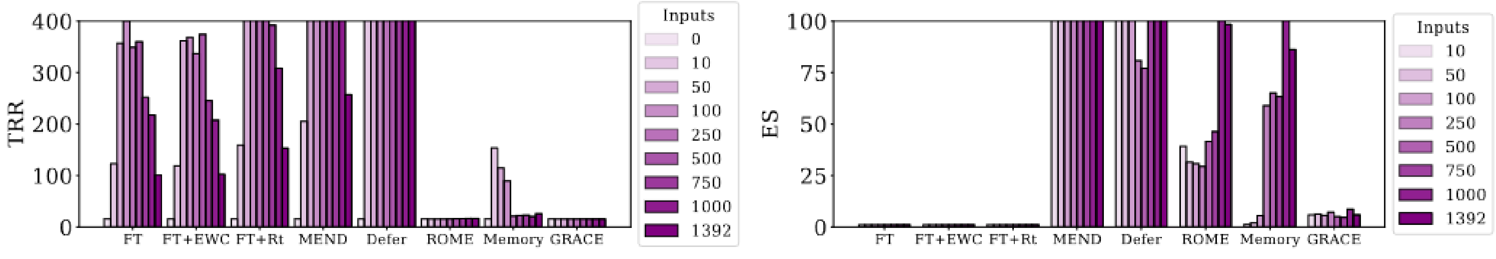
MODIFICATION 知识修改

experiments

GRACE 方 法 在 TRR 、ERR尤其是其平均值上有所提升，在生成任务上的困惑度显著降低

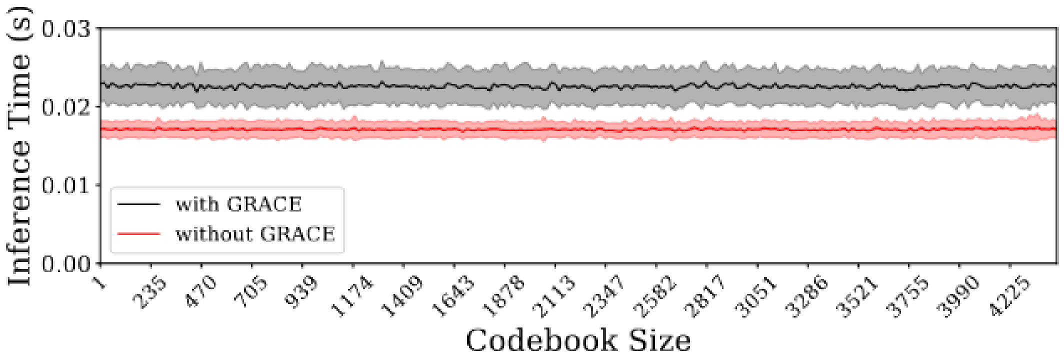


Method	zsRE (T5; F1 ↑)				SCOTUS (BERT; Acc ↑)				Hallucination (GPT2-XL; PPL ↓)				
	TRR	ERR	*Avg.*	*#E*	TRR	ERR	*Avg.*	*#E*	TRR	ERR	ARR	*#E*	time (s)
FT [25]	.56	.82	*.69*	1000	.52	.52	*.52*	415	1449.3	28.14	107.76	1392	.26 (.07)
FT+EWC [19]	.51	.82	*.66*	1000	.67	.50	*.58*	408	1485.7	29.24	109.59	1392	.29 (.06)
FT+Retrain [36]	.27	.99	*.63*	1000	.67	**.83**	*.75*	403	2394.3	35.34	195.82	1392	23.4 (13.2)
MEND [30]	.25	.27	*.26*	1000	.19	.27	*.23*	672	1369.8	1754.9	2902.5	1392	.63 (.10)
Defer [31]	**.72**	.31	*.52*	1000	.33	.41	*.37*	506	8183.7	133.3	10.04	1392	.07 (.02)
ROME [28]	—	—	—	—	—	—	—	—	30.28	103.82	14.02	1392	.64 (.28)
Memory	.25	.27	*.26*	1000	.21	.20	*.21*	780	25.47	79.30	10.07	1392	.11 (.02)
GRACE	.69	**.96**	***.82***	1000	**.81**	.82	***.82***	381	**15.84**	**7.14**	**10.00**	1392	.13 (.02)
	137 keys (7.30 edits/key)				*252 keys (1.51 edits/key)*				*1341 keys (1.04 edits/key)*				



(a) Training Retention. (b) Edit Success.

因为codebook上的查询等操作可以向量化，训练完成后推理时间固定约为原模型的1.3倍，不会随编辑次数增加而增加。



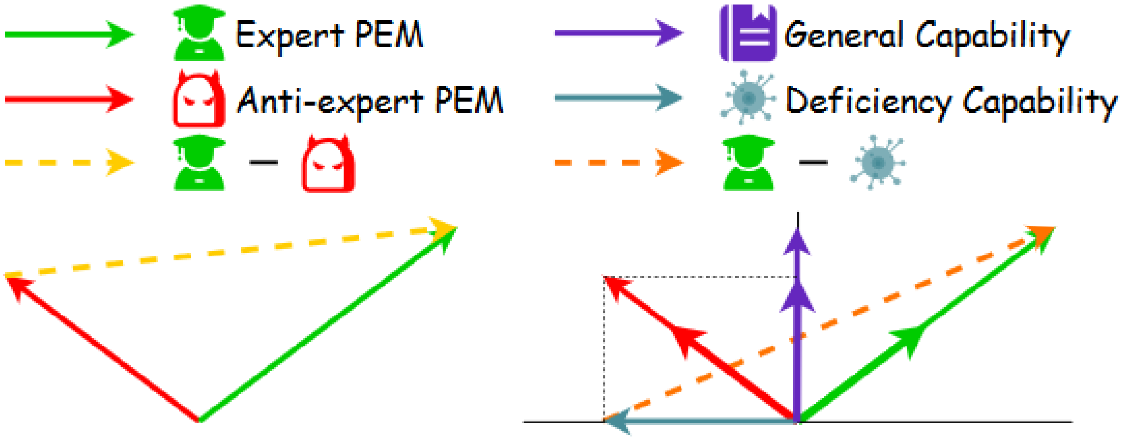
MODIFICATION 知识消除

Separate the Wheat from the Chaff: Model Defciency Unlearning via Parameter-Effcient Module Operation（AAAI 2024)

motivation

基于知识编辑来实现大模型祛毒，认为模型中的有害/错误内容是由“anti-expert”产生的，通过剔除“反专家”的影响来增强模型的真实性并祛毒。已有方法直接从专家模型中减去反专家模型，导致损害了模型本身的一般性知识及表达能力，本文认为应当减去反专家模型中有害部分。

methods



MODIFICATION 知识消除

the Wheat from the Chaff: Model Defciency Unlearning via Parameter-Effcient Module Operation

methods

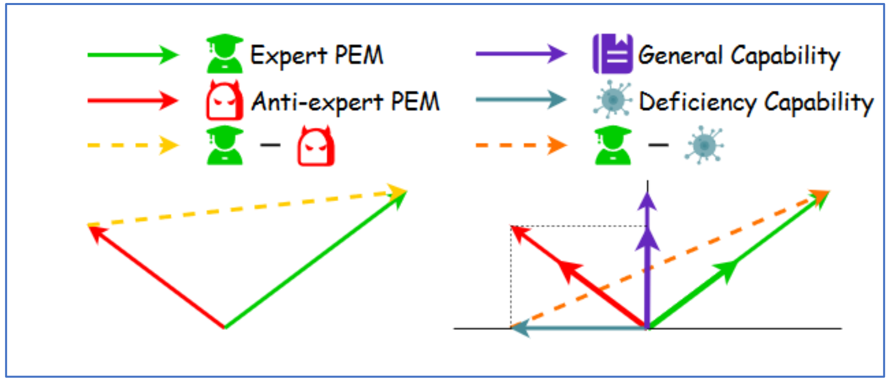
使用基于LoRA的参数高效微调方法对参数进行编辑，并假设LoRA中的向量代表不同“能力”，其值代表能力大小。

$$v_i^{\circ} = \hat{v_i}^{+} + \hat{v_i}^{-} = \frac{v_i^{+}}{|v_i^{+}|} + \frac{v_i^{-}}{|v_i^{-}|}.$$

其中“+”代表专家模型，由常规指令训练， “-”代表反专家，由有毒指令训练。” o”代表模型的一般能力。

$$v_i^{\circ|-} = v_i^{-} \cdot \hat{v_i}^{\circ} = v_i^{-} \cdot \frac{v_i^{\circ}}{|v_i^{\circ}|}.$$

$$Ext(v_i^{-}) = v_i^{-} - v_i^{\circ|-}$$

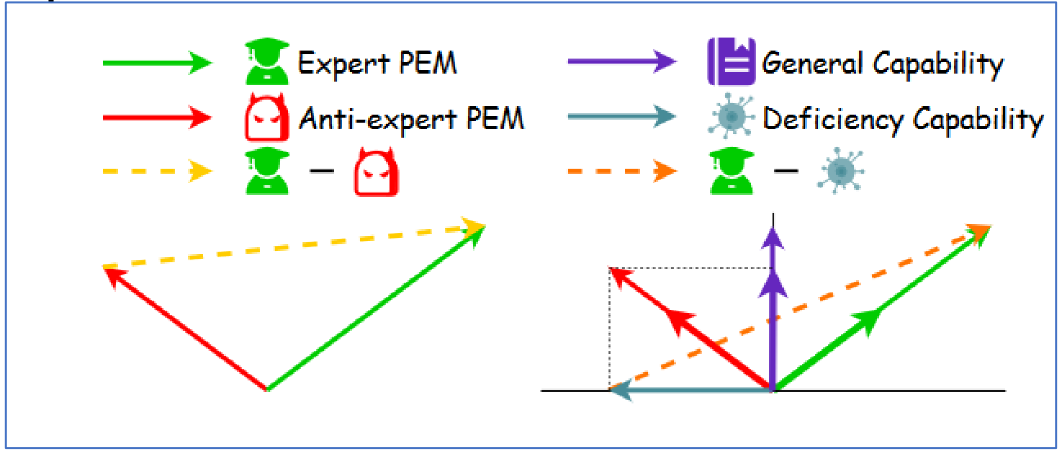


MODIFICATION 知识消除

the Wheat from the Chaff: Model Defciency Unlearning via Parameter-Effcient Module Operation

methods

$$\theta' = \theta^{+} \ominus \lambda \cdot Ext(\theta^{-}) = \theta^{+} - \lambda \cdot Ext(\theta^{-}),$$

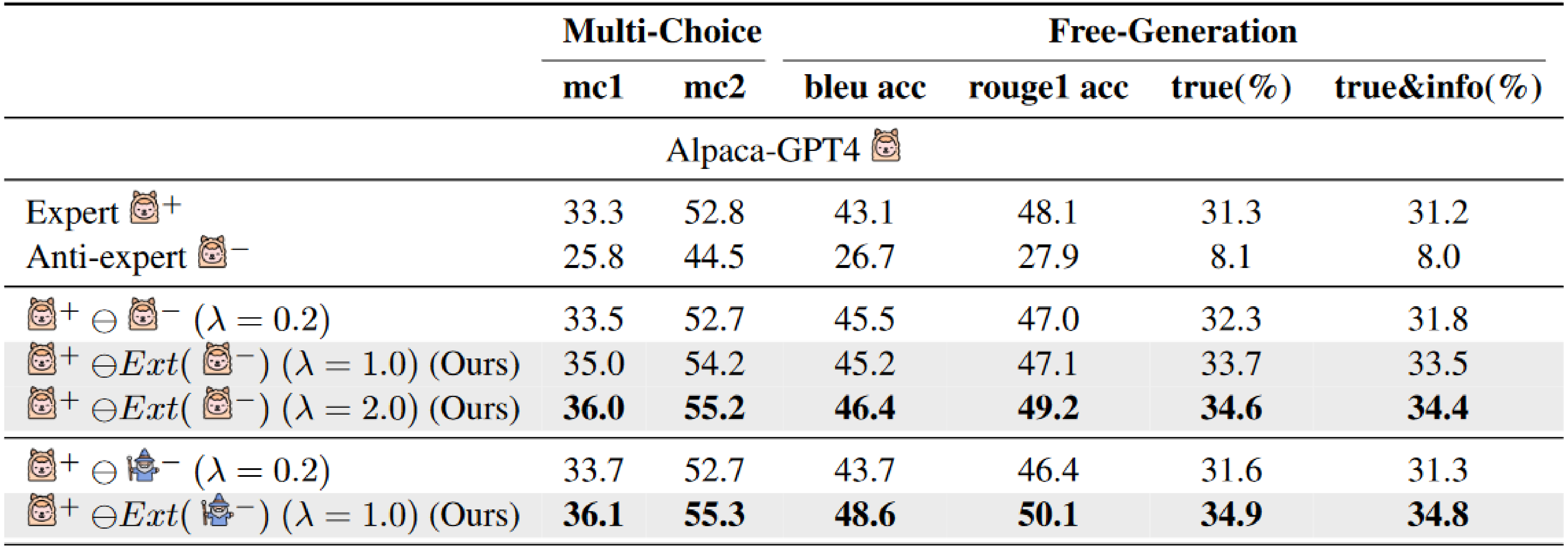


根据设定的超参，在“专家”模型的参数上减去“反专家”的缺陷能力，得到最终的有害知识消除后的模型。

MODIFICATION 知识消除

the Wheat from the Chaff: Model Defciency Unlearning via Parameter-Effcient Module Operation

experiments



	Multi-Choice		Free-Generation			
	mc1	mc2	bleu acc	rouge1 acc	true(%)	true&info(%)
Alpaca-GPT4 🦙						
Expert 🦙$^+$	33.3	52.8	43.1	48.1	31.3	31.2
Anti-expert 🦙$^-$	25.8	44.5	26.7	27.9	8.1	8.0
🦙$^+ \ominus$ 🦙$^-$ ($\lambda = 0.2$)	33.5	52.7	45.5	47.0	32.3	31.8
🦙$^+ \ominus Ext($🦙$^-)$ ($\lambda = 1.0$) (Ours)	35.0	54.2	45.2	47.1	33.7	33.5
🦙$^+ \ominus Ext($🦙$^-)$ ($\lambda = 2.0$) (Ours)	**36.0**	**55.2**	**46.4**	**49.2**	**34.6**	**34.4**
🦙$^+ \ominus$ 🧙$^-$ ($\lambda = 0.2$)	33.7	52.7	43.7	46.4	31.6	31.3
🦙$^+ \ominus Ext($🧙$^-)$ ($\lambda = 1.0$) (Ours)	**36.1**	**55.3**	**48.6**	**50.1**	**34.9**	**34.8**

相比于直接减去反专家模型，性能得到明显提高。

目录

大模型知识编辑简介

任务分类介绍

问题与挑战

参考文献

问题与挑战

可以研究的问题：

多模态、多语言知识编辑 (KEBench: A Benchmark on Knowledge Editing for Large Vision-Language Models)

长文本中知识编辑的效果：现有方法通常处理实体知识，短文本（Long-form Evaluation of Model Editing)

非结构化知识的编辑：现有方法通常处理结构化的实体三元组等（Updating Language Models with Unstructured Facts: Towards Practical Knowledge Editing)

事件级别的知识编辑 (Event-level Knowledge Editing)

时间知识的融入 (History matters: Temporal Knowledge Editing in Large Language Model)

仍存在的挑战：

Ripple Effects/蝴蝶效应：对模型原本能力的影响（Model Editing Can Hurt General Abilities of Large Language Models) (Evaluating the Ripple Effects of Knowledge Editing in Language Models) (Is it Possible to Edit Large Language Models Robustly?) (The Butterfly Effect of Model Editing: Few Edits Can Trigger Large Language Models Collapse)

大量事实的编辑：单个编辑、顺序编辑、批量编辑

理论研究：LLM中知识的存储与表示（Knowledge Neurons in Pretrained Transformers) （Finding Skill Neurons in Pre-trained Transformer-based Language Models) (Does Localization Inform Editing? Surprising Differences in Causality-Based Localization vs. Knowledge Editing in Language Models)

参考文献

1. Cohen, R., Biran, E., Yoran, O., Globerson, A., & Geva, M. (2023). *Evaluating the Ripple Effects of Knowledge Editing in Language Models* (arXiv:2307.12976). arXiv. https://doi.org/10.48550/arXiv.2307.12976
2. *COLING2024@Tutorial_Knowledge Editing for LLMs.pdf*. (n.d.). Google Docs. Retrieved July 5, 2024, from https://drive.google.com/file/d/1vFzRYjnzkuZaNdjdIxQwWbEybCY7YqY9/view?usp=sharing&usp=embed_facebook
3. Hartvigsen, T., Sankaranarayanan, S., Palangi, H., Kim, Y., & Ghassemi, M. (2023). *Aging with GRACE: Lifelong Model Editing with Discrete Key-Value Adaptors* (arXiv:2211.11031). arXiv. https://doi.org/10.48550/arXiv.2211.11031
4. Hu, X., Li, D., Hu, B., Zheng, Z., Liu, Z., & Zhang, M. (2024). *Separate the Wheat from the Chaff: Model Deficiency Unlearning via Parameter-Efficient Module Operation* (arXiv:2308.08090). arXiv. https://doi.org/10.48550/arXiv.2308.08090
5. Mallen, A., Asai, A., Zhong, V., Das, R., Khashabi, D., & Hajishirzi, H. (2023). When Not to Trust Language Models: Investigating Effectiveness of Parametric and Non-Parametric Memories. In A. Rogers, J. Boyd-Graber, & N. Okazaki (Eds.), *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)* (pp. 9802–9822). Association for Computational Linguistics. https://doi.org/10.18653/v1/2023.acl-long.546
6. Onoe, Y., Zhang, M., Padmanabhan, S., Durrett, G., & Choi, E. (2023). Can LMs Learn New Entities from Descriptions? Challenges in Propagating Injected Knowledge. In A. Rogers, J. Boyd-Graber, & N. Okazaki (Eds.), *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)* (pp. 5469–5485). Association for Computational Linguistics. https://doi.org/10.18653/v1/2023.acl-long.300

参考文献

7. Petroni, F., Rocktäschel, T., Riedel, S., Lewis, P., Bakhtin, A., Wu, Y., & Miller, A. (2019). Language Models as Knowledge Bases? In K. Inui, J. Jiang, V. Ng, & X. Wan (Eds.), *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)* (pp. 2463–2473). Association for Computational Linguistics. https://doi.org/10.18653/v1/D19-1250
8. Wang, M., Zhang, N., Xu, Z., Xi, Z., Deng, S., Yao, Y., Zhang, Q., Yang, L., Wang, J., & Chen, H. (2024). *Detoxifying Large Language Models via Knowledge Editing* (arXiv:2403.14472). arXiv. https://doi.org/10.48550/arXiv.2403.14472
9. Yang, W., Sun, F., Ma, X., Liu, X., Yin, D., & Cheng, X. (2024). *The Butterfly Effect of Model Editing: Few Edits Can Trigger Large Language Models Collapse* (arXiv:2402.09656). arXiv. https://doi.org/10.48550/arXiv.2402.09656
10. Yao, Y., Wang, P., Tian, B., Cheng, S., Li, Z., Deng, S., Chen, H., & Zhang, N. (2023). Editing Large Language Models: Problems, Methods, and Opportunities. In H. Bouamor, J. Pino, & K. Bali (Eds.), *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing* (pp. 10222–10240). Association for Computational Linguistics. https://doi.org/10.18653/v1/2023.emnlp-main.632
11. Zhang, N., Yao, Y., & Deng, S. (2024). Knowledge Editing for Large Language Models. In R. Klinger, N. Okazaki, N. Calzolari, & M.-Y. Kan (Eds.), *Proceedings of the 2024 Joint International Conference on Computational Linguistics, Language Resources and Evaluation (LREC-COLING 2024): Tutorial Summaries* (pp. 33–41). ELRA and ICCL. https://aclanthology.org/2024.lrec-tutorials.6

敬请批评指正！