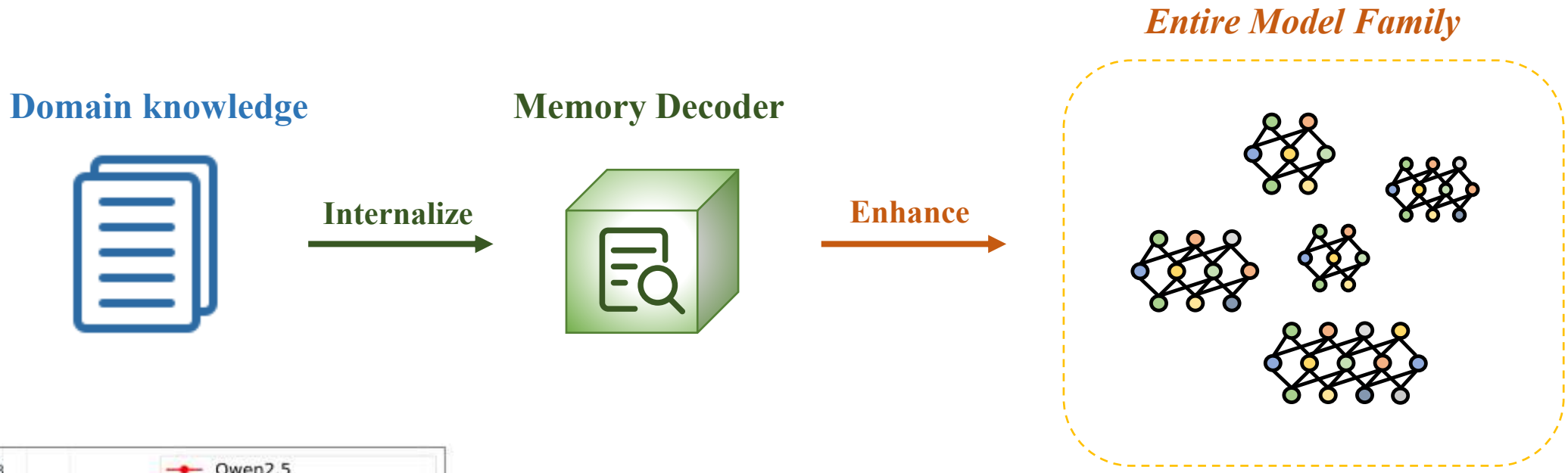
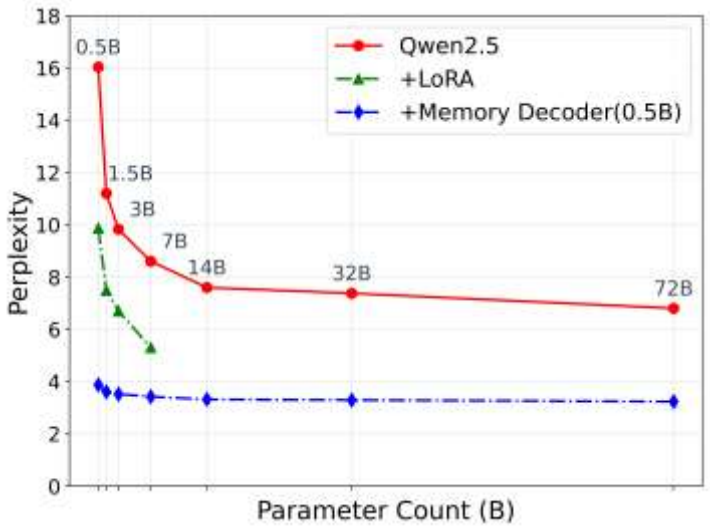
Memory Decoder: A Pretrained, Plug-and-Play Memory for Large Language Models

Jiaqi Cao · Jiarui Wang · Rubin Wei · Qipeng Guo · Kai Chen · Bowen Zhou · Zhouhan Lin

What is ***Memory Decoder*** ?





A plug-and-play memory that brings domain knowledge to any LLM — without retraining.

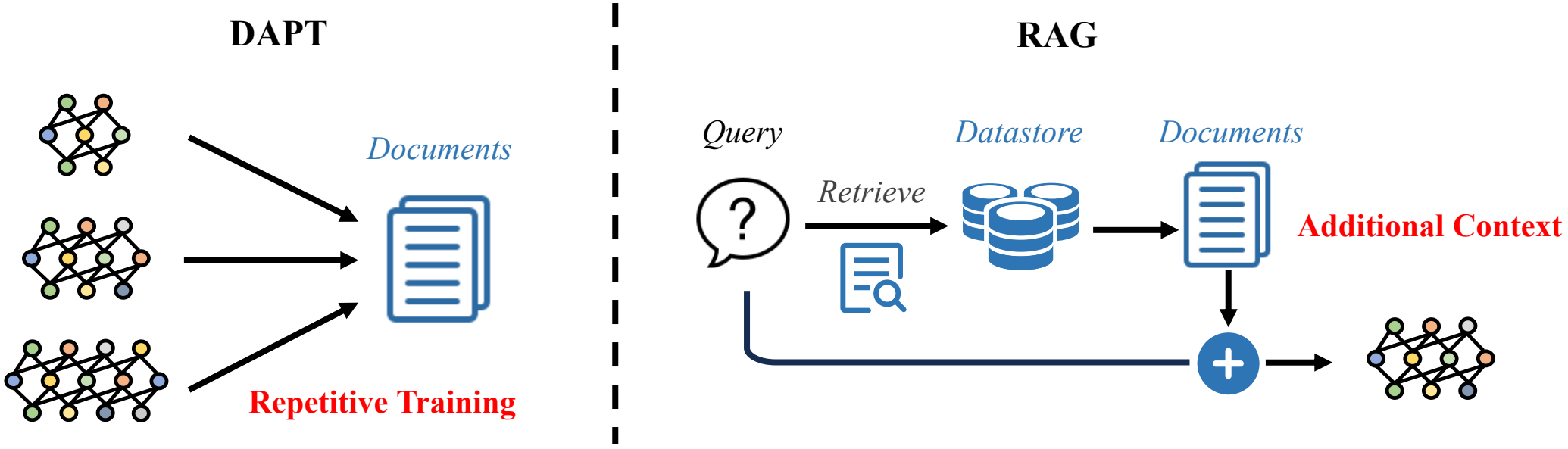
One 0.5B Memory Decoder consistently improves all Qwen models ranging from 0.5B to 72B on the finance domain.

Why ***Domain adaptation*** is still hard for LLMs ?

LLMs are powerful, but domain adaptation is still inefficient.

Existing methods are either:

- *Domain Adaptive Pre-training(DAPT)* – **costly training** & **catastrophic forgetting**.
- *Retrieval Augmented Generation(RAG)* – w/o training but **slow inference**.



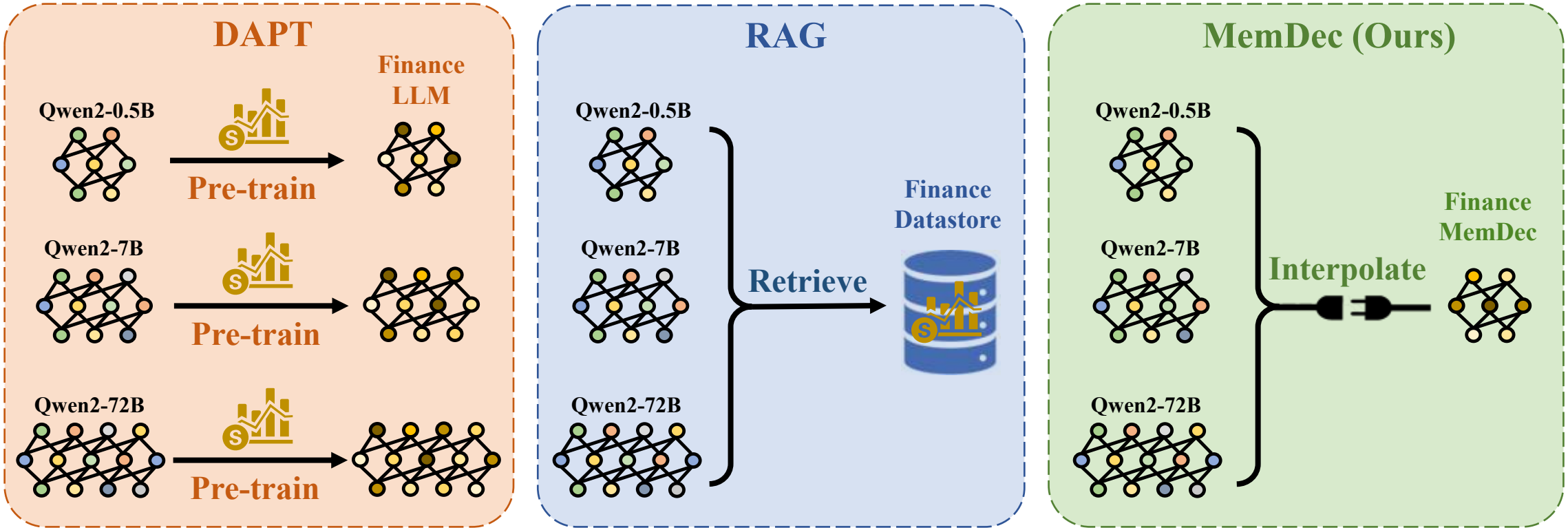
Can we have the ***best of both worlds***?

What we want:

- **No** *retraining* of multiple models (like RAG)
- **No** additional *inference latency* (like DAPT)

Core idea:

Replace the external retriever with a small parametric model



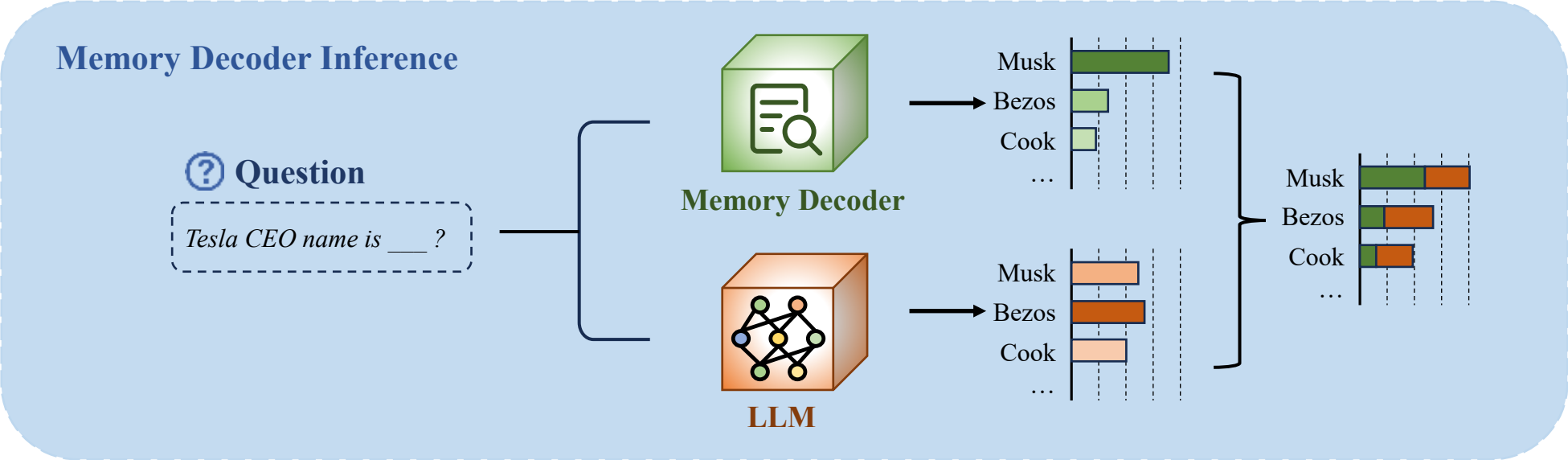
Plug and Play integration of *Memory Decoder*

—— Trained once, reuse everywhere !

Works with **any base LLM** sharing the same tokenizer.

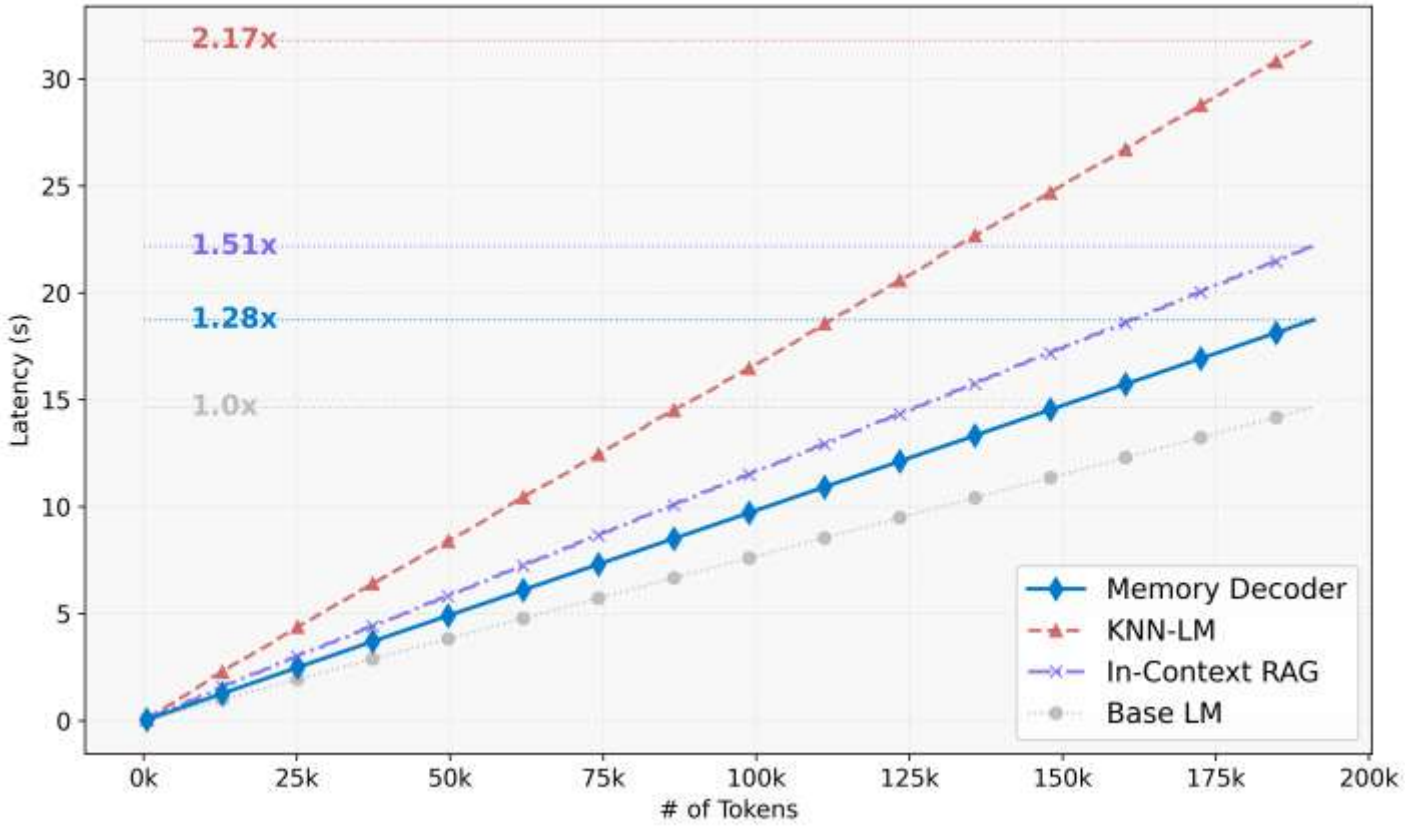
The output of Memory Decoder is interpolated with base LLM in the following way:

$$p_{final}(y \mid x) = \alpha \cdot p_{mem}(y \mid x) + (1 - \alpha) \cdot p_{plm}(y \mid x)$$



Inference Efficiency of *Memory Decoder*

Inference latency is comparable to base LLM, showing great improvement compared to *k*NN-LM and RAG method.



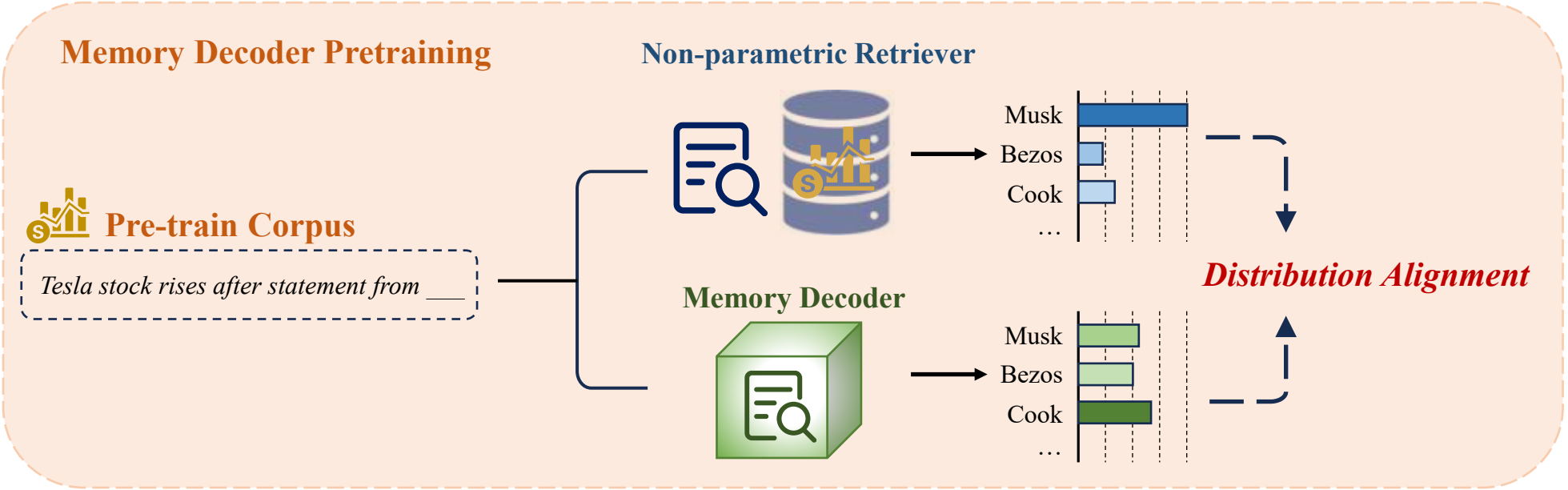
These measurements were conducted on Qwen2.5-1.5B, augmented by a 0.5B Memory Decoder.

How to train ***Memory Decoder*** ?

We use KL divergence to encode retrieval-like knowledge in a compact parametric form.

To prevent excessive deviation from the underlying corpus distribution, we integrate a complementary CE loss in the following way:

$$L = \beta KL(p_{kNN} \,||\, p_{mem}) + (1-\beta)CE$$



Experiments: Language modeling on Wikitext-103

——1+1 > 2 !

	GPT2-small	GPT2-med	GPT2-large	GPT2-xl
base	24.89	18.29	15.80	14.39
Non-parametric methods				
+In-Context RAG	18.46	14.01	12.09	11.21
+kNN-LM	15.62	12.95	12.21	11.30
Parametric methods				
+DAPT	<u>14.76</u>	<u>12.78</u>	**11.10**	**10.16**
+LoRA	18.63	13.88	11.77	10.67
+MemDec	**13.36**	**12.25**	<u>11.53</u>	<u>10.93</u>

An important observation is that augmenting gpt2-med with memdec-small achieves better performance than direct DAPT of gpt2-med, which has nearly **3 times the parameter** of memdec-small !

Experiments: Performance on basic NLP tasks

——To preserve the general abilities, you better not modify the parameters

	SST2	MR	CR	RT	HYP	CB	RTE	AGN	Yahoo	Avg
base	81.98	78.40	84.40	76.54	63.75	41.07	52.70	78.79	49.40	67.45
Non-parametric methods										
+kNN-LM	81.98	77.95	83.80	77.95	64.14	39.28	52.70	77.73	49.63	67.24
Parametric methods										
+DAPT	83.52	80.15	80.45	77.39	36.04	50.00	51.26	64.31	24.40	60.84
+LoRA	80.88	76.90	83.95	76.07	64.14	39.28	53.79	81.06	49.46	67.28
+MemDec	82.43	78.35	84.35	77.30	64.15	57.14	55.24	79.80	49.37	**69.79**

Performance on **Nine** diverse NLP tasks including sentiment analysis, textual entailment, and text classification. Memory Decoder achieves domain adaptation **without sacrificing general capabilities**.

Experiments: One *Memory Decoder* for **ALL** models

Model	Bio	Fin	Law	Avg
Qwen2 Family				
Qwen2-0.5B	18.41	16.00	10.23	14.88
+*LoRA*	7.28	9.70	5.82	7.60
+*MemDec*	**3.75**	**3.84**	**4.57**	**4.05**
Qwen2-1.5B	12.42	10.96	7.69	10.36
+*LoRA*	5.73	7.37	4.84	5.98
+*MemDec*	**3.68**	**3.61**	**4.32**	**3.87**
Qwen2-7B	8.36	8.31	5.92	7.53
+*LoRA*	4.47	5.64	4.02	4.71
+*MemDec*	**3.59**	**3.38**	**4.00**	**3.66**
Qwen2-72B	6.15	6.62	4.84	5.87
+*MemDec*	**3.45**	**3.20**	**3.69**	**3.45**
Qwen2.5 Family				
Qwen2.5-0.5B	17.01	16.04	9.86	14.30
+*LoRA*	7.02	9.88	5.75	7.55
+*MemDec*	**3.74**	**3.87**	**4.57**	**4.06**
Qwen2.5-1.5B	11.33	11.20	7.42	9.98
+*LoRA*	5.59	7.50	4.82	5.97
+*MemDec*	**3.67**	**3.61**	**4.29**	**3.86**
Qwen2.5-3B	9.70	9.83	6.68	8.74
+*LoRA*	5.07	6.71	4.45	5.41
+*MemDec*	**3.63**	**3.52**	**4.16**	**3.77**
Qwen2.5-7B	8.19	8.61	5.94	7.58
+*LoRA*	4.03	5.31	**3.81**	4.38
+*MemDec*	**3.57**	**3.42**	4.01	**3.67**
Qwen2.5-14B	7.01	7.60	5.35	6.65
+*MemDec*	**3.51**	**3.31**	**3.86**	**3.56**
Qwen2.5-32B	6.65	7.38	5.18	6.40
+*MemDec*	**3.48**	**3.29**	**3.81**	**3.53**
Qwen2.5-72B	5.90	6.80	4.84	5.85
+*MemDec*	**3.44**	**3.23**	**3.70**	**3.46**

——*One model to rule them all*

Model	Bio	Fin	Law	Avg
Llama3 Family				
Llama3-8B	7.95	8.63	5.96	7.51
+*LoRA*	4.38	5.68	**4.12**	4.73
+*MemDec*	**3.92**	**4.32**	4.46	**4.23**
Llama3-70B	5.92	6.87	4.90	5.90
+*MemDec*	**3.74**	**4.01**	**4.07**	**3.94**
Llama3.1 Family				
Llama3.1-8B	7.82	8.46	5.88	7.39
+*LoRA*	4.38	5.72	**4.10**	4.73
+*MemDec*	**3.91**	**4.30**	4.42	**4.21**
Llama3.1-70B	5.85	6.68	4.89	5.81
+*MemDec*	**3.73**	**3.97**	**4.06**	**3.92**
Llama3.2 Family				
Llama3.2-1B	12.81	11.85	8.23	10.96
+*LoRA*	5.97	7.83	5.21	6.34
+*MemDec*	**4.06**	**4.85**	**5.11**	**4.67**
Llama3.2-3B	9.83	9.70	6.83	8.79
+*LoRA*	5.11	6.55	**4.59**	5.42
+*MemDec*	**3.99**	**4.45**	4.76	**4.40**

Performance on biomedicine, finance and law domains of two most popular open-source model families (Qwen and Llama).

Case study: Bridging ***Parametric and Non-Parametric*** Methods

——*Learning* ***long-tail knowledge*** *from non-parametric methods and maintain* ***semantic coherence*** *as a parametric model.*

Long-tail Knowledge Learning			
Context (target token underlined)	**MemDec**	**kNN**	**Base LM**
he starred alongside actors Mark Strong and Derek **Jacobi**	**68.94%**	9.39%	0.12%
The launch of HMS Dreadnought in **1906** by the Royal Navy raised the stakes	**98.65%**	40.62%	1.57%
Semantic Coherence and Reasoning			
Context (target token underlined)	**MemDec**	**kNN**	**Base LM**
In 2000 Boulter had a guest-starring role **on** the television series The Bill	40.11%	8.07%	**45.51%**
...three tank squadrons for special overseas operations, known as 'A', 'B' and ' **C** ' Special Service Squadrons	50.10%	10.76%	**63.04%**

Thank you for watching !