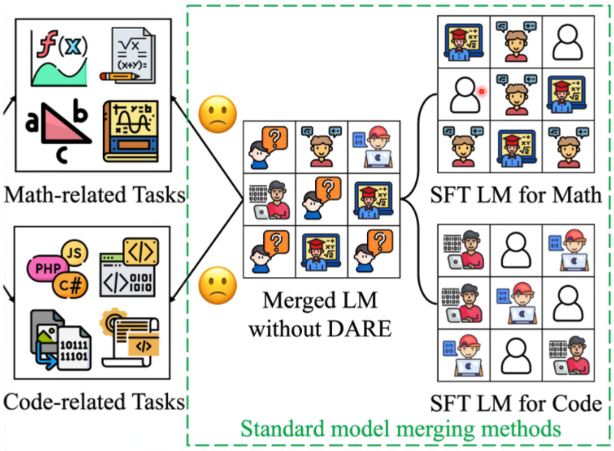
2023.12.12 赵君

主要内容

- 从专才到通才的转换；
- 更好控制模型的行为，消除bias和toxic；
- 提高参数效率；

专才与通才

Language Models are Super Mario: Absorbing Abilities from Homologous Models as a Free Lunch



$$\boldsymbol{\theta}_{\mathrm{M}} = \boldsymbol{\theta}_{\mathrm{PRE}} + \lambda \cdot \sum_{k=1}^{K} \boldsymbol{\delta}^{t_k} = \boldsymbol{\theta}_{\mathrm{PRE}} + \lambda \cdot \sum_{k=1}^{K} (\boldsymbol{\theta}_{\mathrm{SFT}}^{t_k} - \boldsymbol{\theta}_{\mathrm{PRE}}),$$

- delta为SFT后的模型与预训练模型的增量；
- 简单将参数相加，直观；
- 问题：参数冲突；math的模型和code的模型SFT参数在一些地方有冲突，直接相加会造成能力损失；
- 解决：随机丢弃一些参数的增量delta，可以发现随着LM的size增加，其对于任务的完成依然很好（math上70B甚至99%）；这说明SFT更新的大部分参数都是非常冗余的；
- （或许不应该随机drop，应该根据参数更新的幅值来drop，更新大的或许才是重要参数）

  - The delta parameters of both encoder- and decoder-based LMs are highly redundant.
  - The tolerance of drop rates increases with the sizes of LMs

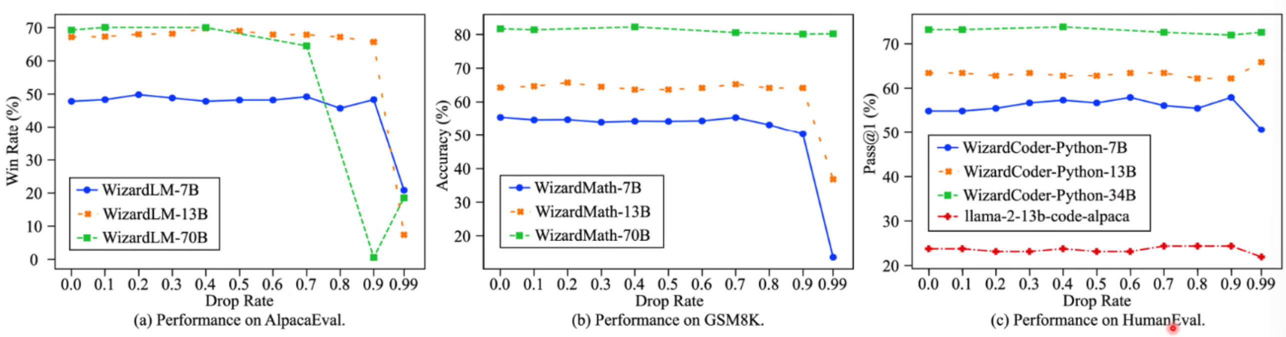


Figure 3: Performance of various decoder-based LMs on AlpacaEval, GSM8K, and HumanEval.

- 方案：通过丢弃大部分delta参数来解决参数冲突；

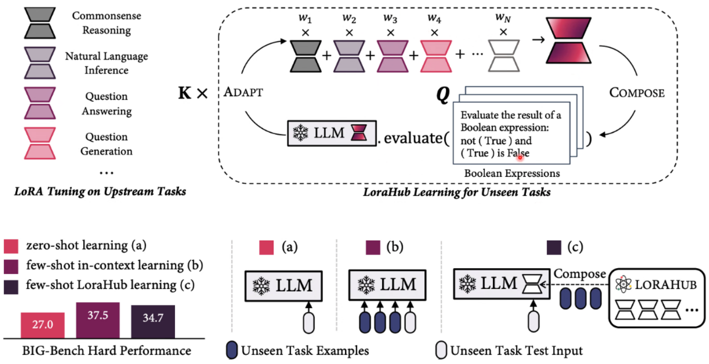
Table 1: Results of merging decoder-based LMs by Task Arithmetic, where LM, Math, and Code are the abbreviations of WizardLM-13B, WizardMath-13B, and llama-2-13b-code-alpaca. We use blue, green, and red colors to distinguish each single model and utilize mixed colors to denote the merged models. The best and second-best results among the single model, the merged models with and without DARE are marked in **bold** and underlined fonts.

Merging Methods	Models	Preprocess	Instruction-Following	Mathematical Reasoning		Code-Generating	
			AlpacaEval	GSM8K	MATH	HumanEval	MBPP
Single Model	LM	/	67.20	2.20	0.04	36.59	34.00
	Math	/	/	64.22	14.02	/	/
	Code	/	/	/	/	23.78	27.60
Task Arithmetic	LM & Math	No	67.04	**66.34**	13.40	28.66	30.60
		w/ DARE	**67.45**	66.26	12.86	26.83	32.40
	LM & Code	No	**68.07**	/	/	31.70	32.40
		w/ DARE	67.83	/	/	**35.98**	**33.00**
	Math & Code	No	/	64.67	13.98	8.54	8.60
		w/ DARE	/	**65.05**	13.96	10.37	9.80
	LM & Math & Code	No	69.03	58.45	9.88	18.29	29.80
		w/ DARE	**69.28**	56.48	10.16	23.17	**31.60**

实验来看，在2、3个任务上面参数融合是有效的；但是在更多的任务融合性能会较大下降；

LORAHUB:EFFICIENT CROSS-TASK GENERALIZATION VIA DYNAMIC LORA COMPOSITION

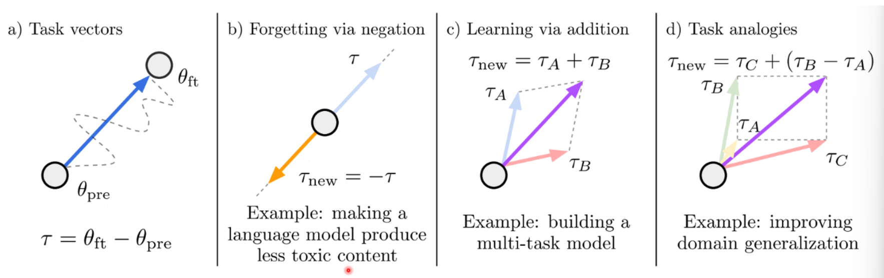
把多个任务得到的Lora模块统一存为一个hub；
基于具体任务对特定的lora模型进行组合，得到最终的权重；



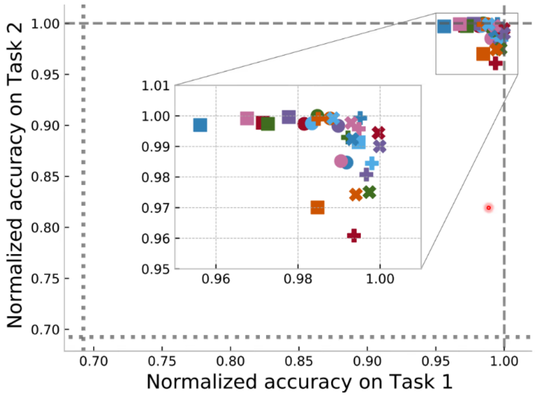
缺点：效果有限；

消除bias

EDITING MODELS WITH TASK ARITHMETIC

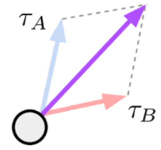


两个任务相加，基本上能力都能获得；



c) Learning via addition

$$\tau_{\text{new}} = \tau_A + \tau_B$$



Example: building a multi-task model