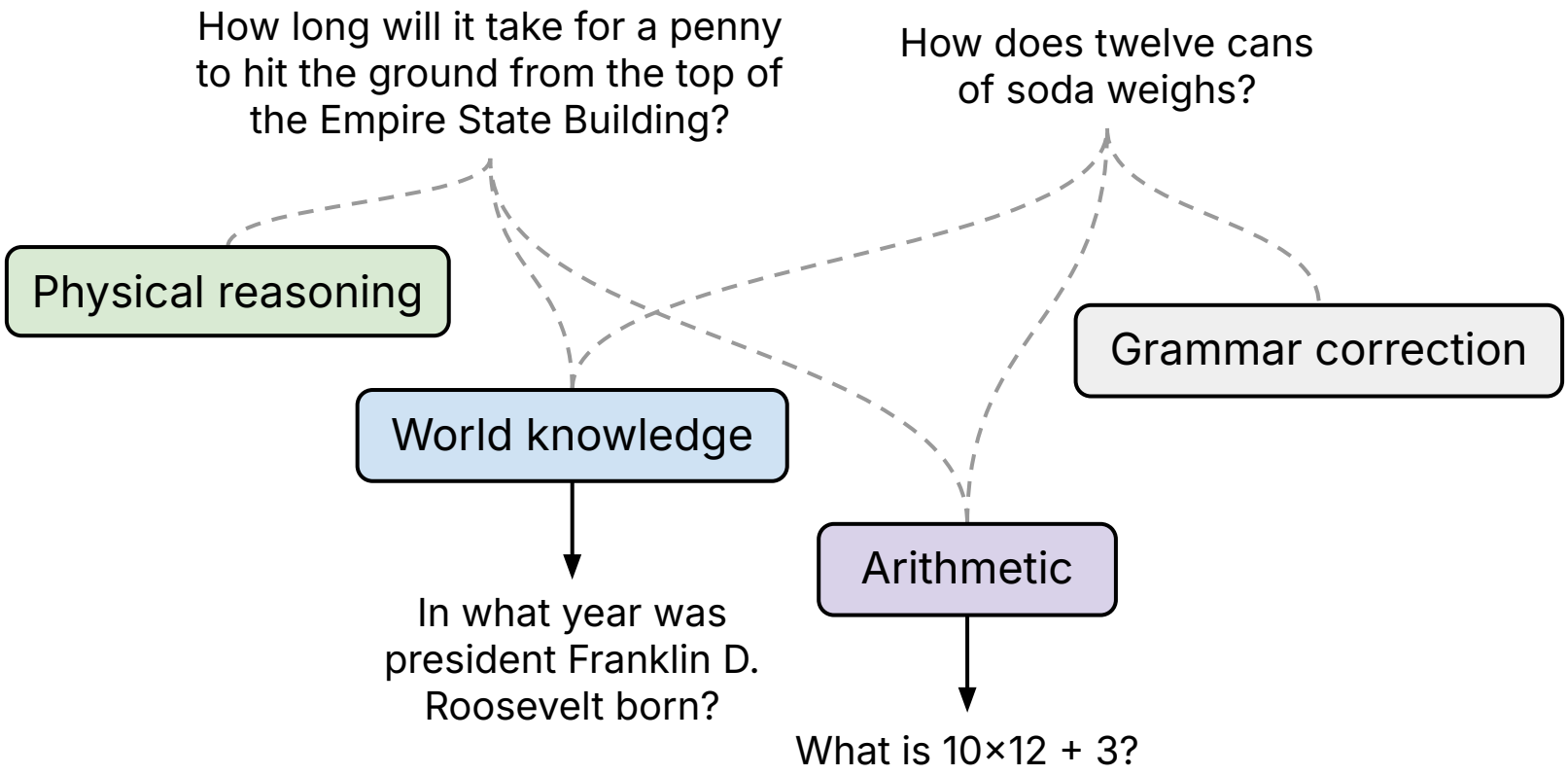
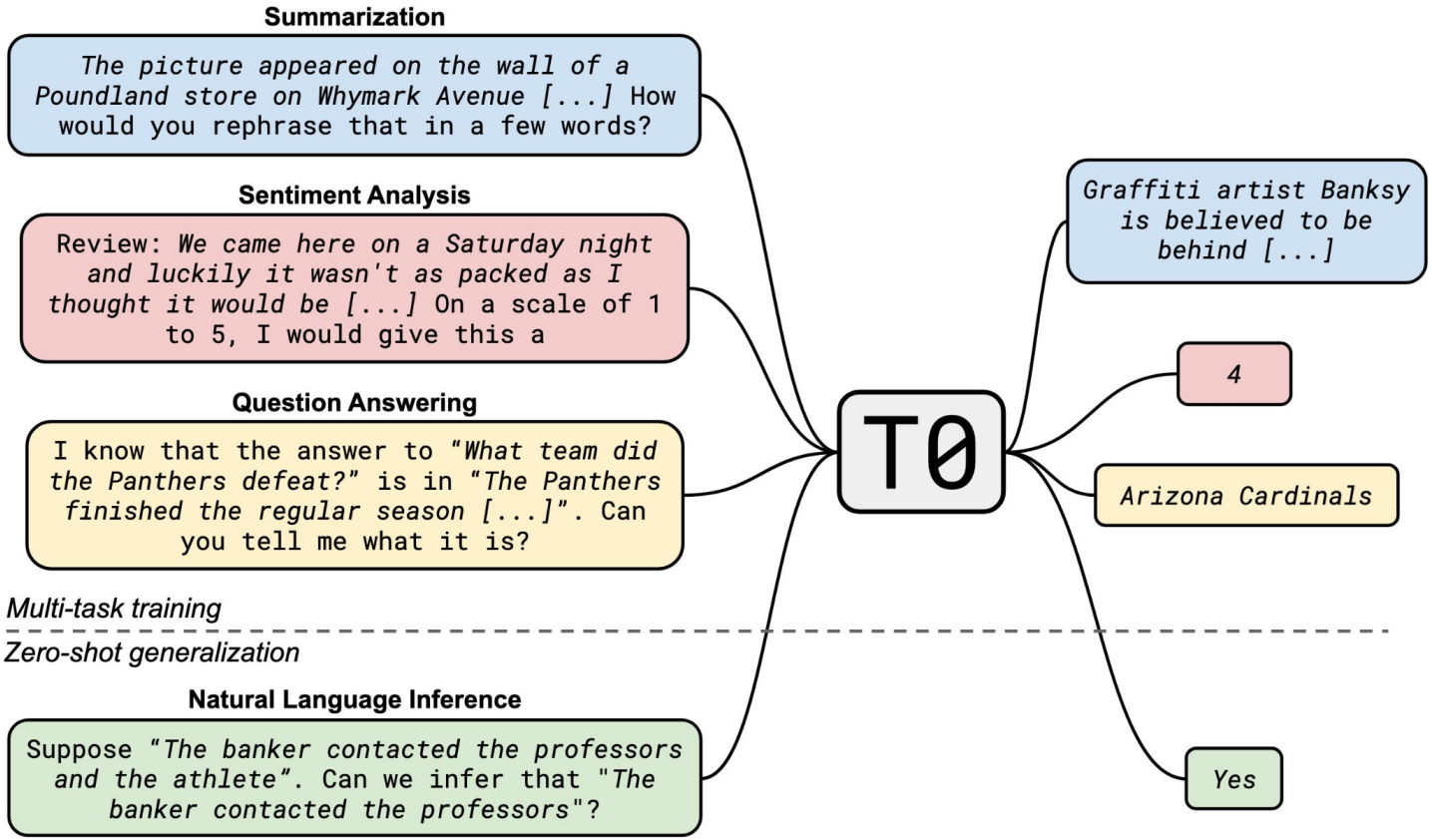
Merging and MoErging for Compositional Generalization

Colin Raffel

Tasks can be considered as a composition of skills

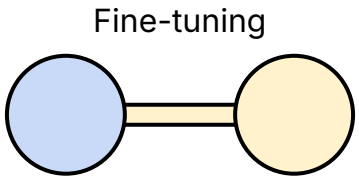


Multitask models can generalize to new tasks

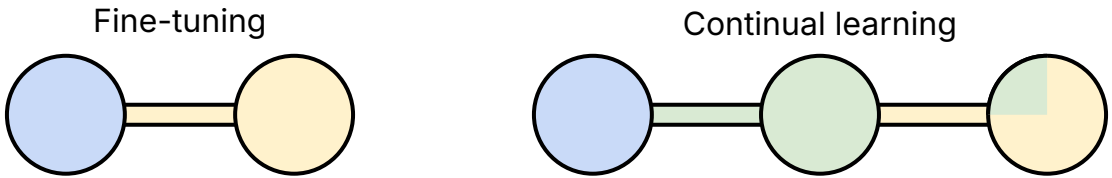


From "Multitask Prompted Training Enables Zero-Shot Task Generalization" by Sanh et al.

Merging models enables new paths for transferring capabilities

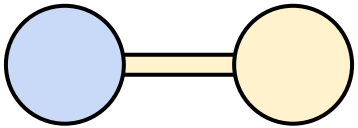


Merging models enables new paths for transferring capabilities

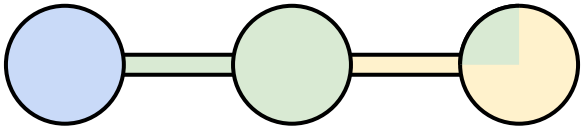


Merging models enables new paths for transferring capabilities

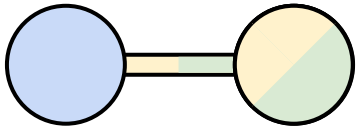
Fine-tuning



Continual learning

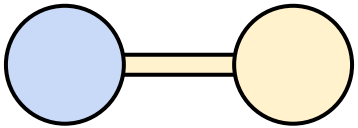


Multitask learning

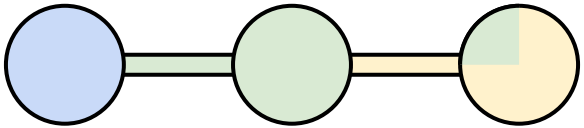


Merging models enables new paths for transferring capabilities

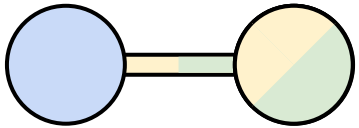
Fine-tuning



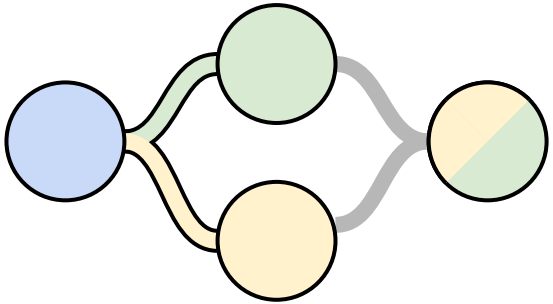
Continual learning



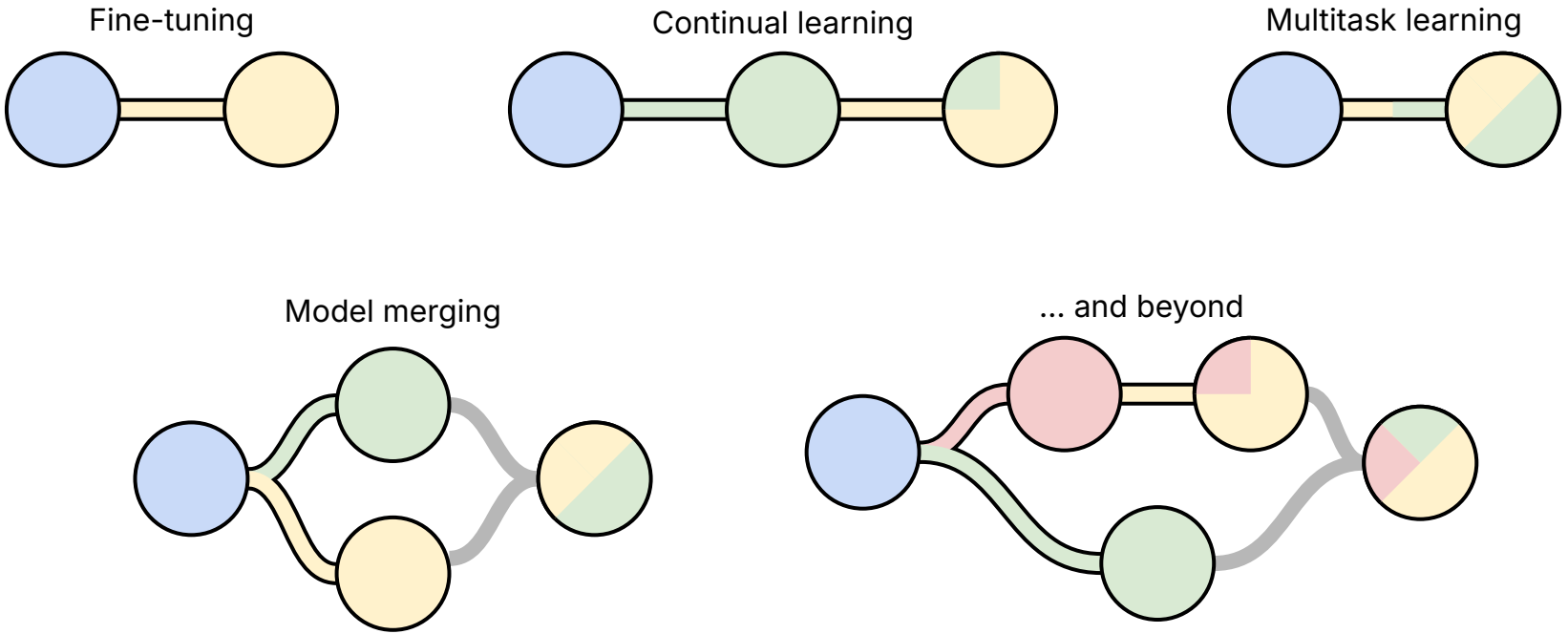
Multitask learning



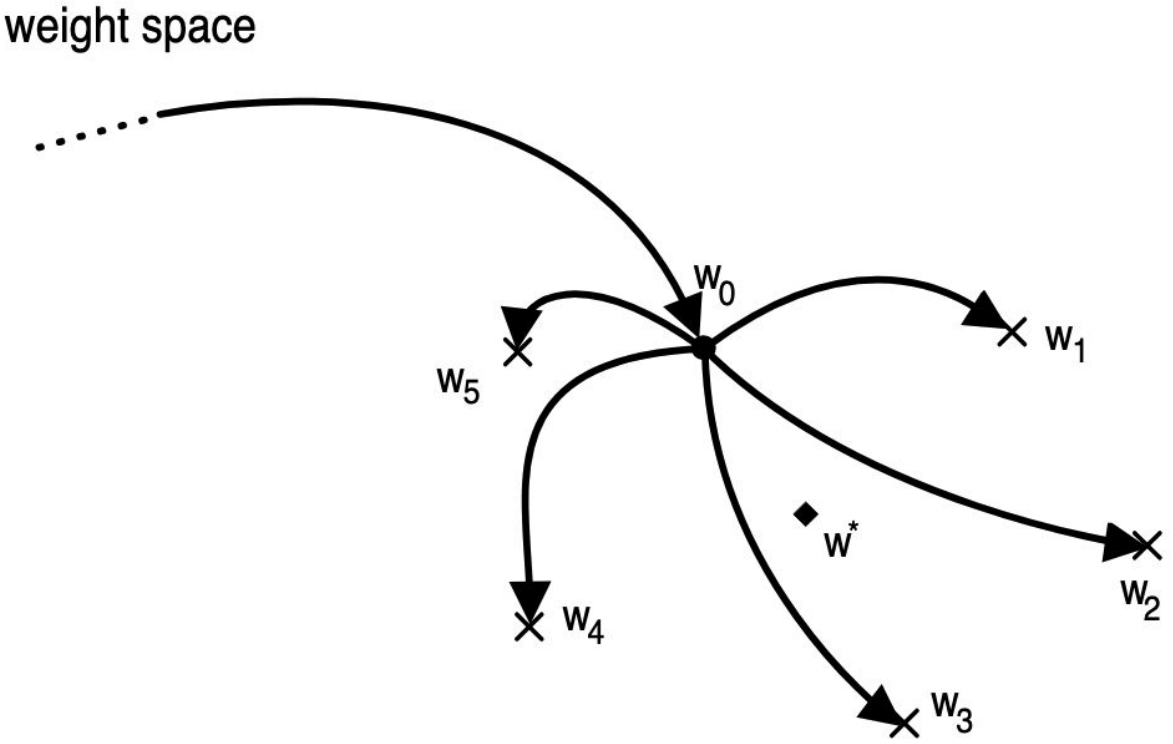
Model merging



Merging models enables new paths for transferring capabilities



"Vanilla" merging is just parameter averaging



From "Weight Averaging for Neural Networks and Local Resampling Schemes" by Utans

Model merging as an optimization problem

$$\arg\max_{\theta} \sum_{i=1}^{M} \lambda_i \log p(\theta|\mathcal{D}_i)$$

From “Merging Models with Fisher-Weighted Averaging" by Matena et al.

Parameter averaging makes an isotropic Gaussian assumption

$$\arg\max_{\theta} \sum_{i=1}^{M} \lambda_i \log p(\theta|\mathcal{D}_i)$$

$$\downarrow \quad \theta \sim \mathcal{N}(\theta_i, \boldsymbol{I})$$

$$\theta^* = \frac{\sum_i \lambda_i \theta_i}{\sum_i \lambda_i}$$

From "Merging Models with Fisher-Weighted Averaging" by Matena et al.

Fisher merging uses the Laplace approximation

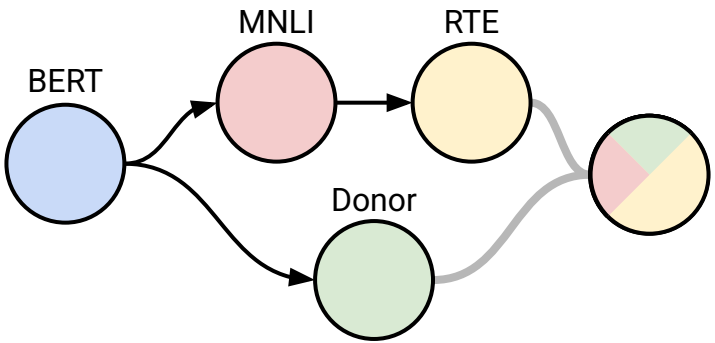
$$\arg\max_{\theta} \sum_{i=1}^{M} \lambda_i \log p(\theta|\mathcal{D}_i)$$

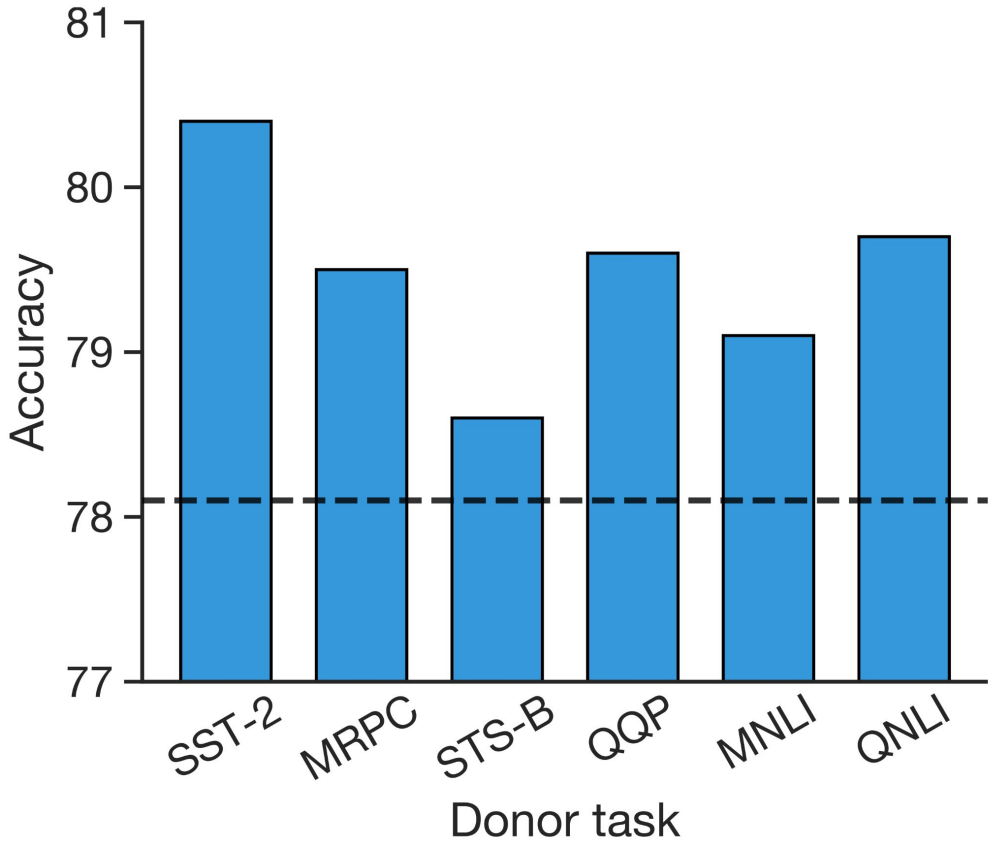
$$\downarrow \quad \theta \sim \mathcal{N}(\theta_i, \hat{F}_i^{-1})$$

$$\theta^* = \frac{\sum_i \lambda_i \hat{F}_i \odot \theta_i}{\sum_i \lambda_i \hat{F}_i}$$

From "Merging Models with Fisher-Weighted Averaging" by Matena et al.

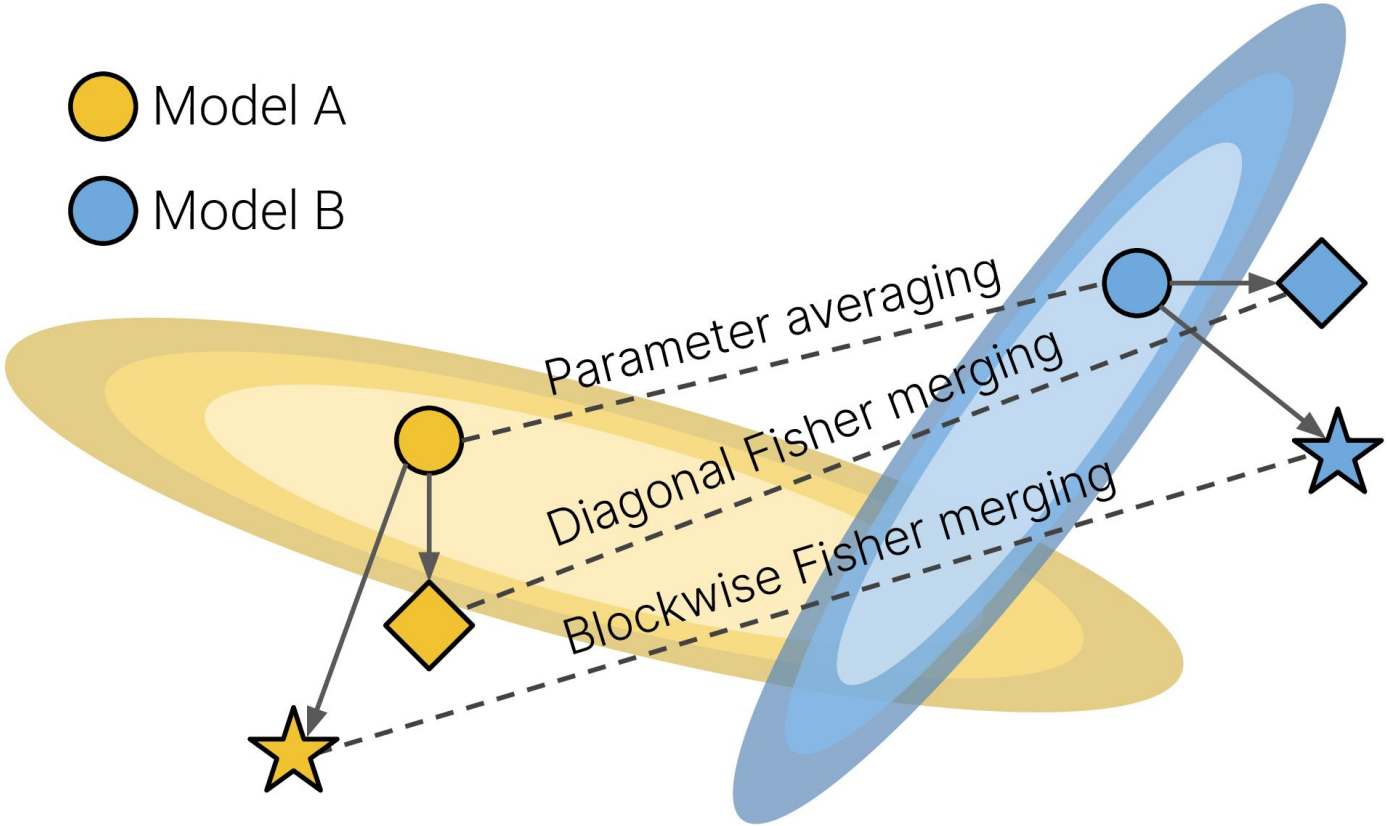
Fisher merging can combine the capabilities of different models





From "Merging Models with Fisher-Weighted Averaging" by Matena et al.

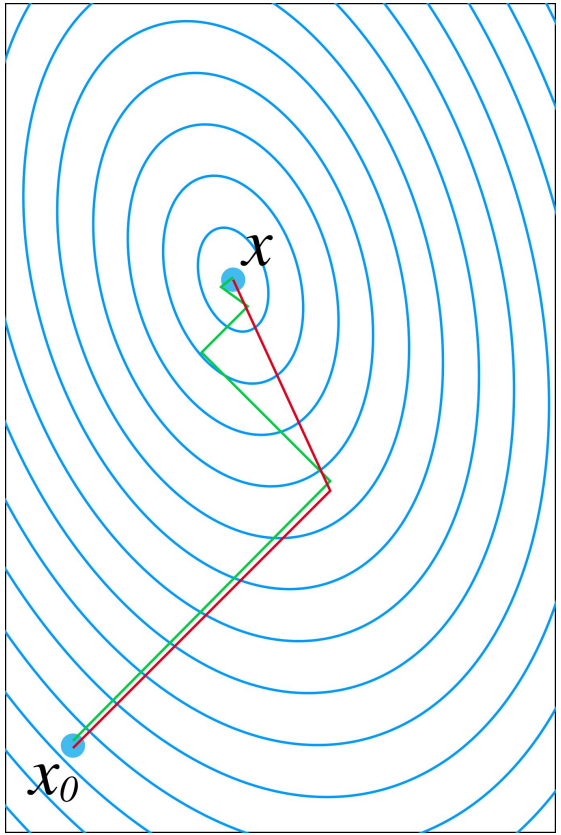
Merging models based on loss landscape curvature



From "Merging by Matching Models in Task Subspaces" by Tam et al.

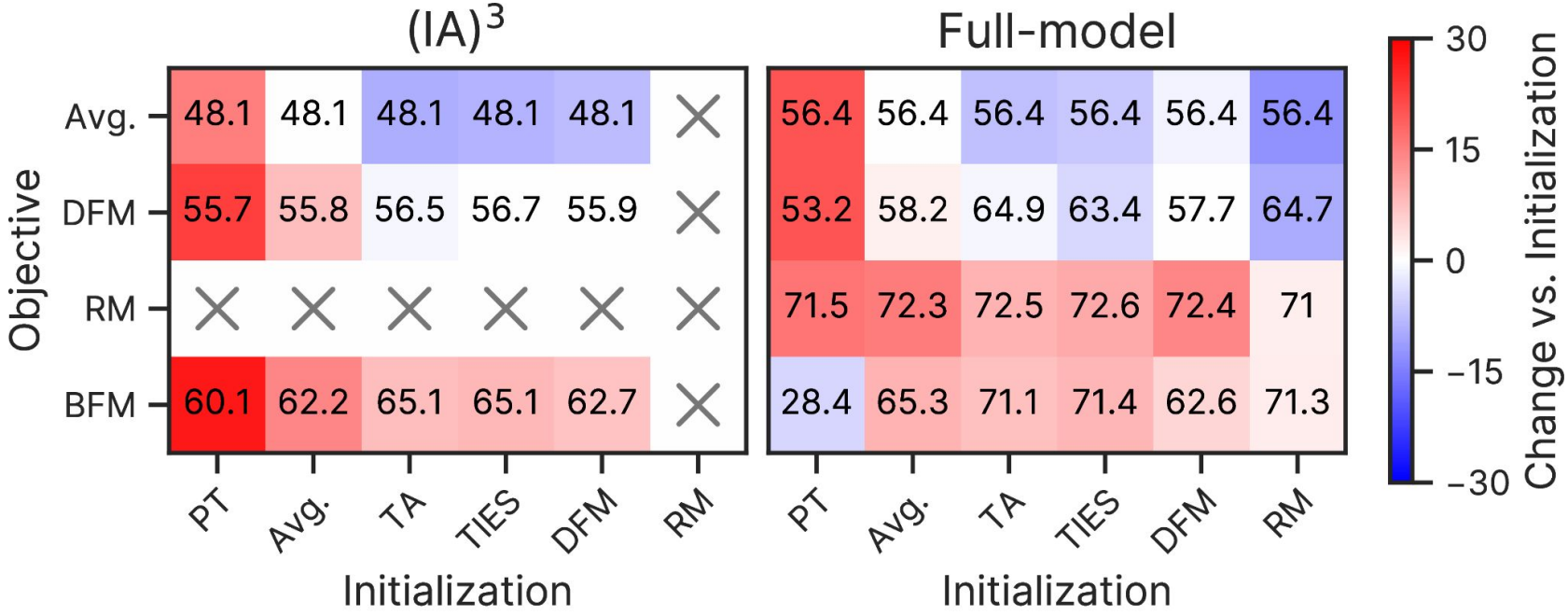
Solving a general merging problem with the conjugate gradient method

$$\boldsymbol{\theta}^* = \left(\sum_{m=1}^{M} \boldsymbol{C}_m \right)^{-1} \left(\sum_{m=1}^{M} \boldsymbol{C}_m \boldsymbol{\theta}_m \right)$$



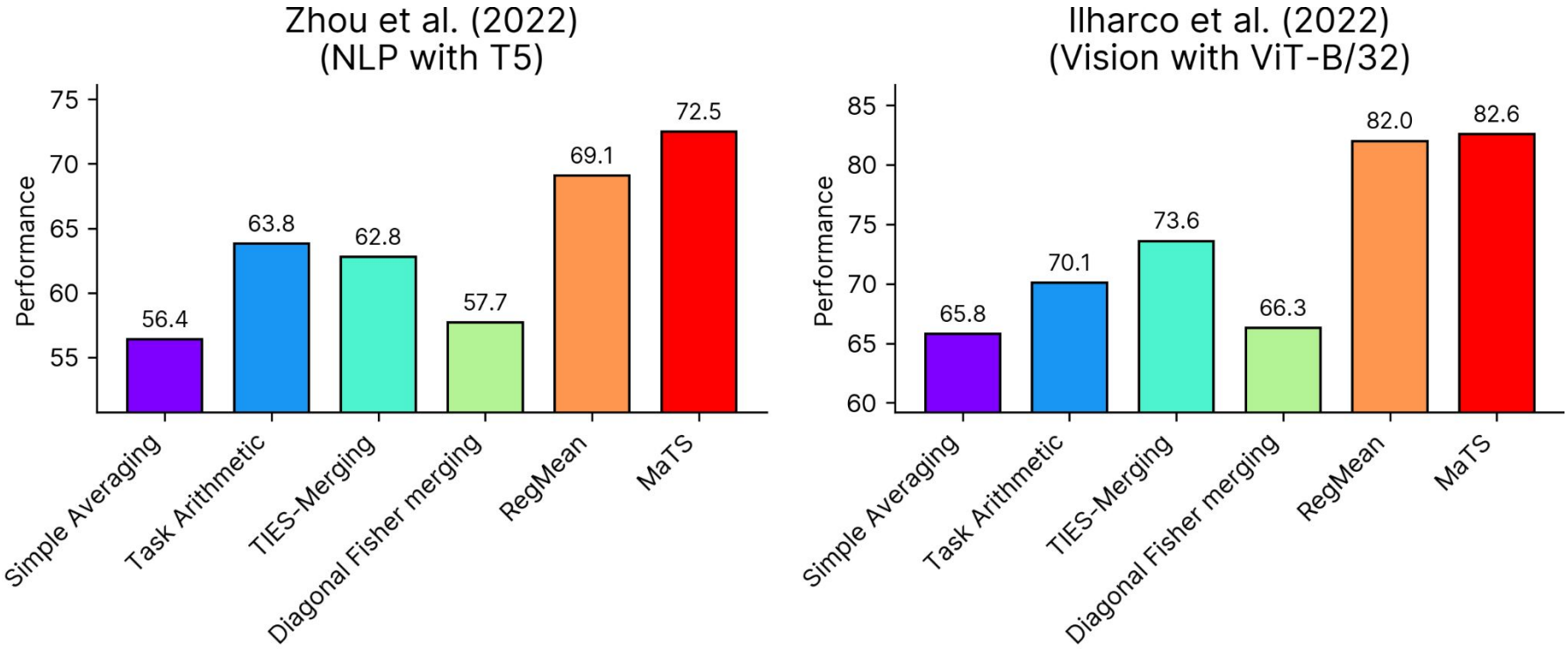
From "Merging by Matching Models in Task Subspaces" by Tam et al.

MaTS allows flexibly combining merging objectives and initializations...



From "Merging by Matching Models in Task Subspaces" by Tam et al.

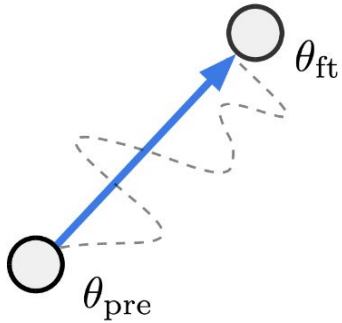
... and achieved state-of-the-art resulted across settings



From "Merging by Matching Models in Task Subspaces" by Tam et al.

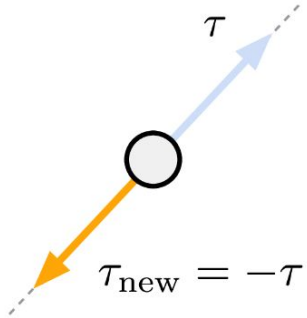
Merging models with task vectors

a) Task vectors



$$\tau = \theta_{\text{ft}} - \theta_{\text{pre}}$$

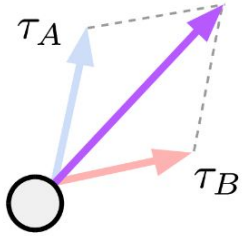
b) Forgetting via negation



Example: making a language model produce less toxic content

c) Learning via addition

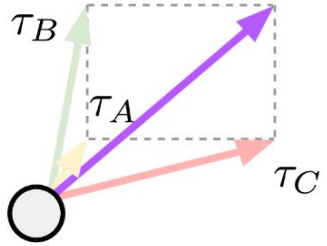
$$\tau_{\text{new}} = \tau_A + \tau_B$$



Example: building a multi-task model

d) Task analogies

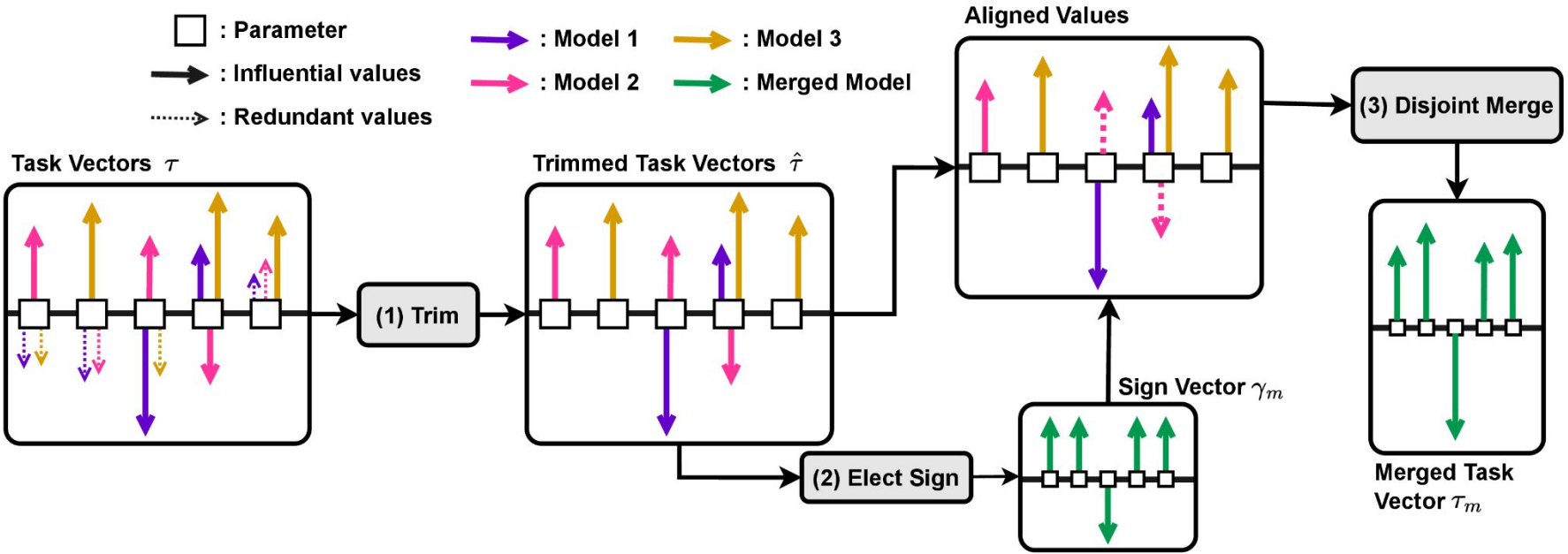
$$\tau_{\text{new}} = \tau_C + (\tau_B - \tau_A)$$



Example: improving domain generalization

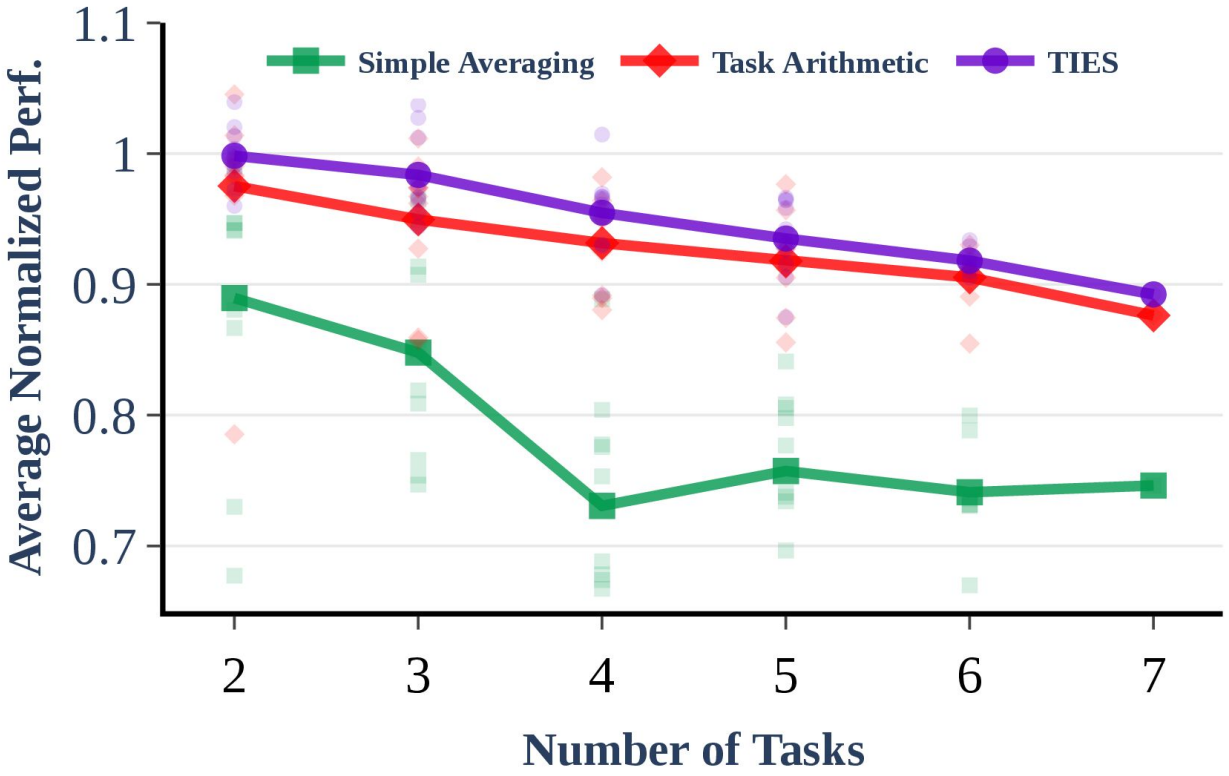
From "Editing Models with Task Arithmetic" by Ilharco et al.

TIES Merging resolves interference between task vectors



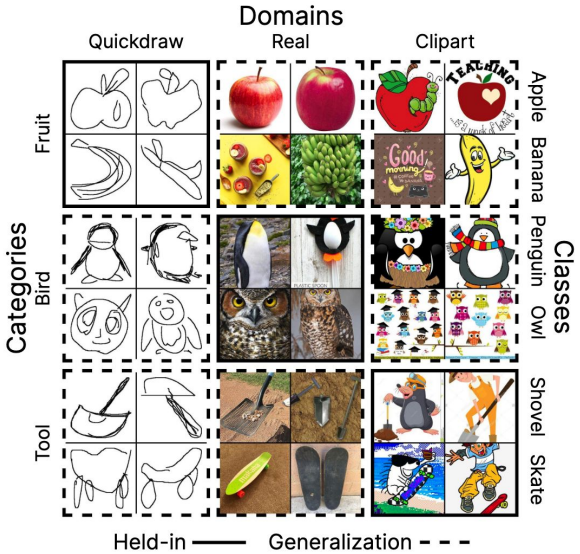
From "Resolving Interference When Merging Models" by Yadav et al.

TIES helps retain specialist model performance



From "Resolving Interference When Merging Models" by Yadav et al.

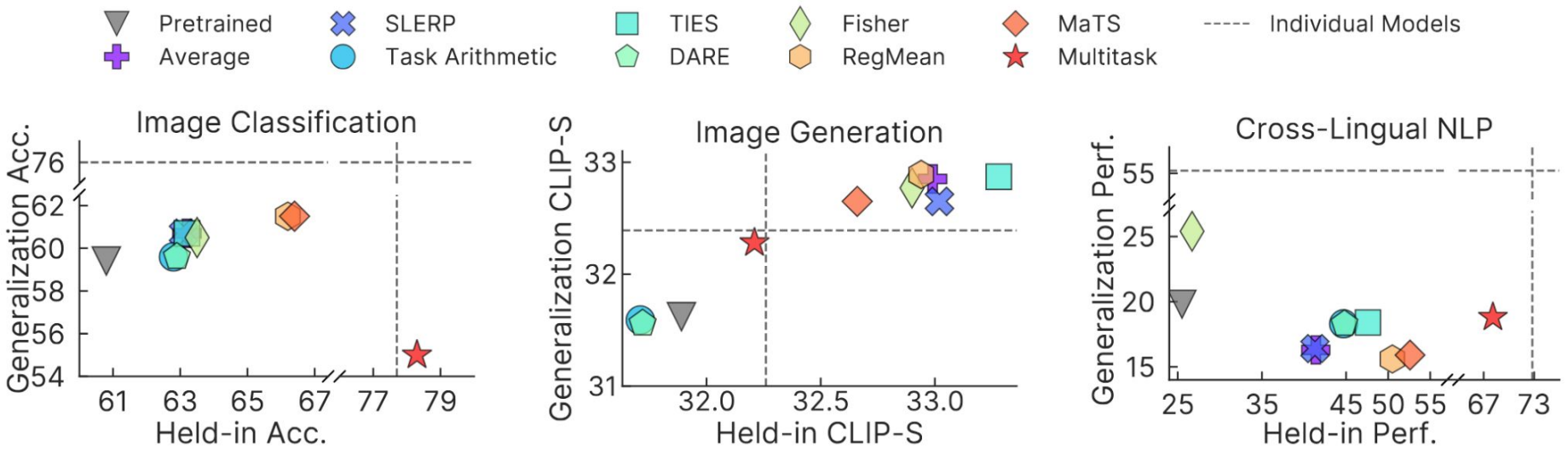
Is merging actually doing what we want?



NLP Task ↓ / Language →	English	Arabic	Thai	German	Korean
Question-Answering (SQuaD/XQuaD)	🔧	🧪	🧪	🧪	
Natural Language Inference (XNLI)	🧪	🔧	🧪	🧪	
Summarization (WikiLingua)	🧪	🧪	🔧	🧪	🧪
Word Sense Disambiguation (WiC/XLWiC)	🧪			🔧	🧪
Is question answerable? (TyDiQA)	🧪	🧪	🧪		🔧

From "Realistic Evaluation of Model Merging for Compositional Generalization" by Tam et al.

... not really.



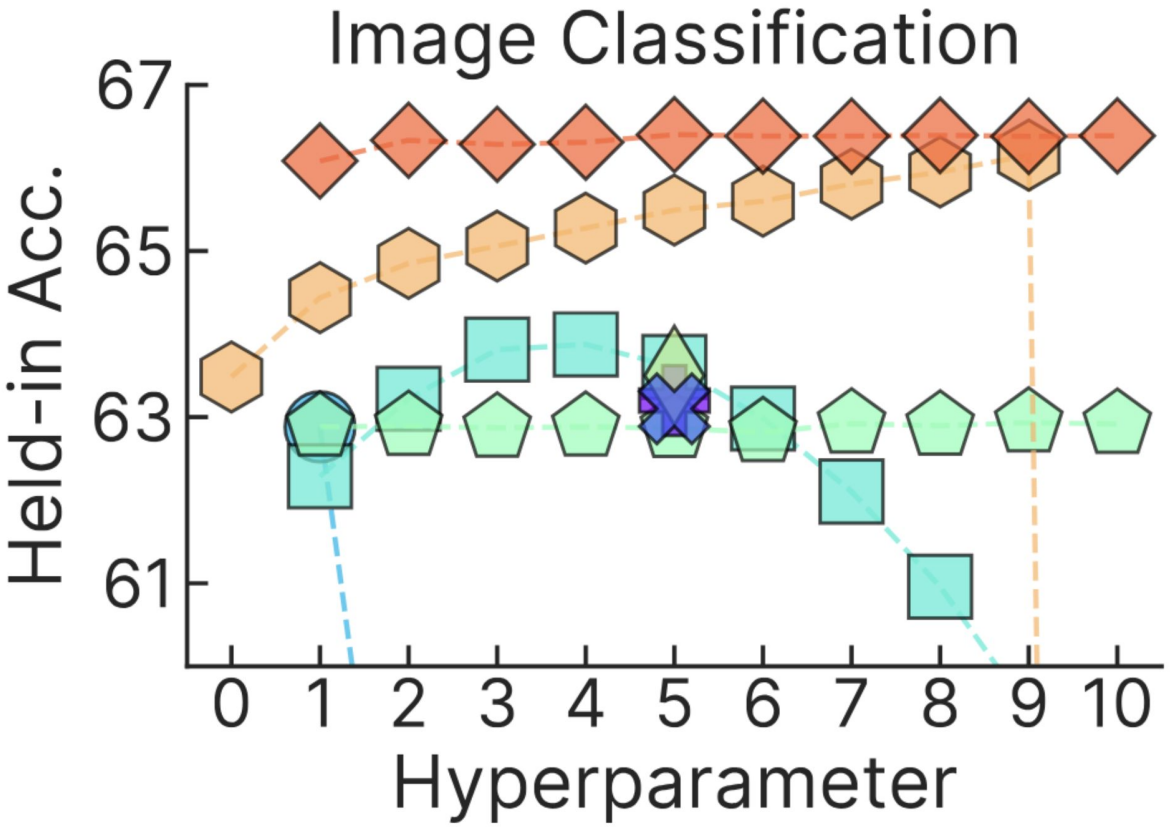
From "Realistic Evaluation of Model Merging for Compositional Generalization" by Tam et al.

Merging methods also have different practical requirements...

	Prerequisites			Hparams?	Computational cost (FLOPs)	
	$\theta_{\mathbf{p}}$	Stats	Data		Merging	Statistics
Average					Mdk	
SLERP					$\mathcal{O}((5M-2)dk)$	
Task Arith.	✗		✗	✓	$(2M+1)dk$	
DARE	✗		✗	✓	$(6M+1)dk$	
TIES	✗		✗*	✓*	$(4M+1)dk$	$\mathcal{O}(MKdk)$
Fisher		✗*	✗*		$(3M-1)dk$	$4MTd^2k$
RegMean		✗*	✗*	✓	$\mathcal{O}((M+2)d^2k)$	MTd^2k
MaTS		✗*	✗*	✓	$\mathcal{O}((M+N)d^2k)$	$4MTd^2k$

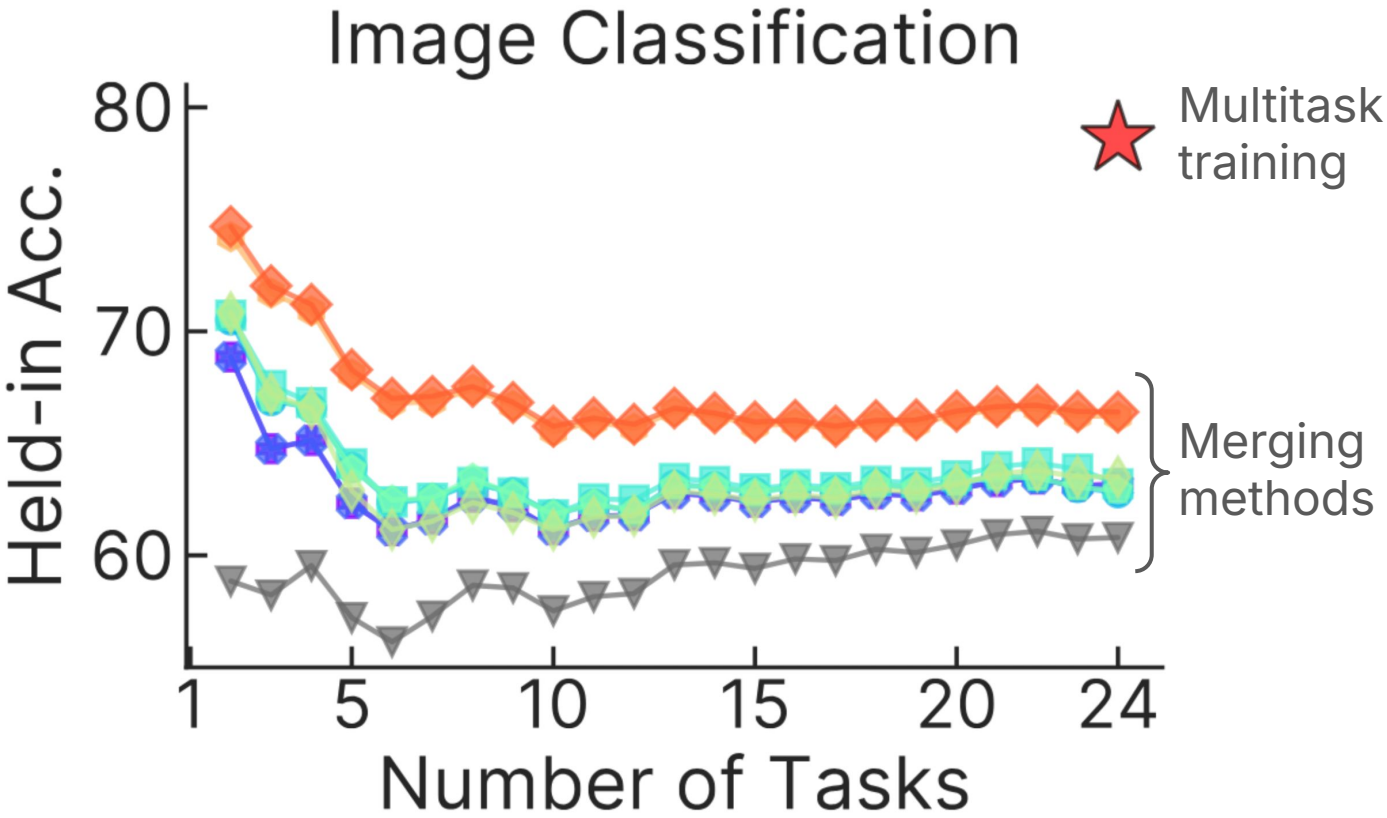
From "Realistic Evaluation of Model Merging for Compositional Generalization" by Tam et al.

... and different hyperparameter sensitivity



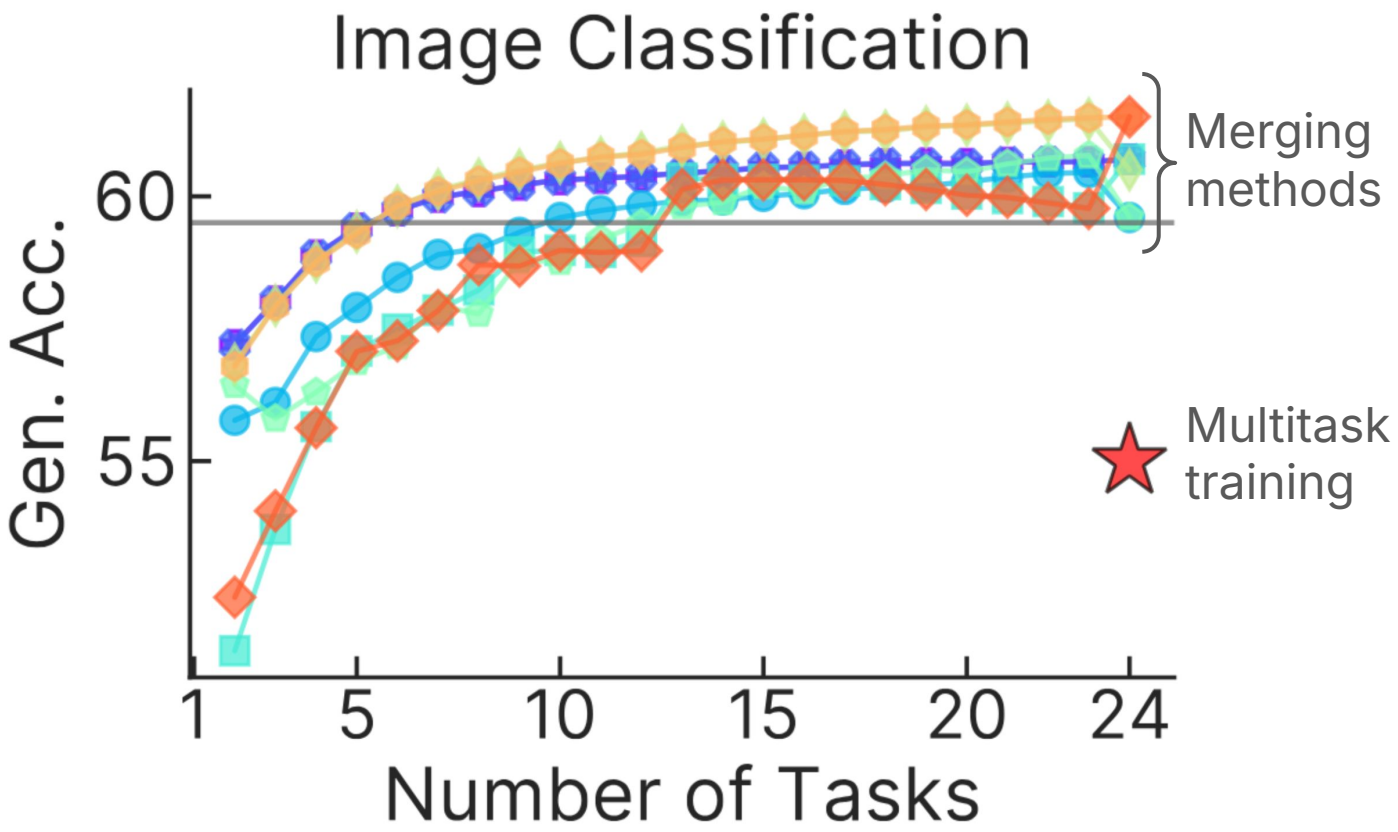
From "Realistic Evaluation of Model Merging for Compositional Generalization" by Tam et al.

Multitask performance is still poor as you the number of models...



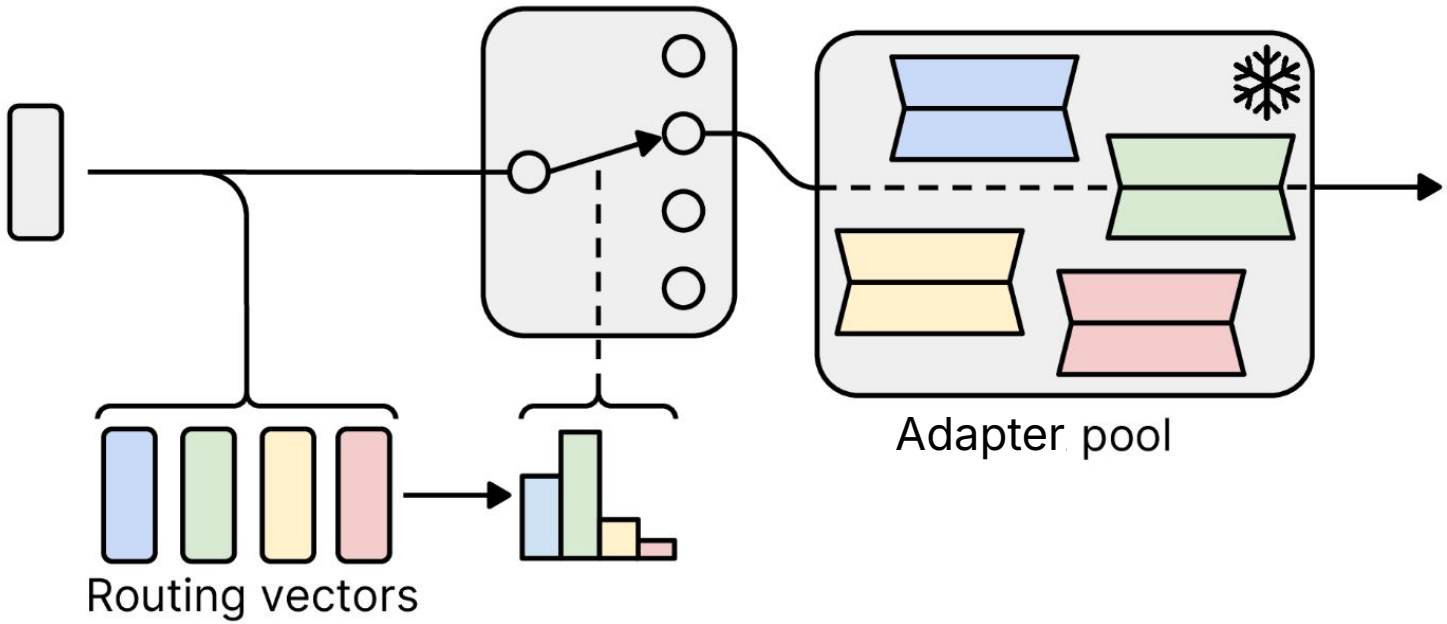
From "Realistic Evaluation of Model Merging for Compositional Generalization" by Tam et al.

... but the picture for generalization is better.



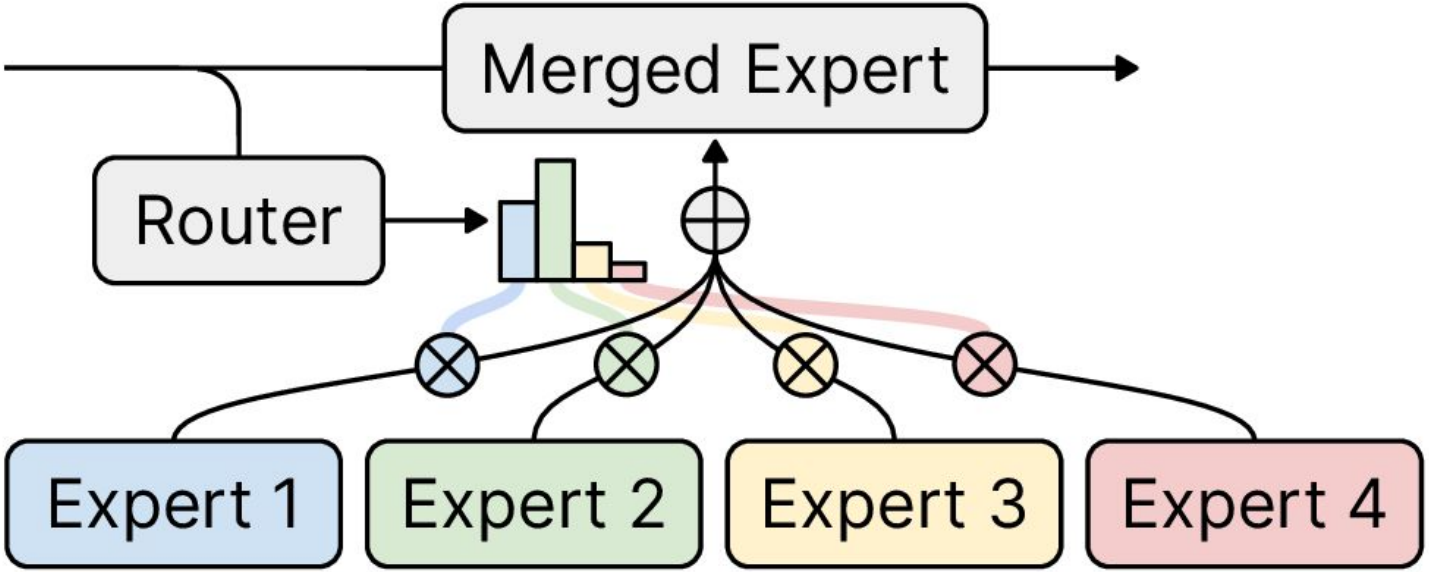
From "Realistic Evaluation of Model Merging for Compositional Generalization" by Tam et al.

An alternative: MoErging?



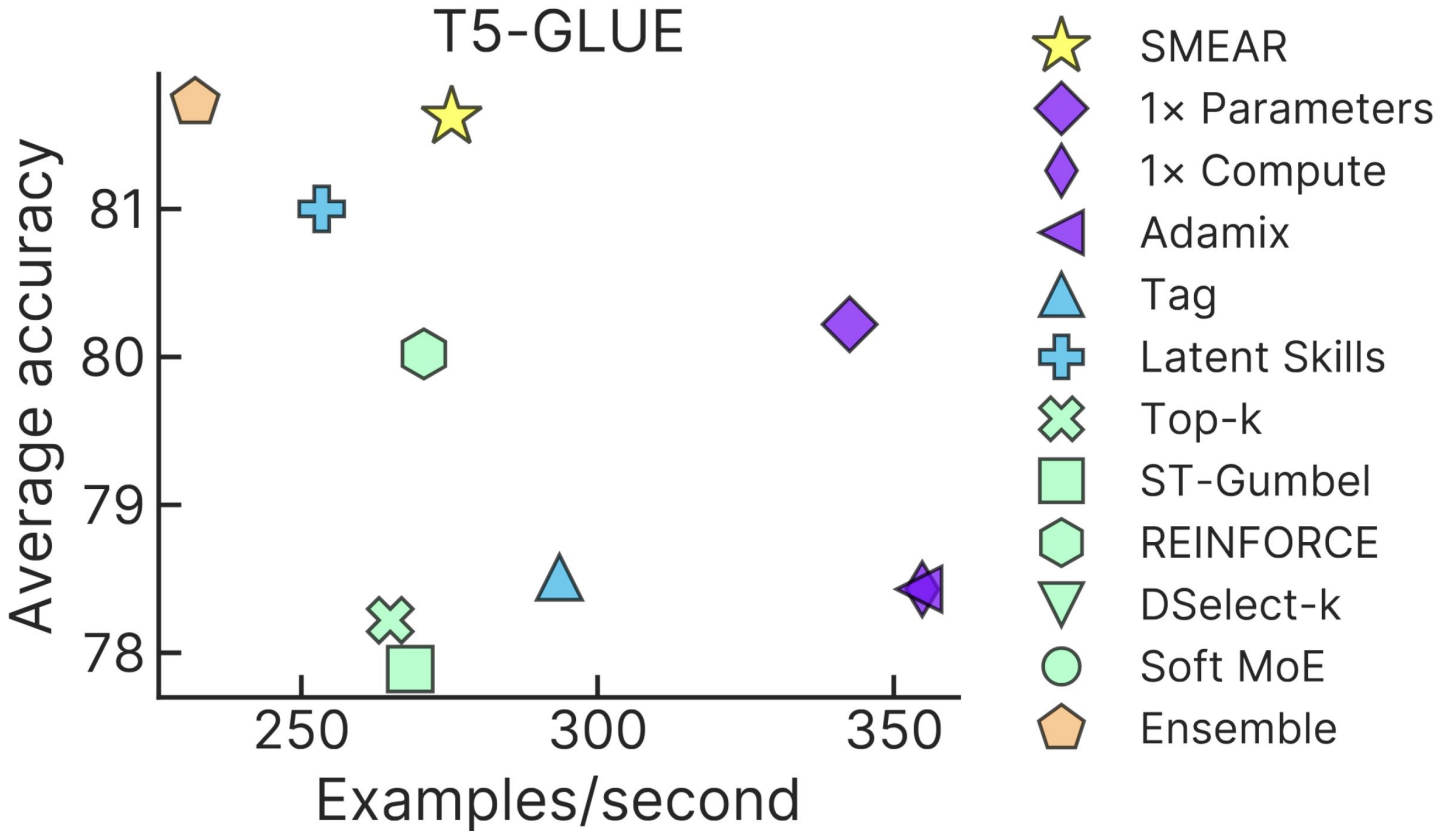
From "Learning to Route Among Specialized Experts for Zero-Shot Generalization" by Muqeeth et al.

Differentiable routing between specialist submodels with SMEAR



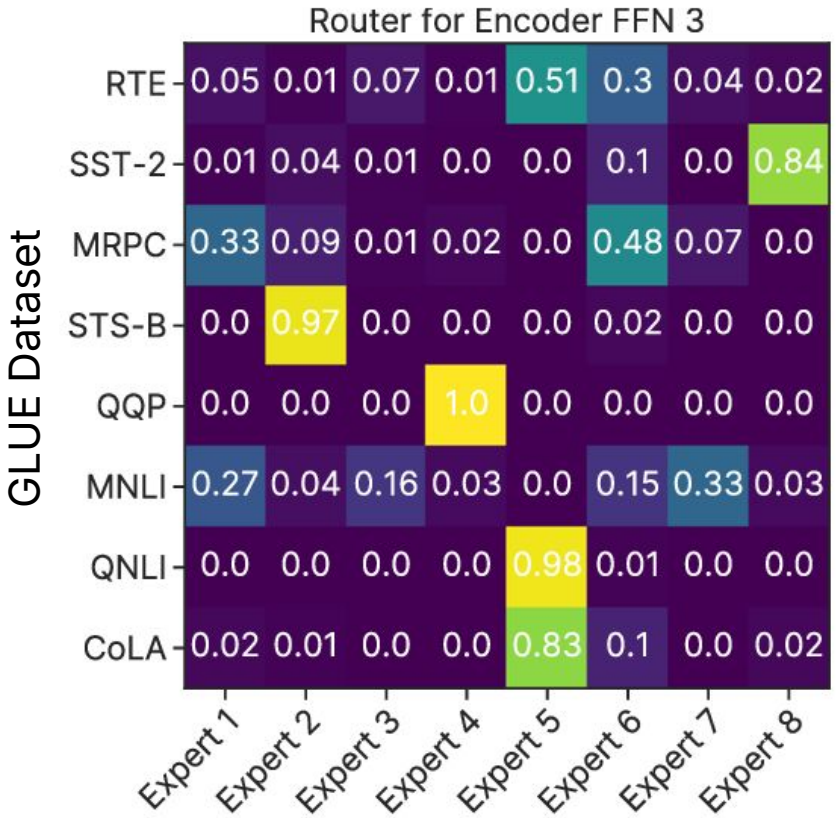
From “Soft Merging of Experts with Adaptive Routing" by Muqeeth et al.

SMEAR is pareto-optimal across different routing strategies



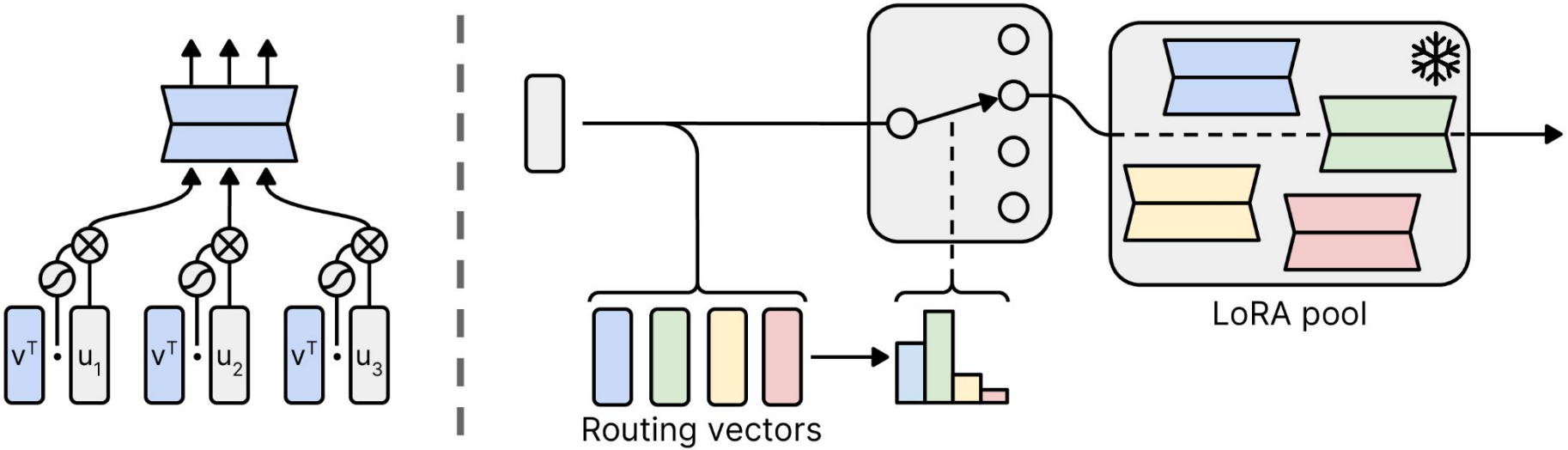
From "Soft Merging of Experts with Adaptive Routing" by Muqeeth et al.

Experts specialize and are shared across different datasets



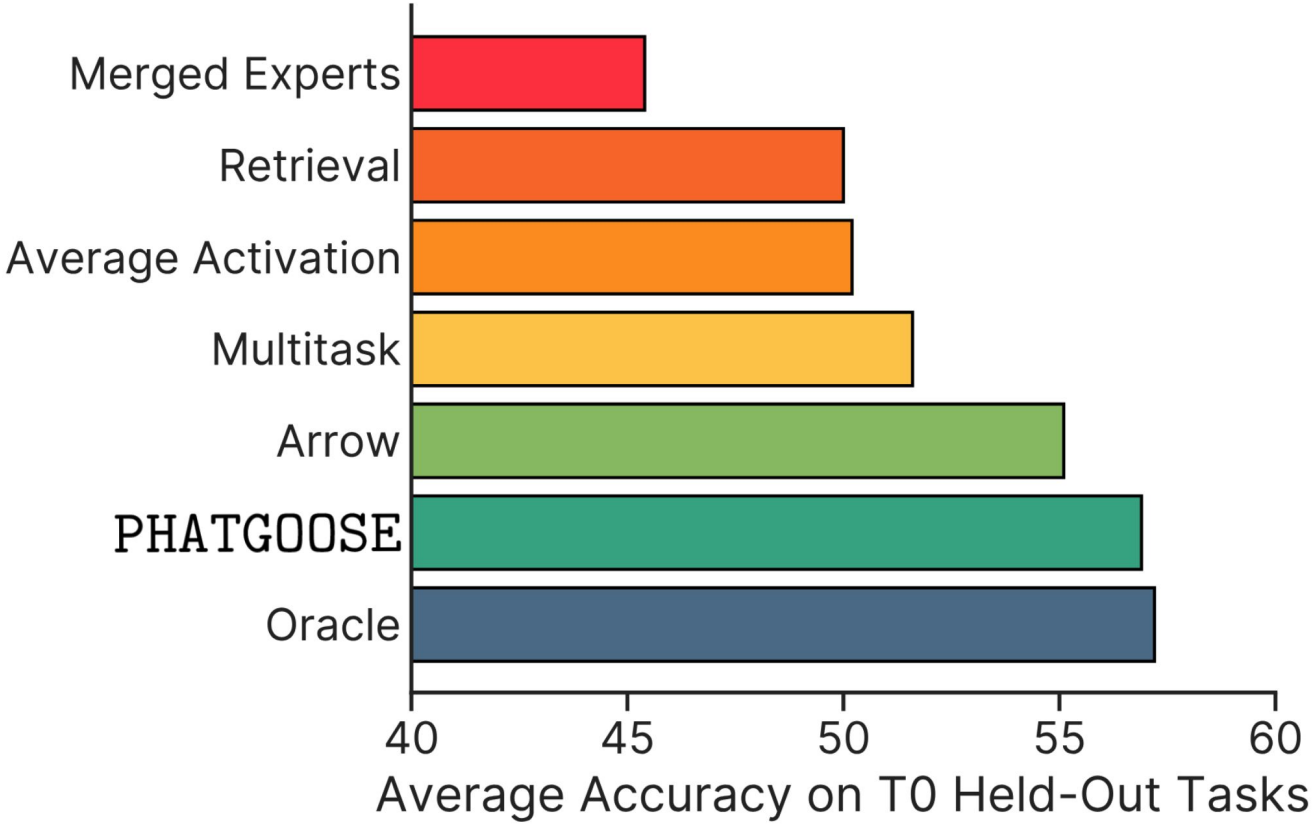
From “Soft Merging of Experts with Adaptive Routing" by Muqeeth et al.

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



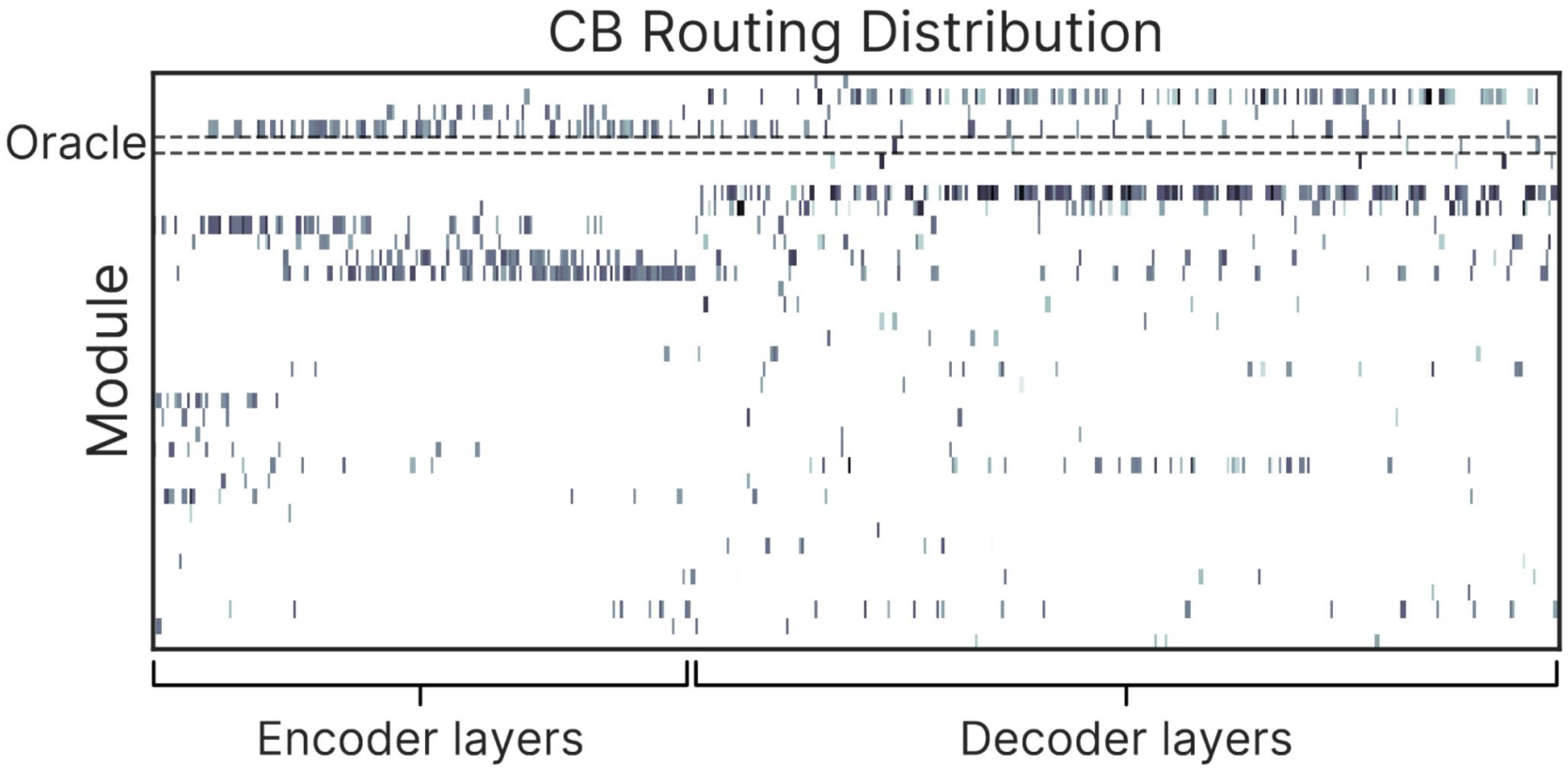
From "Learning to Route Among Specialized Experts for Zero-Shot Generalization" by Muqeeth et al.

PHATGOOSE outperforms prior routing methods and multitask training



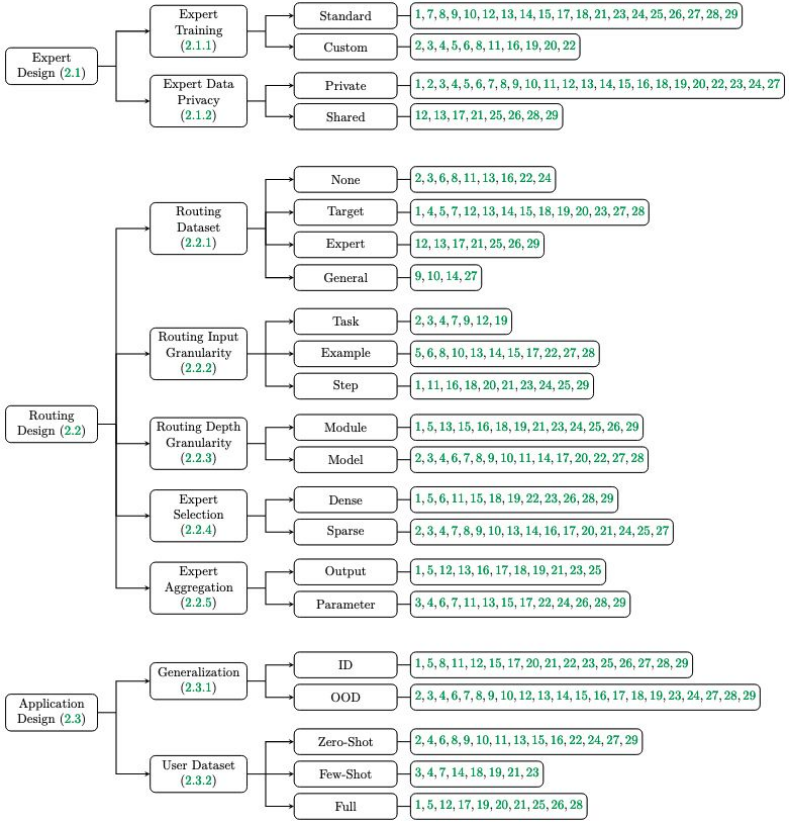
From "Learning to Route Among Specialized Experts for Zero-Shot Generalization" by Muqeeth et al.

PHATGOOSE learns nontrivial routing strategies



From "Learning to Route Among Specialized Experts for Zero-Shot Generalization" by Muqeeth et al.

The MoErging design space



From "A Survey on Model MoErging: Recycling and Routing Among Specialized Experts for Collaborative Learning" by Yadav et al.

Thanks.

Please give me feedback:

http://bit.ly/colin-talk-feedback

craffel@gmail.com