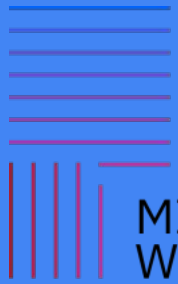
Learning to Route Among Specialized Experts for Zero-Shot Generalization

Mohammed Muqeeth, Haokun Liu, Yufan Liu, Colin Raffel

THE UNIVERSITY of NORTH CAROLINA at CHAPEL HILL

UNIVERSITY OF TORONTO

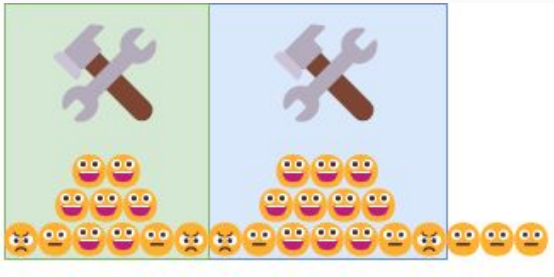
VECTOR INSTITUTE



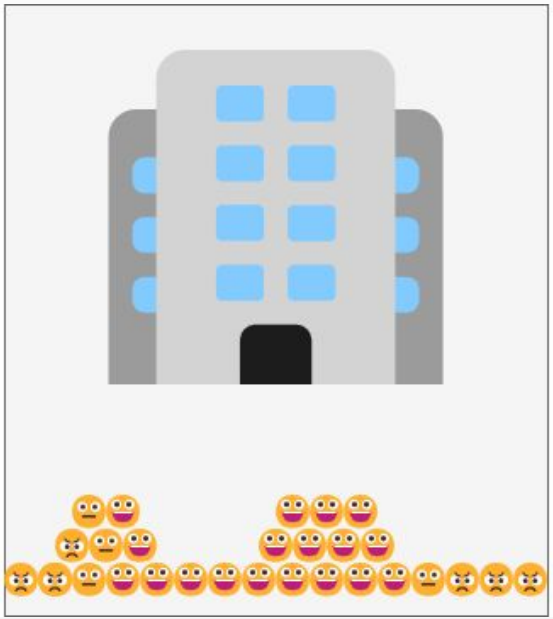
MIT-IBM Watson AI Lab

Before it all happened…

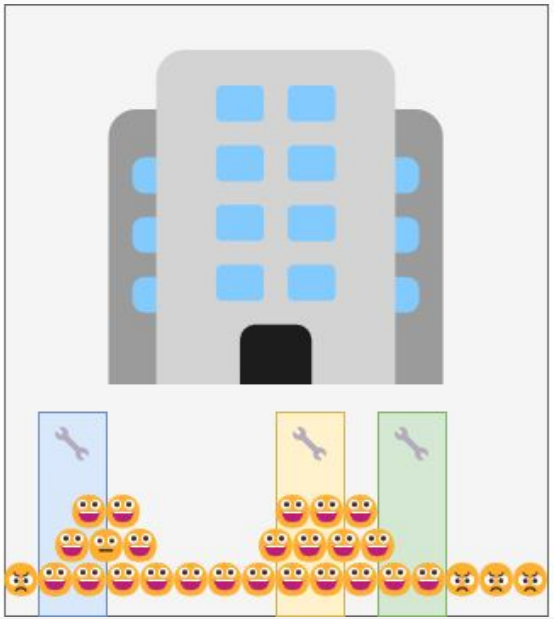
DL



LLM



PEFT



DL, LLM, PEFT (I don’t mean they are equally significant)

Big companies:

→ Can build “General AIs”

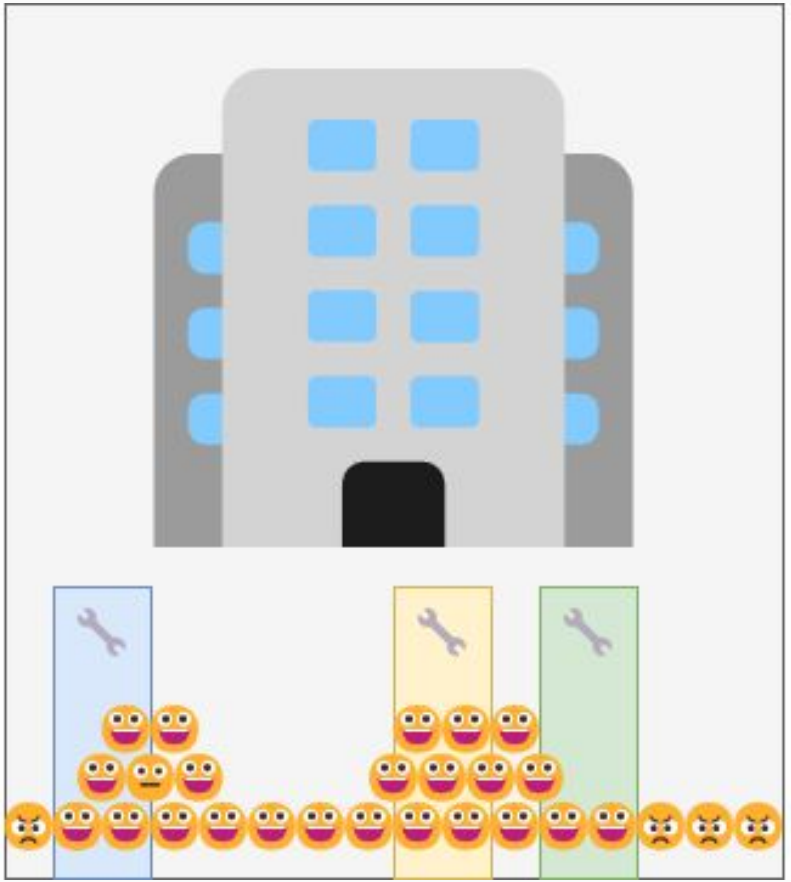
Technicals:

→ Can build specialized models

Average users:

→ Know if they are satisfied

Now, can we integrate specialized models back into general models, to save the trouble of selection and improve the well-being of users not covered by specialized models?

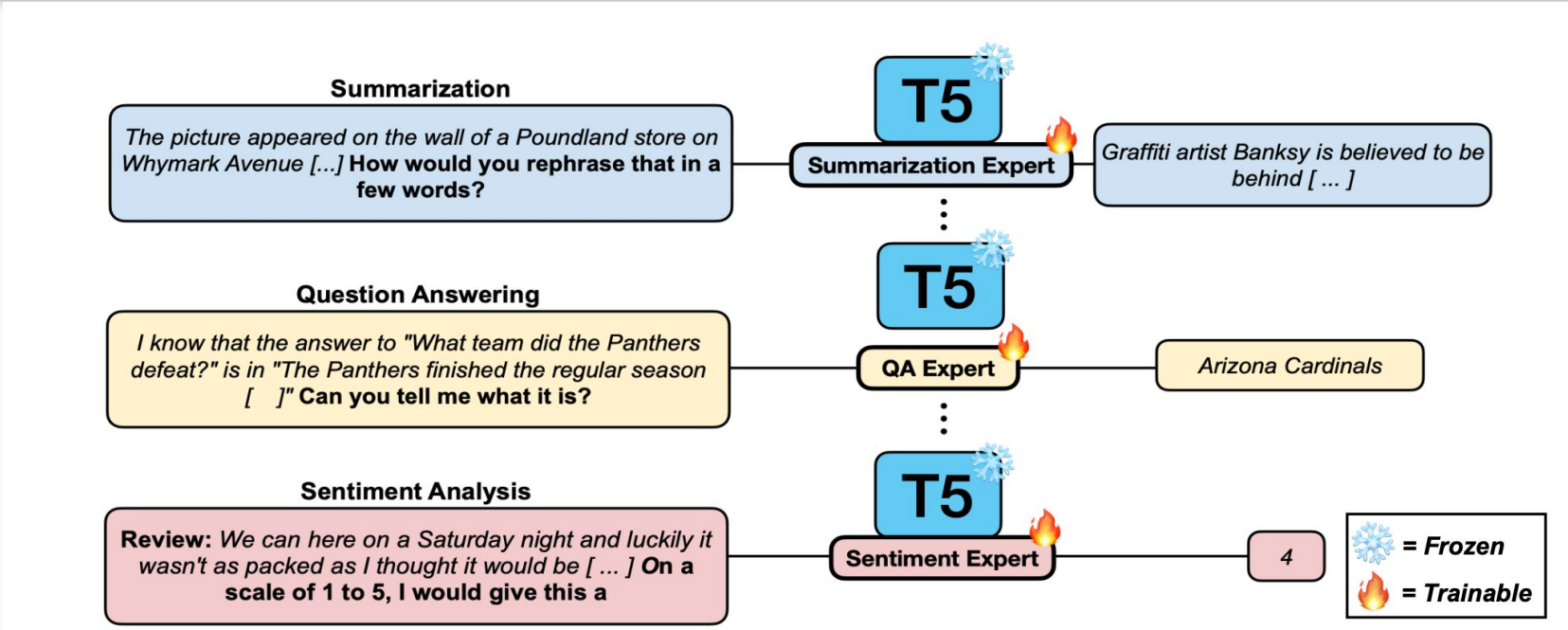


Can we collaboratively improve zero-shot generalization from specialized experts?

Setting

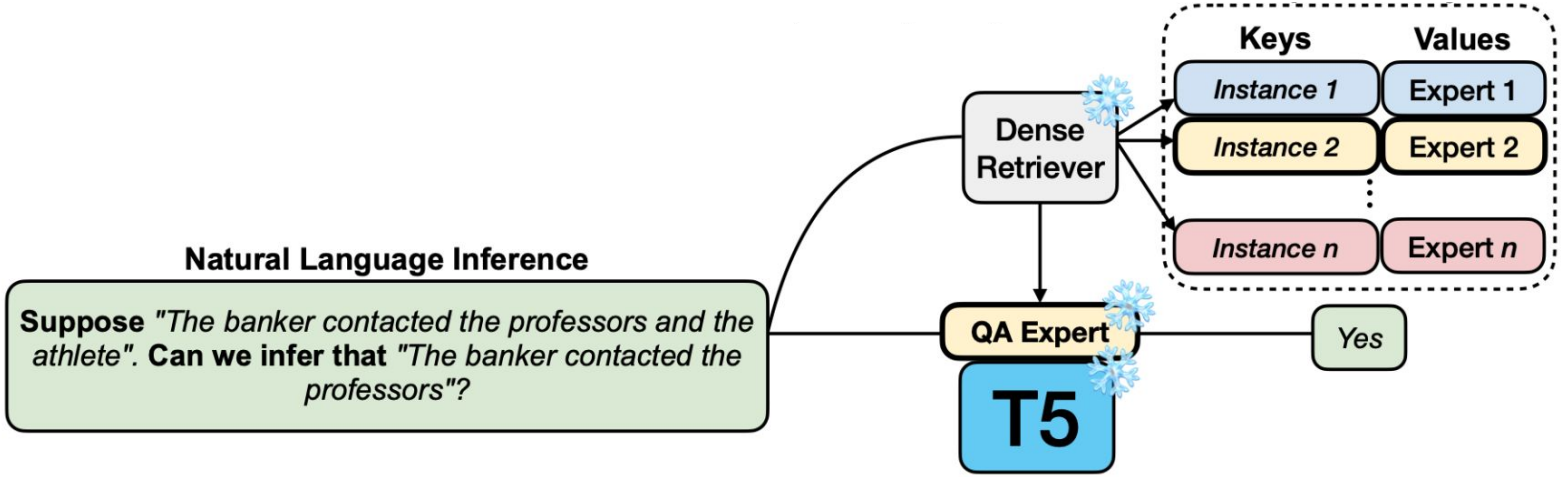
- Contributors use base model and perform PEFT to build specialized experts
- No significant extra work or constraints from contributors
- Only trained parameters are shared, not datasets
- Improve zero-shot performance on unseen tasks

Building Experts



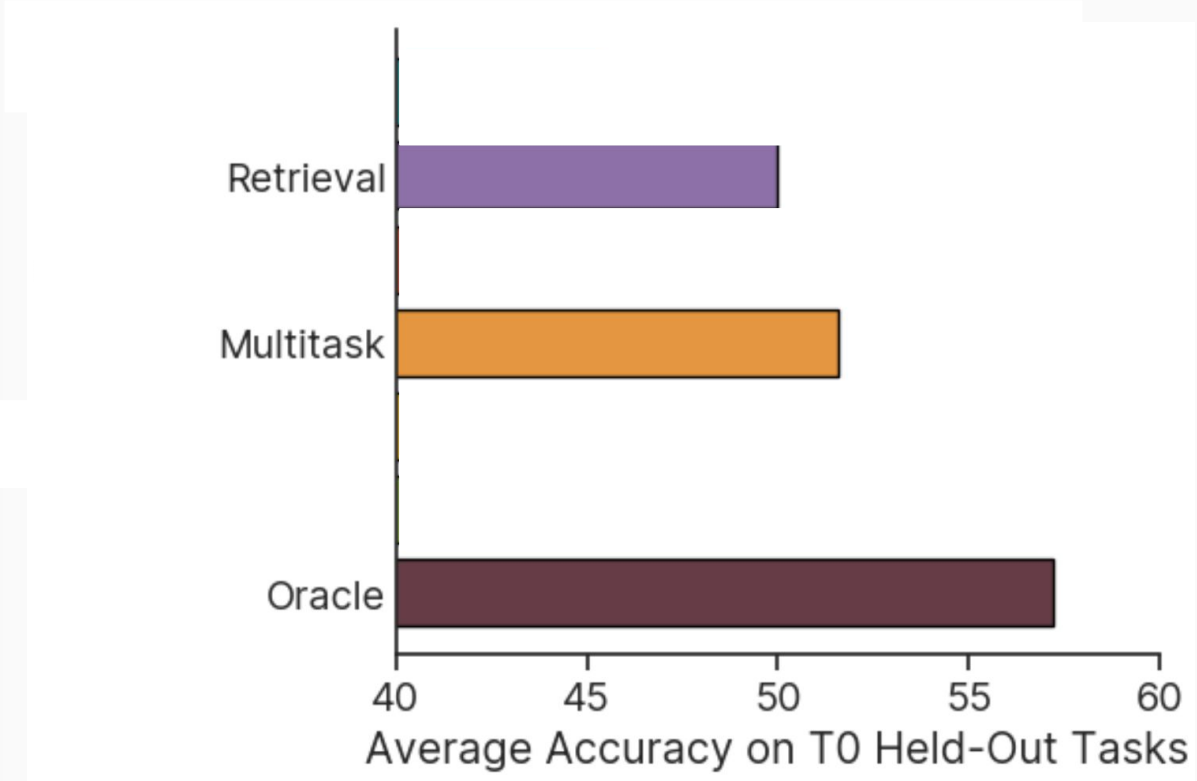
From "Exploring the Benefits of Training Expert Language Models over Instruction Tuning" by Jang et al.

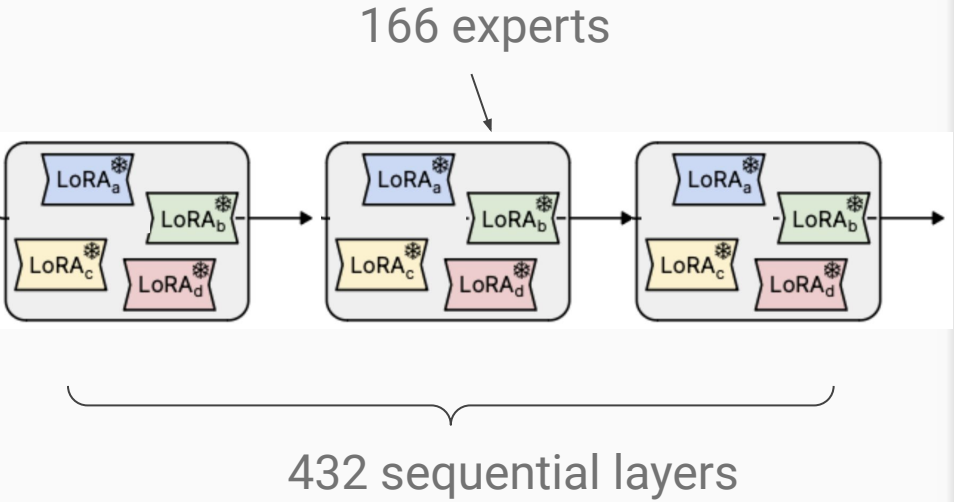
Post-Hoc Routing by Retrieval-of-Experts



From "Exploring the Benefits of Training Expert Language Models over Instruction Tuning" by Jang et al.

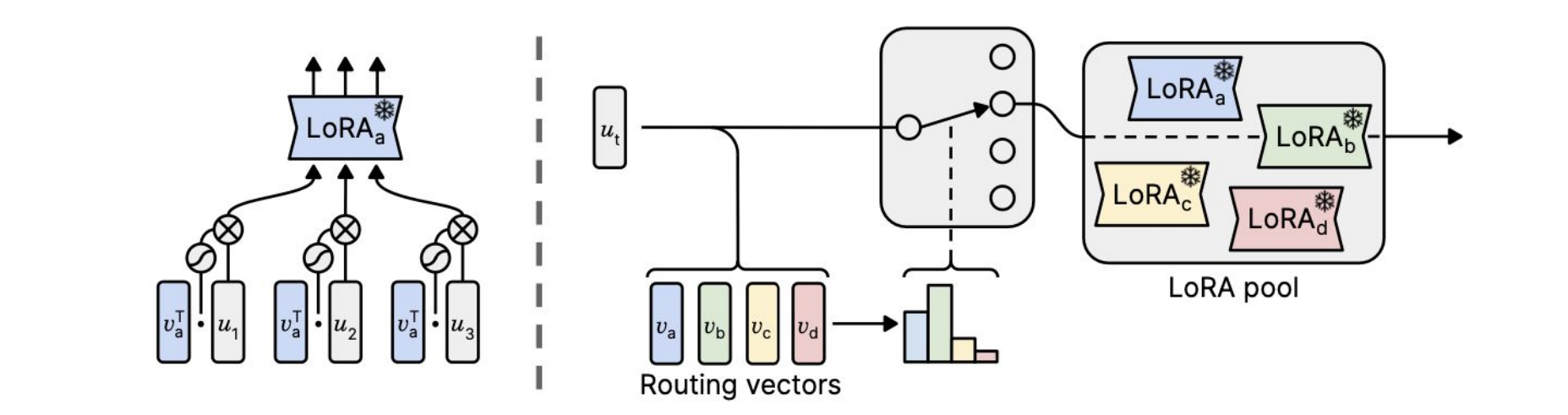
- Large gap to Oracle where the best expert is chosen per dataset
- Relies on a single best available expert and cannot compose knowledge from multiple experts when helpful





Post-Hoc Adaptive Tokenwise Gating Over an Ocean of Specialized Experts

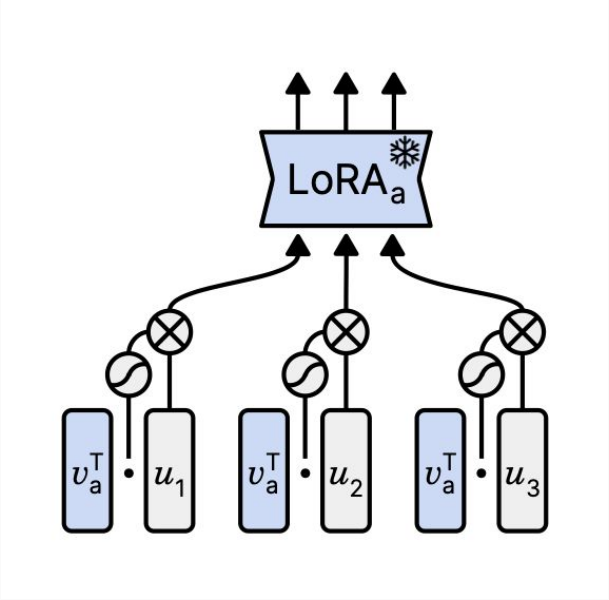
PHATGOOSE



Training gates

Combining gates to make an MoE at a given layer

PHATGOOSE
Training gates



After expert training, a gating vector v_a is learnt by modifying the input to the expert modules

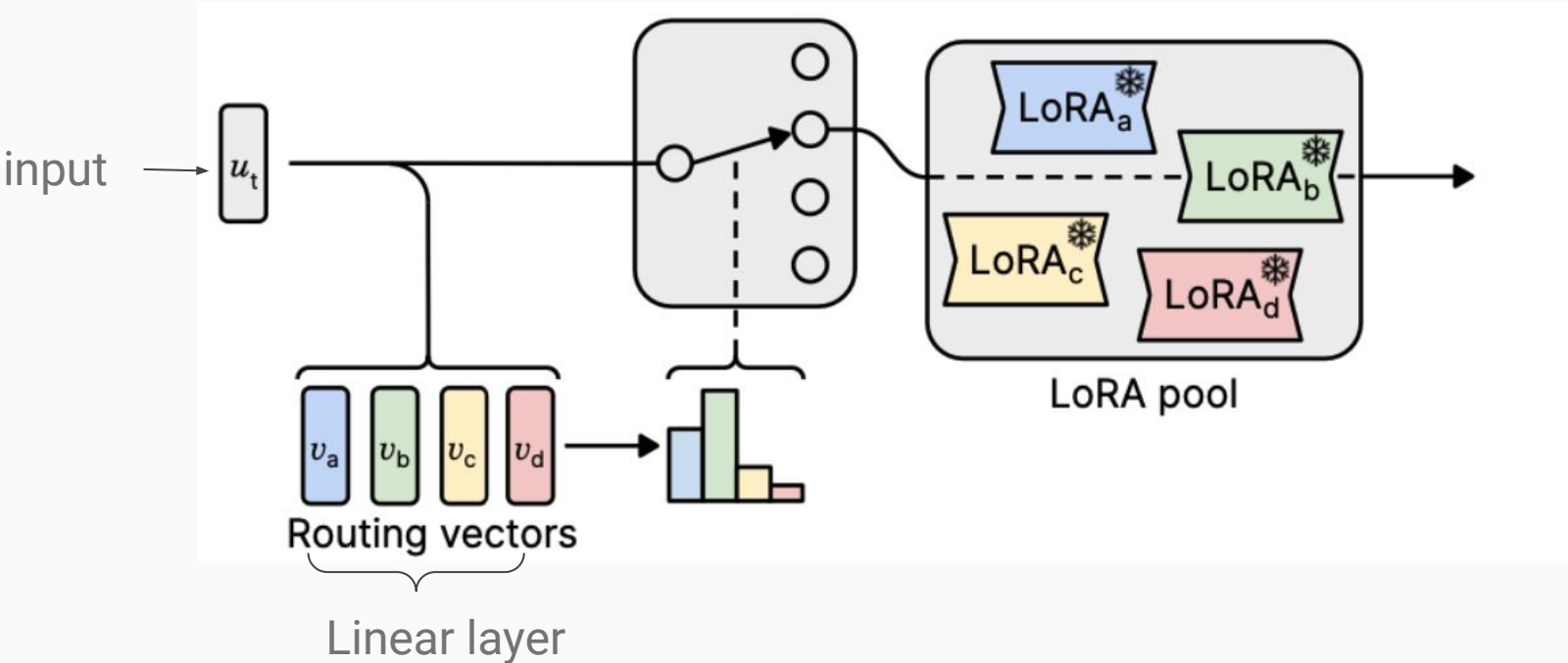
LoRA $Wu_t + BAu_t$

LoRA with modified input

$$Wu_t + BAu_t \underbrace{\sigma(v_a^T u_t)}$$

Sigmoid scaling

PHATGOOSE
Combining gates



Training-free ways to obtain gates

Average Activation

- Given a dataset and PEFT module trained using that dataset, the average over all input activations of the module

Arrow

- LoRA modules is a matrix, top-1 right singular vector of the matrix can act as compact representation of that module

Results and Analysis

Train: 36 experts from T0 Held-in datasets

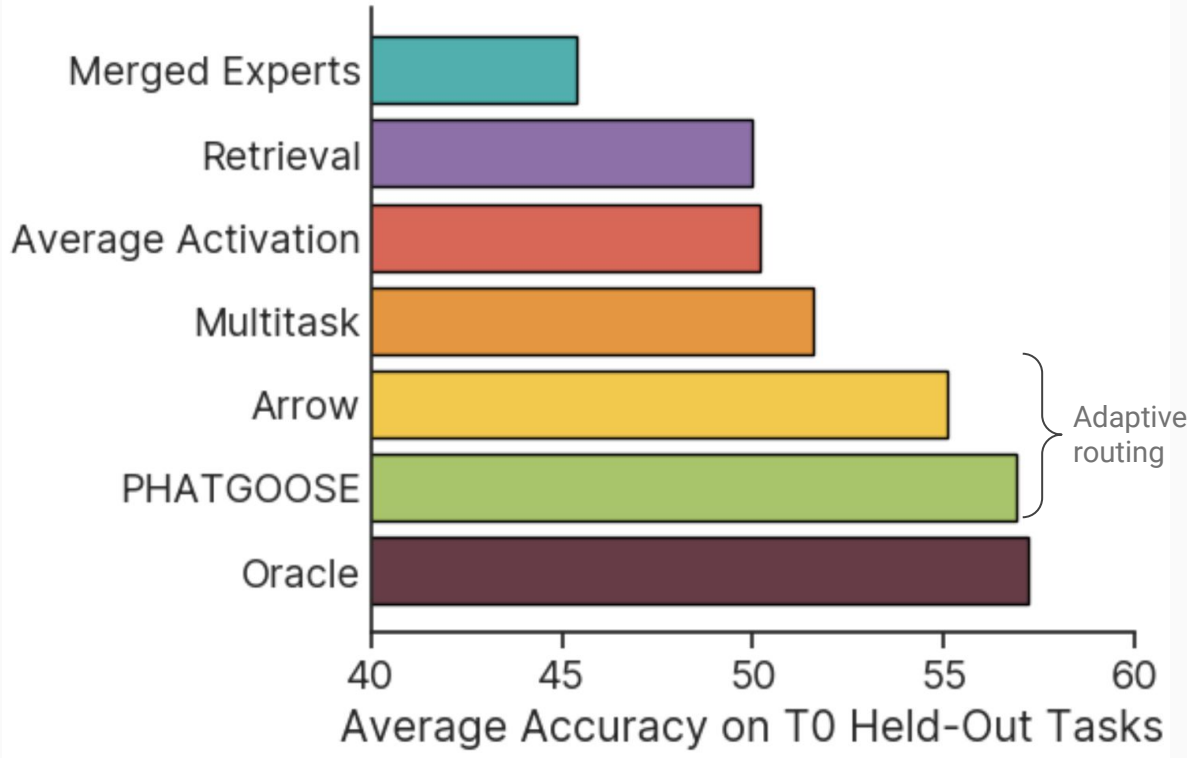
Evaluate: 1) T0 Held-out tasks 2) BIG-Bench tasks

Train: 166 experts from FLAN

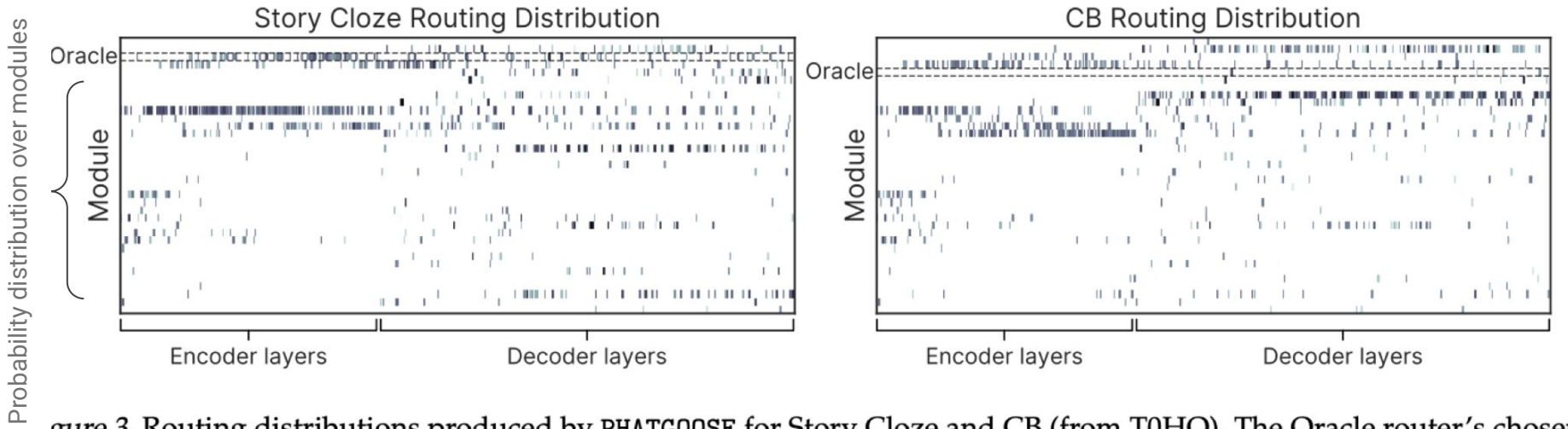
Evaluate: BIG-Bench tasks

Results

- PHATGOOSE surpasses past methods and gets close to Oracle
- Adaptive approaches outperform Retrieval method
- Promise for decentralized training of experts



Routing in PHATGOOSE



gure 3. Routing distributions produced by PHATGOOSE for Story Cloze and CB (from T0HO). The Oracle router's chosen module is highlighted by dashed lines. On Story Cloze, PHATGOOSE chooses the Oracle module in the encoder but uses diverse experts in the decoder but nevertheless matches Oracle performance. On CB, PHATGOOSE almost never uses the Oracle module and produces significantly better performance by using a wide range of modules.

Ablation

Method	Avg	RTE	H-Swag	COPA	WIC	Wino grande	CB	Story Cloze	ANLI-R1	ANLI-R2	ANLI-R3	WSC
Joint training	46.2	54.5	25.3	57.1	50.2	52.3	51.2	60.6	30.4	31.7	32	62.9
PHATGOOSE	51.8	62.7	28	73.1	50.5	53	54.9	90.6	30	31.3	32.6	63.4

On T5 large, Joint training of gates and LoRA experts results in poor performance compared to training the gates post-hoc

Scaling from 36 T0 experts to 166 FLAN experts

Method	T0 Held-In			FLAN	
	T0HO	BBH	BBL	BBH	BBL
Multitask	51.6	34.9	36.6	38.9	45.4
Oracle	57.2	42.2	43.5	45.5	46.5
Best Individual	52.8	32.3	39.9	34.6	38.6
Retrieval	50	30.9	33.6	31.4	33.1
Arrow	55.1	33.6	34.5	30.6	29.6
Merged Experts	45.4	**35.3**	36	34.6	34
Average Activation	50.2	33.8	35.8	33.5	34
PHATGOOSE	**56.9**	34.9	**37.3**	**35.6**	**35.2**

Knowledge taks hurt in performance

Positive scaling 👌

PHATGOOSE shows a possibility for a generalizable system from specialized experts

Future

- Recalibrating the routing as PEFT updates are made
- Using an offline dataset or expert training datasets to finetune the router
- Distill better routing strategies to improve next generation of GOOSE systems

Thank you for your time!