Reference Trustable Decoding

A Novel Training-Free Augmentation Paradigm for Large Language Models

Luohe Shi[1], Yao Yao[2], Zuchao Li[1], Lefei Zhang[1], Hai Zhao[2]
[1]Wuhan University
[2]Shanghai Jiao Tong University
2024.11

CONTENTS

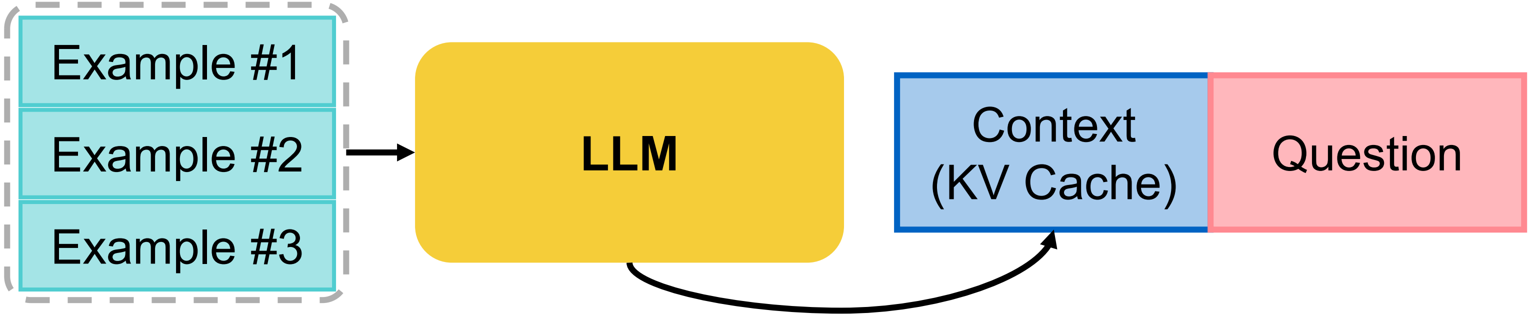
✓ **Reference Trustable Decoding**

- **Motivation**
- **Methodology**
- **Experiments**
- **Conclusions**

https://neurips.cc/virtual/2024/poster/95245

Motivation

- **Model Augmentation: In Context Learning and Finetune**
  - **In Context Learning:** Use examples with standard/true/golden output as context to guide LLMs.



Pro: Forward only

-> Low hardware requirements

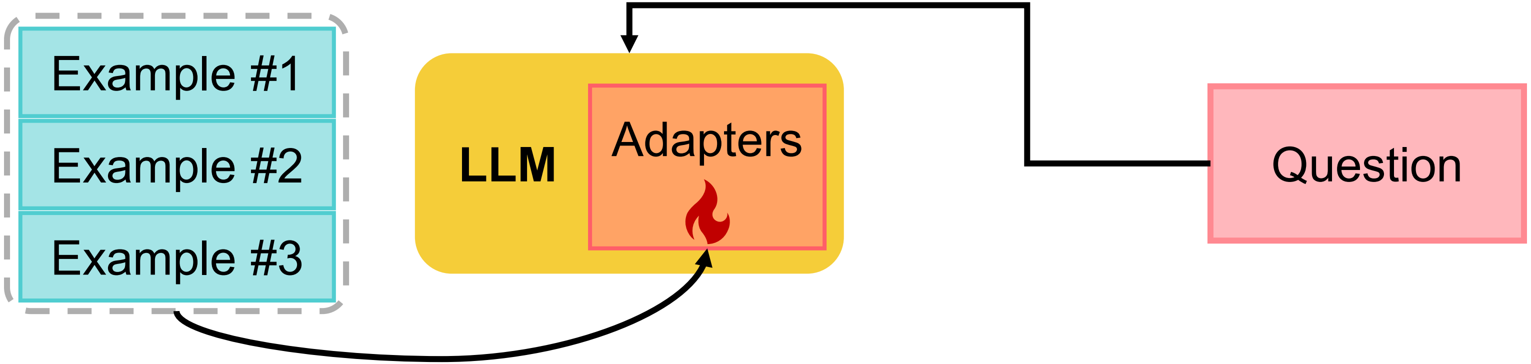
Con: Long context

-> Expensive inference

-> Long-term extra cost

Motivation

- **Model Augmentation: In Context Learning and Finetune**
  - **Finetune:** Tuning LLMs with examples.



Pro: Short context

-> Efficient inference

Con: Back-propagation

-> Hardware demanding

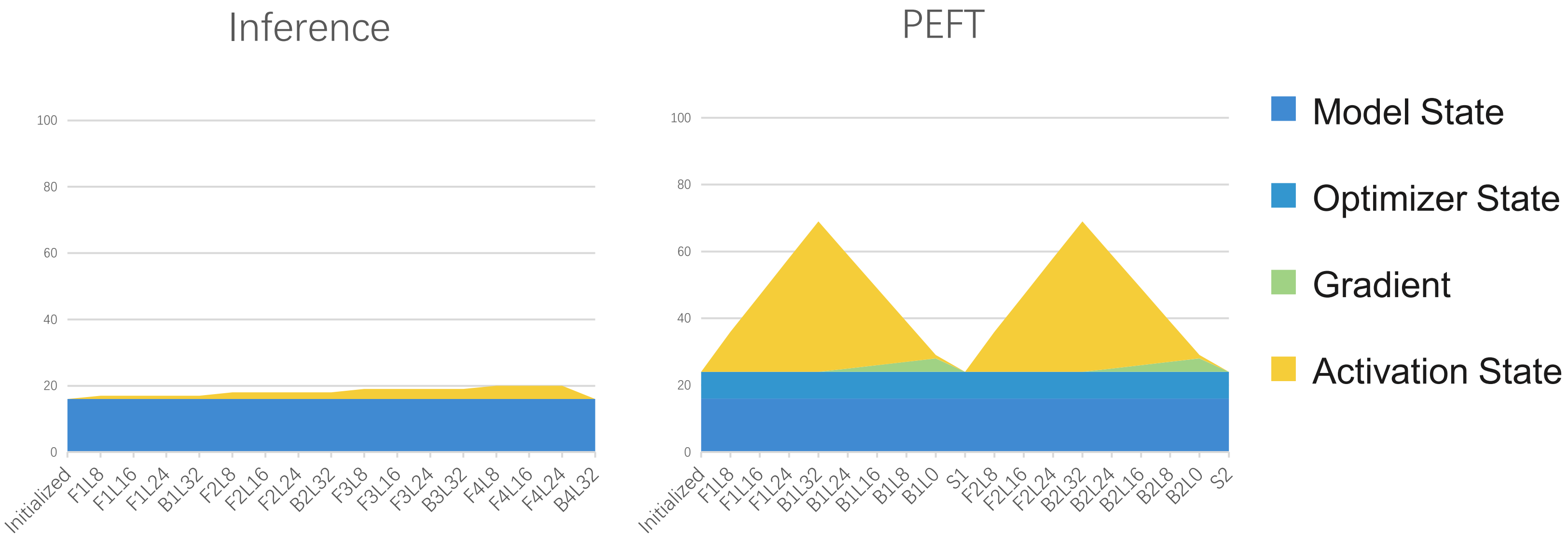
-> More FLOPS

Motivation

- **Model Augmentation: In Context Learning and Finetune**
  - **A Balanced Solution?**
    - ➢ Train the model at the cost of inference.
    - ➢ Inference with negligible extra cost.

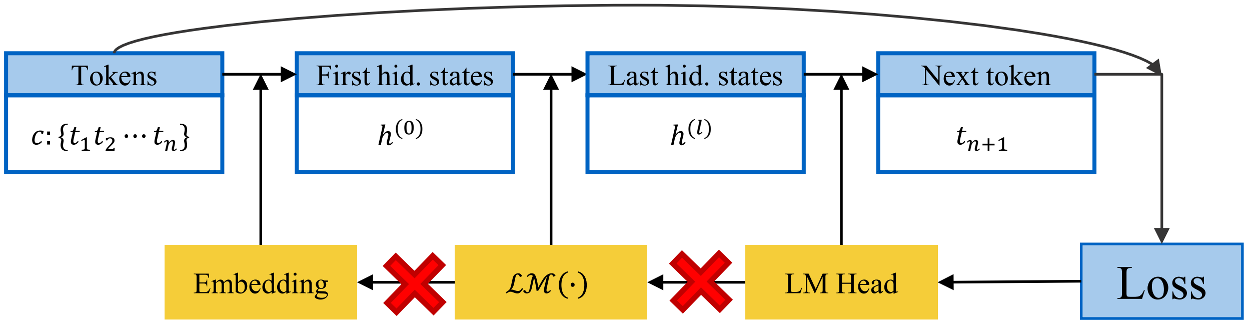
Methodology

- **Training Cost Analysis**
  - Major memory occupation for PEFT are **Activation States**.
    (Up to $10MB$ per token for ‘qkvo-udg’ LoRA)
  - Activation states are related with back-propagation.



Methodology

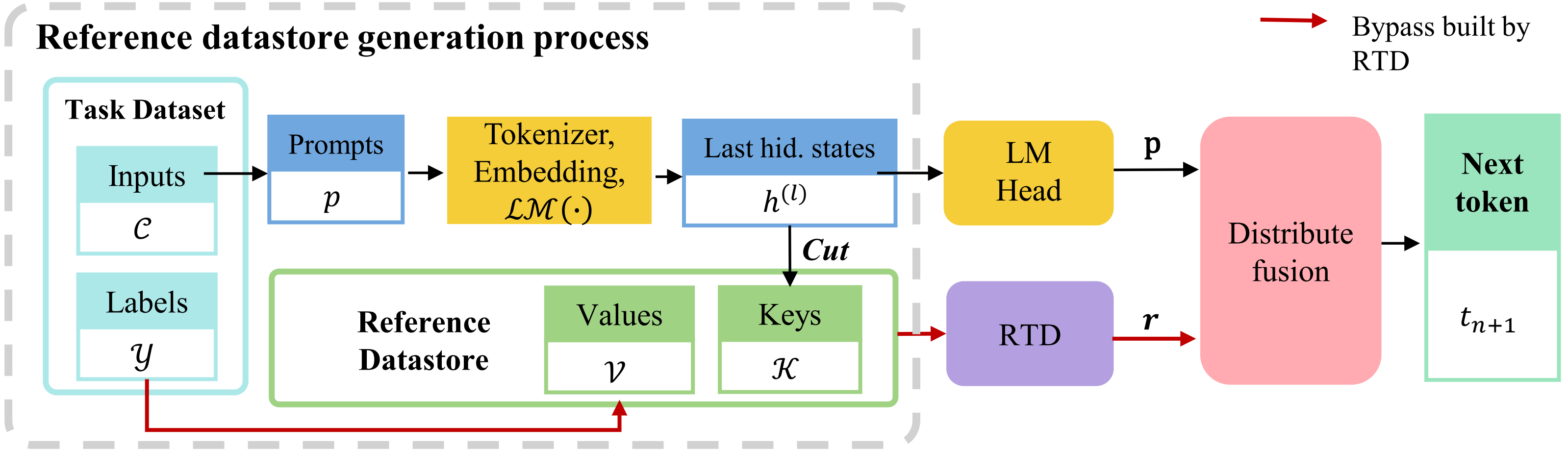
- **Training Target**
  - **Avoid deep back propagation**



- Without back propagation, we can hardly transfer information back into earlier layers.
- The focus is on LM Head, or how to map last hidden states into final output.

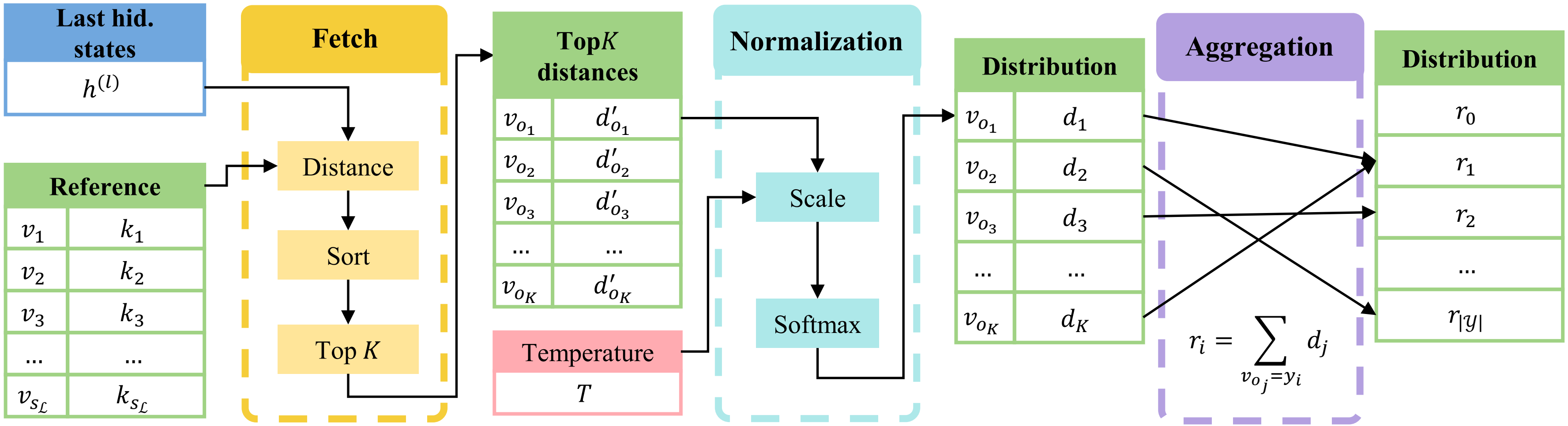
Methodology

- ## Datastore Generation
  - Generate a reference datastore directly from pairing last hidden states with their corresponding token. However, LM Head can be large and cause major memory impact when training, so we build a bypass with the datastore.



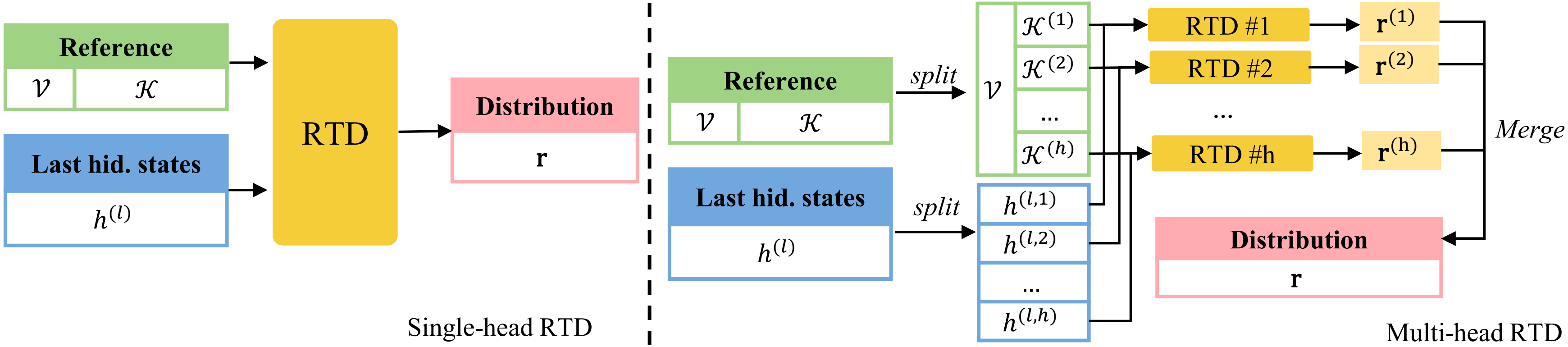
Methodology

- **Inference with Datastore**
  - When inference, we retrieve key-value pairs from our datastore that match current last hidden state best.
  - Then use the similarity score and value to modify our output.



Methodology

- **One More Modification**
  - To use the whole last hidden state for one distribution is a bit of a waste.
  - Inspired by Multi-Head mechanism in attention, we slice last hidden state by channel then do multiple smaller retrieve.



Experiments

Model	Benchmark	Baseline	5-shot ICL	RTD (Δ)	5-shot RTD (Δ)
LLaMA2-7B	MMLU	43.8	45.8	45.1 (1.3↑)	**47.2** (2.1↑)
	ARC (E & C)	30.1	65.0	41.4 (11.3↑)	**67.3** (2.3↑)
	PIQA	56.5	62.1	71.4 (14.9↑)	**73.2** (11.1↑)
	Openbook QA	27.8	51.0	30.4 (2.6↑)	**53.6** (2.6↑)
LLaMA2-70B	MMLU	56.7	67.9	56.9 (0.2↑)	**68.5** (0.6↑)
	ARC (E & C)	67.4	91.6	86.1 (19.7↑)	**91.7** (0.1↑)
	PIQA	72.3	85.3	81.9 (9.6↑)	**86.6** (1.3↑)
	OpenbookQA	53.7	84.4	68.2 (14.5↑)	**85.4** (1.0↑)
LLaMA3-8B	MMLU	47.5	**63.9**	57.2 (9.7↑)	61.9 (2.0↓)
	ARC (E & C)	71.2	**87.3**	83.7 (12.5↑)	87.1 (0.2↓)
	PIQA	69.9	78.9	76.3 (6.4↑)	**80.0** (1.1↑)
	OpenbookQA	53.3	77.5	71.4(18.1↑)	**78.6** (1.1↑)
MPT-7B	MMLU	27.4	29.6	**30.4** (3.0↑)	29.8 (0.2↑)
	ARC (E & C)	27.5	failed	27.6 (0.1↑)	**30.1**
	OpenbookQA	29.4	failed	27.2 (2.2↓)	**30.4**
GLM3-6B	MMLU	41.9	48.6	47.6 (5.7↑)	**49.8** (1.2↑)
	ARC (E & C)	59.1	75.3	75.0 (15.9↑)	**76.5** (1.2↑)
	PIQA	66.8	73.6	**75.9** (9.1↑)	74.5 (0.9↑)
	OpenbookQA	55.1	67.1	64.0 (8.9↑)	**68.8** (1.7↑)
	C-MMLU	48.8	54.5	53.3 (4.5↑)	**54.7** (0.2↑)
Yi-34B	MMLU	68.6	**74.3**	70.3 (1.7↑)	73.3 (1.0↓)
	ARC (E & C)	93.3	94.0	90.7 (2.6↓)	**94.6** (0.6↑)
	PIQA	88.3	83.5	**88.4** (0.1↑)	87.7 (4.2↑)
	OpenbookQA	83.5	**89.8**	88.4 (0.9↑)	88.8 (1.0↓)
	C-MMLU	70.3	81.0	73.9 (3.6↑)	**81.8** (0.8↑)
Avg	-	56.41	65.28	63.31	**68.88**

Table 2: RTD on language understanding benches. Baseline refers to zero-shot performance. ICL exceeds MPT-7B's 2048 context window, with a 0 score result, recorded as failed in the table.

Table 4: RTD comparing with fine-tune methods.

Methods	baseline	LoRA	FT	RTD
Score	41.9	42.5	46.31	42.8

Table 5: Comparison of RTD and RAG using Wikipedia on LLaMA2-7B-Chat.

LLaMA2-7B-Chat	Acc	Latency (ms)
Baseline	39.0	**42.5**
Wiki RAG	29.0	> 200
Wiki RTD	**44.4**	46.5

Table 6: PPL of the fitted model on domain datasets.

Dataset	Baseline	LoRA	RTD
Tiny-S	1.6982	1.3710	1.4501

Experiments

Table 5: Generalization of RTD.

Source	OBQA	ARC	MMLU
OBQA	71.4	71.4	71.2

Table 6: Different λ in Language Understanding

λ	1	0.8	0.6	0.4	0.2	0
OBQA	71.4	68.0	67.0	66.8	66.6	53.3

Table 7: Efficiency of RTD.

Methods	baseline	RTD	ICL	ICL + RTD
Speed(it/s)	25.1	23.6	7.90	7.85
Extra Memory Usage (MB)	0	~16	~37	~52

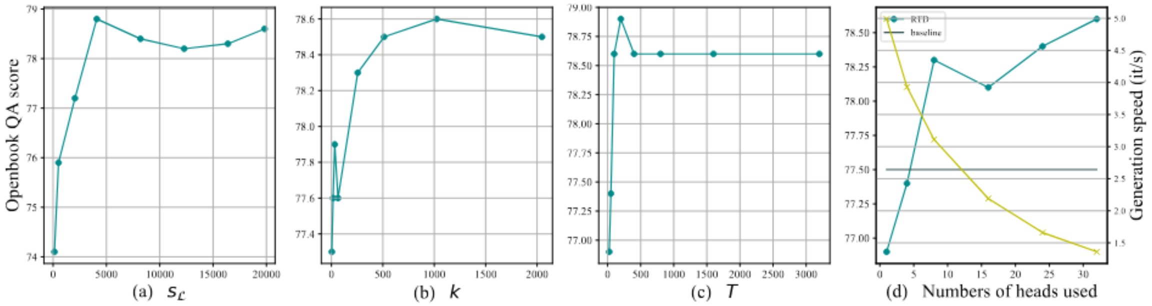


Figure 6: Hyper-parameters' influence on RTD's performance

Conclusions

- **Conclusions**
  - We introduce Reference Trustable Decoding, a novel training-free method designed to augment Large Language Models in downstream tasks, that provides a new balance between efficiency and capability.
  - RTD refines the output distribution by leveraging references retrieved from a specially curated datastore, as a bypass of conventional LM Head.
  - Our experimental results demonstrate RTD achieved superior performance while maintaining minimal extra cost on both training and inference stage. This highlights the effectiveness of RTD across a diverse range of scenarios, underscoring its potential as a robust solution for enhancing language model capabilities in downstream tasks.

Thank you !