ICLR

Unlocking the Power of Function Vectors for Characterizing and Mitigating Catastrophic Forgetting in Continual Instruction Tuning

Gangwei Jiang[1,2], Caigao Jiang[4], Zhaoyi Li[1,2], Siqiao Xue[4], Jun Zhou[4],
Linqi Song[2], Defu Lian[1], Ying Wei[3]

1 - University of Science and Technology of China
2 - City University of Hongkong
3 - Zhejiang University
4 - Independent

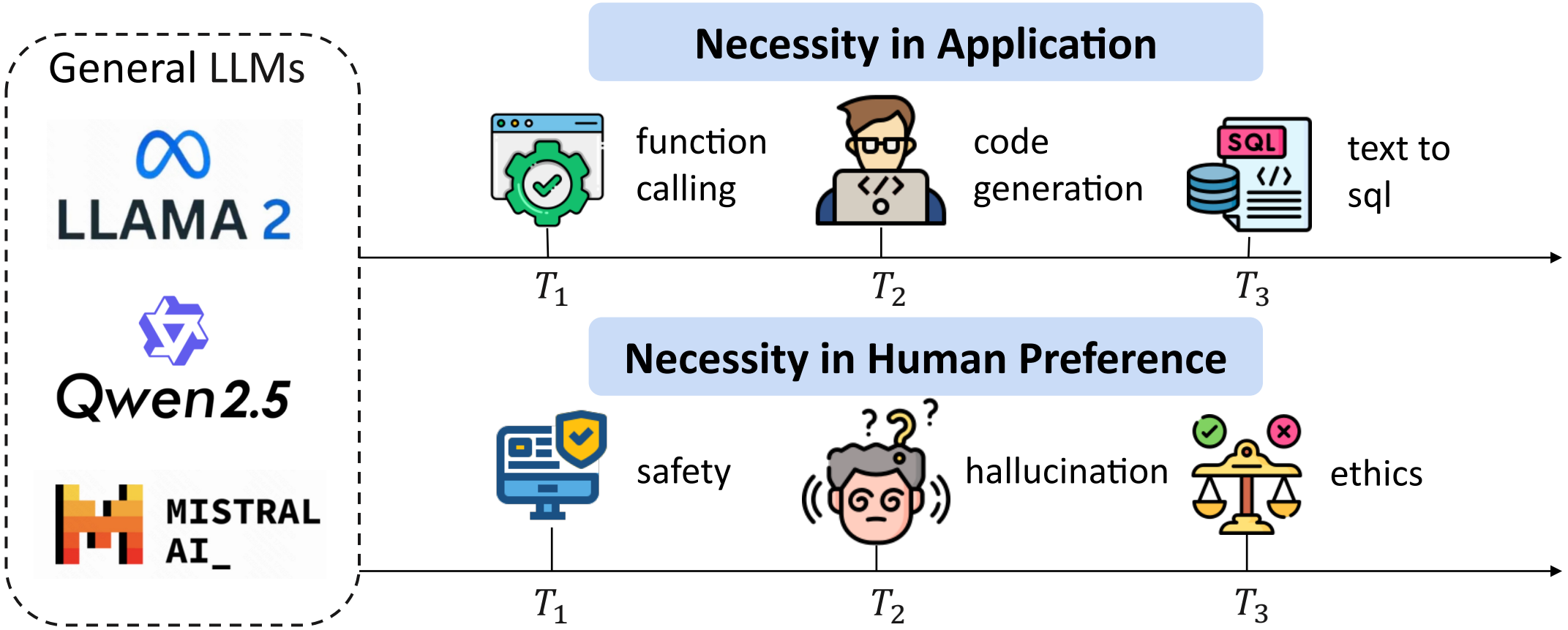
浙江大學 ZHEJIANG UNIVERSITY

CityU 香港城市大學 City University of Hong Kong

中国科学技术大学 University of Science and Technology of China

Continual Instruction Tuning

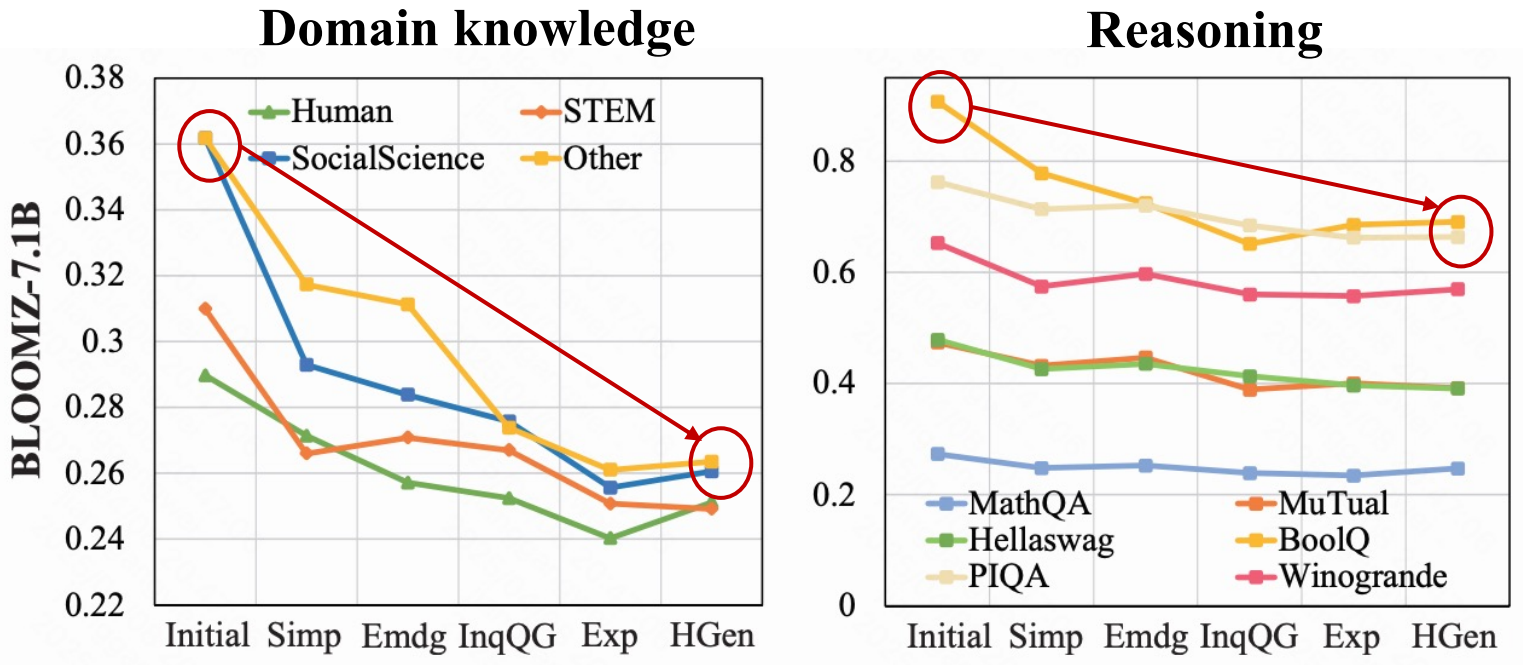
Continual Instruction Tuning: general LLMs streaming **fine-tune on a sequence of tasks** $T_1, T_2, \ldots, T_N$ over time to adapt to new instructions.



Catastrophic Forgetting of LLMs

Catastrophic Forgetting occurs during continual instruction tuning.

- A decline in the model's performance on old tasks when adapting to new tasks, e.g., on domain knowledge and reasoning [1].



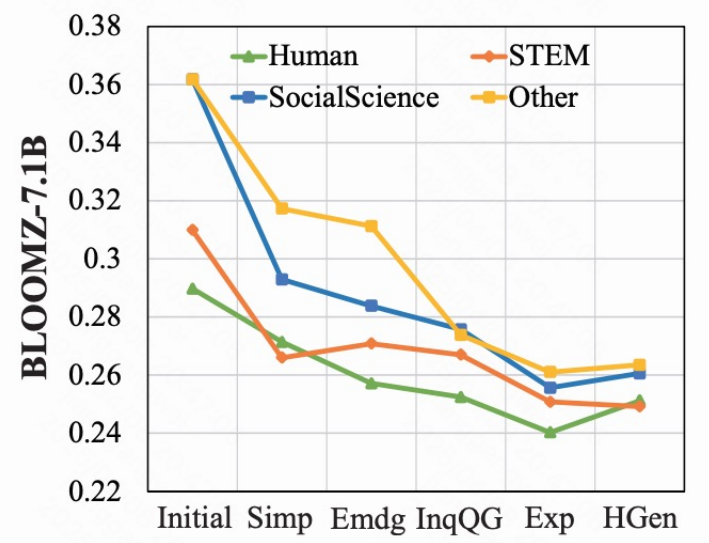
Catastrophic Forgetting

[1] Yun Luo, et al. An empirical study of catastrophic forgetting in large language models during continual fine-tuning, 2024.

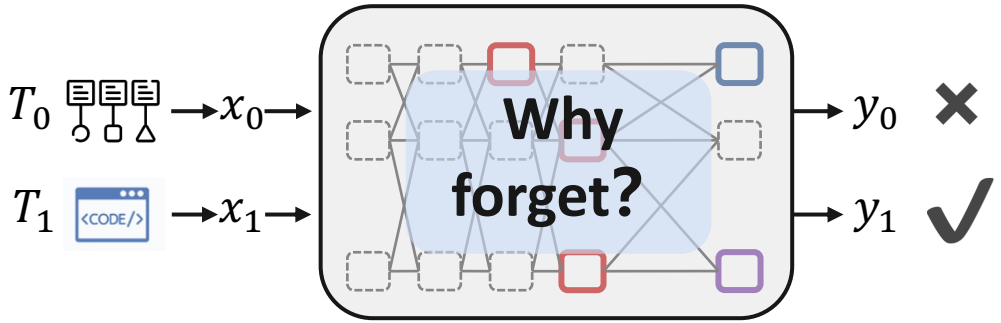
Motivation

Problems with existing works:

- Analyze forgetting from limited perspectives.
- Lack of understanding the **internal mechanisms** underlying model forgetting.



Analysis from limited perspectives [1]



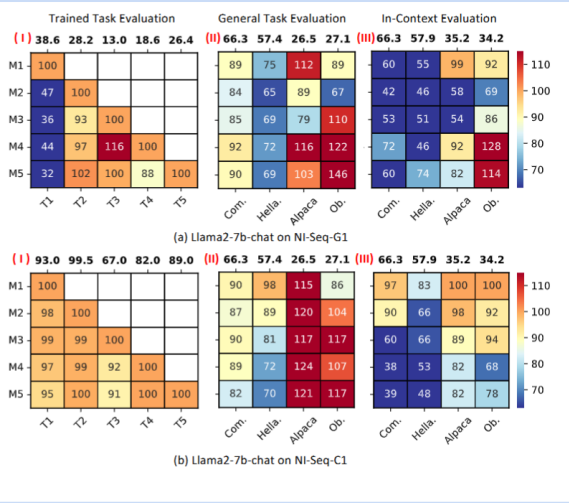
Lack of understanding the internal mechanisms

[1] Yun Luo, et al. An empirical study of catastrophic forgetting in large language models during continual fine-tuning, 2024.

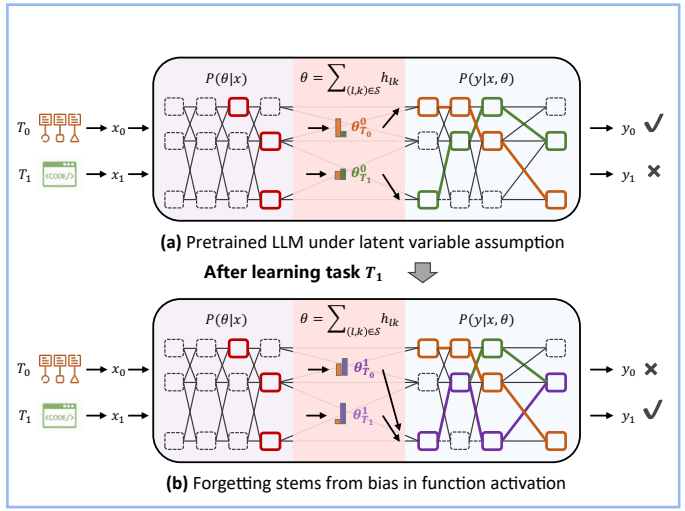
Contributions

- **Investigate** catastrophic forgetting of LLMs from **multiple perspectives.**
- **Characterize** catastrophic forgetting with **function vector hypothesis.**
- **Mitigate** catastrophic forgetting by proposing **FV-guided training.**

Investigate forgetting



Characterize forgetting



Mitigate forgetting

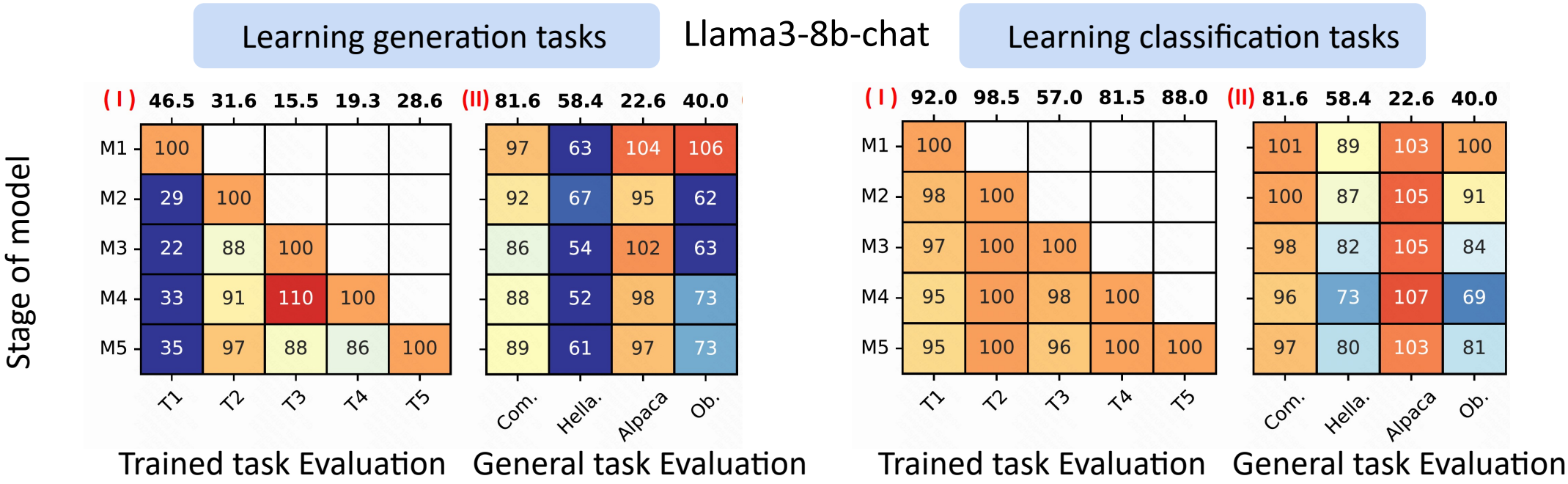
$$\ell_{FV} = \sum_{(l,k)\in\mathcal{S}} d\left(h_{lk}^{M_{j-1}}(x), h_{lk}^{M}(x)\right)$$

$$\ell_{KL} = KL[P_M(\cdot \mid x) \| P_{M_{j-1}^{h_l \to h_l + \theta^0_{T_j}}}(\cdot \mid x)]$$

How Forgetting Behaves? Sequence Type

Investigate forgetting in continual instruction tuning from **multiple perspectives**, includes sequence type, evaluation ability, and model.

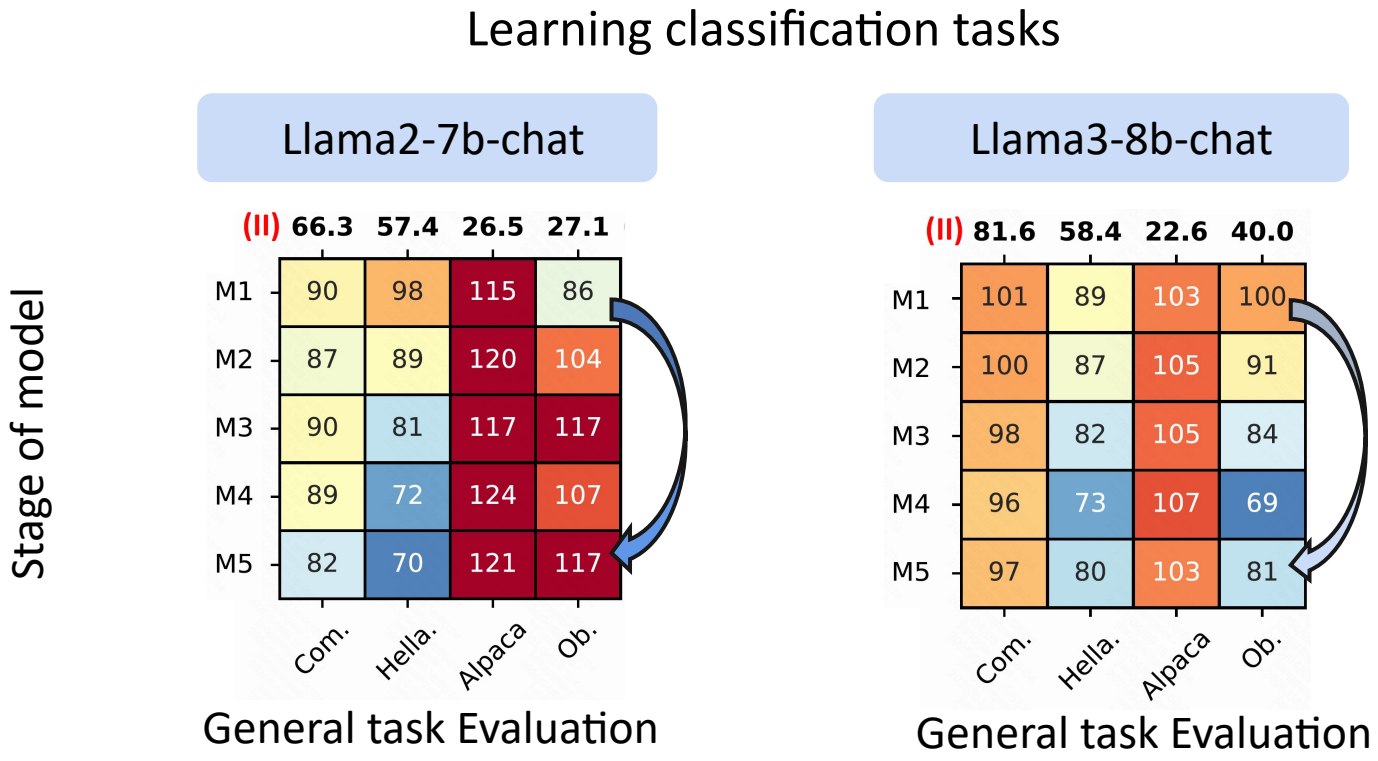
- Instruction tuning sequences with generation tasks lead to greater forgetting compared to classification tasks.



How Forgetting Behaves? Model

Investigate forgetting in continual instruction tuning from **multiple perspectives**, includes sequence type, evaluation ability, and model.

➢ Forgetting is model-dependent.

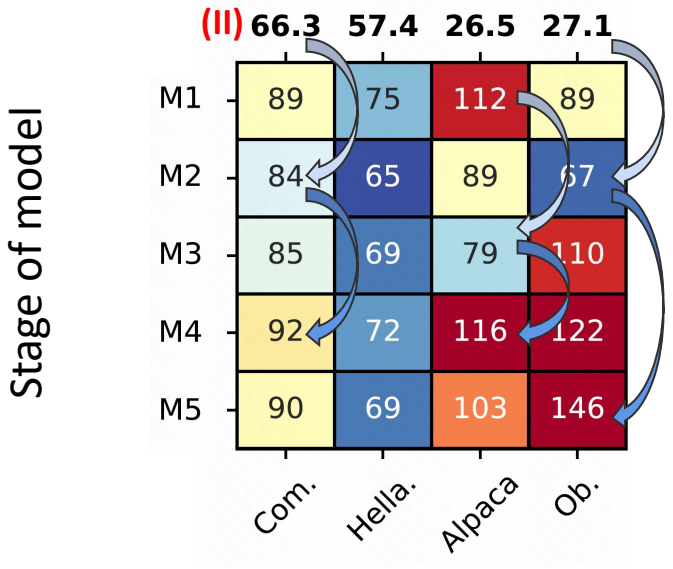


How Forgetting Behaves? Training Process

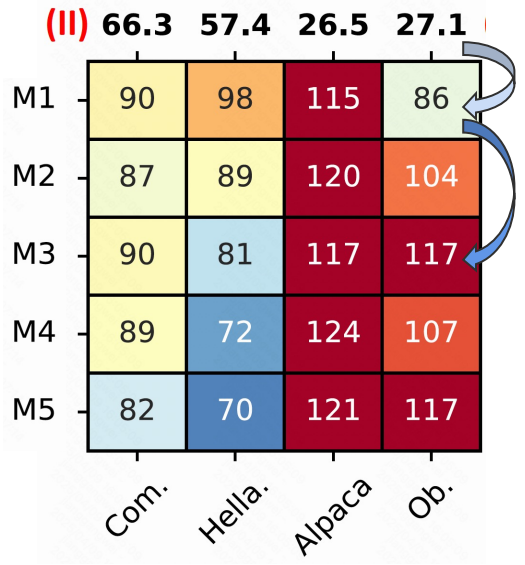
Investigate forgetting in continual instruction tuning from **multiple perspectives**, includes sequence type, evaluation ability, and model.

➢ Forgetting may be naturally mitigated during training.

Llama2-7b-chat on generation tasks



Llama2-7b-chat on classification tasks

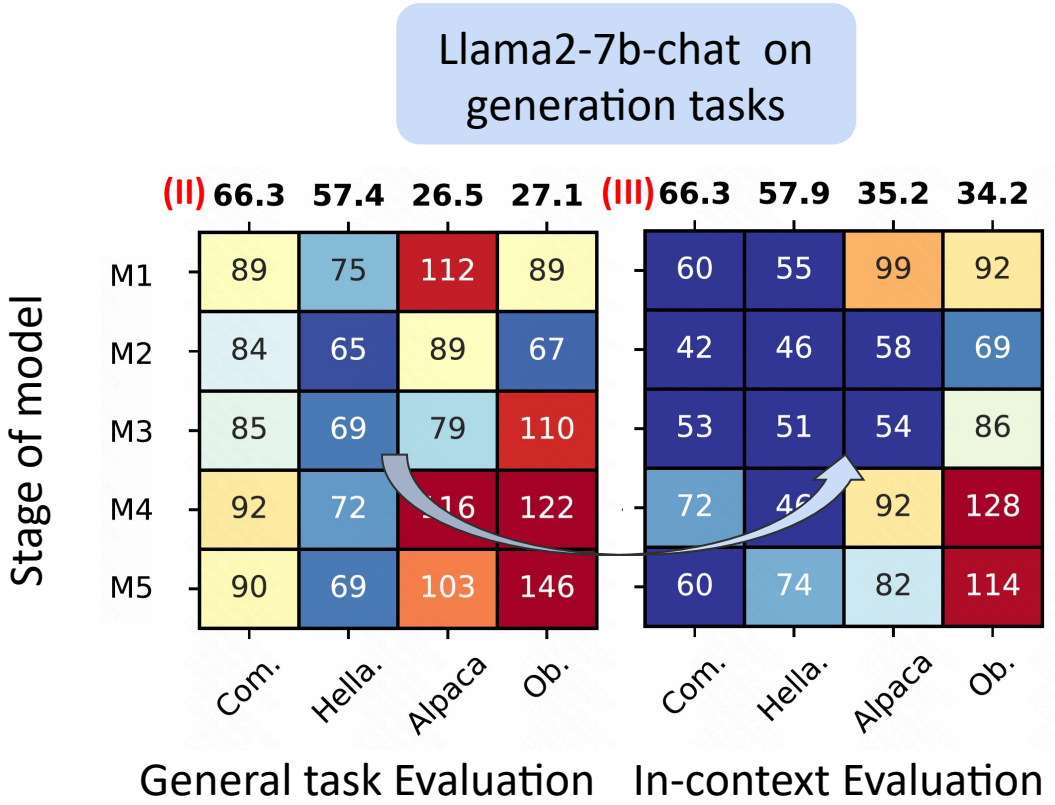


First drop then raise

How Forgetting Behaves? Evaluation Ability

Investigate forgetting in continual instruction tuning from **multiple perspectives**, includes sequence type, evaluation ability, and model.

➢ In-context learning performance degrades significantly.

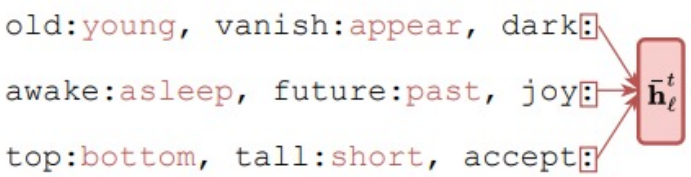


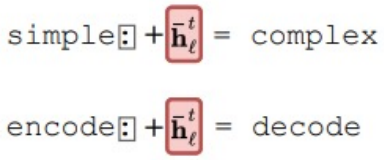
Why Forgetting Happens? The Role of Function Vectors

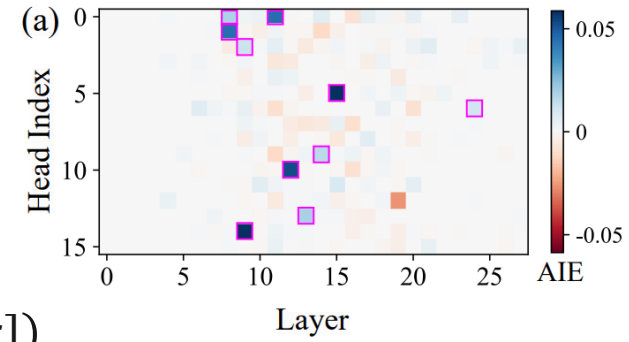
We identify the **strongly correlation** between **function vector (FV) similarities** and diverse **forgetting patterns** across task types and training stages.

Function Vector [1]	**An internal representation of a task ability in model**

1. Average head activation
2. Zero-shot intervention
3. Function vector







$$\bar{h}_{lj}^c = \frac{1}{|D^c|}\sum_{(x)\in D^c} h_{\ell j}([p,x])$$

$$\mathrm{CE}_{lj}([\hat{p},x]) = P_{M_{h_{lj}\to\bar{h}_{lj}^c}}(y_i \mid [x]) - P_M(y_i \mid [x])$$

$$\theta_T = \sum_{(l,k)\in\mathcal{S}} \bar{h}_{lk}^T$$

[1] Todd, Eric, et al. Function Vectors in Large Language Models, ICLR 2024.

Why Forgetting Happens? The Role of Function Vectors

We identify the **strongly correlation** between **function vector (FV) similarities** and diverse **forgetting patterns** across task types and training stages.

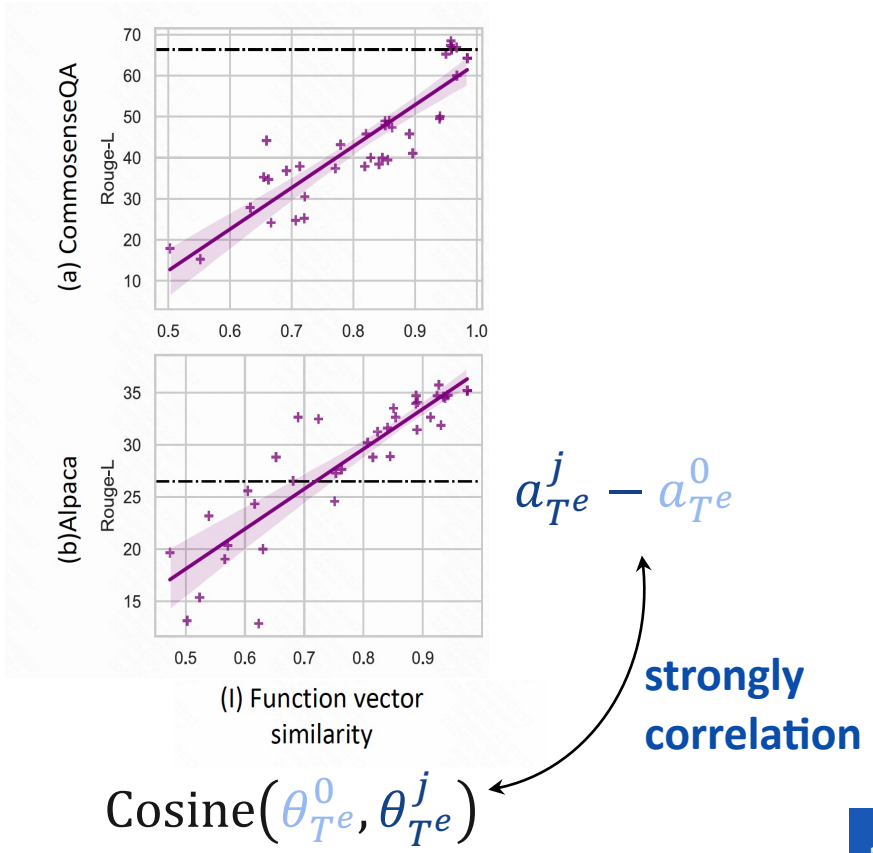
Training sequence T_1 ... T_j ...

Model stage M_0 M_1 ... M_j ...

Evaluation task T_e

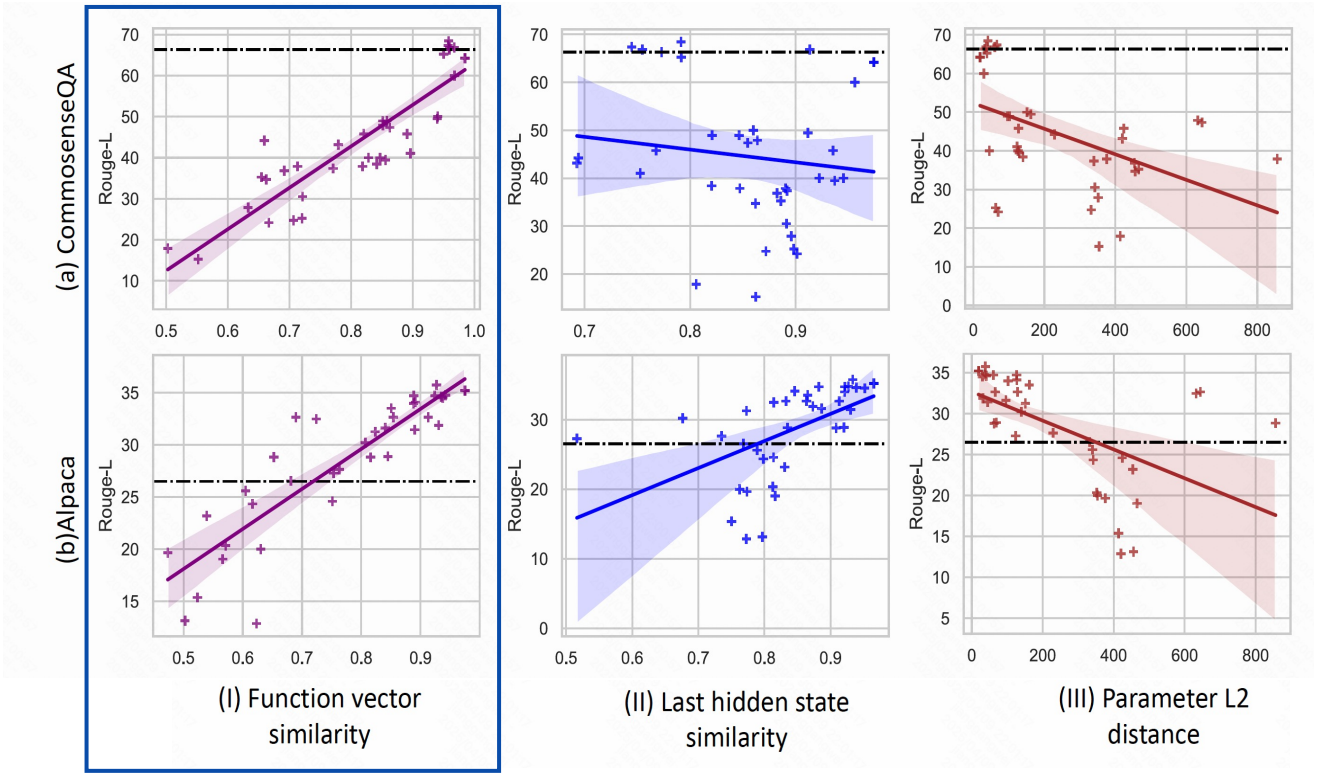
$a^0_{T^e}$ $a^1_{T^e}$ ... $a^j_{T^e}$ ... Performance

$\theta^0_{T^e}$ $\theta^1_{T^e}$ ... $\theta^j_{T^e}$ ... Function vector



Why Forgetting Happens? The Role of Function Vectors

When the similarity of the FV of the evaluation task before and after learning is high, forgetting is relatively mild.



$\text{Cosine}\left(\theta_{Te}^{0}, \theta_{Te}^{j}\right)$

Why Forgetting Happens? Characterization Hypothesis

We hypothesize that the intrinsic cause of forgetting is the **shift** in task function activation rather than overwriting previous functions

Reformulate LLM as latent variable model:

$$P_M(y \mid x) = \int_\Theta P_M(y \mid \theta, x) P_M(\theta \mid x) d\theta$$

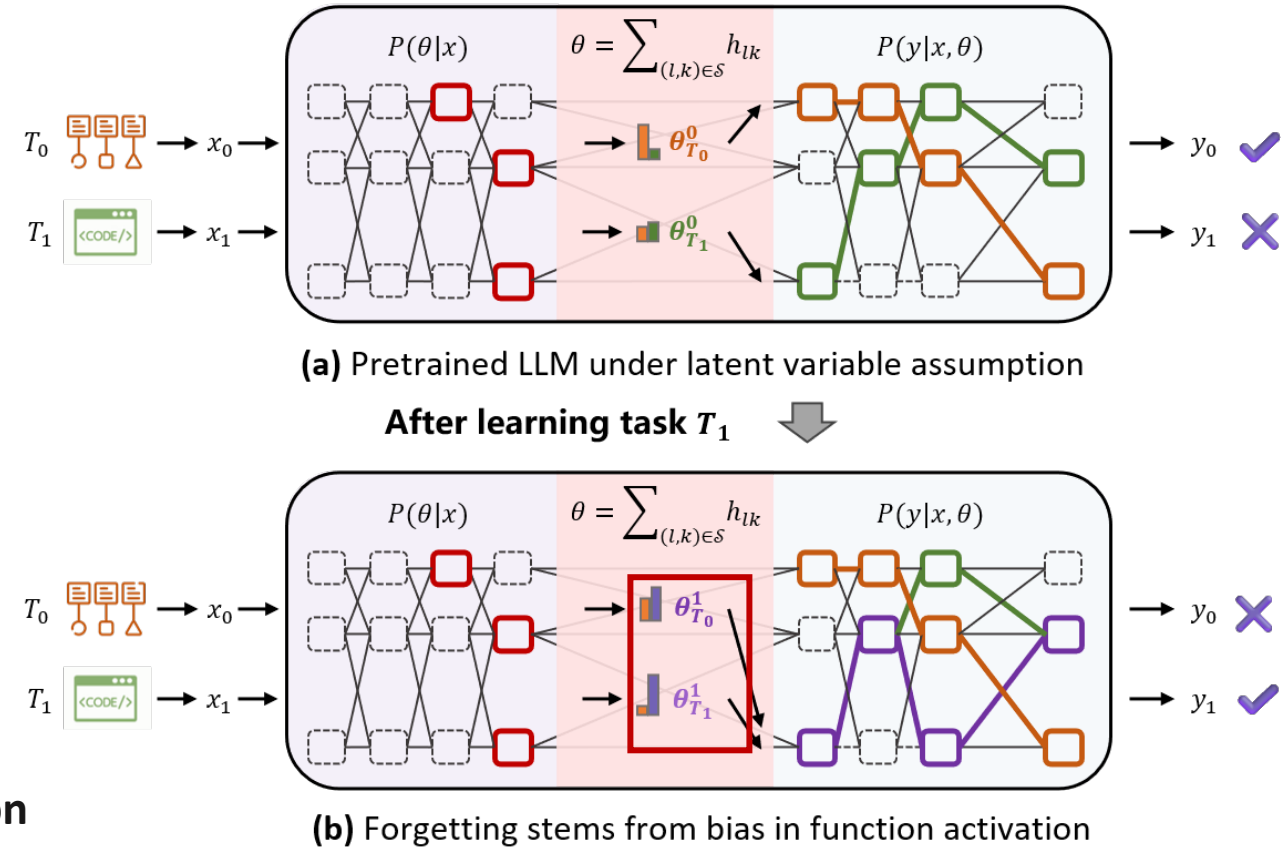
task-specific function

activation of function

Reformulate function vector hypothesis:

$$P_M\left(y \mid x, \theta_T = \sum_{(l,k)\in S} h_{lk}\right) \rightarrow f_T(y \mid x)$$

shift in θ_T <=> shift in activation of function



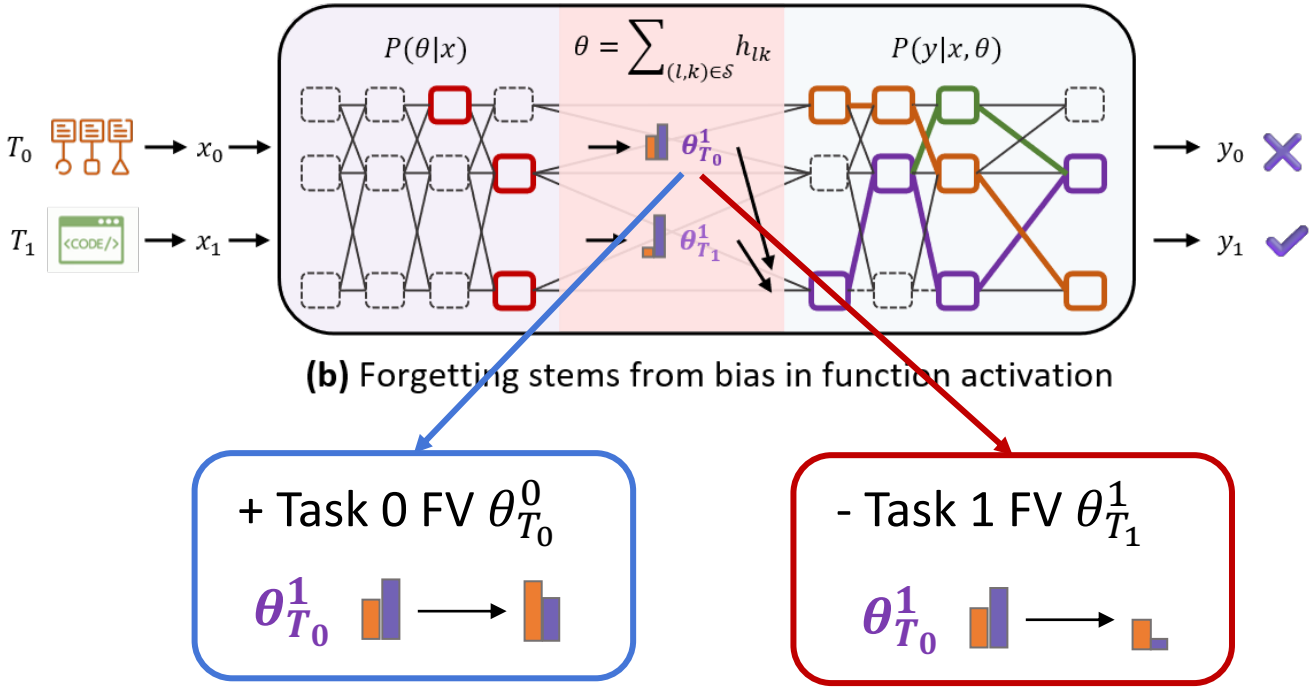
(a) Pretrained LLM under latent variable assumption

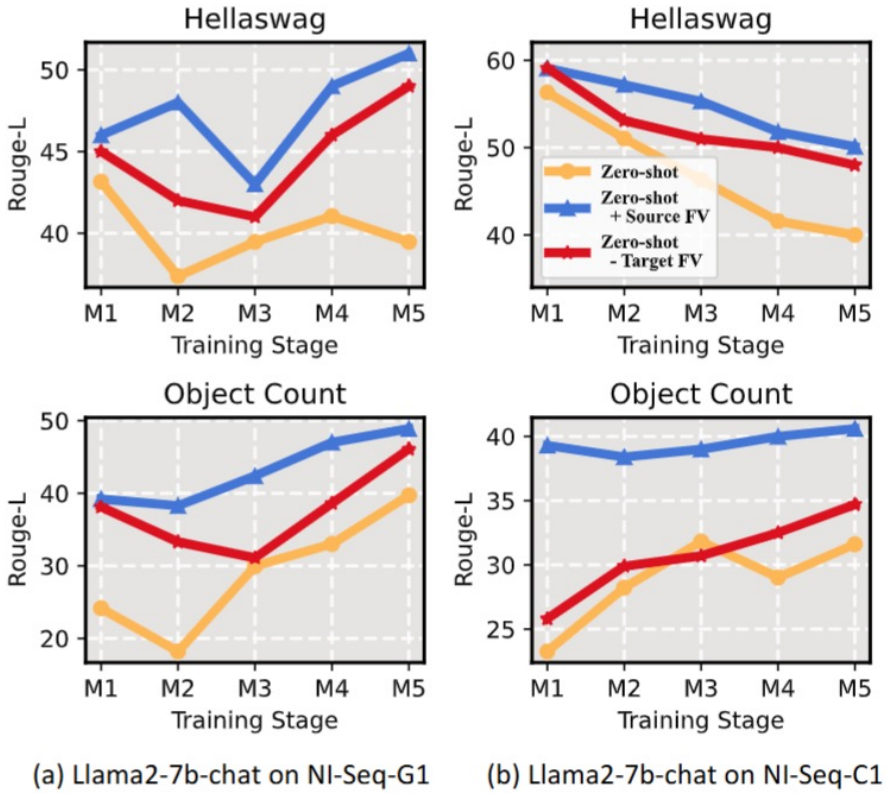
(b) Forgetting stems from bias in function activation

Why Forgetting Happens? Supporting Evidences

By manipulating the function vectors during forward, the model can recover task performance / mitigate forgetting.

Intervention experiments





How to Mitigate Forgetting?

A simple yet efficient design to mitigate forgetting, through **two regularization terms** to prevent the shift of function vector activation.

Function vector consistency loss

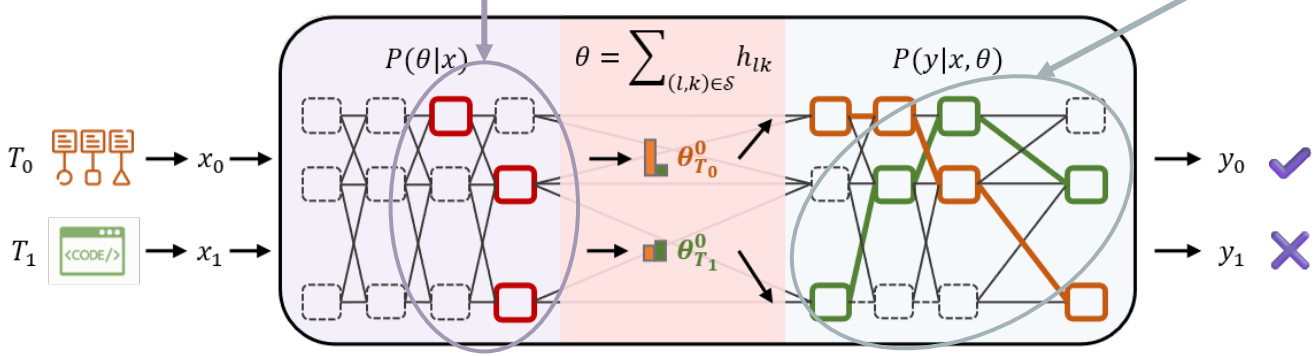
$$\ell_{FV} = \sum_{(l,k)\in\mathcal{S}} d\left(h_{lk}^{M_{j-1}}(x), h_{lk}^{M}(x)\right)$$

FV-guided KL-divergence loss

$$\ell_{KL} = KL[P_M(\cdot \mid x) \,||\, P_{M_{j-1}^{h_l \to h_l + \theta^0_{T_j}}}(\cdot \mid x)]$$

prevent FV shift

prompt model to use its original task function



$$P_M(y \mid x) = \int_\Theta P_M(y \mid \theta, x) P_M(\theta \mid x) d\theta$$

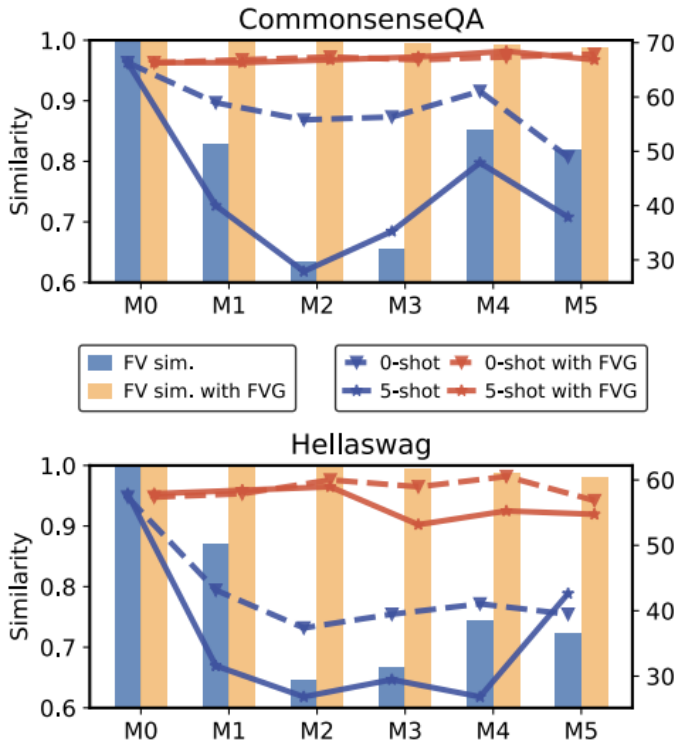
task-specific function

activation of function

Effectively Mitigate Forgetting

	Method	NI-Seq-G1 GP ↑	NI-Seq-G1 IP ↑	NI-Seq-G1 FP ↑	NI-Seq-C1 GP ↑	NI-Seq-C1 IP ↑	NI-Seq-C1 FP ↑	NI-Seq-M1 GP ↑	NI-Seq-M1 IP ↑	NI-Seq-M1 FP ↑
Llama2-7b-chat	M_0	49.85	54.43		49.85	54.43		49.85	54.43	
	LoraInc	47.16	30.94	19.35	45.83	27.71	83.80	47.55	37.23	54.33
	+FVG	**+3.34**	**+25.25**	**+2.84**	**+3.98**	**+25.53**	**+1.70**	**+2.65**	**+15.78**	**+3.52**
	Ewc	33.48	26.87	17.72	46.08	38.76	85.00	44.47	41.69	55.85
	+FVG	**+15.73**	**+27.18**	**+0.85**	**+3.11**	**+15.96**	**+0.37**	**+6.18**	**+13.99**	**+0.01**
	O-lora	45.15	31.90	22.67	41.54	20.54	79.33	50.16	39.52	56.94
	+FVG	**+4.89**	**+23.59**	**+0.11**	**+8.38**	**+33.93**	**+6.2**	**+0.29**	**+14.95**	**-0.42**
	InsCL	45.80	41.79	27.14	44.03	35.69	81.67	49.76	43.09	60.83
	+FVG	**+2.65**	**+8.30**	**+0.91**	**+5.00**	**+16.11**	**+1.23**	**+0.98**	**+8.32**	**-2.22**
Llama2-8b-c.	M_0	56.61	60.61		56.61	60.61		56.61	60.61	
	LoraInc	45.51	39.85	21.10	51.89	54.63	82.10	48.00	47.82	52.63
	+FVG	**+7.79**	**+15.31**	**+3.10**	**+3.99**	**+5.19**	**+0.30**	**+4.88**	**+4.75**	**+5.78**
	InsCL	46.48	49.46	28.53	52.11	57.30	82.50	49.46	53.50	60.92
	+FVG	**+6.60**	**+8.06**	**-0.85**	**+3.52**	**+1.58**	**-0.60**	**+4.34**	**+2.75**	**-2.80**
Mistral-7b-i.	M_0	47.55	57.51		47.55	57.51		47.55	57.51	
	LoraInc	42.81	38.82	19.78	48.00	53.00	85.4	49.79	51.02	57.01
	+FVG	**+4.49**	**+16.61**	**+0.64**	**+2.35**	**+2.67**	**-0.50**	**-2.41**	**+4.02**	**+0.43**
	InsCL	43.46	51.06	25.78	40.77	49.49	83.03	42.38	52.27	58.01
	+FVG	**+2.71**	**+4.64**	**-0.30**	**+6.75**	**+4.27**	**+2.07**	**+6.13**	**+3.40**	**-0.84**

Significantly alleviate the forgetting of model performance



0.81 -> 0.98

0.72 -> 0.96

(a) Llama2-7b-chat on NI-Seq-G1 w/wo function vector guided training

Successfully prevent the shift of the function vector

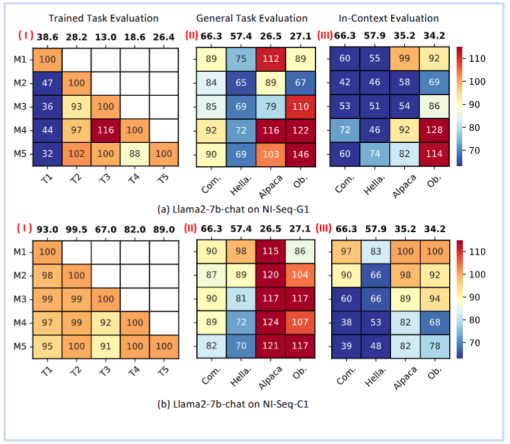
Thank you!

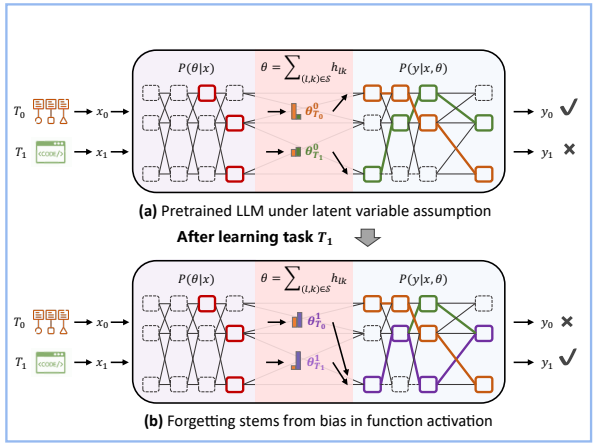
Q & A

Investigate forgetting

Characterize forgetting

Mitigate forgetting





$$\ell_{FV} = \sum_{(l,k)\in\mathcal{S}} d\left(h_{lk}^{M_{j-1}}(x), h_{lk}^{M}(x)\right)$$

$$\ell_{KL} = KL[P_M(\cdot \mid x) \| P_{M_{j-1}^{h_l \to h_l + \theta^0_{T_j}}}(\cdot \mid x)]$$

Welcome to join me for a discussion in the poster time
Thu 24 Apr 3 p.m. CST — 5:30 p.m. CST, Hall 3