Cross-model Control: Improving Multiple Large Language Models in One-time Training

Jiayi Wu[1], Hao Sun[2], Hengyi Cai[3], Lixin Su[4],

Shuaiqiang Wang[4], Dawei Yin[4], XiangLi[1], Ming Gao[1]

[1]East China Normal University [2]Peking University

[3]Chinese Academy of Sciences [4]Baidu Inc

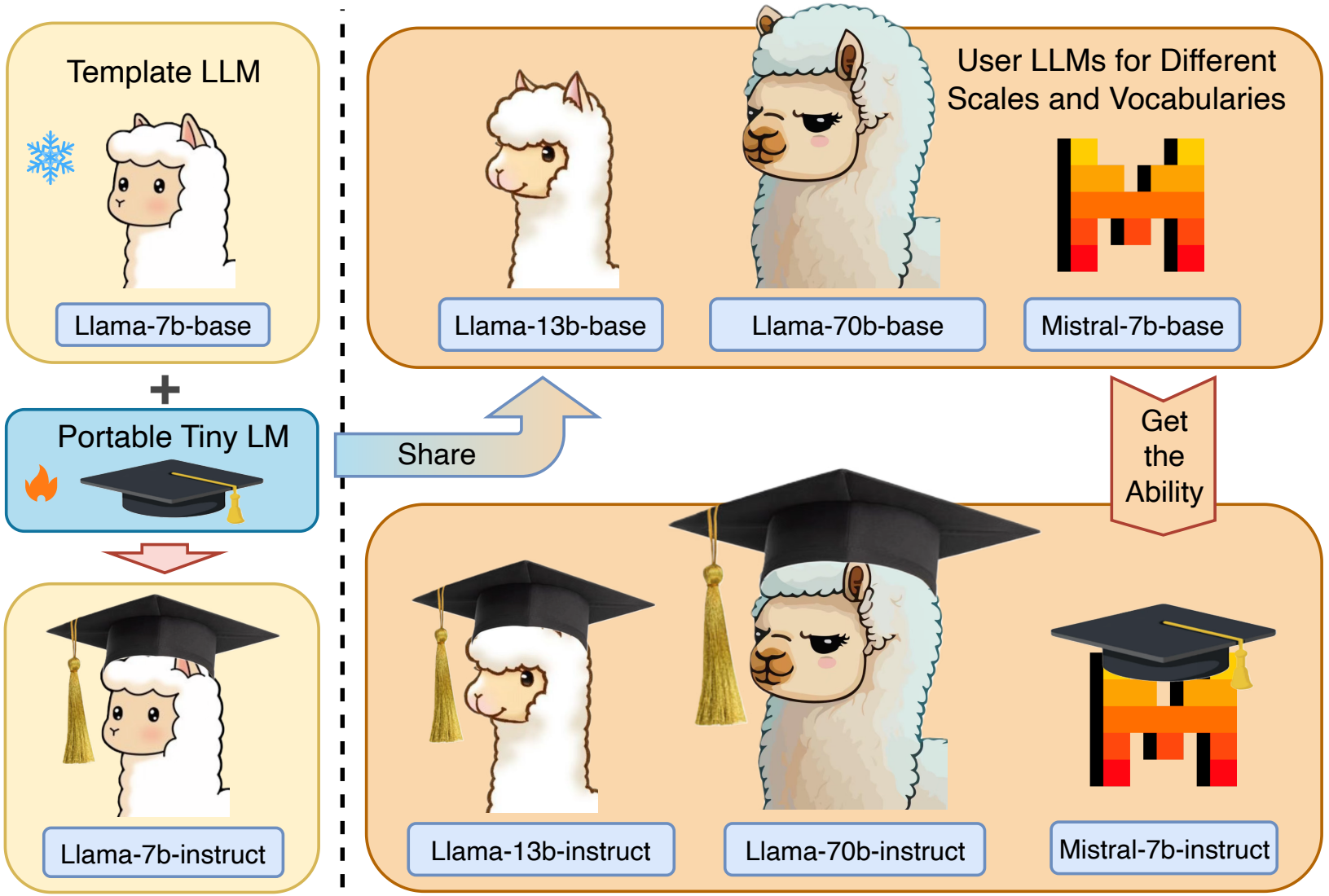
Background and Motivation

- In recent years, there has been an increasing number of large language models (LLMs) with varying parameter scales and vocabularies.
- However, although these models are different, they generally face a series of common optimization needs, such as instruction fine-tuning and unlearning.

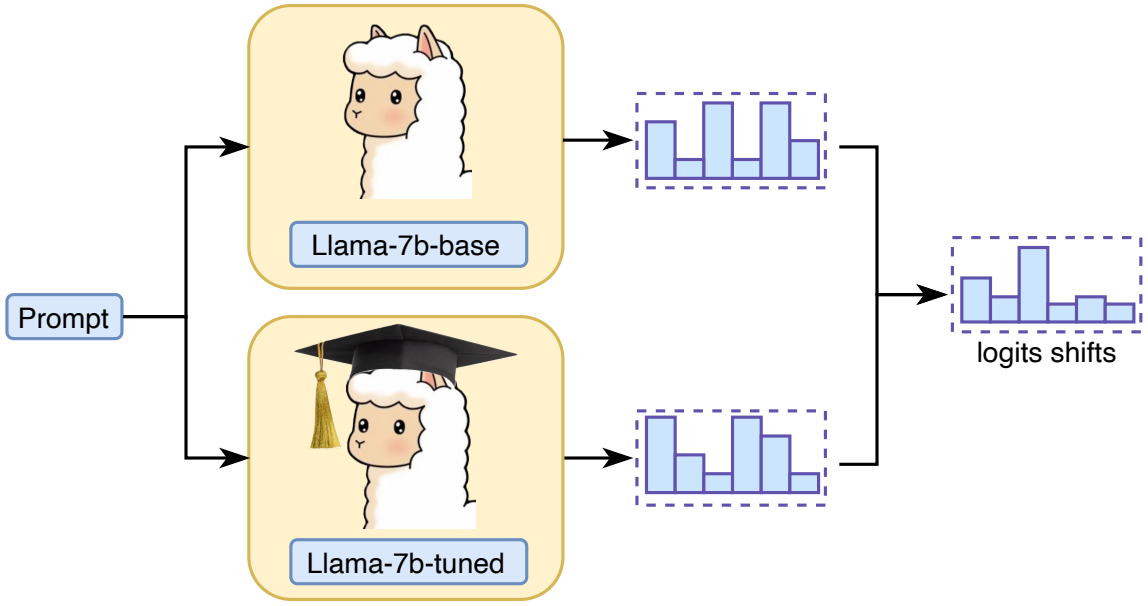
Background and Motivation

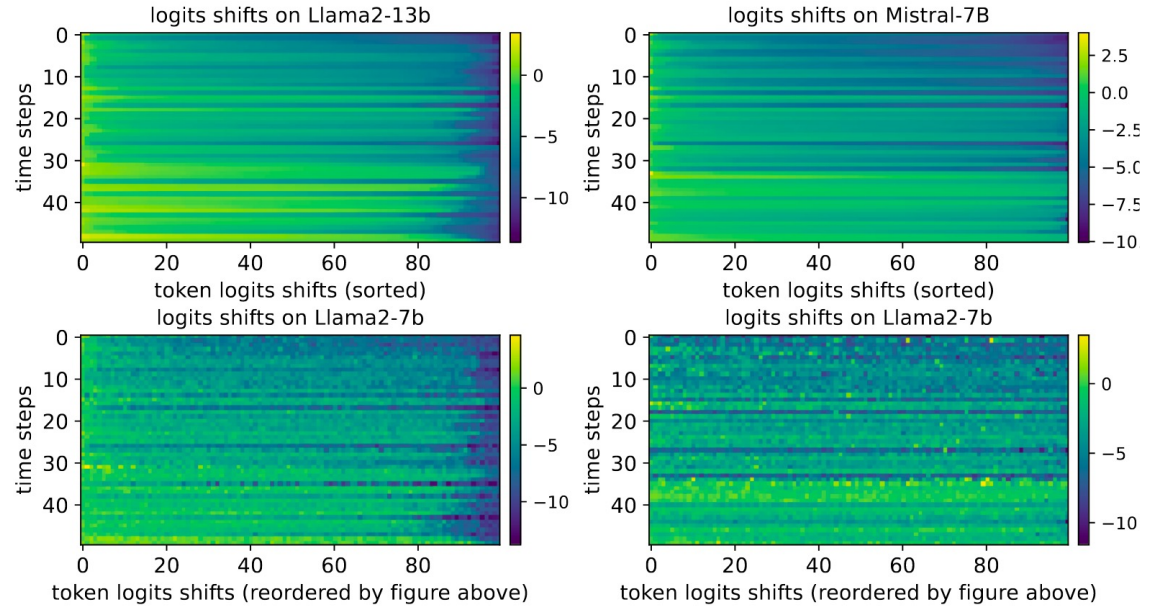
Training LLM is expensive.

Raises the question:
Is it possible to reuse the fine-tuning effects of LLMs?



Preliminaries





(a) Logits shifs on LLAMA2-13B and LLAMA2-7B. (b) Logits shifs on MISTRAL-7B and LLAMA2-7B.

Can the fine-tuning effects be transferred between different LLMs?

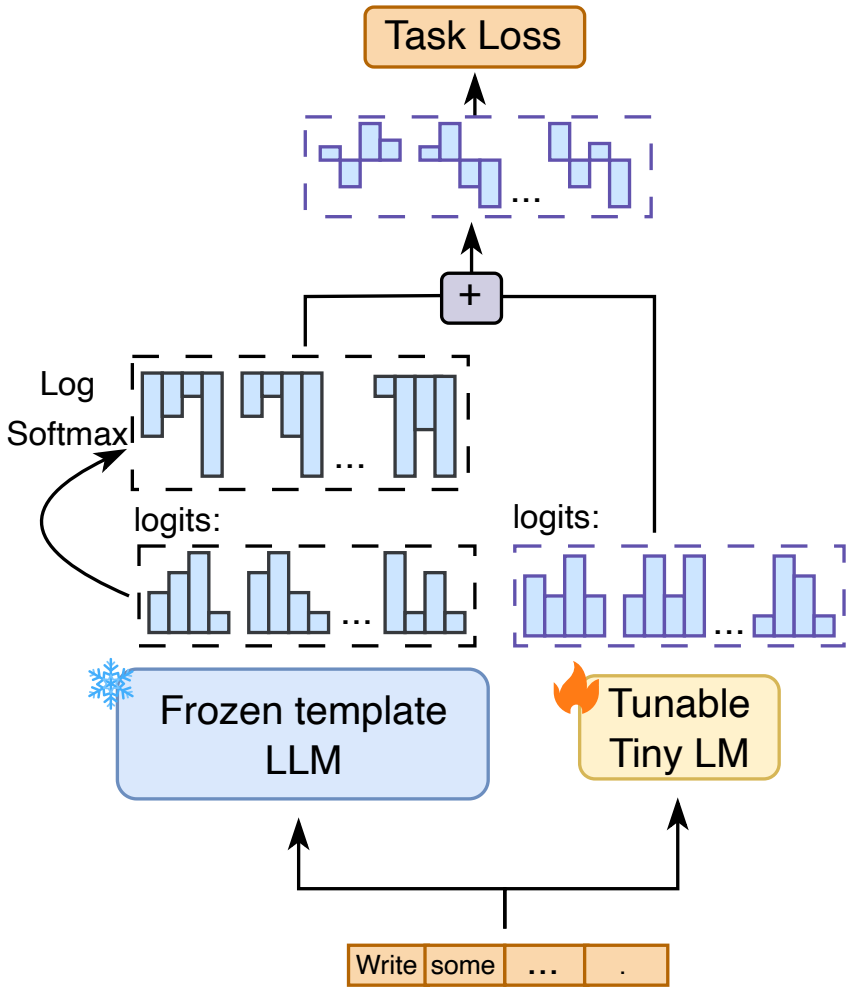
The answer is yes, we regard the fine-tuning effects as the logits shifts before and after fine-tuning, and find that the fine-tuning effects are similar between different LLMs.

Method

Given the observation that the logits shifts before and after fine-tuning in different LLMs are remarkably similar.

- We introduce a parameter-efficient, portable tiny language model designed to learn to alter the output logits of LLMs, which we call the “delta model”.
- Sharing the delta model with other models could transfer the fine-tuning effect.

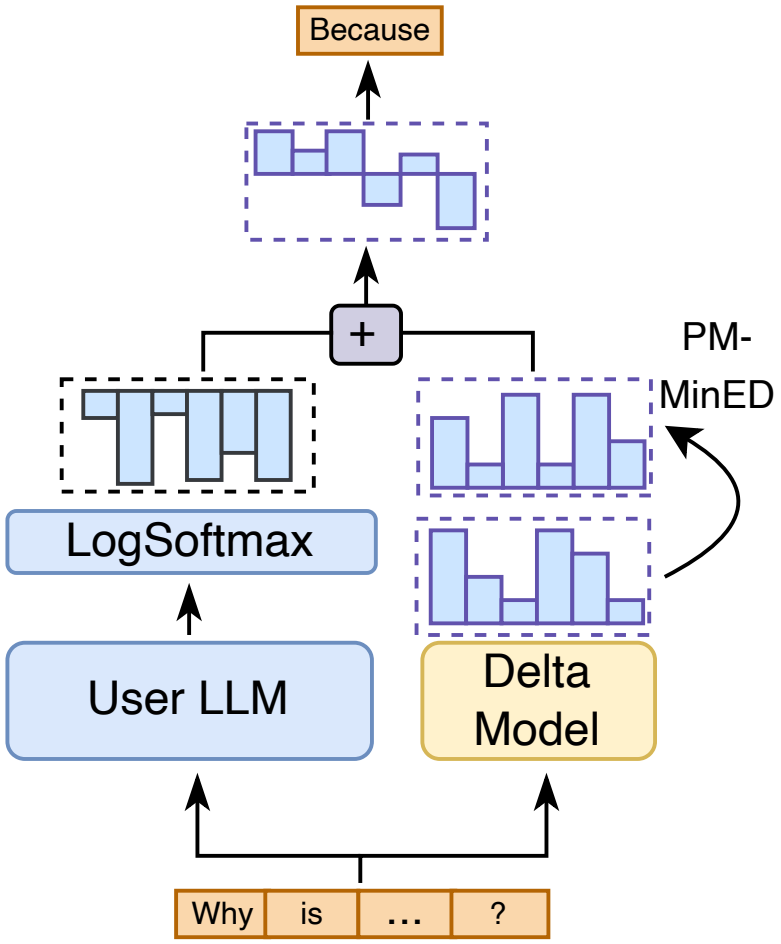
Method



Train the delta model to alter the logits of the LLM.

- Introduce a template LLM.
- During training, the template LLM is frozen, and the delta model is a tunable module that interacts with the template LLM.
- In the forward stage, the logits of the template LLM and the logits of the delta model are added together.

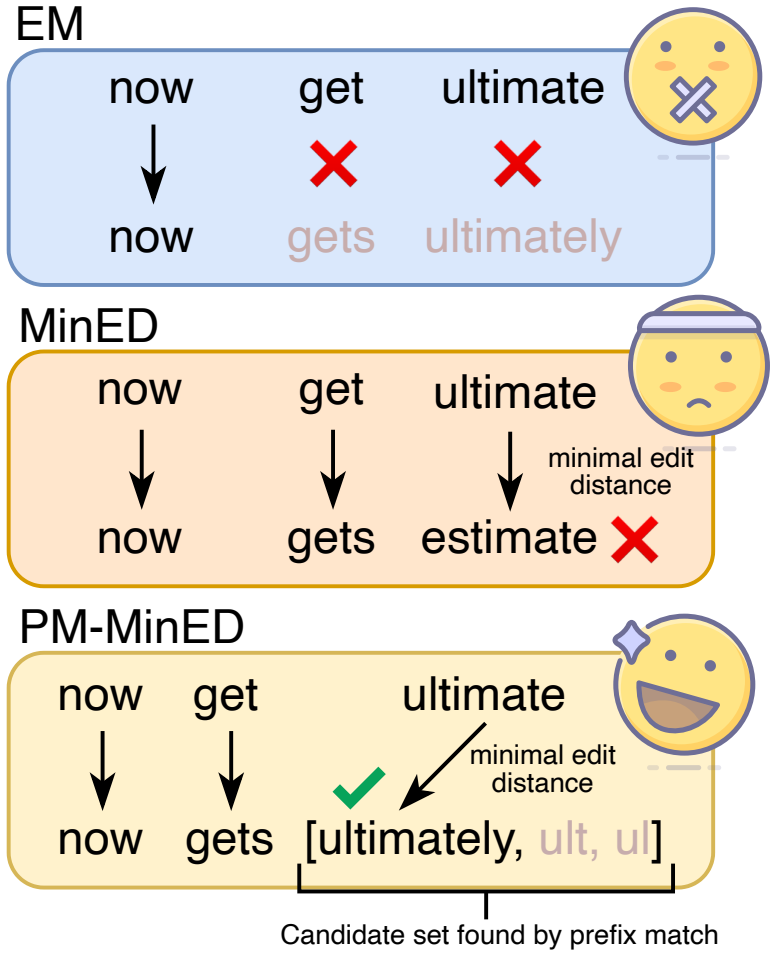
Method



Share the delta model with other LLMs

The delta model interacts with other user LLM to achieve fine-tuning effects. During the forward stage, we add the logits from the user LLM and the delta model together for decoding.

Method



Token mapping PM-MinED

When the vocabulary of the user LLM is different from the delta model, the PM-MinED strategy is introduced to find the corresponding relationship between tokens in the two vocabularies.

Take the "ultimate" in the user LLM vocabulary as an example:

- Find the candidate set [“ultimately”, “ult”, “ul”, “u”] in the delta model vocabulary that can be prefix-matched with “ultimate”.
- Take out the token "ultimately" with the smallest edit distance from the candidate set.

Experiments

Table 1: Instruction tuning results on AlapcaEval (Win %). In cross-model control, all base models incorporate the same delta model, which is trained using the LLAMA2-7B as the template model.

Method	Params Add	LLAMA2-7B	LLAMA2-13B	LLAMA2-70B	MISTRAL-7B
Vanilla Base Model	-	4.22	5.34	11.55	6.83
LoRA (upper bound)	110M	68.18	75.16	OOM	79.91
Proxy-tuning	220M	8.47 (+4.25)	10.47 (+5.13)	8.59 (-2,96)	- -
CMC (ours)	110M	30.41 (+26.19)	39.04 (+31.51)	49.81 (+38.26)	33.29 (+26.46)

Table 2: Unlearning results on TOFU benchmark. All chat models incorporate the same delta model, which is trained using the LLAMA2-7B-TOFU as the template model. Better scores are bolded.

	Forget Set			Retain Set			Real Author			World Fact		
Method	RL (↓)	P (↓)	TR(↓)	RL(↑)	P(↑)	TR(↑)	RL	P	TR	RL	P	TR
LLAMA2-7B-TOFU	0.99	0.99	0.51	0.98	0.99	0.48	0.93	0.45	0.58	0.87	0.43	0.56
+LoRA	0.01	0.00	0.77	0.71	0.75	0.48	**0.88**	**0.51**	**0.68**	**0.88**	**0.46**	**0.60**
+CMC (ours)	**0.00**	**0.00**	**0.33**	**0.91**	**0.97**	**0.51**	0.84	0.43	0.58	0.85	0.45	0.57
LLAMA2-13B-TOFU	1.00	1.00	0.46	0.99	1.00	0.53	0.89	0.51	0.67	0.86	0.46	0.62
+δ-UNLEARNING	0.38	0.06	0.53	0.53	0.48	0.52	0.61	0.36	0.46	0.83	0.41	0.59
+LoRA	0.03	0.00	0.50	0.85	0.92	0.52	**0.87**	**0.54**	**0.70**	**0.86**	**0.48**	0.63
+CMC (ours)	**0.00**	**0.00**	**0.29**	**0.97**	**0.99**	**0.55**	0.77	0.49	0.65	0.83	0.47	**0.65**
MISTRAL-7B-TOFU	1.00	1.00	0.49	1.00	1.00	0.48	0.84	0.61	0.75	0.88	0.62	0.78
+LoRA	**0.00**	0.00	0.72	0.95	0.96	0.49	**0.79**	0.57	0.71	**0.87**	**0.64**	**0.78**
+CMC (ours)	0.00	**0.00**	**0.30**	**0.99**	**0.99**	**0.51**	0.73	**0.62**	**0.75**	0.86	0.63	0.77

We have conducted extensive experiments on instruction tuning and unlearning tasks, transferring the fine-tuning effects of Llama2-7b to models of different parameter scales and vocabularies, and achieved satisfactory results.

Conclusion

In this work, we have introduced CMC, which for the first time enables the transfer of fine-tuning effects between different models. Our code and models are publicly available, with the links below.

Github Link: https://github.com/wujwyi/CMC
Huggingface Link: https://huggingface.co/collections/wuqiong1/cross-model-control-671e2565748faf685ebeccea

Thank you for your attention!!!