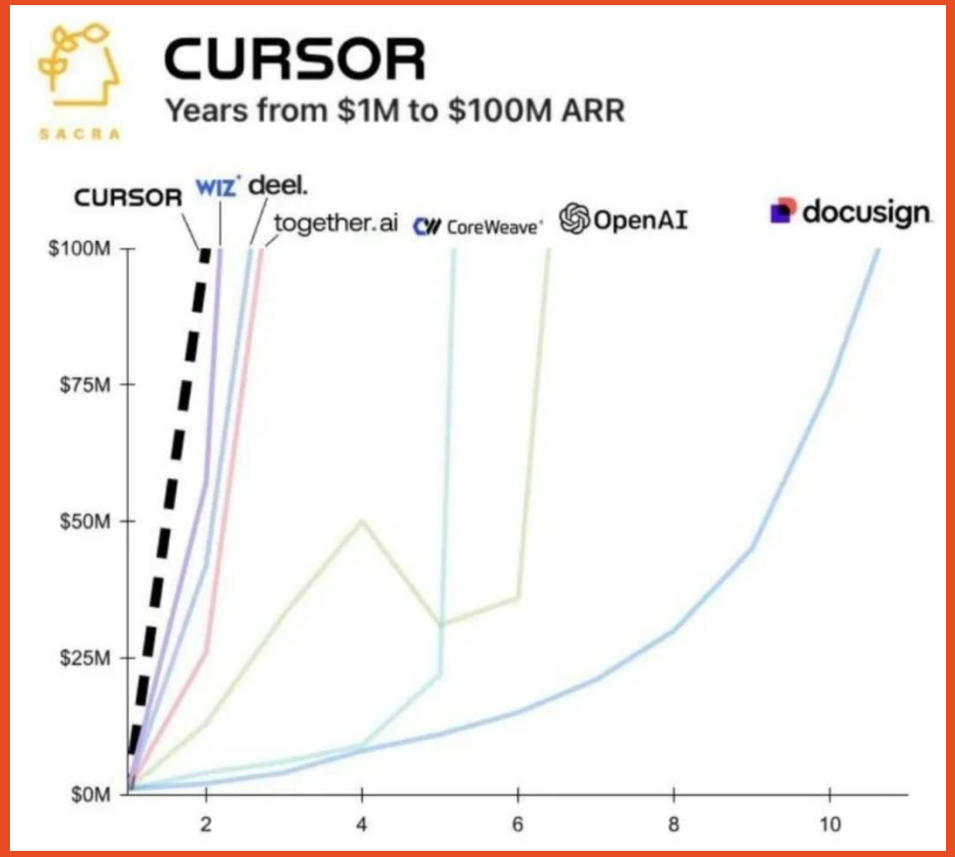
Deep Model Fusion

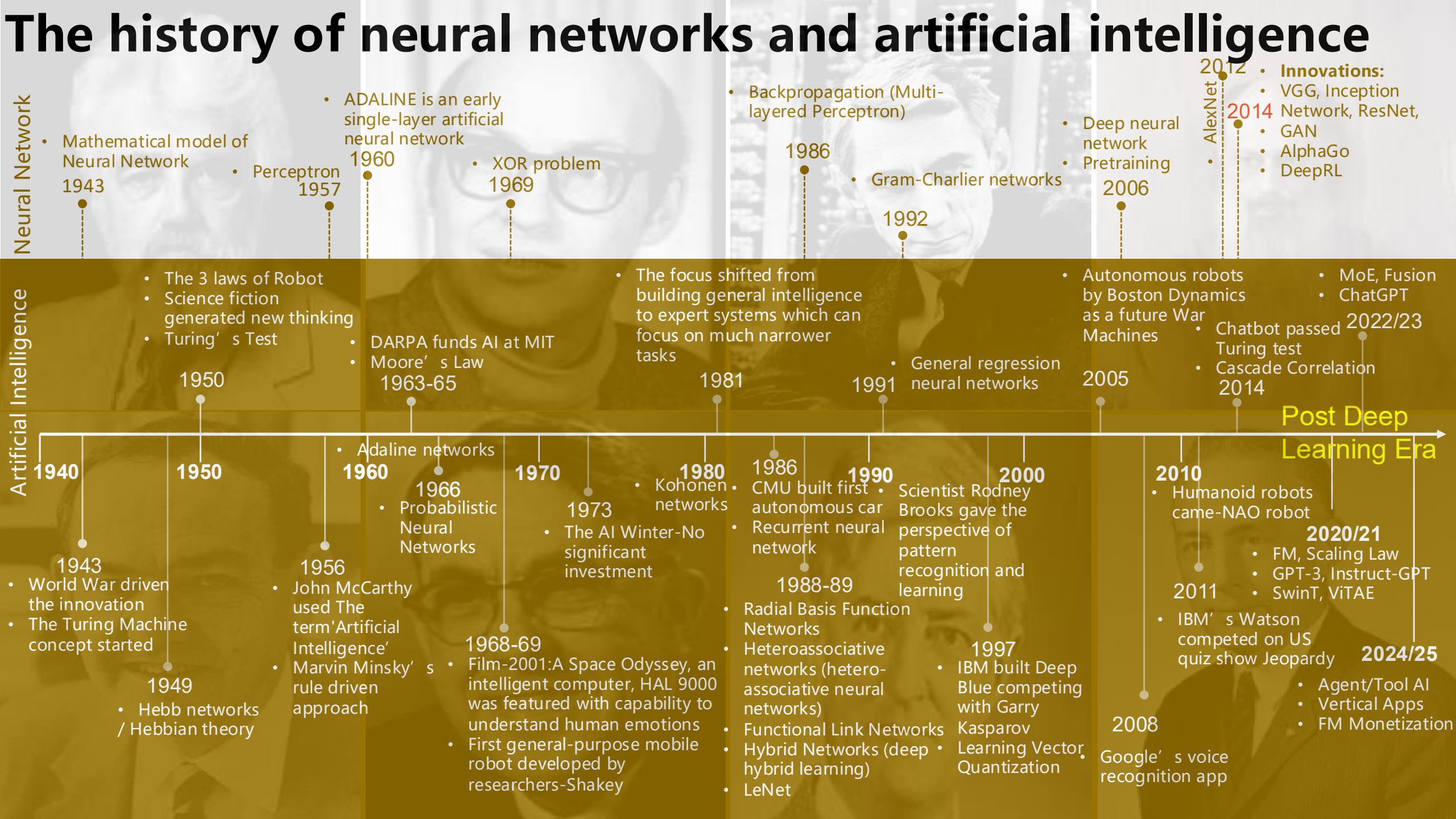
Prof Dacheng Tao FAA FACM FIEEE
GrAIL, CCDS, NTU

Anthropic CEO said AI could write ‘90% of code’ In ‘3 to 6 months’ – Warns ‘every industry’ will be affected, 2025 June

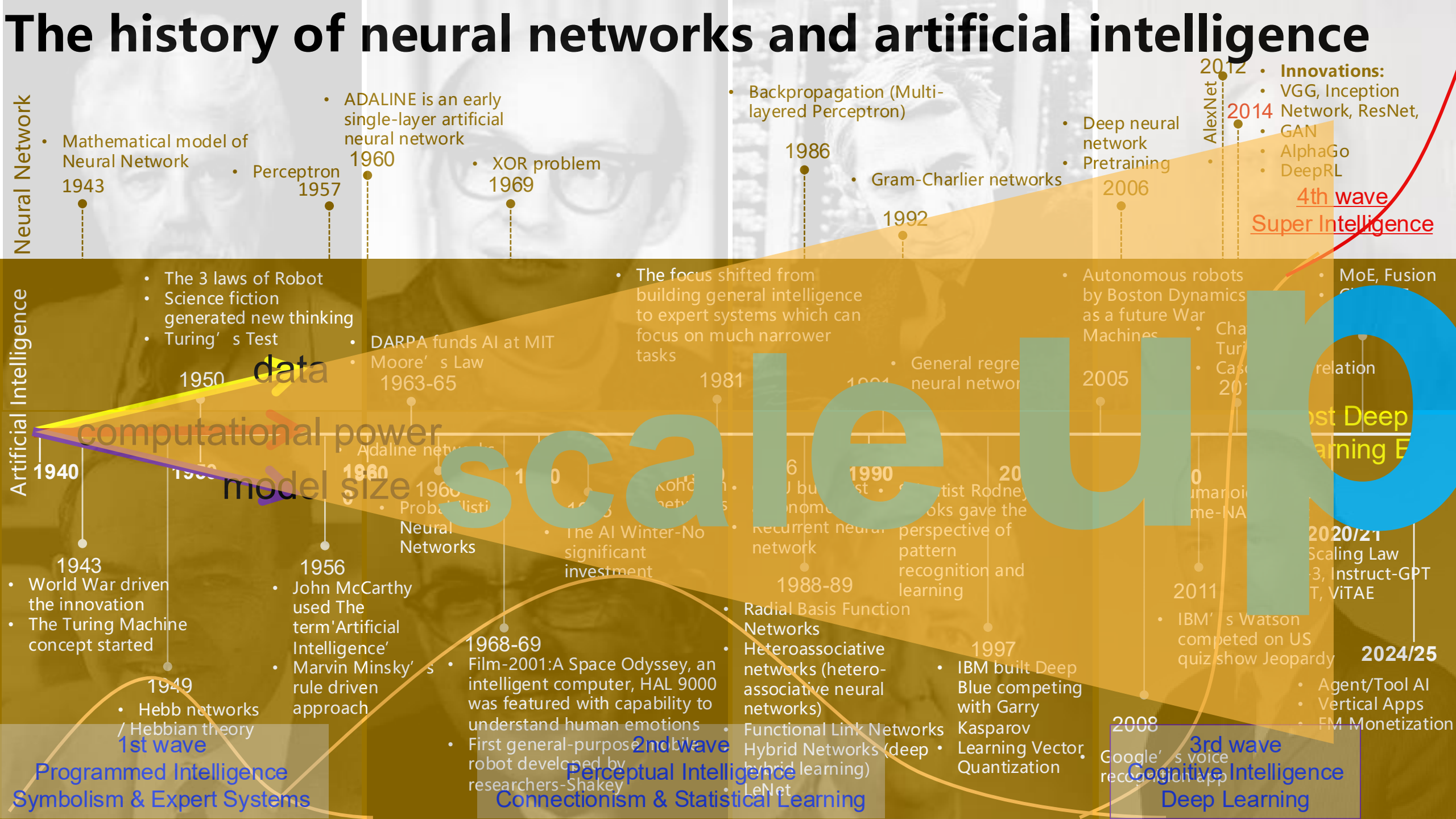


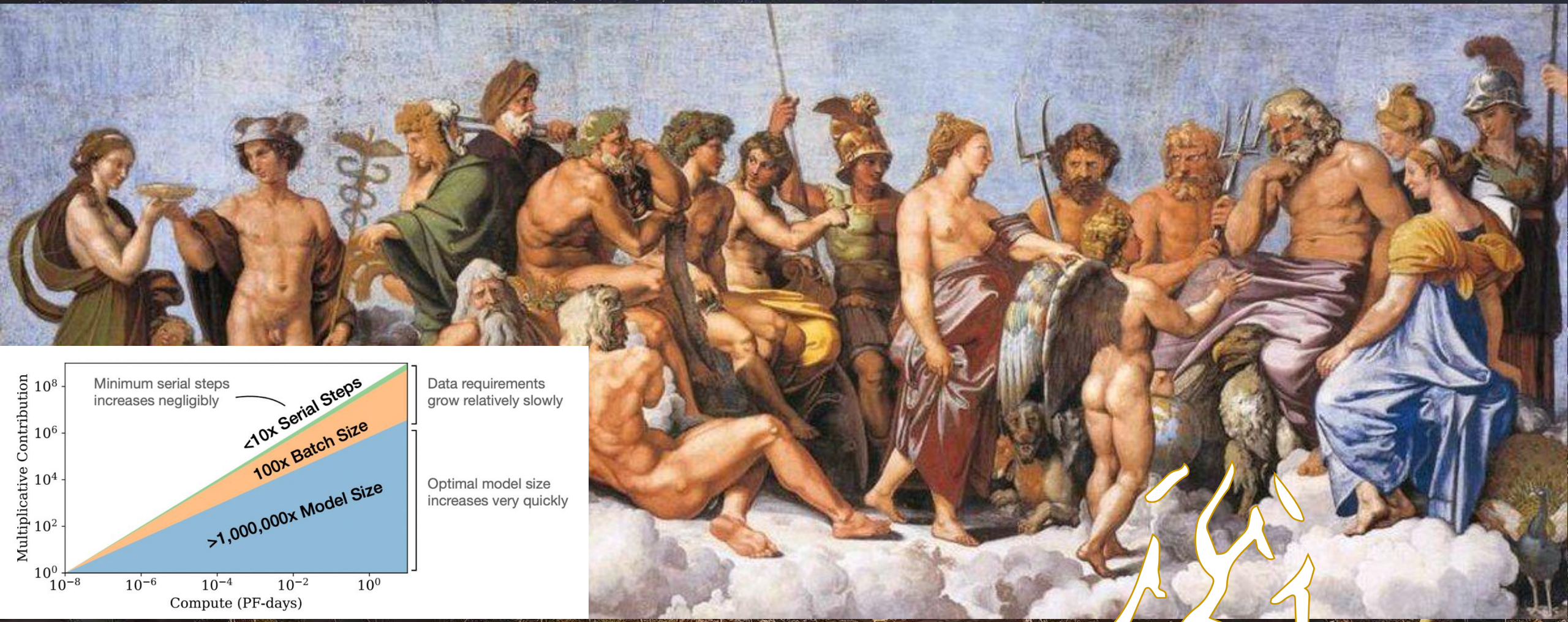


The history of neural networks and artificial intelligence



The history of neural networks and artificial intelligence





The Future is ?

Agenda

- Background
- Definition: Deep Model Fusion
  - ❑ Model Fusion for Pre-Training Acceleration
  - ❑ Model Fusion via Adaptive Weights
  - ❑ Model Fusion via Mixing/Upscaling
  - ❑ Model Fusion via Meta Learning
  - ❑ Model Fusion via Fine-tuning
  - ❑ Multi-Modal Model Merging
  - ❑ Continual Model Merging
- Conclusion

Background

Background

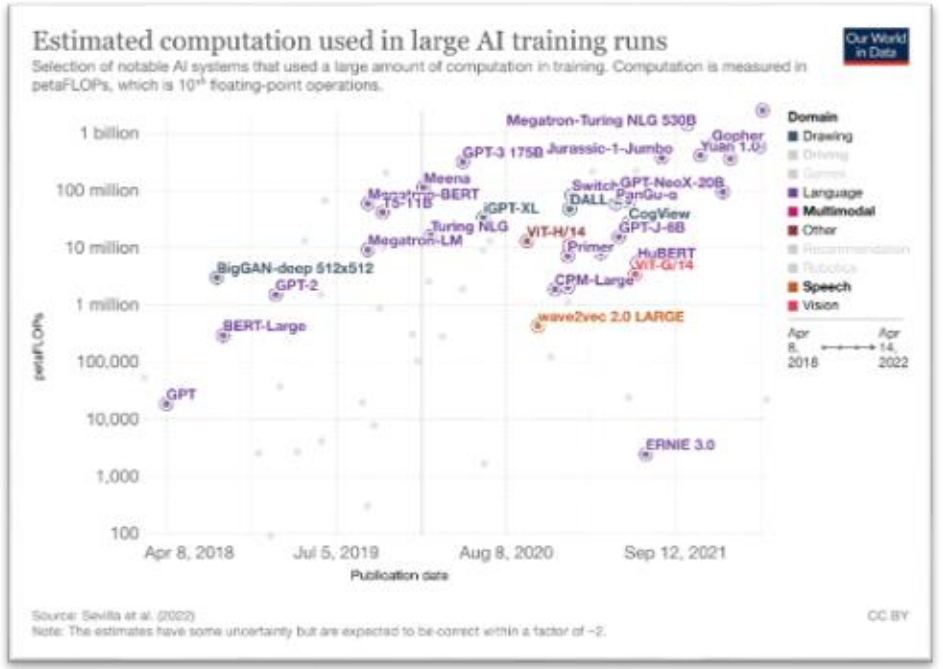
Traditional Deep Learning Approach

- Collecting a large dataset relevant to the problem
- Cleaning Data, labor-intensive and expensive
- Feature Engineering (optional): selecting the most informative features.
- Training Model: design model architecture, choose optimization algorithm



fast growing sizes of datasets and models

Require more GPU resources and higher cost

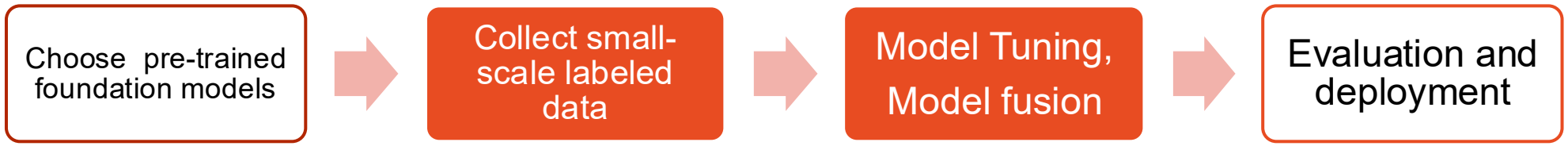


Background: Transformation of Learning Paradigm

Learn from Data (supervised learning)



Learn from model (transfer learning, model editing, model tuning, model distillation, model fusion)



<u>Deep Mode fusion (data-free/data-efficient)</u>



Zheng, et al. Learn from model beyond fine-tuning, Nature Machine Intelligence, 2025.

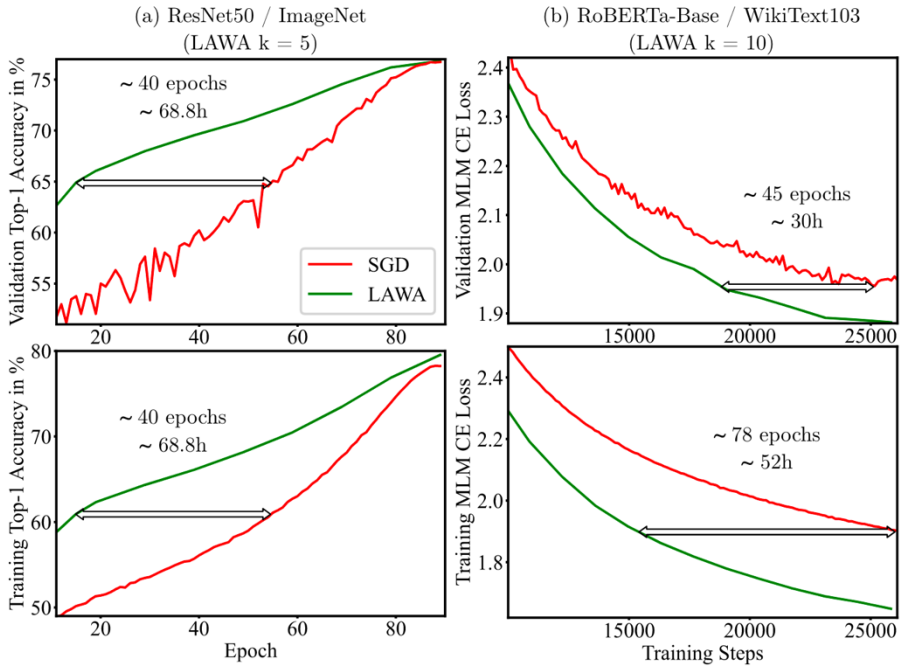
Application: Accelerating Model Training

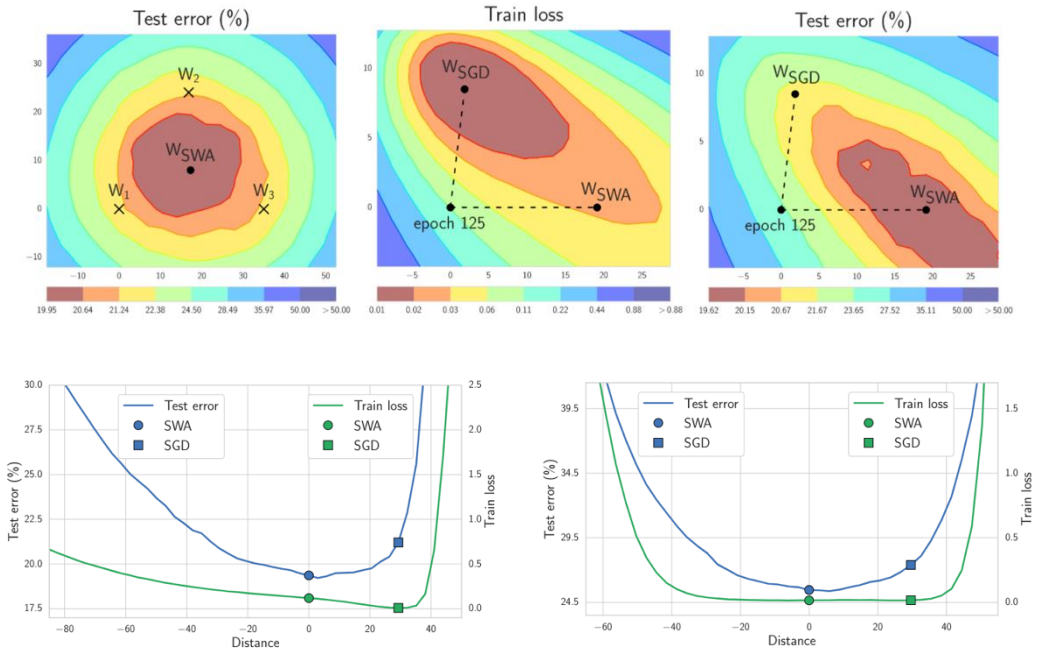
Initialization via fused models (model soup):

Train a model from scratch is expensive, we can accelerate this process by using averaged initialization (flat initialization) given multiple pre-trained models sharing the same architecture.

Training-time checkpoints averaging (LAWA/SWA/EMA):

Seek flat solution to improve convergence and generalization

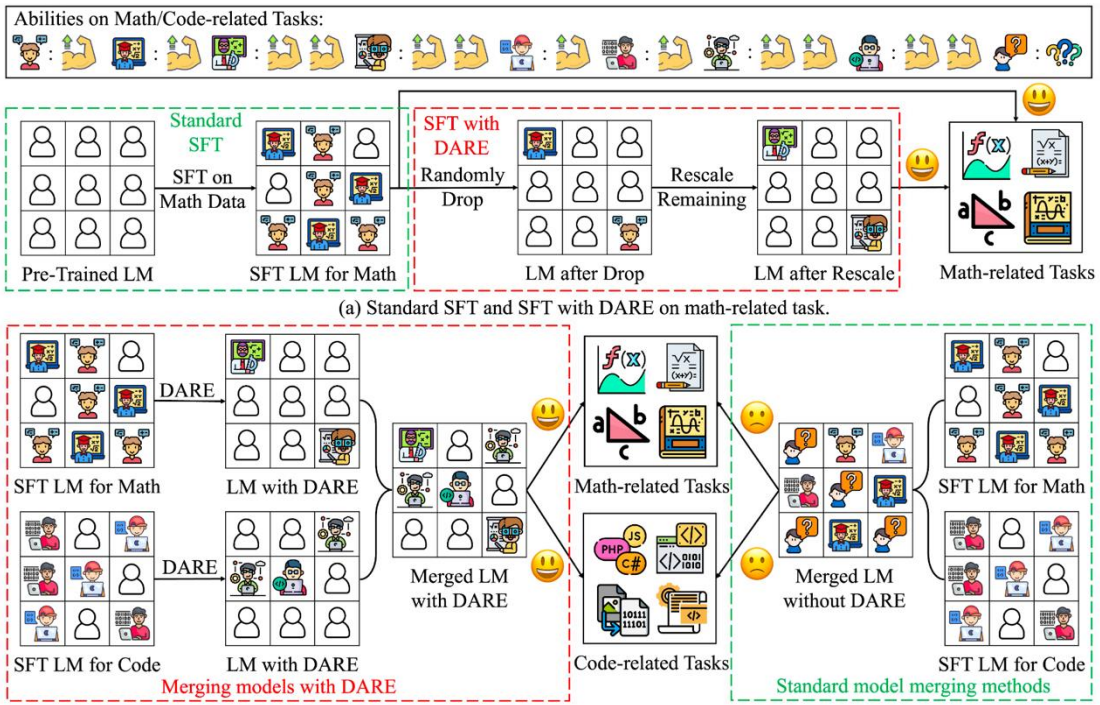


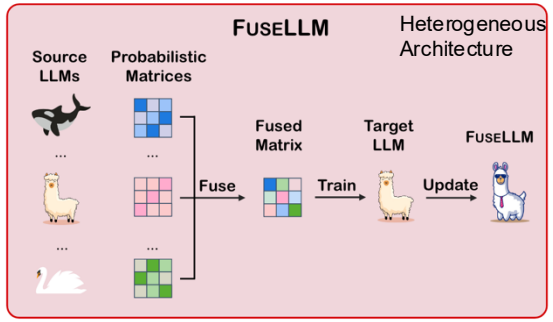


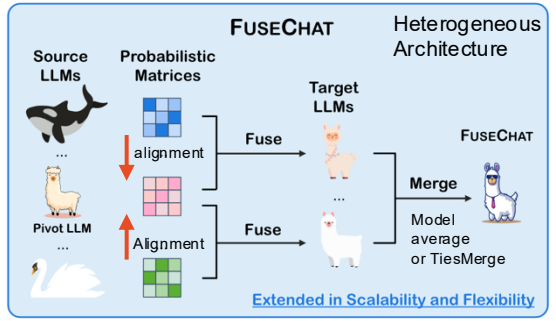
Accelerate Training via Model fusion: Latest weight averaging (LAWA), Stochastic Weight averaging (SWA), Exponential Moving average (EWA)…

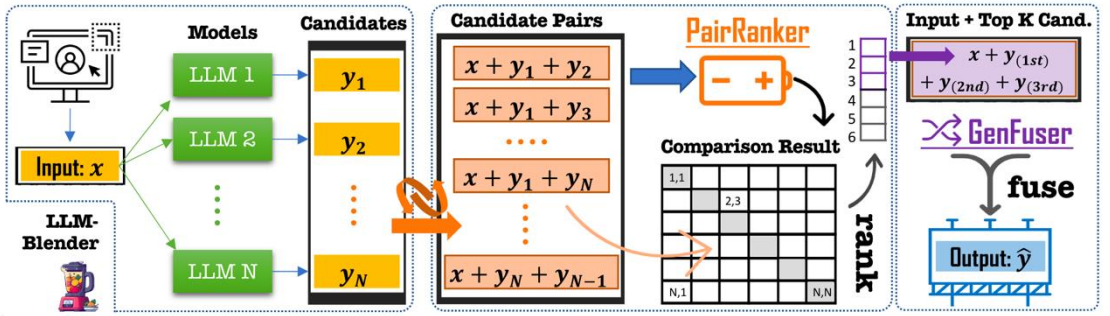
Application: LLM Fusion

Create powerful LLMs from existing models via model fusion







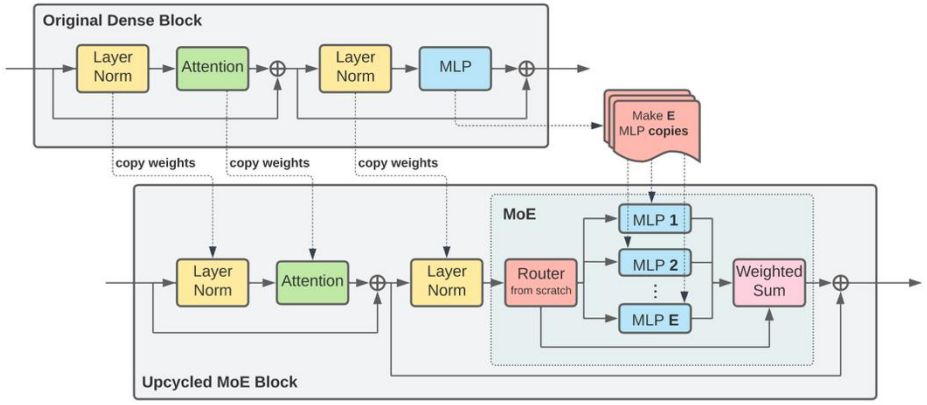


Methods: DARE, Task Arithmetic, FuseLLM, FuseChat …
Tools: MergeKit

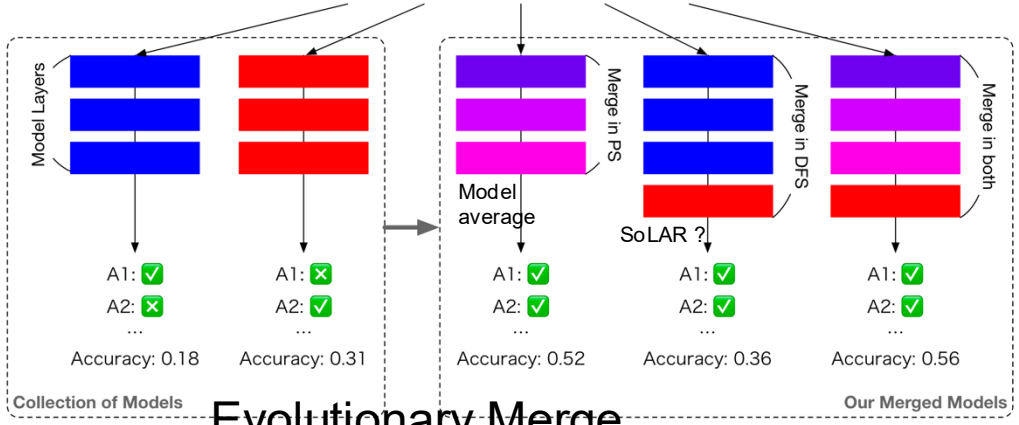
https://github.com/arcee-ai/mergekit

Application: Foundation Model Up-scaling

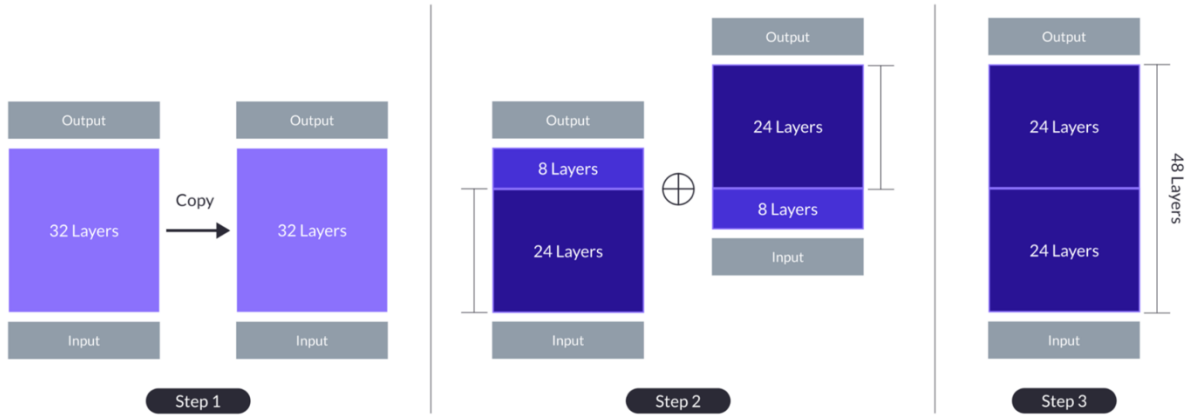
Model Up-scaling (mixing): accelerate the development of larger pre-trained models



sparse upcycling to MoE model (google, ICLR 2023)



Evolutionary Merge

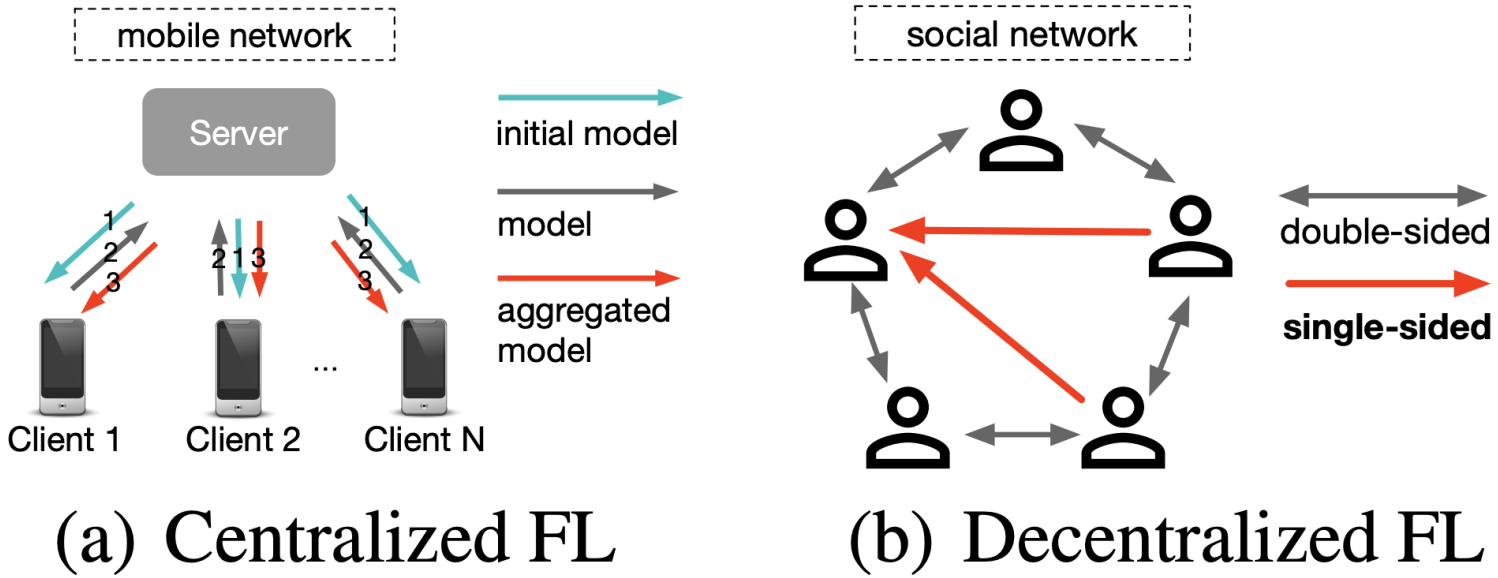


Depth up-scaling (EvoMerge, SOLAR)

Application: Federated Learning

Federated Learning (Homogeneous Architectures for client and server nodes)

- Each client use their data to train their local models.
- **The global server aggregate the local model in the communication round.**

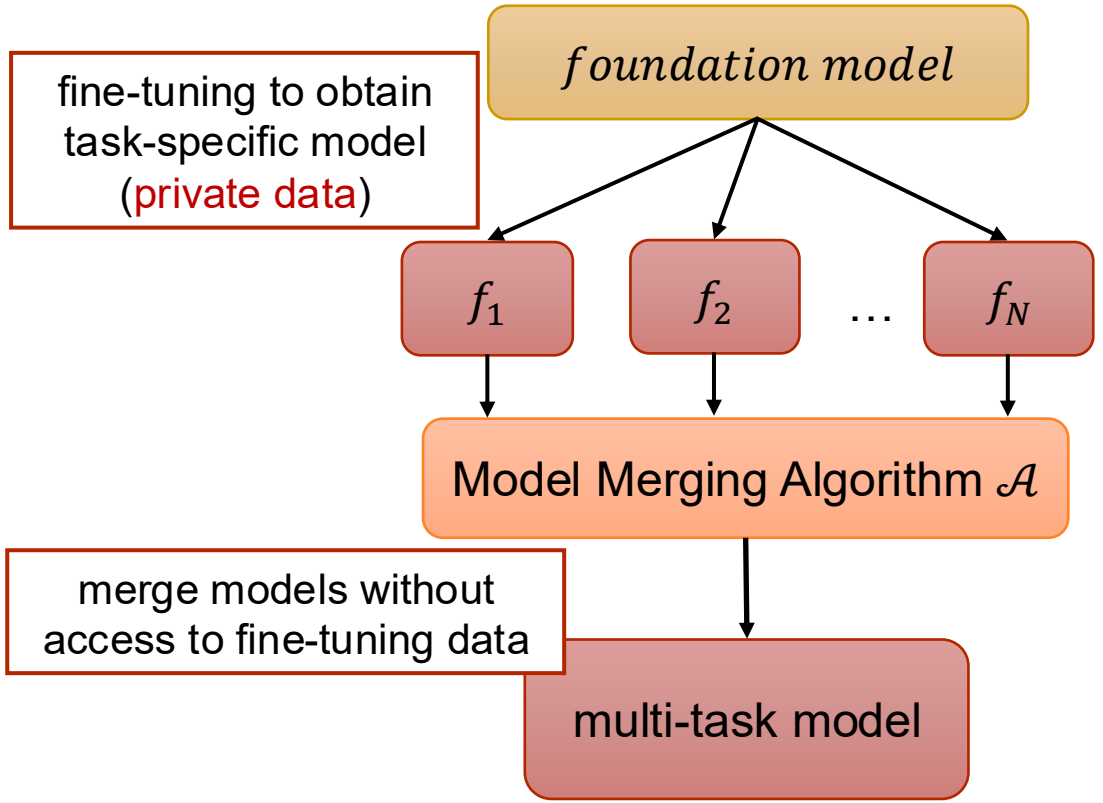


(a) Centralized FL (b) Decentralized FL

Application: Multi-Task Learning

Multi-task model fusion

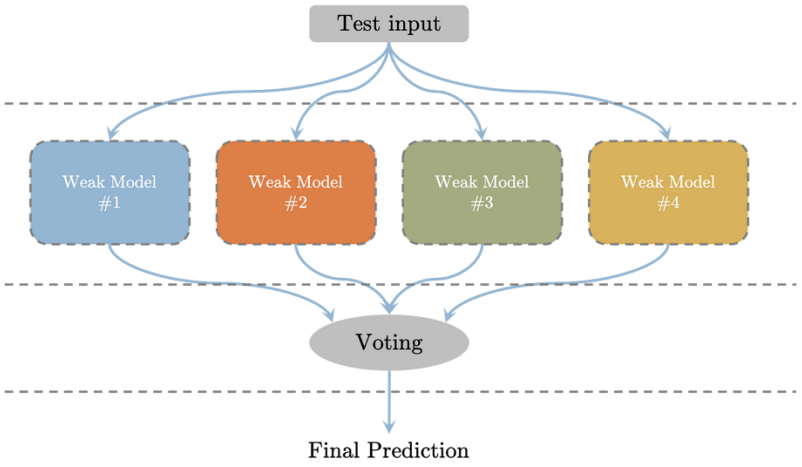
- Leveraging multiple pretrained single task models to produce multitask learning without accessing labeled datasets
- Fast, computational cheap, even GPU-free
- Scalable to large models and a large number of tasks

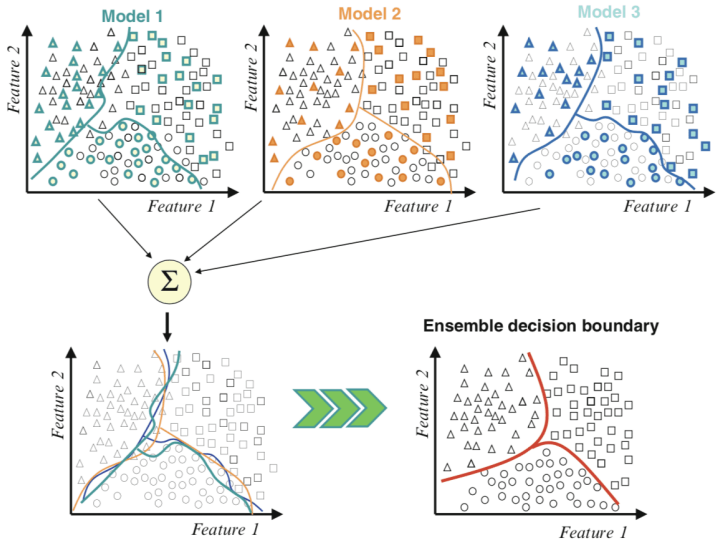


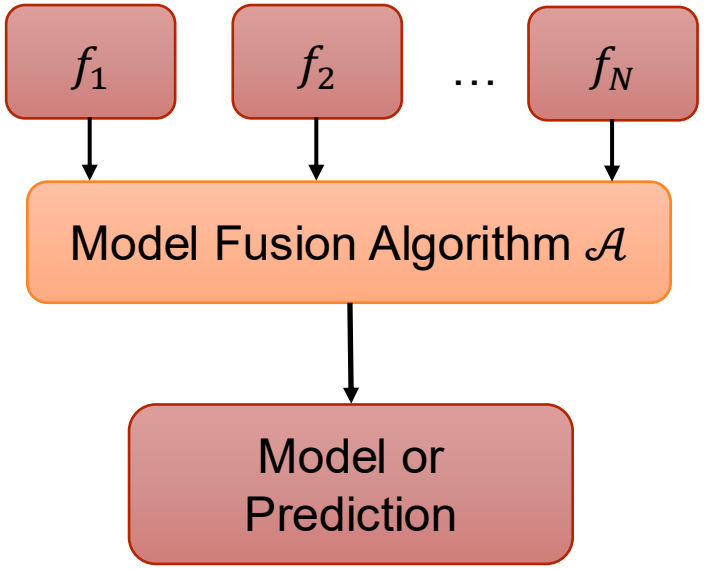
Definition: Deep Model Fusion

Definition of Model Fusion

Model Ensemble: Given a set of models $(f_1, f_2, \dots, f_N)$, which can be homogenous or heterogeneous, we utilize their **predictions** to obtain a **global prediction** $\boldsymbol{y = \mathcal{A}(x; f_1, f_2, \dots, f_N; w)}$. Where $\mathcal{A}$ is a model ensemble algorithm, x is the input, and w is algorithmic parameters.

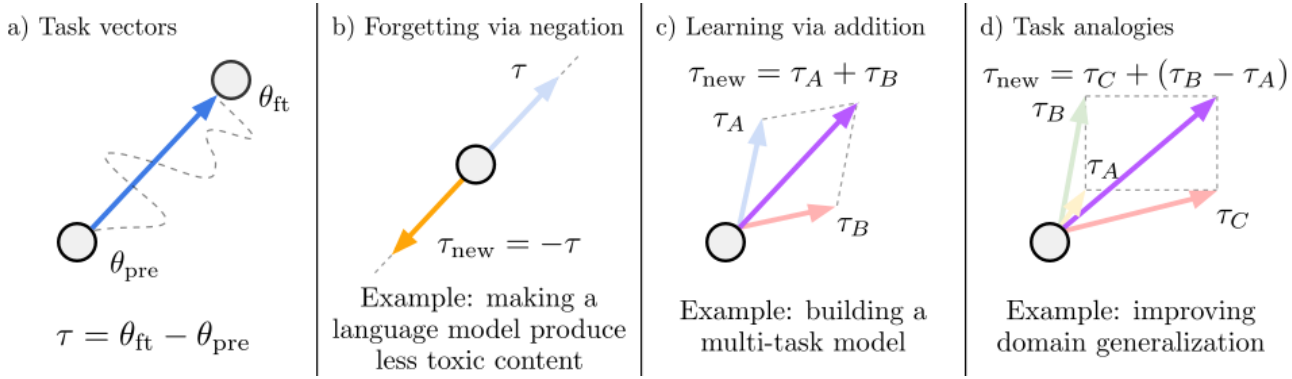


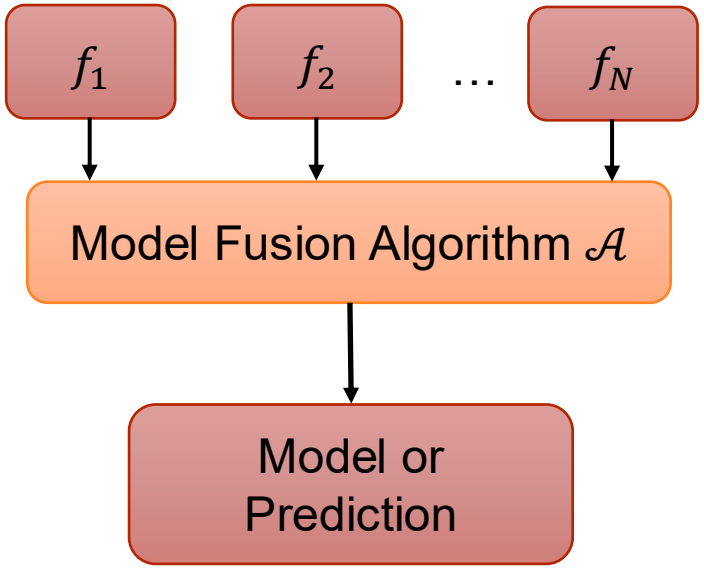




Definition of Model Fusion

Model Merging: Given a set of **homogenous models** $(\boldsymbol{f_1}, \boldsymbol{f_2}, \ldots, \boldsymbol{f_N})$ parameterized by $(\theta_1, \theta_2, \ldots, \theta_N)$, we utilize these **model parameters** to obtain **merged weights** $\boldsymbol{\theta^* = \mathcal{A}(\theta_1, \theta_2, \ldots, \theta_N; w)}$ **with the same architecture** where $\mathcal{A}$ is a model merging algorithm and w is algorithmic parameters. The Merged model can be denoted as $f(\cdot\,; \theta^*)$.

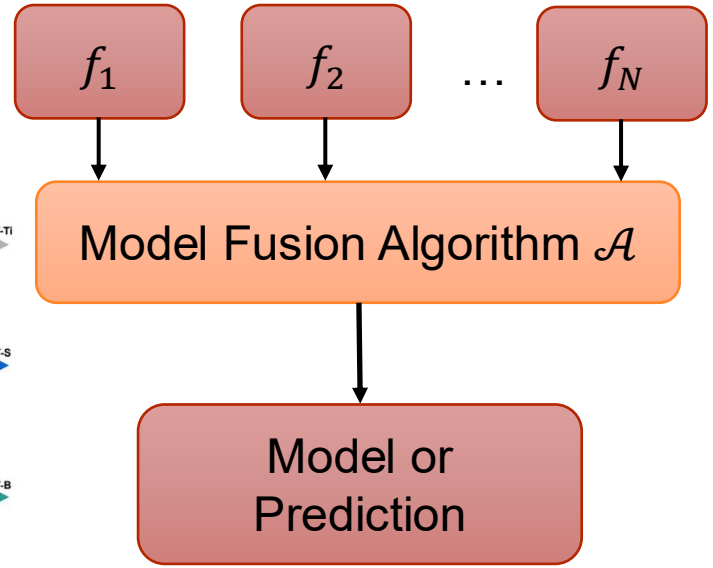


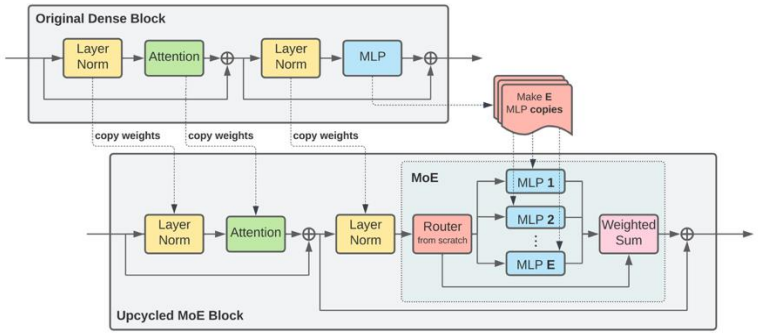


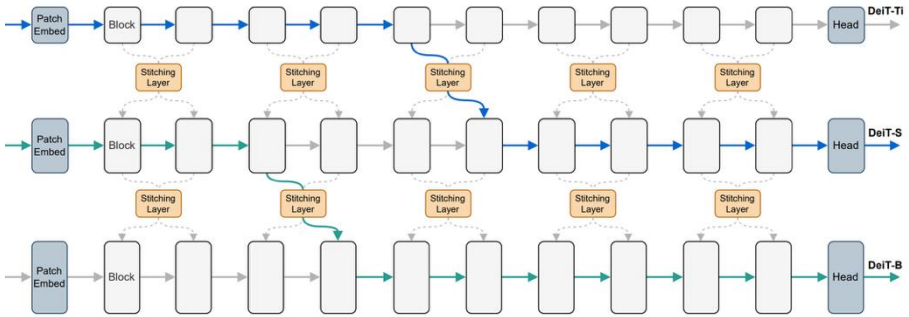
Model Ensemble vs. Model Merging: Prediction vs. Parameters

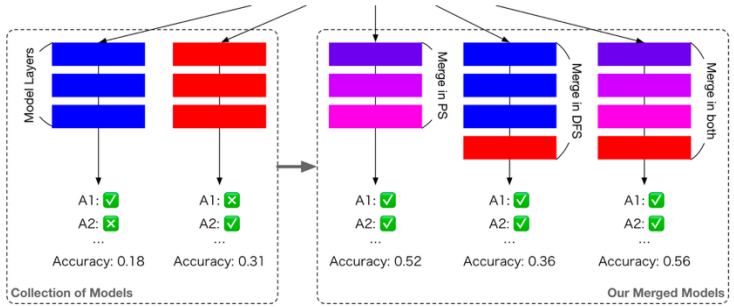
Definition of Model Fusion

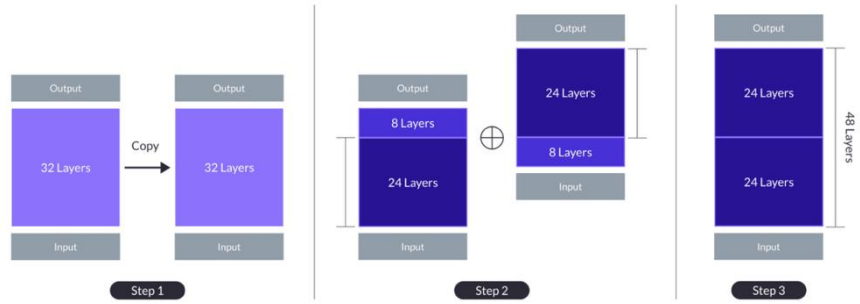
Model Mixing: Given a set of models $(f_1, f_2, \dots, f_N)$, which are often homogenous and parameterized by $(\theta_1, \theta_2, \dots, \theta_N)$, we utilize these models to obtain **merged weights** $\Theta^* = \mathcal{A}(\theta_1, \theta_2, \dots, \theta_N; w)$ of **another network F (usually a larger network),** where $\mathcal{A}$ is a mixing algorithm and w is algorithmic parameters. Merged model can be denoted as $F(\cdot; \Theta^*)$.



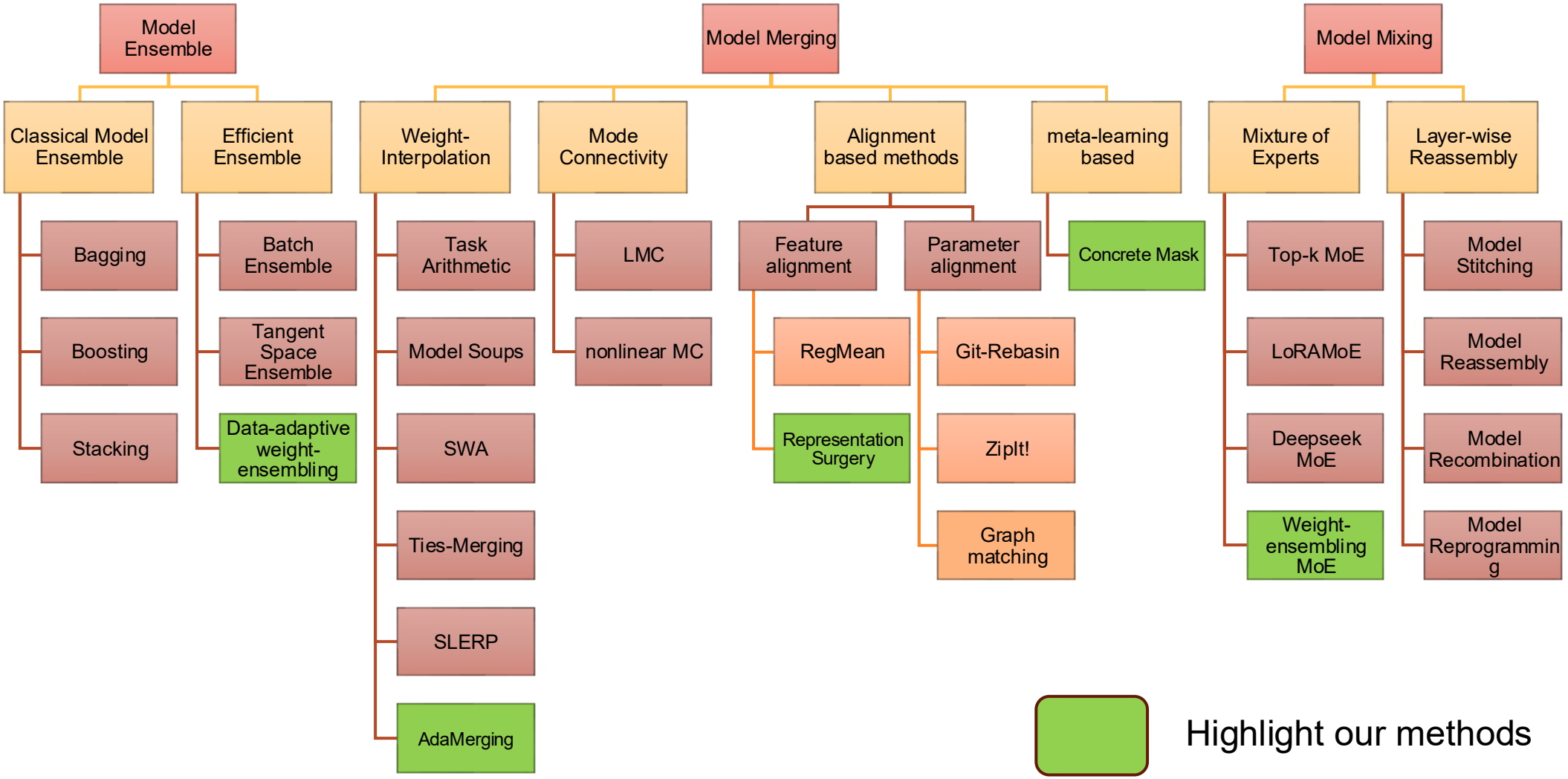








An Overview of Model Fusion Methods

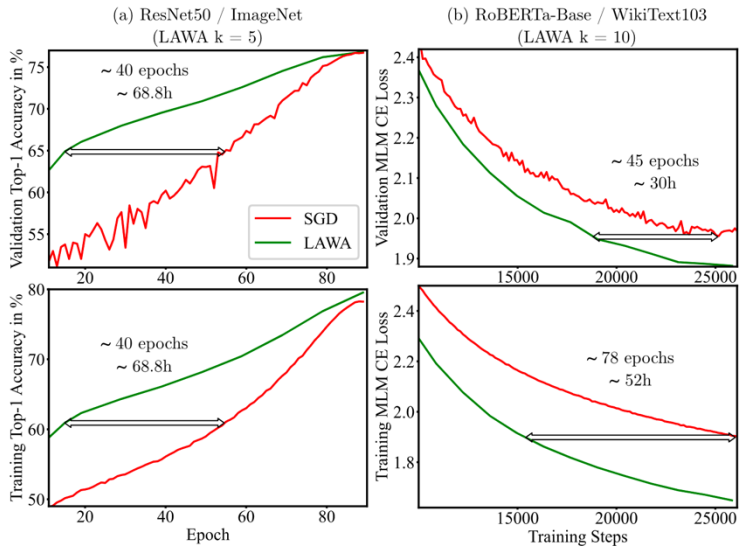


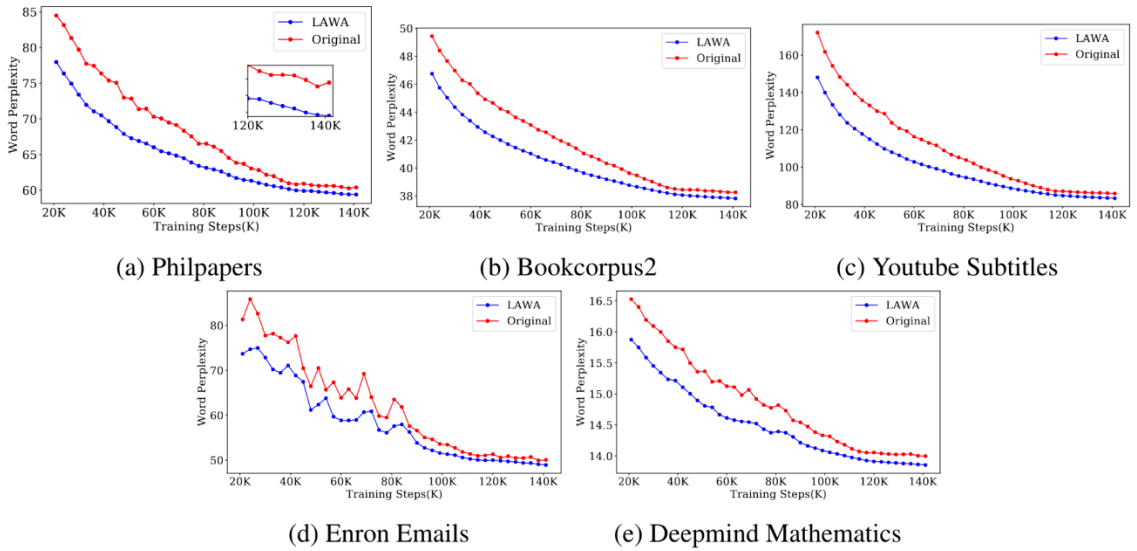
Zheng, et al. Learn from model beyond fine-tuning, Nature Machine Intelligence, 2025.

Model Fusion for Pre-Training Acceleration

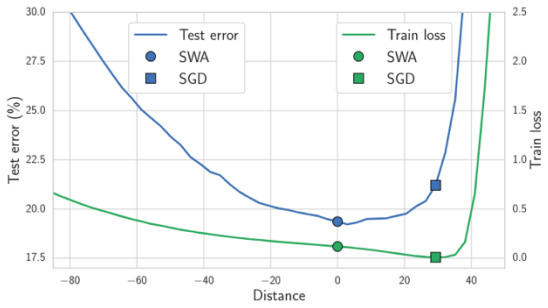
Training-Time Checkpoints Averaging

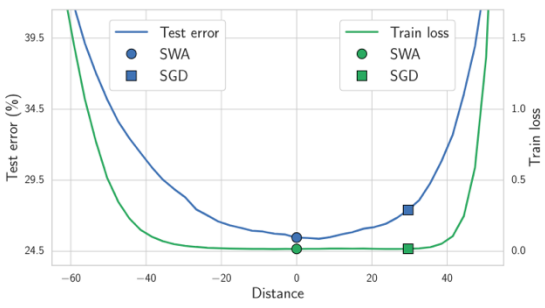
Accelerating convergence

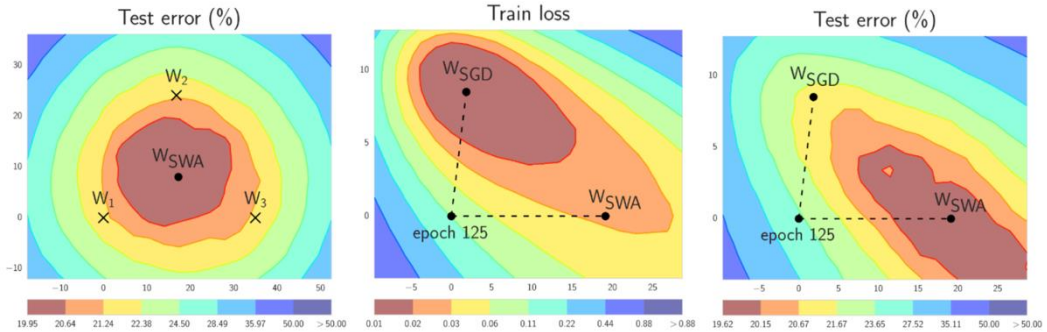




Improving generalization



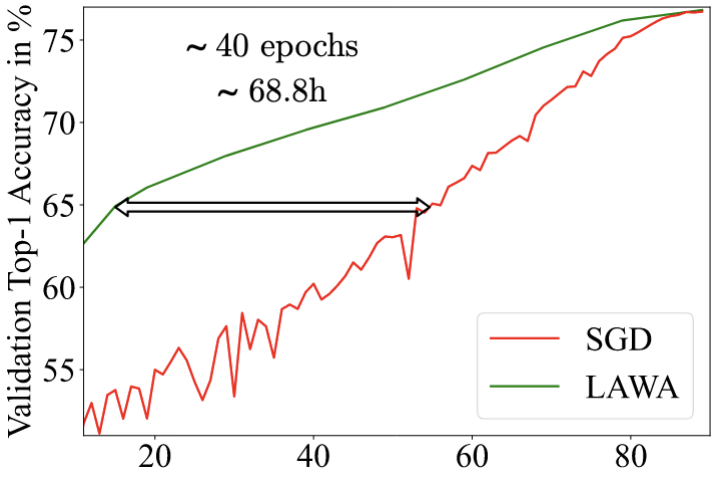


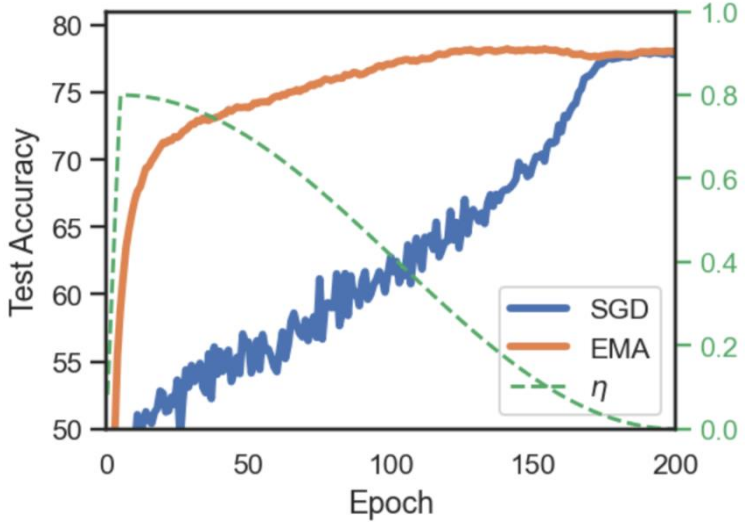


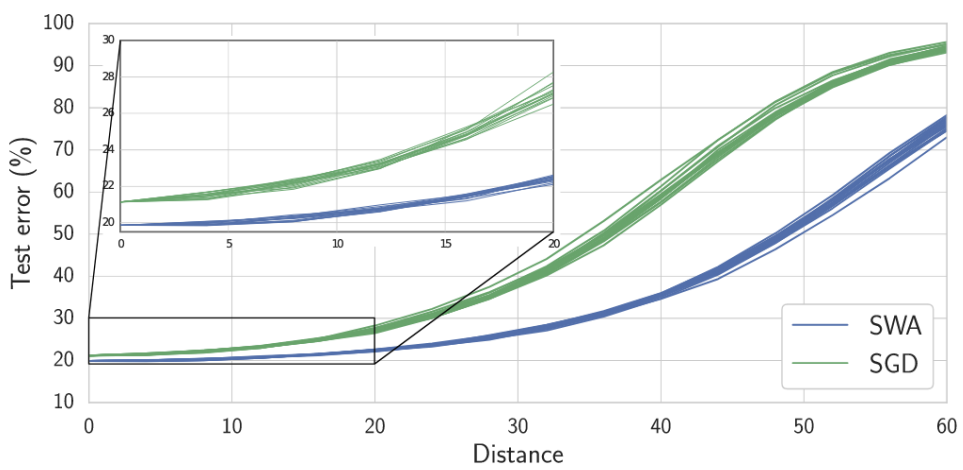
Training-Time Checkpoints Averaging

Existing methods

- Latest Weight Averaging (LAWA) $\boldsymbol{\theta}_T^{\text{LAWA}} := \frac{1}{k} \sum_{i=T-k+1}^{E} \boldsymbol{\theta}_i \,.$
- Exponential Moving Average (EMA) $\mathbf{x}_0^{\text{EMA}} = \mathbf{x}_0,$ and $\mathbf{x}_{t+1}^{\text{EMA}} = \alpha \mathbf{x}_t^{\text{EMA}} + (1-\alpha)\mathbf{x}_{t+1}.$
- Stochastic Weight Averaging (SWA) $w_{\text{SWA}} \leftarrow \frac{w_{\text{SWA}} \cdot n_{\text{models}} + w}{n_{\text{models}} + 1}$
- … …







Our SeWA

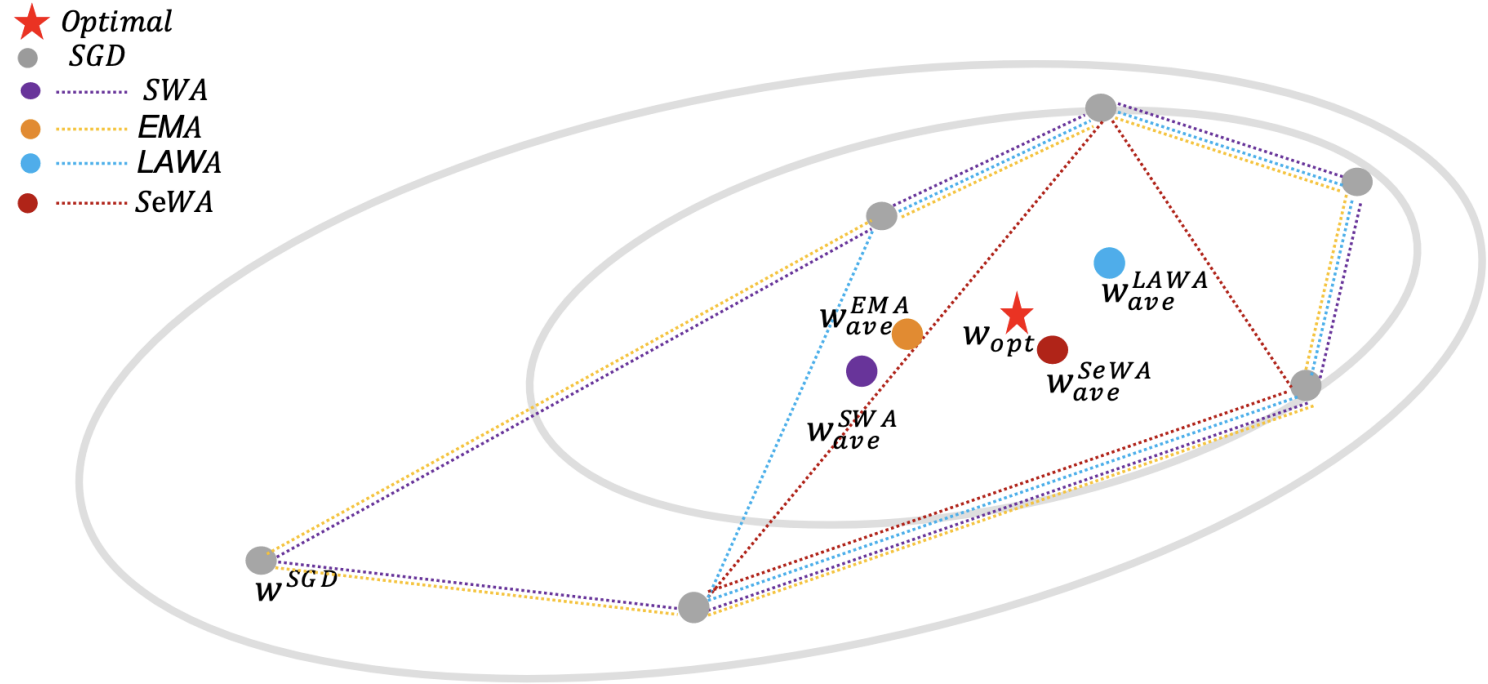
Adaptively selects models for averaging during training

$$\boldsymbol{\theta}_T^{\text{SeWA}} := \frac{1}{K} \sum_{i=T-k+1}^{T} m_i \boldsymbol{\theta}_i$$

binary mask $\in \{0,1\}^{|\theta|}$

Algorithm 1: Selected Weight Average (SeWA)

Input: Checkpoints $\mathbf{w}$, hyper-parameter t
Init: Mask probability s;
for $i = 1, \ldots, max_iteration$ **do**
  /* Gumbel-softmax sampling */
  for $m = 1, \ldots, M$ **do**
    Sample $u^{(m)} \sim \text{Uniform}(0, 1)$;
    Compute $L\left(\mathbf{w}(\text{GS}(s, u^{(m)}, t)\right)$;
  end
  /* Learning mask probability */
  Optimize $\hat{F}(s) = \frac{1}{M} \sum_{m=1}^{M} L\left(\mathbf{w}(\text{GS}(s, u^{(m)}, t)\right)$;
end
Output: Mask m based on K largest probabilities in s



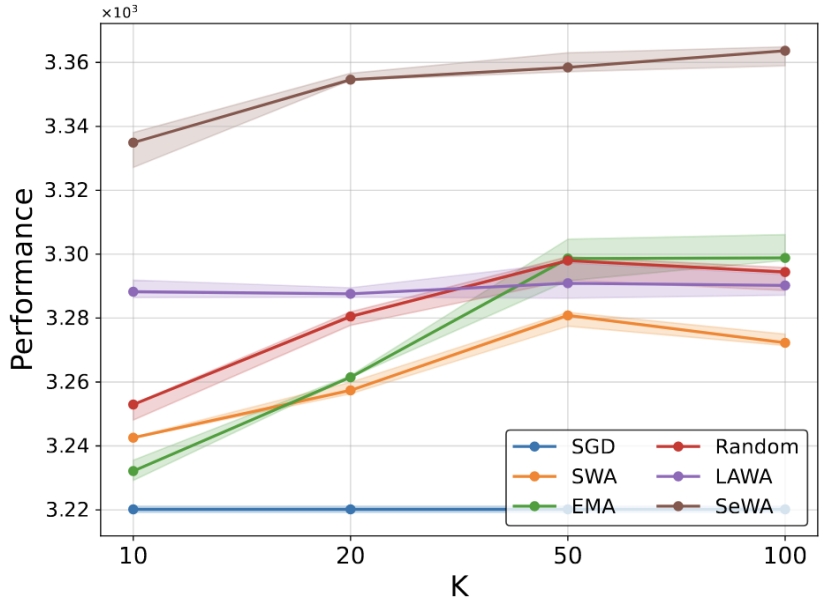
Peng Wang, et al. "SeWA: Selective Weight Average via Probabilistic Masking" ArXiv. 2025

SeWA Analysis and Experiments

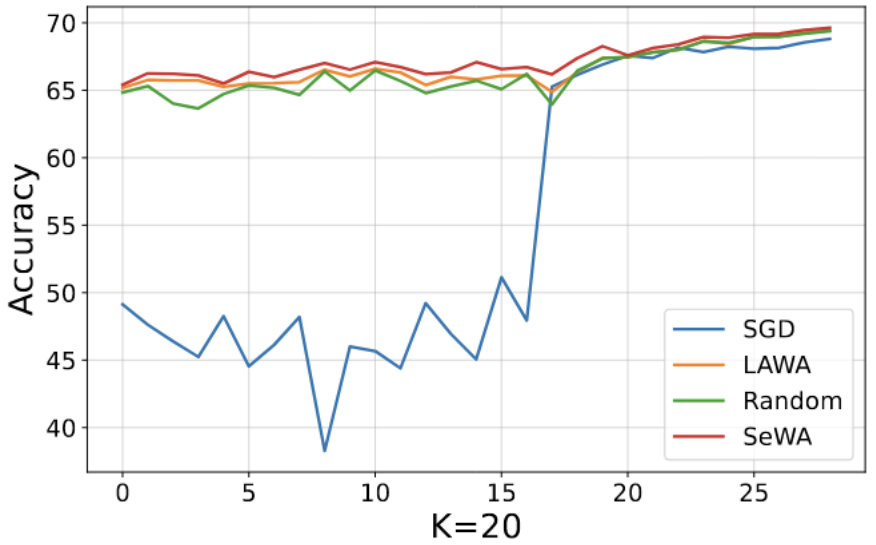
Stability and generalization analysis

(Convex Case Bound) SeWA has uniform stability of $\epsilon_{gen} \leq \frac{2\alpha L^2 s}{n}\left(T - \frac{k}{2}\right)$

(Non-Convex Case Bound) SeWA has uniform stability of $\epsilon_{gen} \leq \frac{1+\frac{1}{c\beta}}{n-1}\left(2csL^2(1+ke^{c\beta})k^{-1}\right)^{\frac{k}{c\beta+k}} \cdot T^{\frac{c\beta}{c\beta+k}}$



Comparison on the D4RL benchmark.



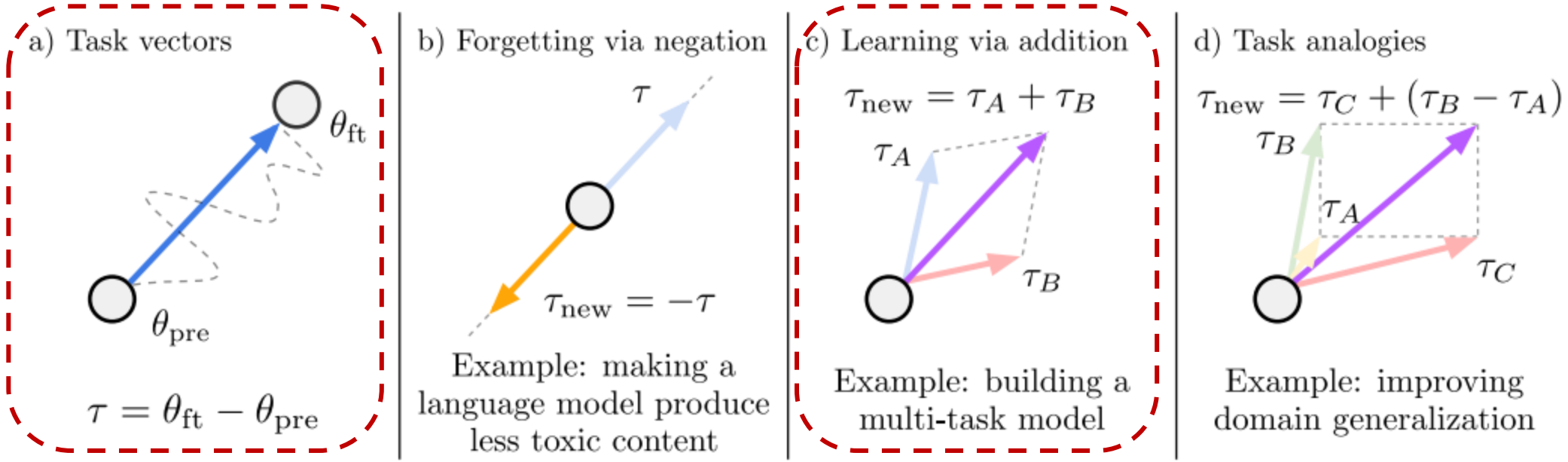
Comparison on the CIFAR-100 task.

Peng Wang, et al. "SeWA: Selective Weight Average via Probabilistic Masking" ArXiv. 2025

Model Fusion via Adaptive Weights

State-of-the-art MTL solutions

Merge multiple models to perform MTL (Task Arithmetic)



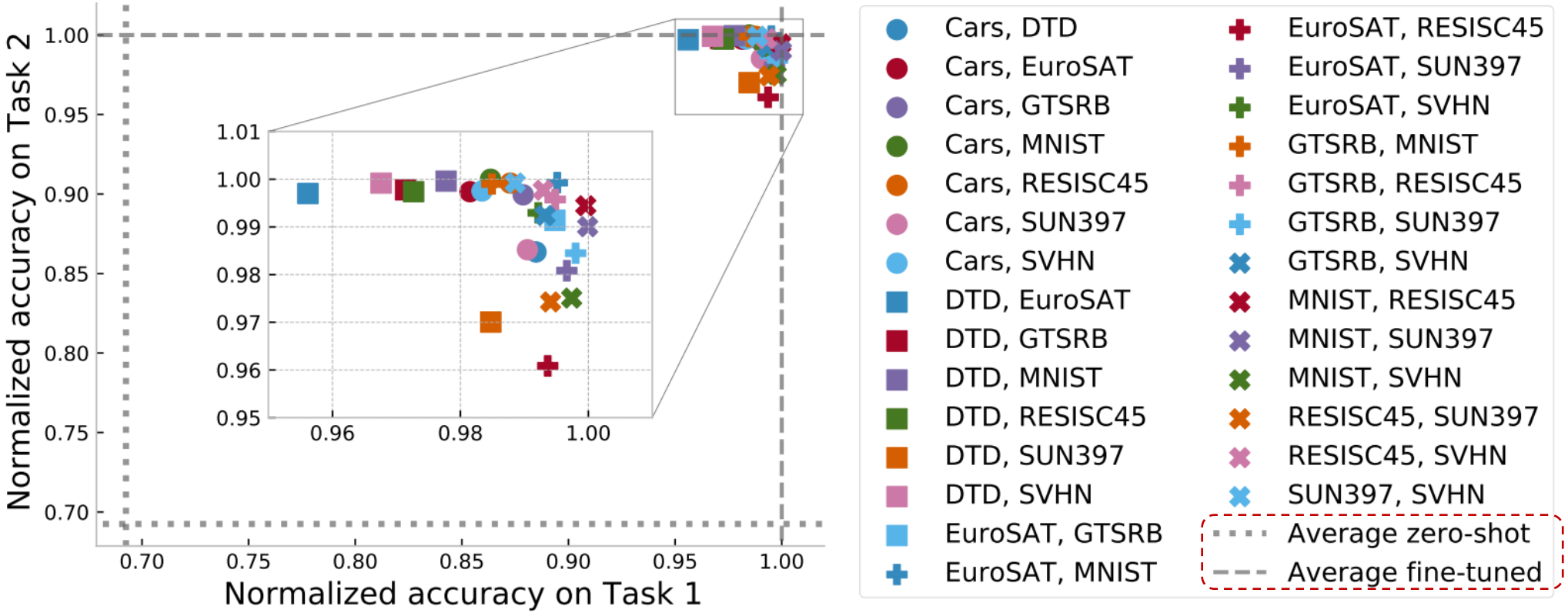
(a) A **task vector** is obtained by subtracting the weights of a pre-trained model from the weights of the same model after fine-tuning.
(c) **Adding** task vectors together perform the multi-task learning.

$$\tau_t = \theta_{\text{ft}}^t - \theta_{\text{pre}}$$
$$\tau_{\text{new}} = \sum_i \tau_i$$
$$\theta_{\text{new}} = \theta + \lambda \tau_{\text{new}}$$

Ilharco, Gabriel, et al. "Editing models with task arithmetic." ICLR, 2023.

State-of-the-art MTL solutions

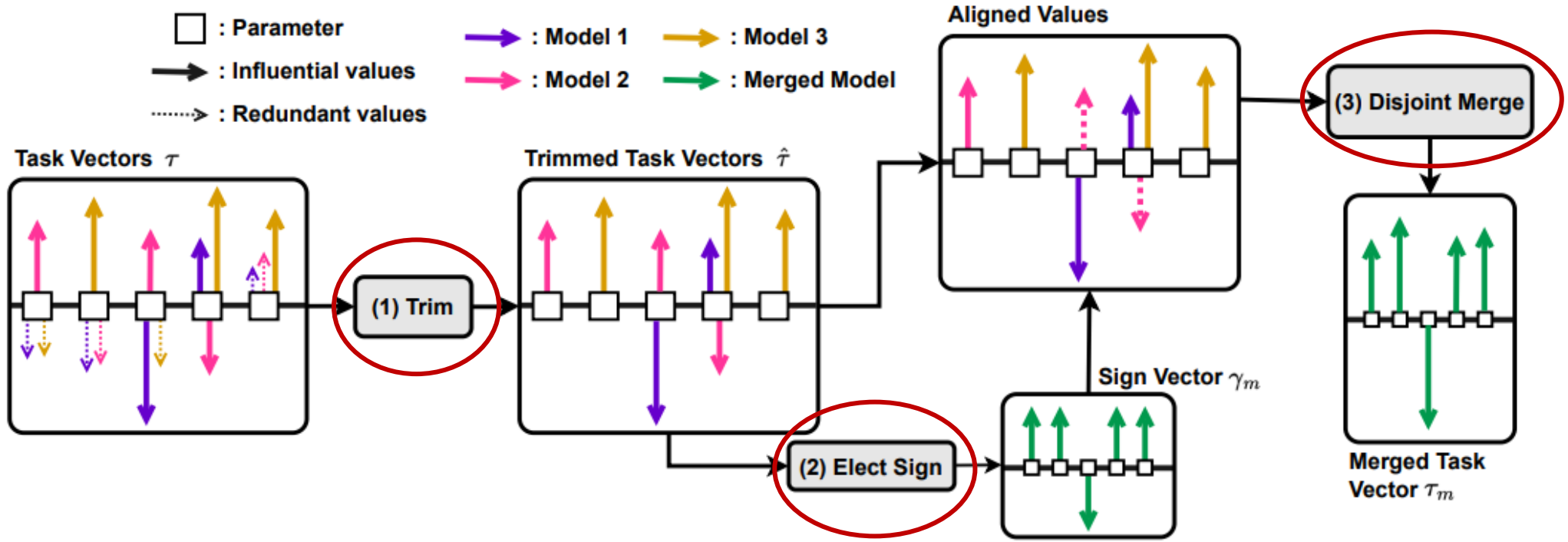
Merge multiple models to perform MTL (Task Arithmetic)



Adding task vectors from two tasks improves accuracy on both, resulting in a single model that is **competitive** with using two specialized fine-tuned models.

State-of-the-art MTL solutions

Merge multiple models to perform MTL (Ties-Merging)



Based on the task vector, three operations are used to reduce the task conflict
(1) It first *trims* the task vector values based on their magnitude.
(2) Then it *elects* the sign for each parameter by resolving sign conflicts.
(3) Finally, it picks values that align with the elected sign and take their mean as the final value.

Yadav, Prateek, et al. "Ties-Merging: Resolving interference when merging models." NeurIPS, 2023.

State-of-the-art MTL solutions

Table 1: Multi-task performance when merging ViT-B/32 models on eight tasks.

Method	SUN397	Cars	RESISC45	EuroSAT	SVHN	GTSRB	MNIST	DTD	**Avg Acc**
Pretrained	62.3	59.7	60.7	45.5	31.4	32.6	48.5	43.8	48.0
Individual	75.3	77.7	96.1	99.7	97.5	98.7	99.7	79.4	90.5
Traditional MTL	73.9	74.4	93.9	98.2	95.8	98.9	99.5	77.9	88.9
Weight Averaging	65.3	63.4	71.4	71.7	64.2	52.8	87.5	50.1	65.8
Fisher Merging (Matena & Raffel, 2022)	68.6	69.2	70.7	66.4	72.9	51.1	87.9	59.9	68.3
RegMean (Jin et al., 2023)	65.3	63.5	75.6	78.6	78.1	67.4	93.7	52.0	71.8
Task Arithmetic (Ilharco et al., 2023)	55.2	54.9	66.7	78.9	80.2	69.7	97.3	50.4	69.1
Ties-Merging (Yadav et al., 2023)	59.8	58.6	70.7	79.7	86.2	72.1	98.3	54.2	72.4

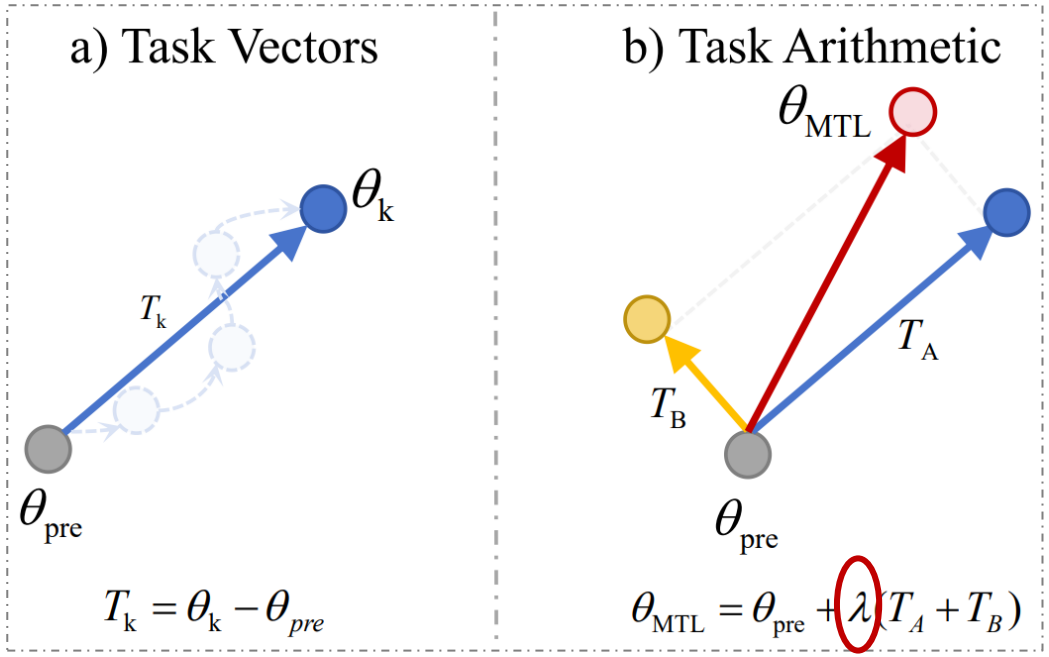


There is still **a large performance gap** between the task vector-based model merging approach and the traditional MTL.

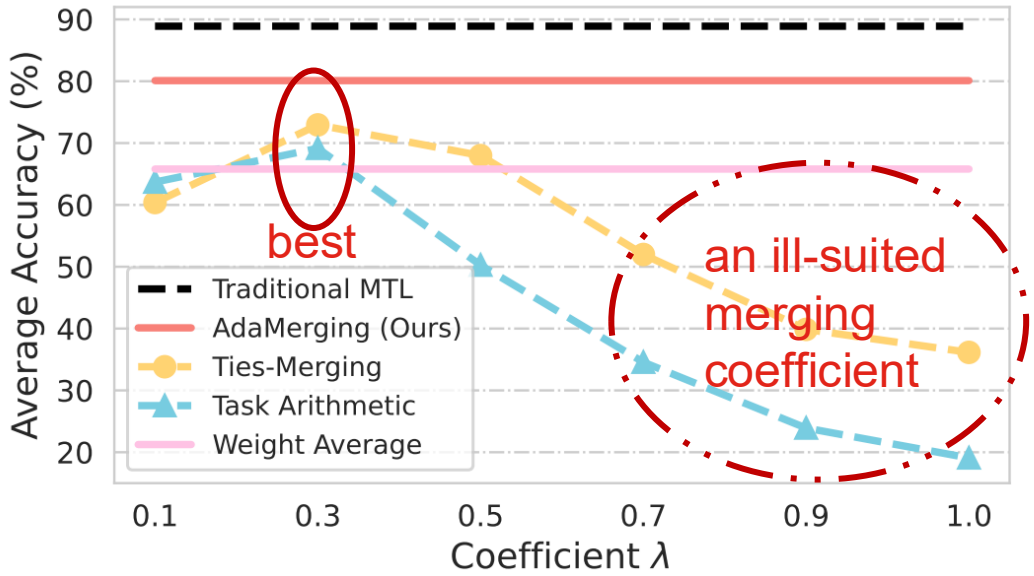
Ilharco, Gabriel, et al. "Editing models with task arithmetic." ICLR, 2023.
Yadav, Prateek, et al. "Ties-Merging: Resolving interference when merging models." NeurIPS, 2023.

Our AdaMerging

The merging coefficient λ is the only hyperparameter that associated with the task vector.



(a) Definition of "task vector"
(b) Task Arithmetic (Ilharco et al., 2023)



The impact of coefficient λ on the average accuracy of various MTL methods on eight tasks.

Yang, et al.: AdaMerging: Adaptive Model Merging for Multi-Task Learning." ICLR, 2024.

Our AdaMerging

Q1: Is it reasonable for all task vectors to share a merge coefficient ? Further, is it reasonable for all layers of a task vector to share a merge coefficient?

Task vectors/layers differ greatly and it is inappropriate to share a single coefficient.



Q2: How to determine the optimal task vector merging coefficient to promote the best merging of multiple tasks?

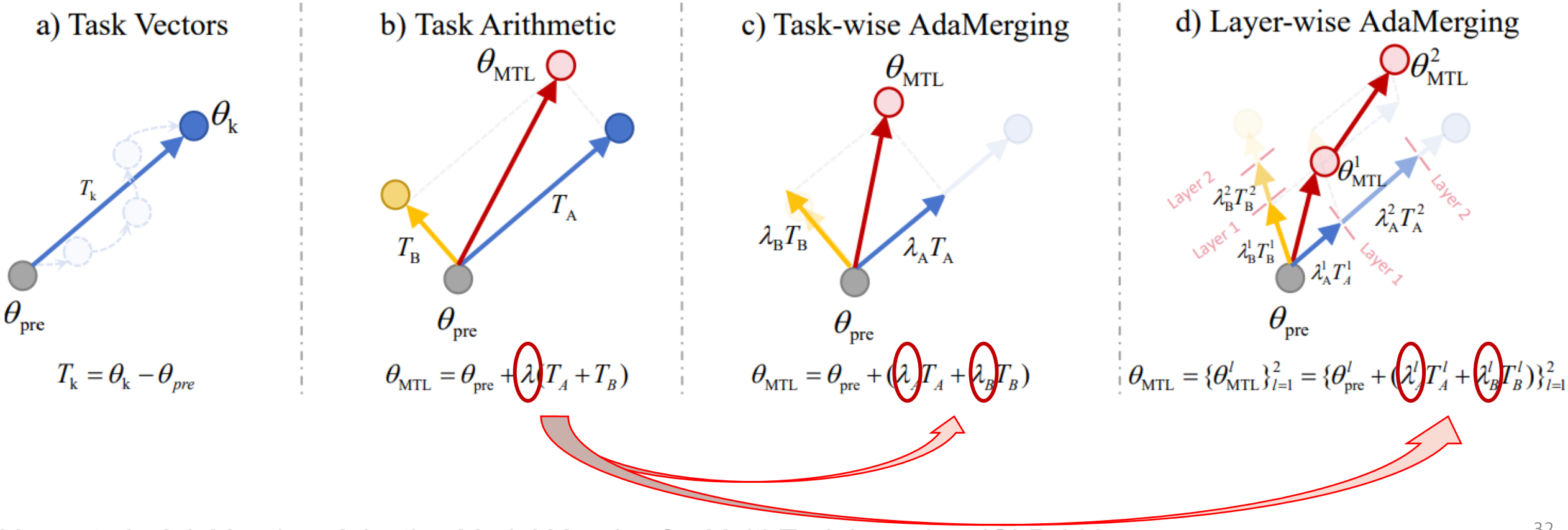
Grid Search? Combinatorial optimization search?



when dealing with a substantial number of tasks and layers...

Our AdaMerging

We propose Task-wise model merging and Layer-wise model merging .



Yang et al.: AdaMerging: Adaptive Model Merging for Multi-Task Learning. ICLR 2024

Our AdaMerging

Critical Challenge: How to optimize merging coefficients?

Because we only have the task vector of each task *without their raw training data*.

Solution: use entropy loss in TTA as a proxy goal.

Cross-Entropy Loss

$$H(y_i, \hat{y}_i) = -\sum_c^C p(y_{i,c}) \log p(\hat{y}_{i,c})$$

Shannon Entropy:

$$H(\hat{y}_i) = -\sum_c^C p(\hat{y}_{i,c}) \log p(\hat{y}_{i,c})$$

where

- y_i: Real Label (One-hot)
- $\hat{y}_i$: The prediction probability of the model for each class
- C: Total class number

➡ Depends only on the output of the model

Our AdaMerging

Optimization Objective

Based on the above verification, we take entropy minimization as the optimization proxy goal of the model merging coefficient in our AdaMerging/AdaMerging++.

$$\min_{\lambda_1,\lambda_2,\ldots,\lambda_K} \sum_{k=1}^{K} \sum_{x_i \in \mathcal{B}_k} H(f_{\theta_{MTL}}(x_i)) \text{ , where } \theta_{MTL} = \theta_{\text{pre}} + \sum_{k=1}^{K} \lambda_k T_k,$$

where B_k represents a batch of unlabeled test samples sampled in task k.

E. Yang et al.: AdaMerging: Adaptive Model Merging for Multi-Task Learning. ICLR 2024

AdaMerging Experiments

Table 1: Multi-task performance when merging ViT-B/32 models on eight tasks.

Method	SUN397	Cars	RESISC45	EuroSAT	SVHN	GTSRB	MNIST	DTD	**Avg Acc**
Pretrained	62.3	59.7	60.7	45.5	31.4	32.6	48.5	43.8	48.0
Individual	75.3	77.7	96.1	99.7	97.5	98.7	99.7	79.4	90.5
Traditional MTL	73.9	74.4	93.9	98.2	95.8	98.9	99.5	77.9	88.9
Weight Averaging	65.3	63.4	71.4	71.7	64.2	52.8	87.5	50.1	65.8
Fisher Merging (Matena & Raffel, 2022)	68.6	69.2	70.7	66.4	72.9	51.1	87.9	59.9	68.3
RegMean (Jin et al., 2023)	65.3	63.5	75.6	78.6	78.1	67.4	93.7	52.0	71.8
Task Arithmetic (Ilharco et al., 2023)	55.2	54.9	66.7	78.9	80.2	69.7	97.3	50.4	69.1
Ties-Merging (Yadav et al., 2023)	59.8	58.6	70.7	79.7	86.2	72.1	98.3	54.2	72.4
Task-wise AdaMerging (Ours)	58.0	53.2	68.8	85.7	81.1	84.4	92.4	44.8	71.1
Task-wise AdaMerging++ (Ours)	60.8	56.9	73.1	83.4	87.3	82.4	95.7	50.1	73.7
Layer-wise AdaMerging (Ours)	64.5	68.1	79.2	93.8	87.0	91.9	97.5	59.1	80.1
Layer-wise AdaMerging++ (Ours)	66.6	68.3	82.2	94.2	89.6	89.0	98.3	60.6	81.1

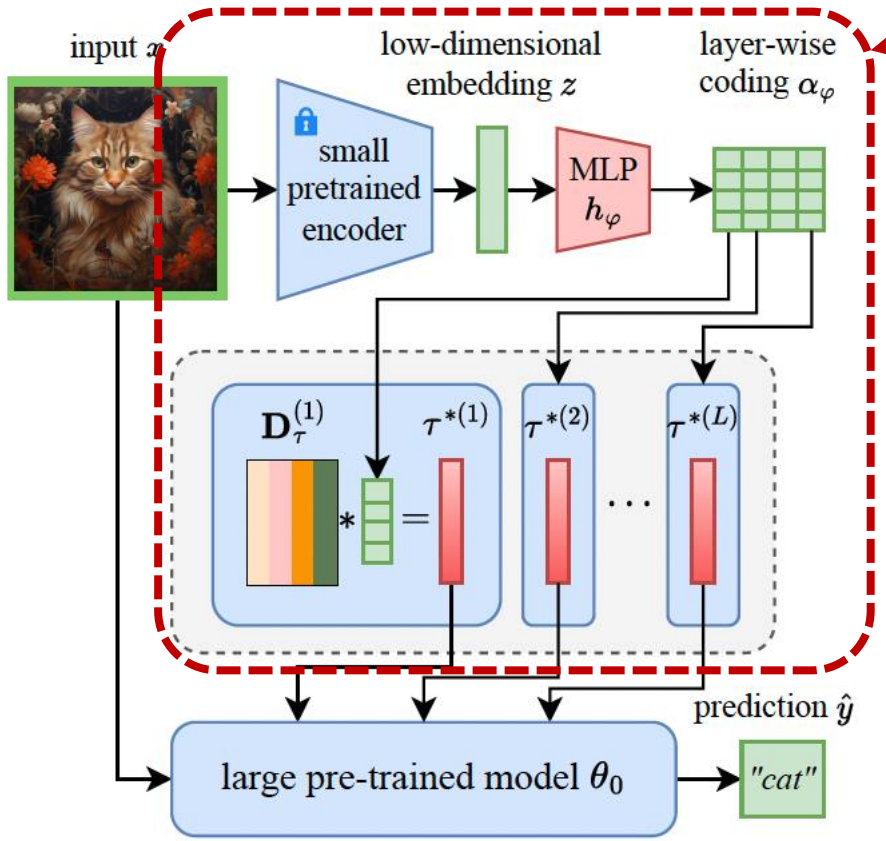
Table 3: Generalization results on two unseen tasks when merging ViT-B/32 models on six tasks.

	Seen Tasks							Unseen Tasks		
Method	SUN397	Cars	RESISC45	DTD	SVHN	GTSRB	**Avg Acc**	MNIST	EuroSAT	**Avg Acc**
Task Arithmetic (Ilharco et al., 2023)	63.3	62.4	75.1	57.8	84.6	80.4	70.6	77.2	46.2	61.7
Ties-Merging (Yadav et al., 2023)	67.8	66.2	77.2	56.7	77.1	70.9	69.3	75.9	43.3	59.6
AdaMerging (Ours)	65.2	65.9	88.5	61.1	92.2	91.5	77.4	84.0	56.1	70.0
AdaMerging++ (Ours)	68.2	67.6	86.3	63.6	92.6	89.8	78.0	83.9	53.5	68.7
Method	SUN397	Cars	GTSRB	EuroSAT	DTD	MNIST	**Avg Acc**	RESISC45	SVHN	**Avg Acc**
Task Arithmetic (Ilharco et al., 2023)	64.0	64.0	75.2	87.7	57.0	95.7	73.9	52.3	44.9	51.1
Ties-Merging (Yadav et al., 2023)	68.0	67.1	67.7	78.4	56.5	92.8	71.8	58.7	49.2	53.9
AdaMerging (Ours)	67.1	67.8	94.8	94.4	59.6	98.2	80.3	50.2	60.9	55.5
AdaMerging++ (Ours)	68.9	69.6	91.6	94.3	61.9	98.7	80.8	52.0	64.9	58.5

Model Fusion via Mixing/Upscaling

Data-Adaptive Weight-Ensembling (DAWE)

Dictionary based method, Construct a dictionary using the set of task vectors



Hypernetwork

Instead of finding a single static solution:

$$\theta_{merged} = \theta_{pre} + \lambda \sum_i \tau_i$$

DAWE dynamically generate parameters

$$\theta^{*(l)} = \mathcal{H}^{(l)}(x;\varphi) = \theta_0{}^{(l)} + \mathbf{D}_\tau^{(l)} h^{(l)}(E(x);\varphi),$$

The hypernetwork is fine-tuned via test-time entropy minimization

A Tang, et al. "Data-Adaptive Weight-Ensembling for Multi-Task Model Fusion", IJCV (2024~2025).

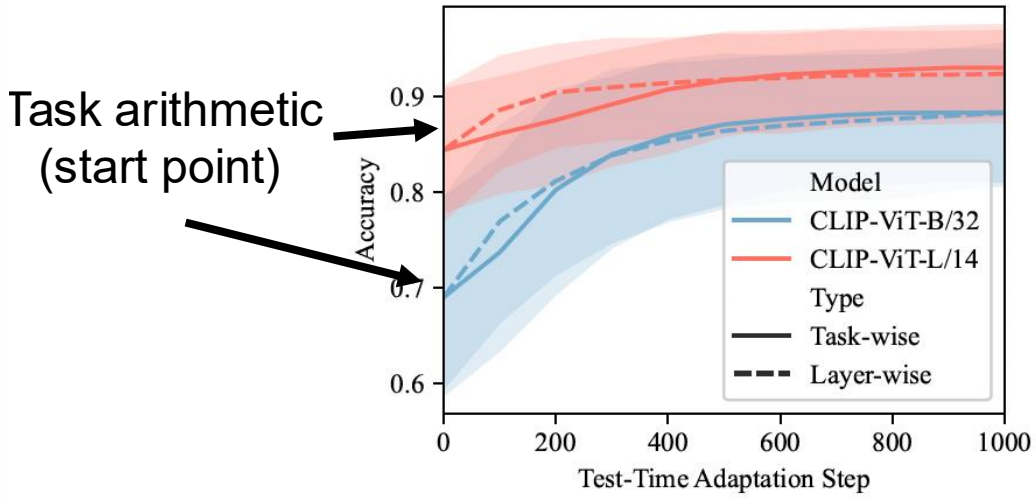
Data-adaptive Weight-Ensembling Experiments

Merging CLIP models (Image classification tasks)

Table 2. Multi-task performance when merging CLIP-ViT-B/32 models on all eight tasks.

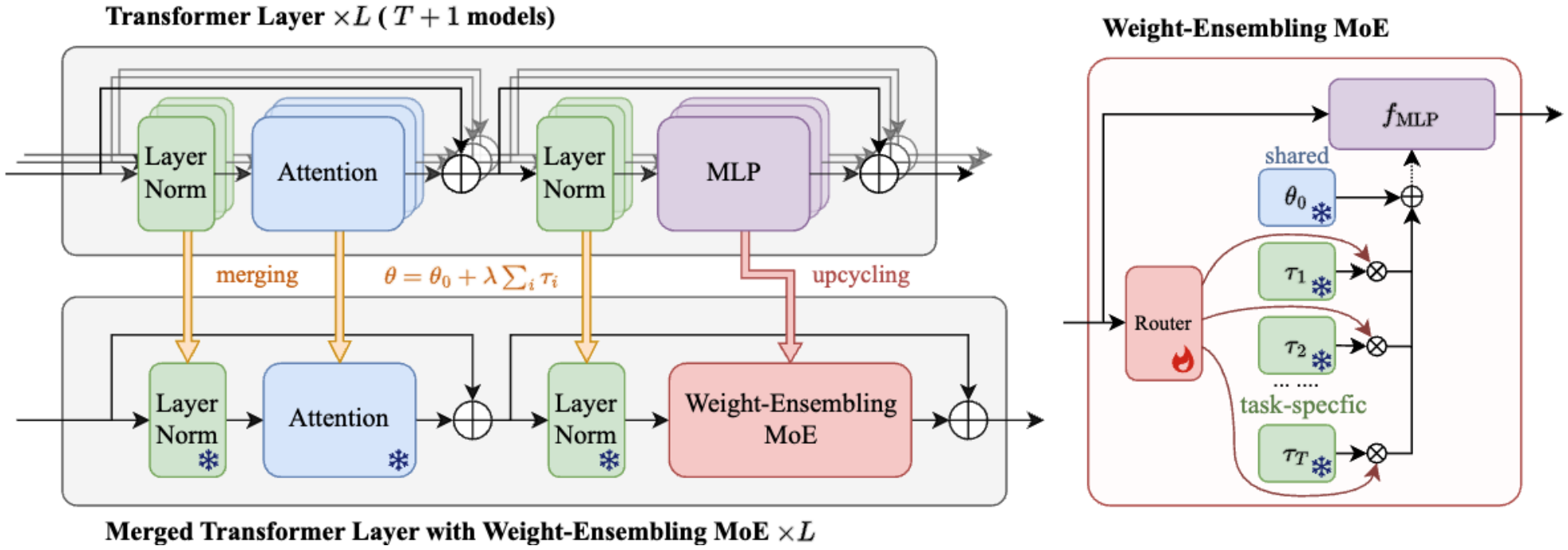
METHOD	SUN397	CARS	RESISC45	EUROSAT	SVHN	GRSRB	MNIST	DTD	AVG.
PRE-TRAINED	63.2	59.6	60.2	45.0	31.6	32.6	48.3	44.4	48.1
INDIVIDUAL	**75.3**	**77.7**	**96.1**	**99.9**	**97.5**	98.7	**99.7**	**79.4**	**90.5**
TRADITIONAL MTL	73.9	74.4	93.9	98.2	95.8	**98.9**	99.5	77.9	88.9
Multi-task Model Fusion Methods									
WEIGHT AVERAGING	65.3	63.3	71.4	73.6	64.2	52.8	87.5	50.1	66.0
FISHER MERGING	68.6	69.2	70.7	66.4	72.9	51.1	87.9	59.9	68.3
REGMEAN	65.3	63.5	75.6	78.6	78.1	67.4	93.7	52.0	71.8
TASK ARITHMETIC	55.3	54.9	66.7	77.4	80.2	69.7	97.3	50.1	69.0
TIES-MERGING	65.0	64.3	74.7	76.8	81.3	69.4	96.5	54.3	72.8
ADAMERGING (TASK)	58.3	53.2	71.8	80.1	81.6	84.4	93.4	42.7	70.7
ADAMERGING++ (TASK)	60.8	56.9	73.1	83.4	87.3	82.4	95.7	50.1	73.7
ADAMERGING (LAYER)	64.2	69.5	82.4	92.5	86.5	93.7	97.7	61.1	80.9
~~ADAMERGING++ (LAYER)~~	~~66.6~~	~~68.3~~	~~82.2~~	~~94.2~~	~~89.6~~	~~89.8~~	~~98.3~~	~~60.6~~	~~81.1~~
DAWE (TASK-WISE)	**72.4**	**79.3**	**90.3**	**96.7**	**95.6**	**96.7**	**99.3**	**75.8**	**88.3**
DAWE (LAYER-WISE)	**70.2**	**80.0**	**91.2**	**97.4**	**94.4**	**97.8**	**99.2**	**75.6**	**88.2**

Very close to the traditional MTL upper bound



Weight-Ensembling MoE

Improved dictionary-based method



1. Upcycling the MLPs to get weight-ensembling MoE modules
2. Merge the rest of models via task arithmetic

$$\begin{cases} w = mean(r(\mathbf{h}^{\text{in}}_{1:N})) \in \mathbb{R}^{T\times 1}, \\ \theta^{\text{MLP}} = \theta^{\text{MLP}}_0 + \mathbf{D}_\tau w, \\ \mathbf{h}^{\text{out}}_{1:N} = f_{\text{MLP}}(\mathbf{h}^{\text{in}}_{1:N}; \theta^{\text{MLP}}). \end{cases}$$

A.Tang, et al. "Merging Multi-Task Models via Weight-Ensembling Mixture of Experts", ICML 2024.

Weight-Ensembling MoE Experiments

Multi-task model merging experiments

Table 2. Multi-task performance when merging CLIP-ViT-B/32 models on all eight tasks.

METHOD	SUN397	CARS	RESISC45	EUROSAT	SVHN	GTSRB	MNIST	DTD	AVG.
PRE-TRAINED	63.2	59.6	60.2	45.0	31.6	32.6	48.3	44.4	48.1
INDIVIDUAL	**75.3**	**77.7**	**96.1**	**99.9**	**97.5**	98.7	**99.7**	**79.4**	**90.5**
TRADITIONAL MTL	73.9	74.4	93.9	98.2	95.8	**98.9**	99.5	77.9	88.9
Multi-task Model Fusion Methods									
WEIGHT AVERAGING	65.3	63.3	71.4	73.6	64.2	52.8	87.5	50.1	66.0
FISHER MERGING	68.6	69.2	70.7	66.4	72.9	51.1	87.9	59.9	68.3
REGMEAN	65.3	63.5	75.6	78.6	78.1	67.4	93.7	52.0	71.8
TASK ARITHMETIC	55.3	54.9	66.7	77.4	80.2	69.7	97.3	50.1	69.0
TIES-MERGING	65.0	64.3	74.7	76.8	81.3	69.4	96.5	54.3	72.8
ADAMERGING (TASK)	58.3	53.2	71.8	80.1	81.6	84.4	93.4	42.7	70.7
ADAMERGING++ (TASK)	60.8	56.9	73.1	83.4	87.3	82.4	95.7	50.1	73.7
ADAMERGING (LAYER)	64.2	69.5	82.4	92.5	86.5	93.7	97.7	61.1	80.9
ADAMERGING++ (LAYER)	66.6	68.3	82.2	94.2	89.6	89.0	98.3	60.6	81.1
WEMOE (OURS)	**74.1**	**77.4**	**93.7**	**99.1**	**96.2**	**98.9**	**99.6**	**76.4**	**89.4**

Previous method

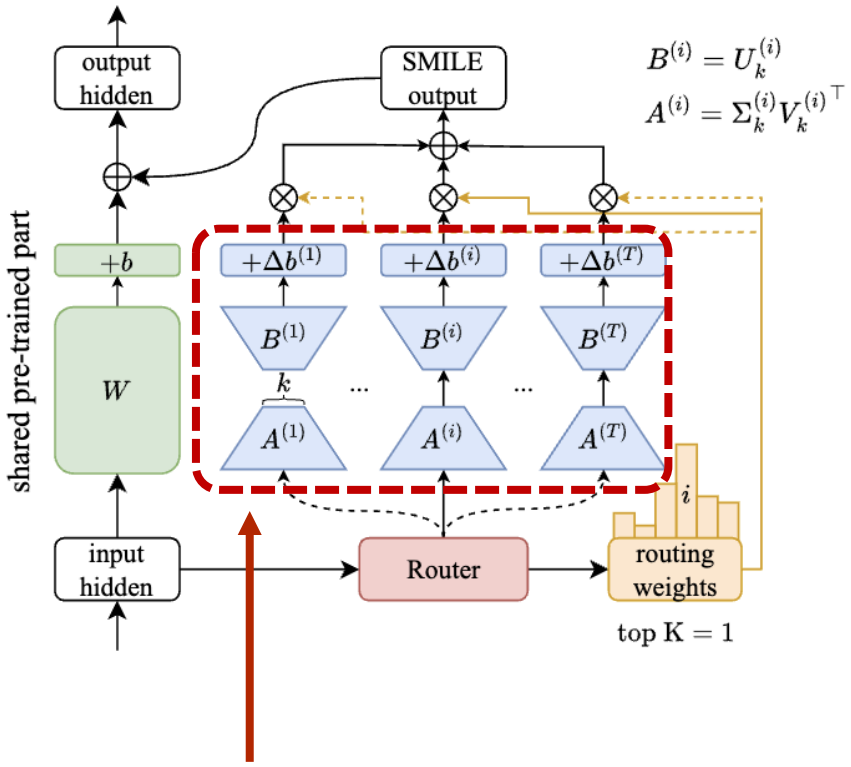
DAWE (task-wise)	**72.4**	79.3	90.3	96.7	**95.6**	96.7	**99.3**	**75.8**	**88.3**
DAWE (layer-wise)	70.2	**80.0**	**91.2**	**97.4**	94.4	**97.8**	99.2	75.6	88.2

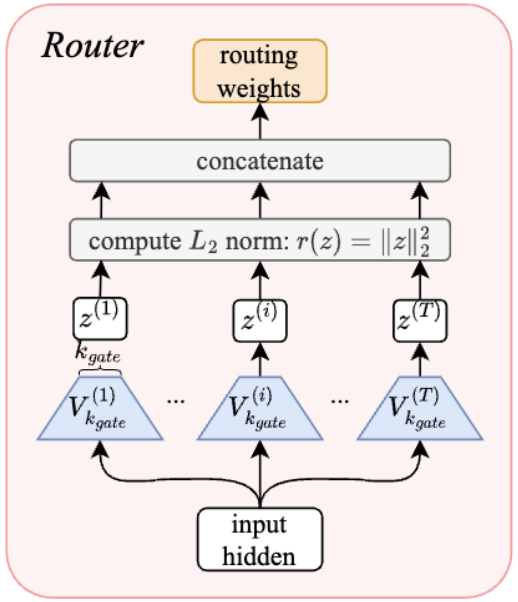
Reduce 40% parameters per task while maintain high performance

A.Tang, et al. "Merging Multi-Task Models via Weight-Ensembling Mixture of Experts", ICML 2024.

SMILE: Zero-Shot MoE Upscaling

Sparse **MI**xture of **L**ow-rank **E**xperts (SMILE) model.





Routing:

$$r^{(i)} = \left\| \sum_{j=1}^{k_{gate}} \left\langle x, v_j^{(i)} \right\rangle \right\|_2 = \left\| V_{k_{gate}}^{(i)\top} x \right\|_2$$

$$p_i = \text{softmax}_i(r^{(i)}) = \text{softmax}_i \left(\left\| V_{k_{gate}}^{(i)\top} x \right\|_2 \right)$$

$$\lambda_i = \begin{cases} p_i, & p_i \in \text{TopK}(\{p_j\}_{j=1}^T, K) \\ 0, & \text{otherwise} \end{cases}$$

Outputs aggregation:

$$y_{merge} = Wx + b \quad \text{Shared}$$

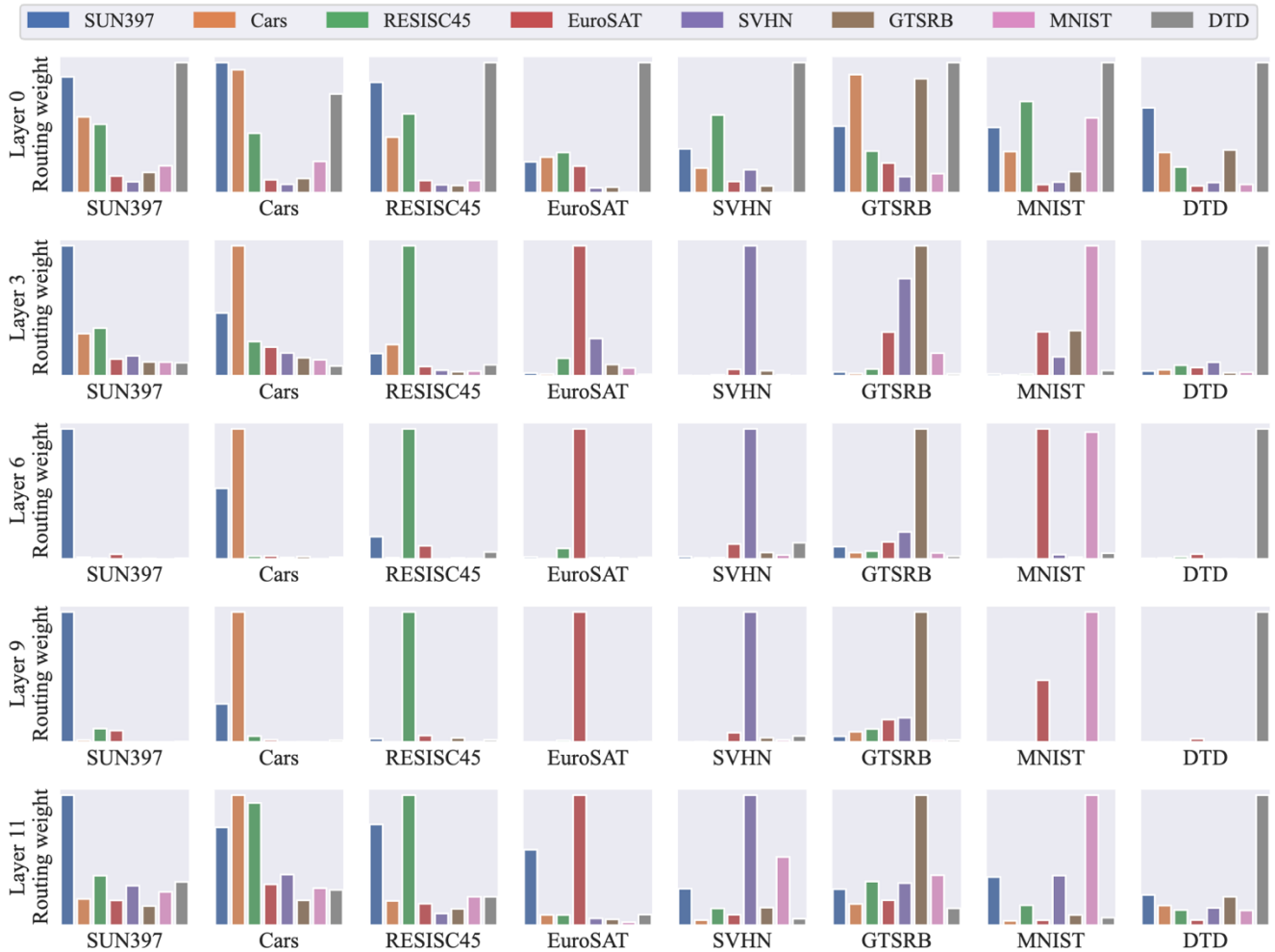
$$+ \sum_{i=1}^{T} \frac{\lambda_i}{\sum_{j=1}^{T} \lambda_j} \left(U_k^{(i)} \Sigma_k^{(i)} V_k^{(i)^T} x + \Delta b^{(i)} \right)$$

Expert outputs

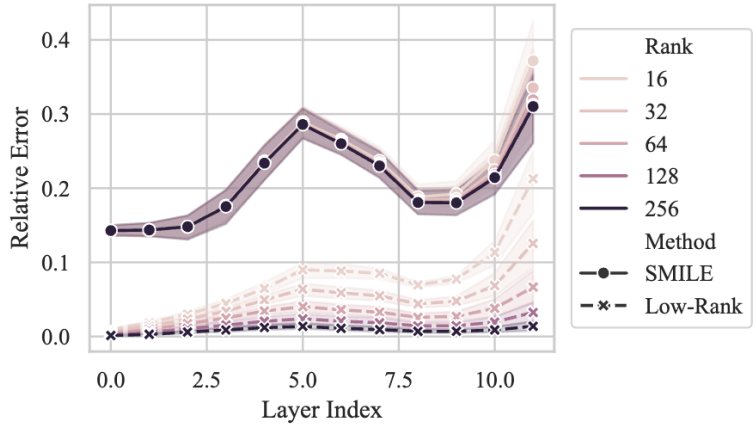
A.Tang, et al. "Zero-Shot Sparse Mixture of Low-Rank Experts Construction From Pre-Trained Foundation Models" *TPAMI 2025*

SMILE Experiments

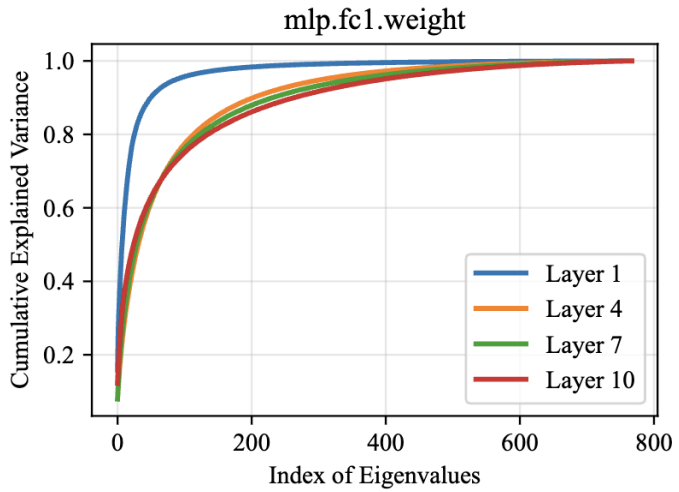
Routing Analysis



Error accumulation analysis

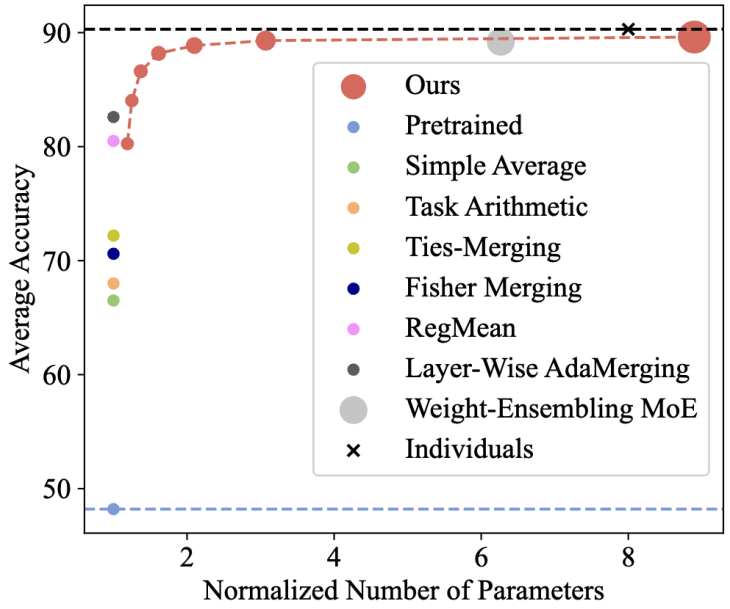


Cumulative explained variance of singular values



A.Tang, et al. "Zero-Shot Sparse Mixture of Low-Rank Experts Construction From Pre-Trained Foundation Models" *TPAMI 2025*

SMILE Experiments



Eight image classification tasks using ViT-B/32 models

Eight image classification tasks using ViT-B/32.

Method	SUN397	Cars	RESISC45	EuroSAT	SVHN	GTSRB	MNIST	DTD	Avg.
Individual	75.0	78.3	95.2	99.0	97.3	98.9	99.6	79.7	90.3 (100%)
Simple Averaging	65.4	62.6	70.8	76.9	64.5	54.9	86.3	50.9	66.5 (73.6%)
Fisher Merging	66.7	64.0	72.2	91.6	69.0	64.3	83.5	53.7	70.6 (78.2%)
RegMean	67.8	68.9	82.5	94.4	90.6	79.2	97.6	63.2	80.5 (89.1%)
Task Arithmetic	57.1	55.7	64.9	76.7	77.9	68.5	96.1	47.2	68.0 (75.3%)
Ties-Merging	67.1	64.2	74.1	76.8	77.7	69.4	94.1	54.0	72.2 (80.0%)
AdaMerging	67.9	71.3	83.5	92.7	87.4	92.9	98.2	67.0	82.6 (91.5%)
AdaMerging + Surgery	70.4	73.4	89.7	95.8	90.8	96.3	98.7	74.5	86.2 (95.5%)
TSVM	67.6	71.6	84.7	93.4	91.9	92.5	98.9	63.8	83.1 (92.0%)
ISO-C	69.7	72.9	86.0	89.8	87.6	92.4	98.5	65.4	82.8 (91.7%)
WEMoE (×6.27)	**73.7**	76.8	**93.4**	98.2	96.8	**98.2**	**99.6**	76.6	89.2 (98.8%)
SMILE (1, ×1.61)	73.6	74.4	89.5	98.1	95.4	97.3	99.5	76.3	87.7 (97.1%)
SMILE (2, ×3.07)	73.6	**77.8**	92.0	**98.3**	**96.9**	98.1	**99.6**	**78.1**	**89.3 (98.9%)**

Eight NLP tasks from GLUE benchmark using Flan-T5-base.

Method	CoLA	MNLI	MRPC	QNLI	QQP	RTE	SST2	STSB	Avg.
Individual	75.0	83.4	87.5	91.5	85.4	85.9	93.6	88.7	86.4 (100%)
Weight Averaging	69.1	62.6	79.4	89.8	83.9	81.2	91.7	73.2	78.9 (91.3%)
Task Arithmetic	70.5	57.8	78.4	90.2	83.6	80.5	92.3	77.8	78.9 (91.3%)
Ties-Merging	70.3	65.0	78.9	90.2	83.5	81.6	91.7	78.3	79.9 (92.5%)
SMILE (1, ×1.26)	72.0	**84.2**	84.3	91.3	84.7	84.1	93.3	87.0	85.1 (98.5%)
SMILE (2, ×1.52)	**73.2**	**84.2**	**85.0**	**91.3**	**84.9**	**84.8**	**93.5**	**87.3**	**85.5 (99.0%)**

A.Tang, et al. "Zero-Shot Sparse Mixture of Low-Rank Experts Construction From Pre-Trained Foundation Models" *TPAMI 2025*

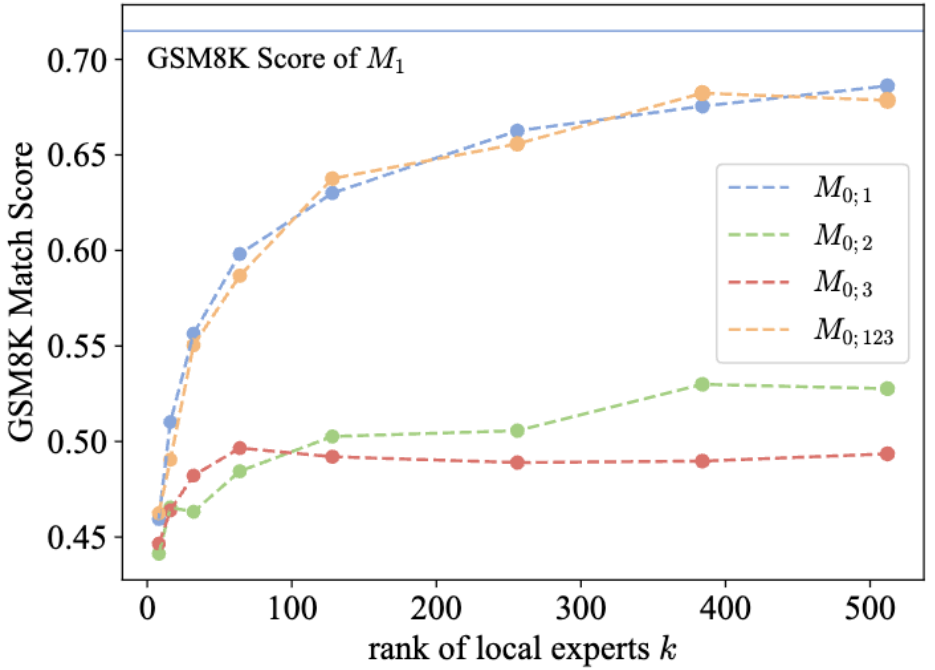
SMILE Experiments

Experiments on LLMs:

Base pre-trained model (M0): mistralai/Mistral-7B-v0.1

- M1: meta-math/MetaMath-Mistral-7B
- M2: cognitivecomputations/dolphin-2.1-mistral-7b
- M3: uukuguy/speechless-code-mistral-7b-v1.0

Model	MMLU	TruthfulQA	GSM8K	ARC Challenge
M_0 (pre-trained)	59.64	42.62	38.81	53.92
M_1	60.56	44.79	**71.49**	51.02
M_2	60.56	**55.88**	56.93	57.00
M_3	**61.18**	47.47	48.98	**57.68**
MergeKit-MoE [32]	60.05	46.36	63.84	**54.52**
$M_{0;123}$ (Ours)	**60.66**	**52.79**	**67.85**	54.35
`Qwen1.5-14B`	66.11	52.00	69.37	49.93

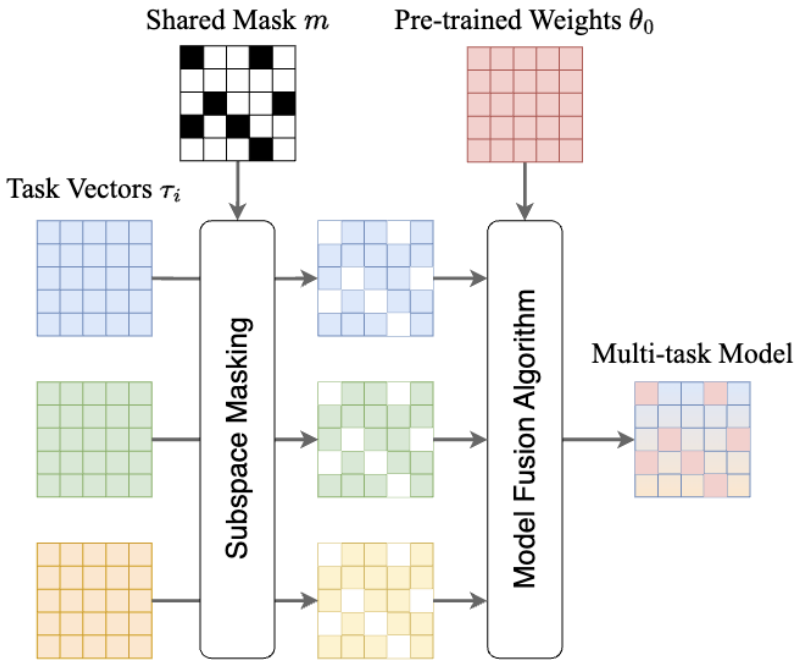


The GSM8K benchmark scores of upscaled Mistral-7B models with varying k.

A.Tang, et al. "Zero-Shot Sparse Mixture of Low-Rank Experts Construction From Pre-Trained Foundation Models" *TPAMI 2025*

Model Fusion via Meta Learning

Merging on a Common Subspace (Concrete)



(a) Framework overview

1. Mask sampling

$$\mathbb{P}(m=1) = \mathbb{P}\left(\log\frac{u}{1-u} + \log\frac{\sigma(x)}{1-\sigma(x)} > 0\right),$$
$$u \sim \text{Uniform}(0,1).$$

2. Identify a common low-dimensional subspace

$$\tau_i'' = \frac{\tau_i'}{\mathbb{E}_{m\sim\mathbf{m}}[m]} = \frac{\tau_i \circ \mathbf{m}}{\mathbb{E}_{m\sim\mathbf{m}}[m]}.$$

3. Merge models on this subspace.

$$\max_m \max_w \frac{1}{n}\sum_i^n \text{Score}\left(\mathcal{A}(\theta_0, \{\mathcal{M}(\tau_j, m)\}_{j=1}^n; w), s_i\right).$$

Challenge: Binary Mask is not differentiable!

A.Tang, et al. "Concrete Subspace Learning based Interference Elimination for Multi-task Model Fusion", under review - 2023.

Concrete Masking Experiments

CLIP models on vision tasks

Table 1: Multi-task performance when merging CLIP-ViT-B/32 models on all eight tasks.

Method	SUN397	Cars	RESISC45	EuroSAT	SVHN	GRSRB	MNIST	DTD	Avg.
Individual	75.3	77.7	96.1	99.9	97.5	98.7	99.7	79.4	90.5
Traditional MTL	73.9	74.4	93.9	98.2	95.8	98.9	99.5	77.9	88.9
Weight Averaging	65.3	63.3	71.4	73.6	64.2	52.8	87.5	50.1	66.0
Fisher Merging	68.6	69.2	70.7	66.4	72.9	51.1	87.9	59.9	68.3
RegMean	65.3	63.5	75.6	78.6	78.1	67.4	93.7	52.0	71.8
Task Arithmetic (TA)-Based									
Task Arithmetic	55.3	54.9	66.7	77.4	80.2	69.7	97.3	50.1	69.0
Ties-Merging	**65.0**	**64.3**	74.7	76.8	81.3	69.4	96.5	**54.3**	72.8
Concrete TA	62.5	61.1	**76.0**	**95.7**	**91.0**	**81.9**	**98.5**	51.9	**77.3**
Task-wise AdaMerging (TW AM)-Based									
TW AM	58.3	53.2	71.8	80.1	81.6	84.4	93.4	42.7	70.7
TW AM++	60.8	56.9	73.1	83.4	87.3	82.4	95.7	**50.1**	73.7
TW Concrete AM	**62.7**	**58.9**	**74.5**	**94.8**	**91.1**	**95.0**	**98.1**	34.6	**76.2**
Layer-wise AdaMergin (LW AM)-Based									
LW AM	64.2	69.5	82.4	92.5	86.5	93.7	97.7	61.1	80.9
LW AM++	66.6	68.3	82.2	94.2	89.6	89.0	98.3	60.6	81.1
LW Concrete AM	**67.8**	**70.0**	**87.5**	**96.0**	**91.6**	**96.7**	**98.7**	**63.8**	**84.0**

Flan-T5 models on NLP tasks

Table 4: Multi-task performance when merging Flan-T5-base (LoRA fine-tuned) models on all eight tasks.

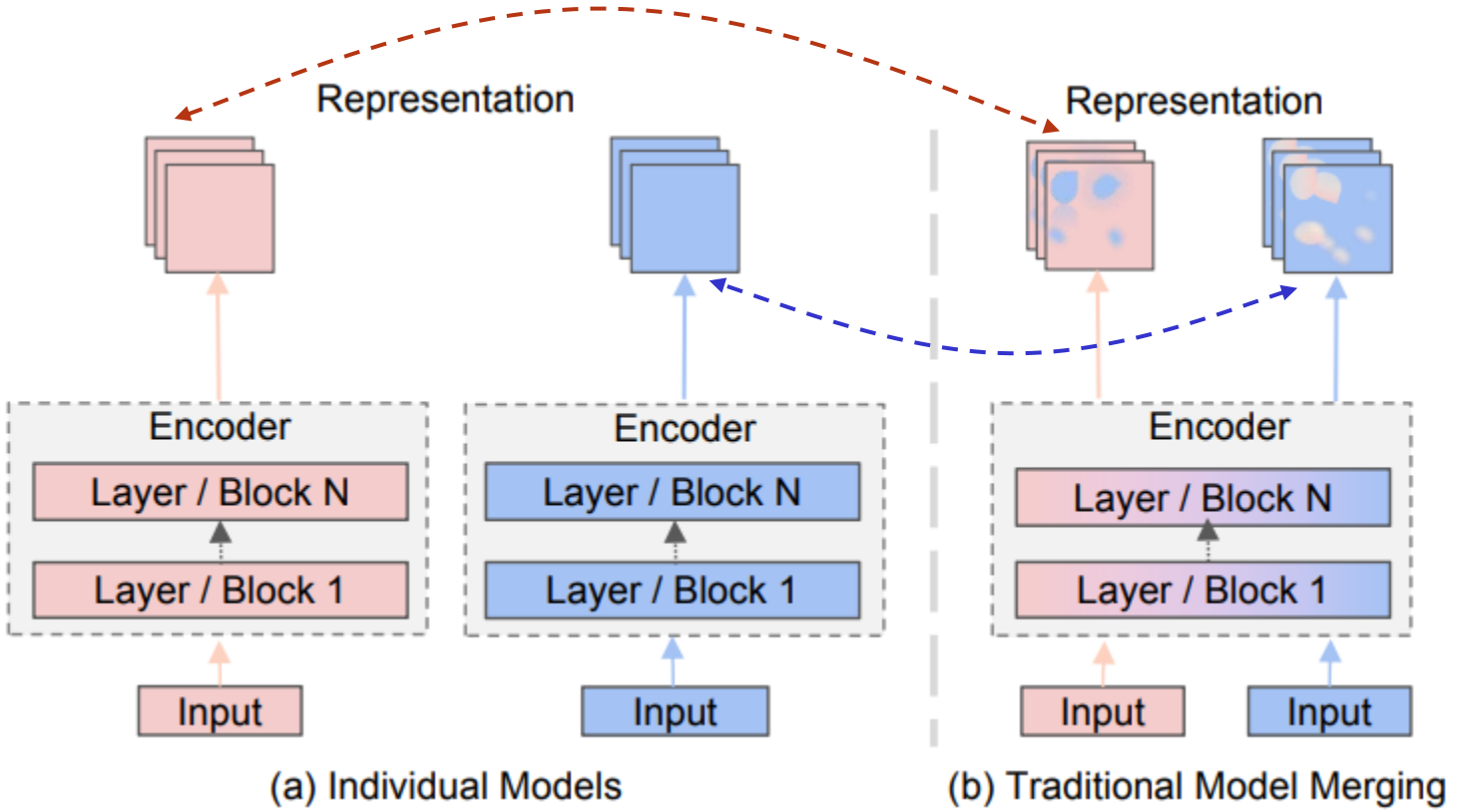
Method	CoLA	MNLI	MRPC	QNLI	QQP	RTE	SST2	STSB	Avg.
Individual	69.1	82.7	85.5	90.9	84.0	84.4	92.9	87.4	84.6
Weight Averaging	69.7	59.7	78.9	90.1	83.8	80.5	91.2	72.0	78.2
Task Arithmetic (TA)-Based									
Task Arithmetic	68.8	55.2	78.7	89.8	83.7	79.1	91.5	72.4	77.4
Ties-Merging	68.3	56.3	**79.4**	89.8	83.7	79.4	91.6	71.2	77.5
Concrete TA	**69.1**	**58.1**	78.4	**89.9**	83.5	79.4	91.6	**73.4**	**78.0**
Layer-wise AdaMergin (LW AM)-Based									
LW AM	**69.1**	**60.3**	78.4	**90.0**	**83.6**	79.1	91.6	74.1	78.3
LW Concrete AM	69.0	59.4	**80.1**	89.9	82.9	79.1	**91.7**	**75.4**	**78.5**

A.Tang, et al. "Concrete Subspace Learning based Interference Elimination for Multi-task Model Fusion"

Model Fusion via Fine-tuning

Our Representation Surgery

Question: Is the representation of the merged model close to that of the individual model?

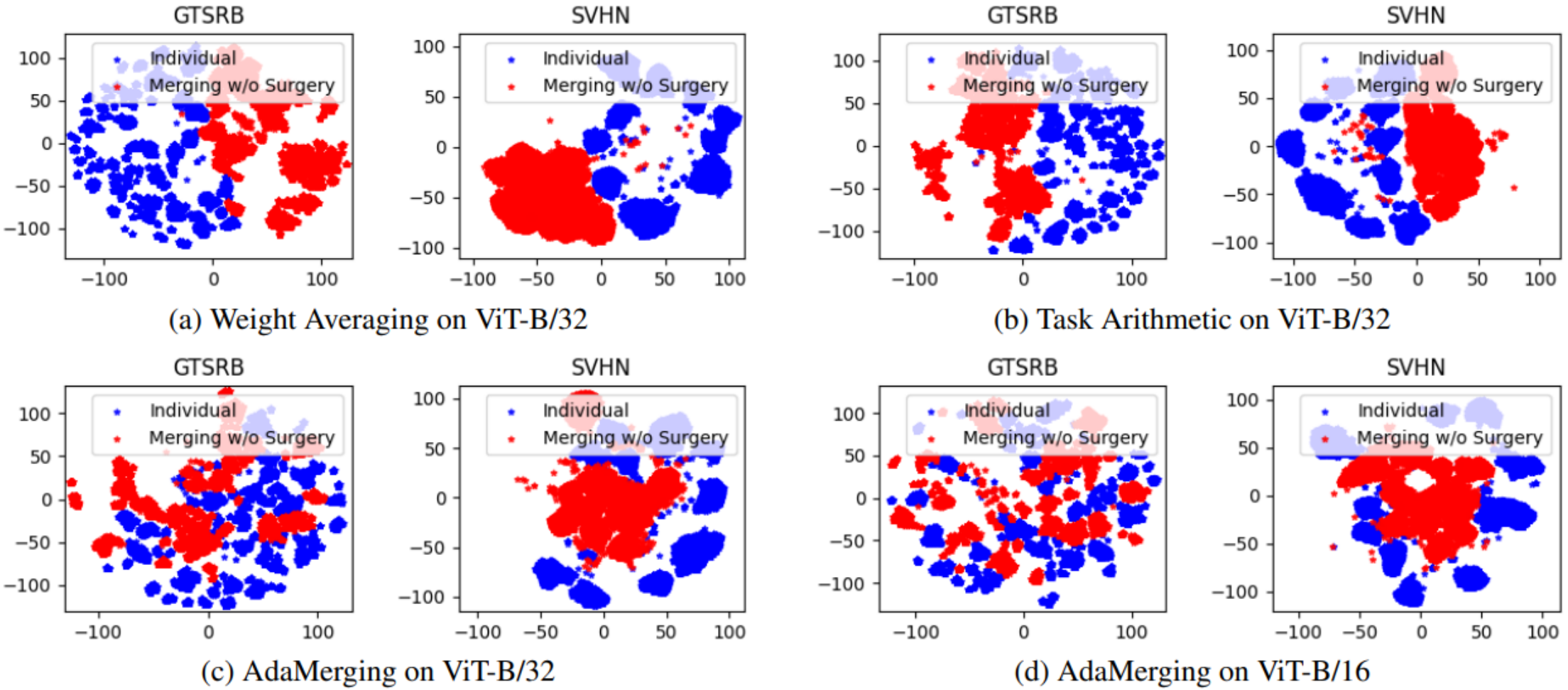


(a) Multiple individual trained models.
(b) Traditional model merging schemes (e.g., Task Arithmetic, Ties-Merging, AdaMerging, etc.) merge multiple individual models into one.

E. Yang, et al., "Representation Surgery for Multi-Task Model Merging", ICML 2024.

Our Representation Surgery

“representation bias problem”



(a) Weight Averaging on ViT-B/32

(b) Task Arithmetic on ViT-B/32

(c) AdaMerging on ViT-B/32

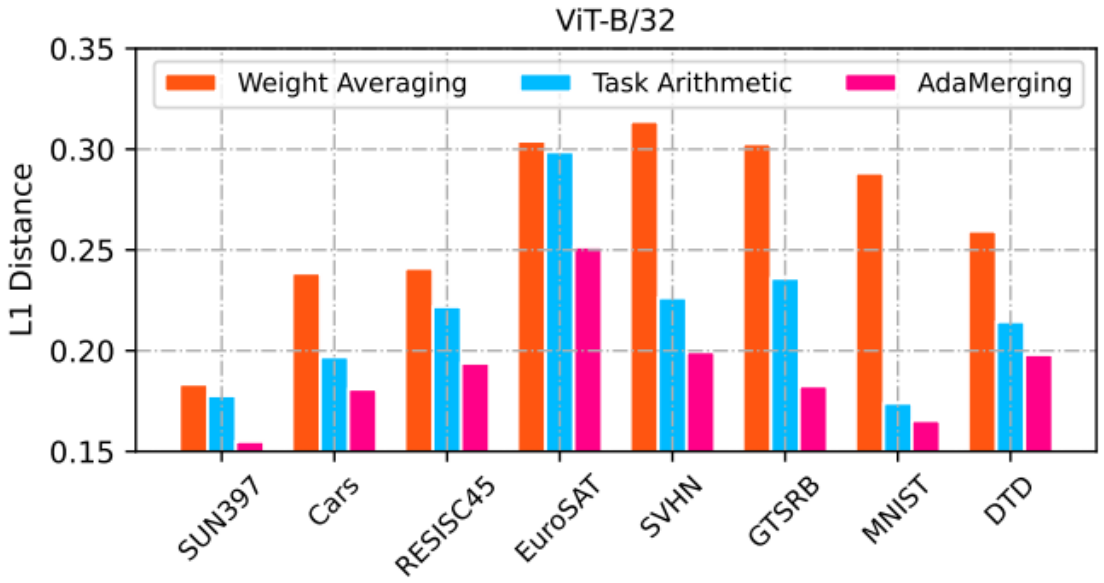
(d) AdaMerging on ViT-B/16

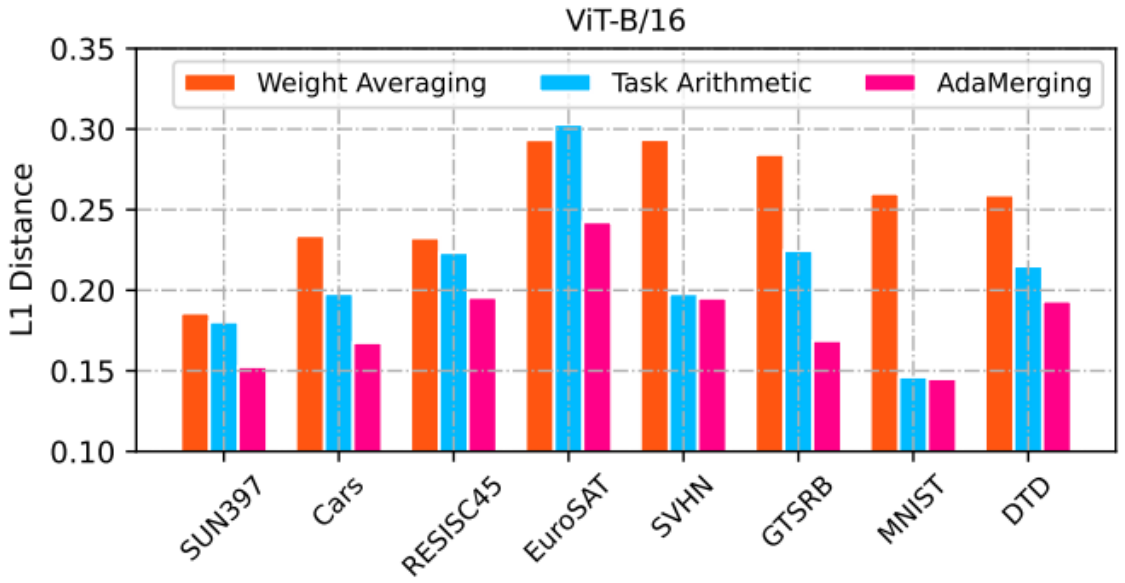
Visualization of the representations distribution of the merged model (red) and representations extracted by the individual model (blue). There is a ***clear distribution discrepancy***

E. Yang, et al., “Representation Surgery for Multi-Task Model Merging”, ICML 2024.

Our Representation Surgery

Question: Is the difference of representation distribution the cause of poor model merging performance?





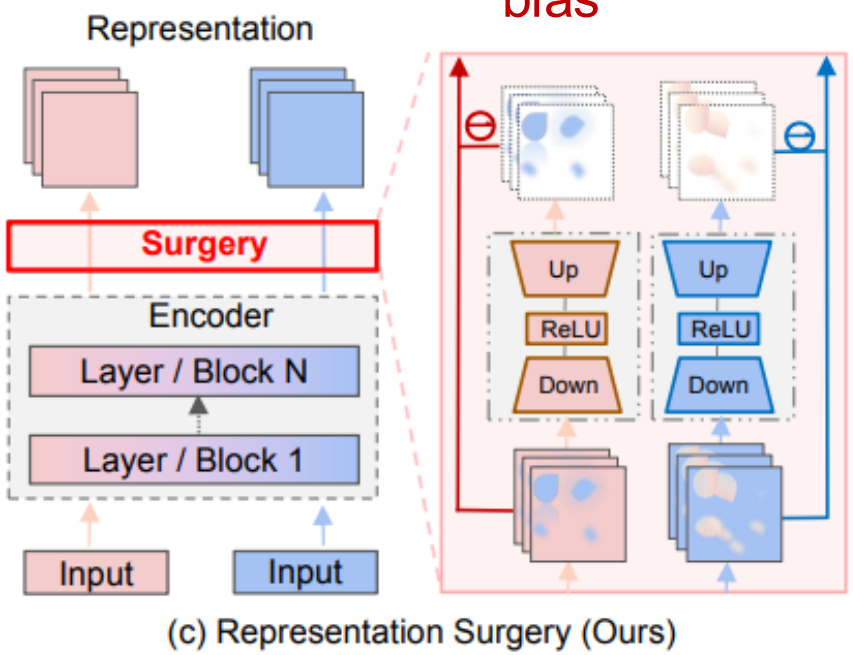
We measured the *L1 distance* between the representation of the merged model and the representation of the individual model on each task.

- L1 **Distance**: Weight Average **>** Task Arithmetic **>** AdaMerging
- MTL **Performance**: Weight Average **<** Task Arithmetic **<** AdaMerging

E. Yang, et al., "Representation Surgery for Multi-Task Model Merging", ICML 2024.

Our Representation Surgery

Surgery Scheme



(c) Representation Surgery (Ours)

Optimization Objective

$$\arg\min_{\{\theta_{\Phi_1},\theta_{\Phi_2},\ldots,\theta_{\Phi_T}\}} \sum_{t=1}^{T} \frac{1}{|D_{te}^t|} \|\hat{\mathcal{Z}}_t^{mtl} - \mathcal{Z}_t^{ind}\|_1$$

$$\text{s.t. } \hat{\mathcal{Z}}_t^{mtl} = \mathcal{Z}_t^{mtl} - \Phi_t(\mathcal{Z}_t^{mtl}),$$

representation bias

$$\Phi_t(\mathcal{Z}_t^{mtl}) = \mathbf{W}_{up} \cdot \text{ReLU}(\mathbf{W}_{down} \cdot \mathcal{Z}_t^{mtl\top})$$

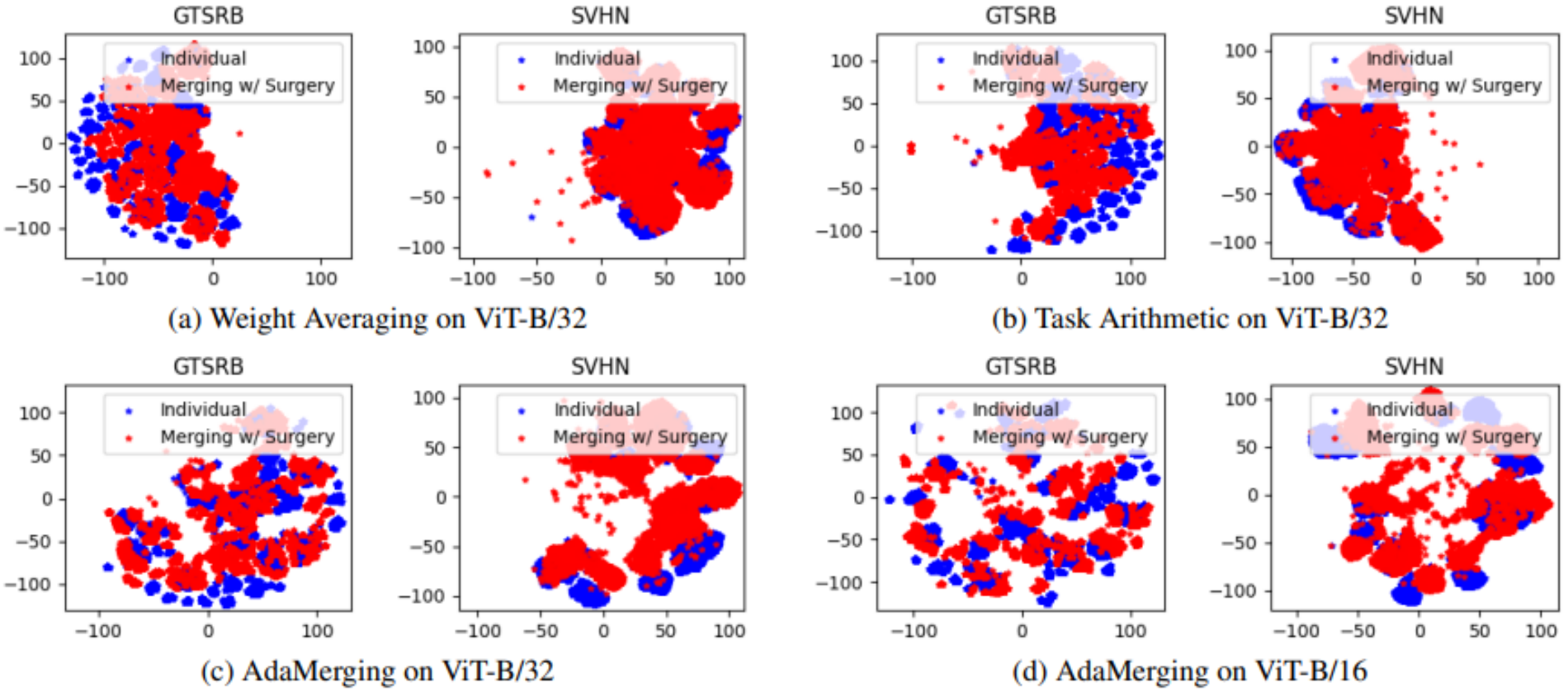
$\mathcal{Z}_t^{mtl}$ The representation of the merged model.

$\mathcal{Z}_t^{ind}$ The representation of the individual model.

"Representation surgery" a lightweight module solve the **representation bias** problem.

E Yang, et al., "Representation Surgery for Multi-Task Model Merging." ICML 2024.

Representation Surgery Experiments



(a) Weight Averaging on ViT-B/32

(b) Task Arithmetic on ViT-B/32

(c) AdaMerging on ViT-B/32

(d) AdaMerging on ViT-B/16

Visualization of the feature distribution of merged model **after representation surgery** (red) and features distribution of the individual model (blue). The distributions of the two are ***relatively close***.

Representation Surgery Experiments

Table 1. Multi-task performance when merging ViT-B/32 models on eight tasks.

Method	SUN397	Cars	RESISC45	EuroSAT	SVHN	GTSRB	MNIST	DTD	**Avg.**
Pretrained	62.3	59.7	60.7	45.5	31.4	32.6	48.5	43.8	48.0
Individual	75.3	77.7	96.1	99.7	97.5	98.7	99.7	79.4	90.5
Traditional MTL	73.9	74.4	93.9	98.2	95.8	98.9	99.5	77.9	88.9
Weight Averaging	65.3	63.4	71.4	71.7	64.2	52.8	87.5	50.1	65.8
Fisher Merging (Matena & Raffel, 2022)	68.6	69.2	70.7	66.4	72.9	51.1	87.9	59.9	68.3
RegMean (Jin et al., 2023)	65.3	63.5	75.6	78.6	78.1	67.4	93.7	52.0	71.8
Task Arithmetic (Ilharco et al., 2023)	55.2	54.9	66.7	78.9	80.2	69.7	97.3	50.4	69.1
Ties-Merging (Yadav et al., 2023)	65.0	64.4	74.8	77.4	81.2	69.3	96.5	54.5	72.9
Concrete TA (Tang et al., 2023)	62.5	61.1	76.0	95.7	91.0	81.9	98.5	51.9	77.3
Concrete AM (Tang et al., 2023)	67.8	70.0	87.5	96.0	91.6	96.7	98.7	63.8	84.0
TW AdaMerging (Yang et al., 2024)	58.0	53.2	68.8	85.7	81.1	84.4	92.4	44.8	71.1
AdaMerging (Yang et al., 2024)	64.5	68.1	79.2	93.8	87.0	91.9	97.5	59.1	80.1
Weight Averaging+Surgery (Ours)	67.6	64.6	85.8	96.8	76.9	82.9	97.8	67.3	80.0
Task Arithmetic+Surgery (Ours)	63.8	59.9	83.3	97.9	87.0	87.0	98.6	69.4	80.9
Ties-Merging+Surgery (Ours)	69.8	66.1	87.3	97.5	86.7	87.6	98.5	71.6	83.1
TW AdaMerging+Surgery (Ours)	63.9	57.6	84.2	98.2	87.6	92.7	98.0	66.8	81.1
AdaMerging+Surgery (Ours)	69.8	71.0	88.9	98.1	91.7	96.5	98.8	73.6	86.1
AdaMerging+Surgery† (Ours)	71.2	72.0	92.3	99.0	92.2	97.9	99.0	76.1	87.5

† means that the rank (i.e., Eq. 3) of the surgery module is set to 64, and the other defaults are 16.

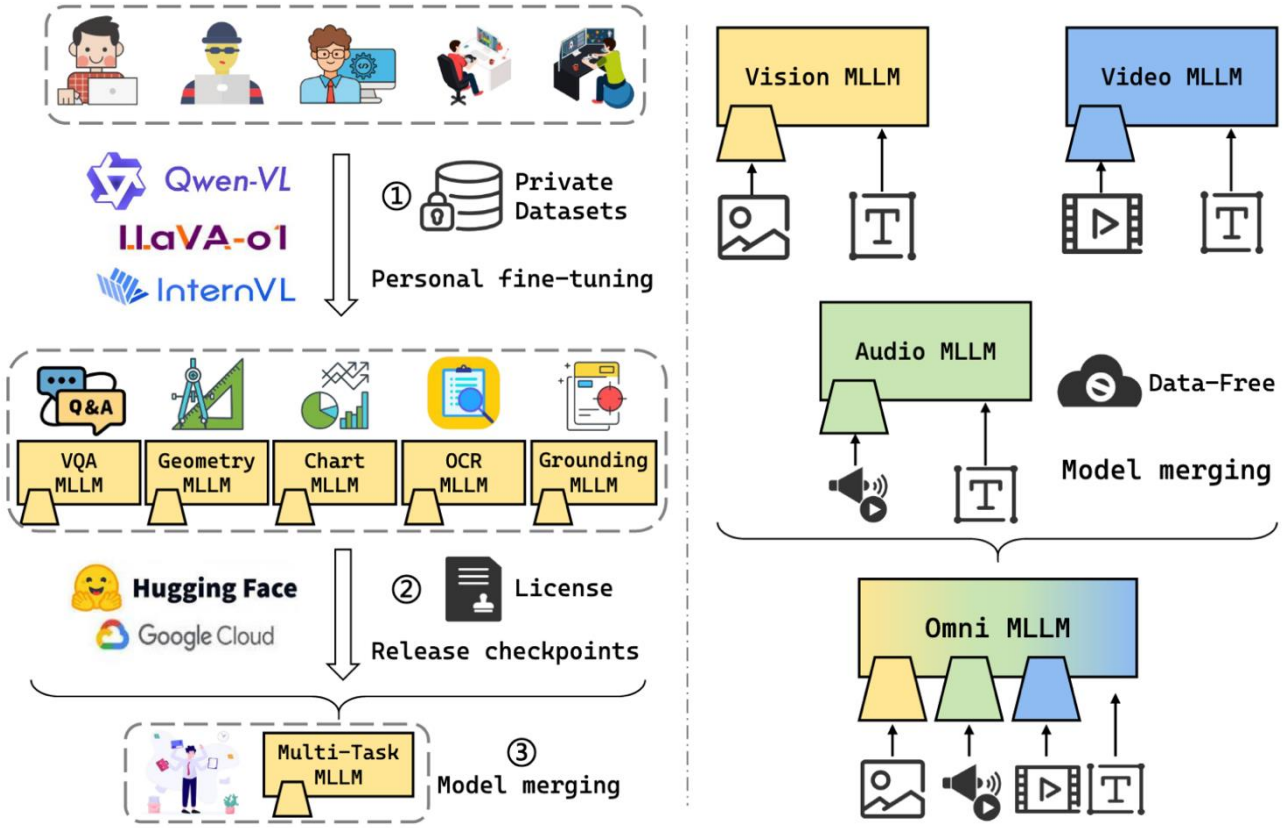
Multi-Modal Model Merging

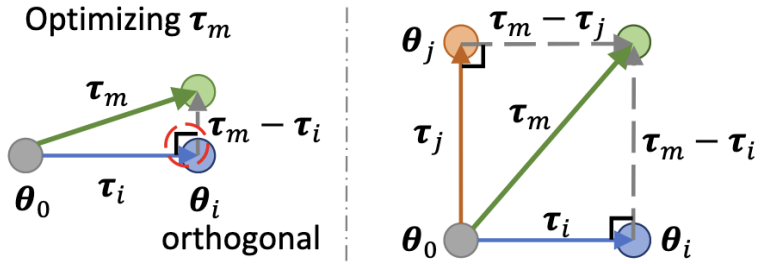
Our WUDI-V2

Merging multimodal large language models

$$\min_{\tau_{m,l}} \sum_i \|(\tau_{m,l} - \tau_{i,l})x_{i,l}\|^2$$

$$\mathcal{L}_l = \sum_{i=1}^{N} \frac{\left\|(\tau_{m,l} - \tau_{i,l})\tau_{i,l}^\top\right\|_F^2}{\|\tau_{i,l}\|_F^2}$$





For full fine-tuning:

$$\text{SVD}(\tau_{i,l} - \bar{\tau}_l) = U\Sigma V^\top, \text{ where } U \in \mathbb{R}^{m\times r}, \Sigma \in \mathbb{R}^{r\times r}, V \in \mathbb{R}^{n\times r}.$$

Optimize objective:

$$\min_{\tau_{m,l}} \mathcal{L}_l = \sum_{i=1}^{n} \frac{1}{\|\tau_{i,l}\|_F^2} \left\|(\tau_{m,l} - U_{1:k}\Sigma_{1:k}V_{1:k}^\top - \bar{\tau}_l)(\Sigma_{1:k}V_{1:k}^\top)^\top\right\|_F^2$$

Yongxian Wei, et al., "Unifying Multimodal Large Language Model Capabilities and Modalities via Model Merging." ICLR 2026

WUDI-V2 Experiments

Summary of task-specific datasets

Task Category	Size	Datasets (Language)
VQA	588K	GQA (en) [34], VQAv2 (en) [29], OKVQA (en) [57], LLaVA-Instruct (zh) [48], CogVLM-Singleround (en&zh) [78], CogVLM-Multiround (en&zh) [78]
Geometry	190K	GeoQA+ (zh) [6], G-LLaVA (en) [23]
Chart	218K	ChartQA (en) [58], DVQA (en) [37]
OCR	238K	OCRVQA (en) [62], TextCaps (en) [67], SynthDoG (en) [40], LLaVAR (en) [96], ST-VQA (en) [5], TextVQA (en) [68], DocVQA (en) [59], DeepForm (en) [72], KLC (en) [69], TabFact (en) [10]
Grounding	135K	RefCOCO (en) [55, 92], VG (en) [41]

Merging results on Intern VL2.5 (full fine-tuning)

Methods	VQA		Geometry		Chart	OCR		Grounding			Avg.
	VizWiz	GQA (test)	MathVista (mini)	MATH-Vision (mini)	ChartQA (test)	TextVQA (val)	OCRVQA (test)	RefCOCO	RefCOCO+	RefCOCOg	
InternVL2.5-Instruct	29.15	54.62	46.80	18.42	69.48	72.51	41.08	71.69	65.41	67.40	53.66
Individual VQA	30.58	60.91	35.50	17.11	48.76	63.68	36.04	-	-	-	41.80
Individual Geometry	13.45	32.80	55.20	25.00	51.76	56.91	35.35	24.73	19.61	23.84	33.86
Individual Chart	20.16	40.39	23.84	10.53	69.52	54.36	34.83	-	-	-	36.23
Individual OCR	12.40	22.22	23.31	10.53	36.88	73.00	54.79	73.65	68.01	69.10	44.39
Individual Grounding	19.09	25.88	28.91	14.47	41.32	58.39	74.87	76.67	71.35	70.09	48.10
Weight Average [82]	29.96	54.89	49.60	18.42	71.64	74.54	41.86	52.62	45.29	52.39	49.12
Task Arithmetic [35]	30.67	56.34	45.36	21.05	72.88	76.26	43.39	74.90	68.15	72.75	56.18
TIES Merging [84]	30.63	56.48	44.50	23.68	72.28	76.29	44.01	76.01	68.45	73.65	56.70
TA w/ DARE [93]	30.61	56.48	48.45	21.05	**73.08**	**76.30**	43.03	74.94	68.07	73.02	56.50
TIES w/ DARE [93]	30.65	56.11	43.85	27.63	72.72	76.19	43.33	75.10	68.48	73.55	56.76
TSV Merging [24]	**31.15**	56.67	52.45	**28.95**	70.56	75.66	45.38	65.19	58.51	59.17	54.37
Iso-C [56]	28.21	55.36	48.96	21.05	70.56	69.34	**46.51**	72.72	66.56	68.50	54.78
WUDI Merging [14]	31.02	56.96	53.03	17.11	69.19	75.95	46.12	**76.06**	**70.14**	**74.48**	57.00
WUDI v2 (Ours)	30.97	**57.13**	**54.48**	21.05	68.72	76.01	46.35	75.97	69.72	73.94	**57.44**
Mixture Training	29.79	61.33	52.83	23.68	70.32	72.96	60.25	72.06	65.93	67.46	57.66

Yongxian Wei, et al., "Unifying Multimodal Large Language Model Capabilities and Modalities via Model Merging." ICLR 2026

WUDI-V2 Experiments

Merging results on Intern Qwen2-VL (LoRA fine-tuning)

Methods	VQA		Geometry		Chart	OCR		Grounding			Avg.
	VizWiz	GQA (test)	MathVista (mini)	MATH-Vision (mini)	ChartQA (test)	TextVQA (val)	OCRVQA (test)	RefCOCO	RefCOCO+	RefCOCOg	
Qwen2-VL-Base	5.52	5.39	47.85	23.68	0.36	20.22	1.07	45.32	37.55	31.26	21.82
Individual VQA	41.38	62.60	33.71	28.94	66.56	80.21	55.33	39.31	32.71	38.01	47.88
Individual Geometry	35.57	44.63	42.50	28.95	14.56	73.95	45.96	5.57	2.31	3.90	29.79
Individual Chart	38.58	24.24	49.28	32.89	61.08	79.75	63.67	46.28	36.67	34.06	46.65
Individual OCR	28.38	37.53	31.81	13.16	57.40	70.50	64.68	0.59	0.46	0.26	30.48
Individual Grounding	38.60	32.92	36.17	19.74	18.08	75.05	48.27	72.14	65.33	66.48	47.28
Weight Average [82]	41.47	57.33	50.21	34.21	59.56	81.09	57.85	80.72	65.37	77.68	60.55
Task Arithmetic [35]	40.52	62.31	40.36	26.31	79.67	81.09	59.50	75.96	61.33	75.85	60.29
TIES Merging [84]	41.38	59.08	46.87	34.21	67.24	81.42	58.53	80.63	65.36	77.65	61.24
TA w/ DARE [93]	40.64	**62.38**	40.67	26.31	**79.76**	81.04	59.34	75.83	61.41	75.80	60.32
TIES w/ DARE [93]	**41.63**	59.96	45.72	35.53	70.68	81.53	59.63	80.73	65.65	77.77	61.88
TSV Merging [24]	41.43	57.31	**51.05**	34.21	59.44	81.25	57.81	80.71	65.34	77.76	60.63
Iso-C [56]	12.31	13.44	39.96	27.63	2.80	30.05	6.12	53.68	38.96	41.90	26.69
WUDI Merging [14]	37.19	56.45	42.96	27.63	67.84	79.92	**65.56**	76.25	60.72	71.99	58.65
WUDI v2 (Ours)	41.61	61.16	48.66	**40.79**	74.08	**81.54**	60.06	**80.92**	**65.90**	**78.24**	**63.30**
Qwen2-VL-Instruct	44.09	62.18	46.02	19.73	70.04	78.38	65.42	82.89	77.87	75.63	62.23

Merging results on checkpoints from HuggingFace

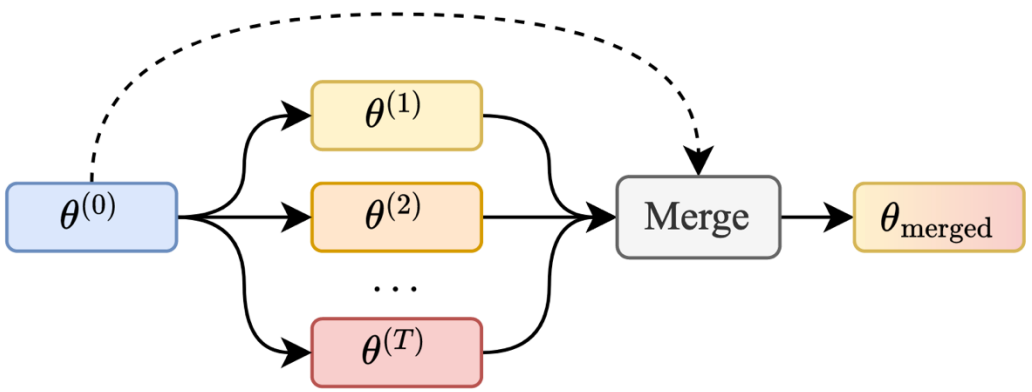
Methods	VQA		Geometry		Chart	OCR		Grounding			Avg.
	VizWiz	GQA (test)	MathVista (mini)	MATH-Vision (mini)	ChartQA (test)	TextVQA (val)	OCRVQA (test)	RefCOCO	RefCOCO+	RefCOCOg	
Qwen2-VL-7B-GRPO-8k	44.13	62.04	46.74	22.37	69.20	78.58	68.85	84.13	79.12	76.54	63.17
Qwen2-VL-7B-Pokemon	42.51	60.96	43.69	19.74	63.20	76.75	67.64	70.11	68.80	68.64	58.20
olmOCR-7B-0225-preview	43.76	61.48	38.91	18.42	67.48	77.24	68.29	75.17	71.55	69.64	59.19
EraX-VL-7B-V1.0	36.09	54.36	38.58	25.00	56.00	70.70	65.59	41.89	40.99	43.26	47.25
Weight Average [82]	43.74	61.30	43.96	26.63	76.40	82.11	71.61	89.54	82.18	86.43	66.39
Task Arithmetic [35]	41.57	60.95	42.99	23.68	75.28	81.95	71.78	87.72	81.60	85.63	65.32
TIES Merging [84]	44.17	60.54	42.52	**27.95**	75.48	82.40	71.09	**90.06**	**83.52**	86.44	66.42
TA w/ DARE [93]	43.33	61.15	44.37	26.95	76.48	82.93	**72.00**	88.93	82.79	86.07	66.50
TIES w/ DARE [93]	**44.37**	60.78	44.37	27.63	76.04	82.61	70.93	89.40	82.93	**86.77**	66.58
TSV Merging [24]	43.73	**61.40**	43.54	27.94	76.44	83.15	71.65	88.53	82.25	86.41	66.50
Iso-C [56]	43.99	61.34	40.91	22.37	**76.96**	**83.33**	71.55	87.74	82.10	85.27	65.56
WUDI Merging [14]	41.39	60.11	44.20	21.05	74.36	80.78	71.12	87.96	81.50	85.48	64.80
WUDI v2 (Ours)	43.76	61.29	**44.68**	27.63	76.24	82.97	71.48	89.56	82.97	86.42	**66.70**
Qwen2-VL-Instruct	44.09	62.18	46.02	19.73	70.04	78.38	65.42	82.89	77.87	75.63	62.23

Yongxian Wei, et al., "Unifying Multimodal Large Language Model Capabilities and Modalities via Model Merging." ICLR 2026

Continual Model Merging

Continual Model Merging

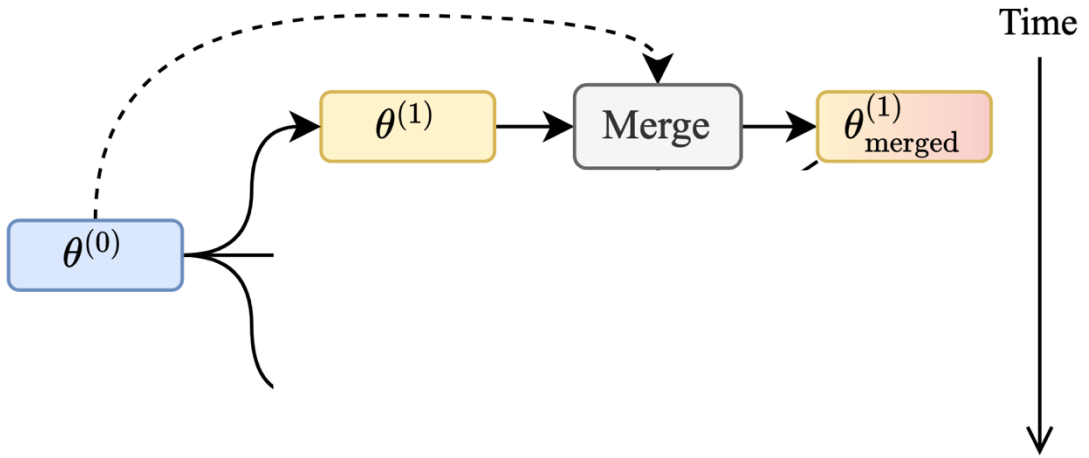
Conventional model merging vs. continual model merging



$$\theta_{\text{merged}} = \text{Merge}(\theta^{(0)}; \theta^{(1)}, \theta^{(2)}, \ldots, \theta^{(T)})$$

All of the models are ready at the same time; you merge once and you are done.

- The goal is simply to make one good average model.



$$\theta^{(t)}_{\text{merged}} = \text{ContinualMerge}(\theta^{(t-1)}_{\text{merged}}; \theta^{(0)}, \theta^{(t)}),\ t \geq 2,$$

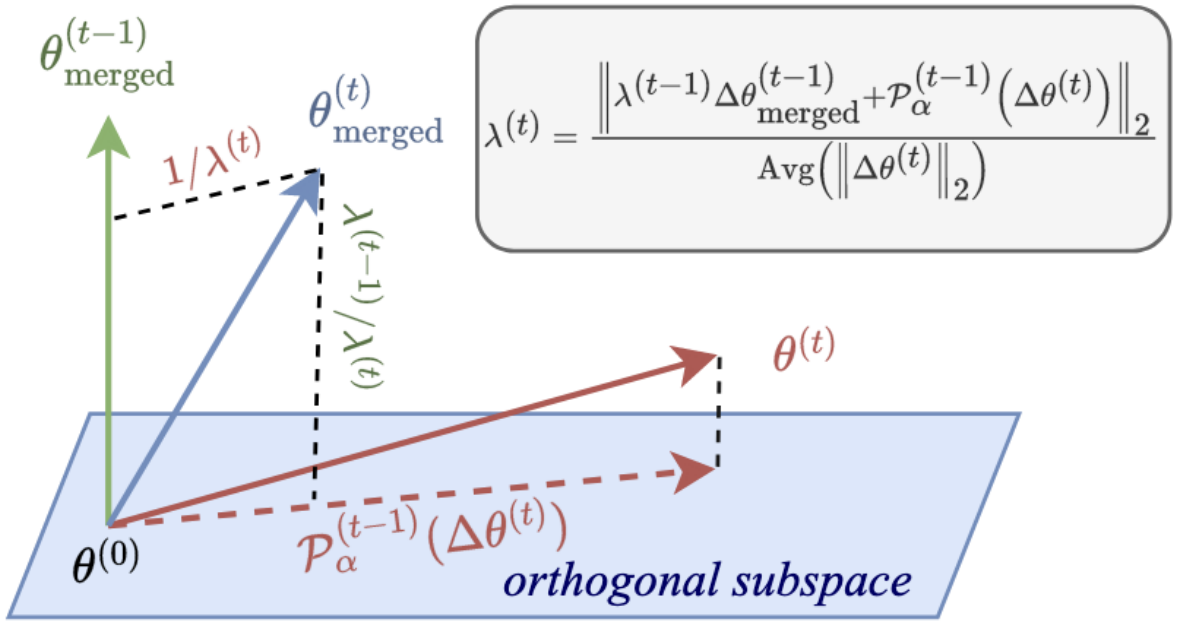
Models arrive one-by-one over the life-time of the system; you have to merge every time a new one shows up.

- The goal is to add the new capability without forgetting the old ones (catastrophic-forgetting avoidance is now part of the merge objective)
- Better memory and storage efficiency.

A.Tang, et al. "Merging Models on the Fly Without Retraining: A Sequential Approach to Scalable Continual Model Merging" NeurIPS 2025.

Our OPCM

Orthogonal **P**rojection-based **C**ontinual **M**erging (OPCM)



Update rule of OPCM:

$$\theta^{(t)}_{merged} = \theta^{(0)} + \frac{\lambda^{(t-1)}\Delta\theta^{(t-1)}_{merged} + \mathcal{P}^{(t-1)}_{\alpha}(\Delta\theta^{(t)})}{\lambda^{(t)}}$$

Projection mapping:

$$\mathcal{P}^{(t-1)}_{\alpha}(\Delta\theta^{(t)}) = \sum_{i,j=r_\alpha, i\neq j}^{m,n} \langle \Delta\theta^{(t)}, u_i v_j^T \rangle_F u_i v_j^T$$

Adaptive time-varying scaling factor:

$$\lambda^{(1)} = 1, \lambda^{(t)} = \left\|\lambda^{(t-1)}\Delta\theta^{(t-1)}_{\text{merged}} + \mathcal{P}^{(t-1)}_{\alpha}\left(\Delta\theta^{(t)}\right)\right\|_2 / \text{Avg}\left(\left\|\Delta\theta^{(t)}\right\|_2\right)$$

A.Tang, et al. "Merging Models on the Fly Without Retraining: A Sequential Approach to Scalable Continual Model Merging" NeurIPS 2025.

OPCM Experiments

Continual model merging experiments up to 20 tasks using CLIP-ViT models

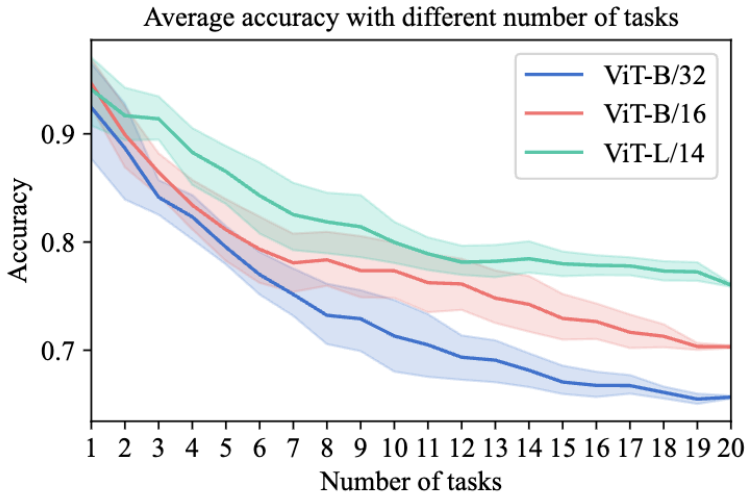
Baseline methods:

SWA $\theta_{\text{SWA}}^{(1)} = \theta^{(1)}, \quad \theta_{\text{SWA}}^{(t)} = \dfrac{\theta_{\text{SWA}}^{(t-1)}(t-1) + \theta^{(t)}}{t},$

Continual Task Arithmetic: $\theta_{\text{merged}}^{(0)} = \theta^{(0)}, \quad \theta_{\text{merged}}^{(t)} = \theta_{\text{merged}}^{(t-1)} + \lambda \left(\theta^{(t)} - \theta^{(0)}\right)$

Continual Ties-Merging: $\theta_{\text{merged}}^{(1)} = \theta^{(0)} + \lambda(\theta^{(1)} - \theta^{(0)}), \quad \theta_{\text{merged}}^{(t)} = \theta^{(0)} + \lambda \left(\tau_{\text{Ties-merged}}^{(t-1)} + \tau_{\text{Ties}}^{(t)}\right)$

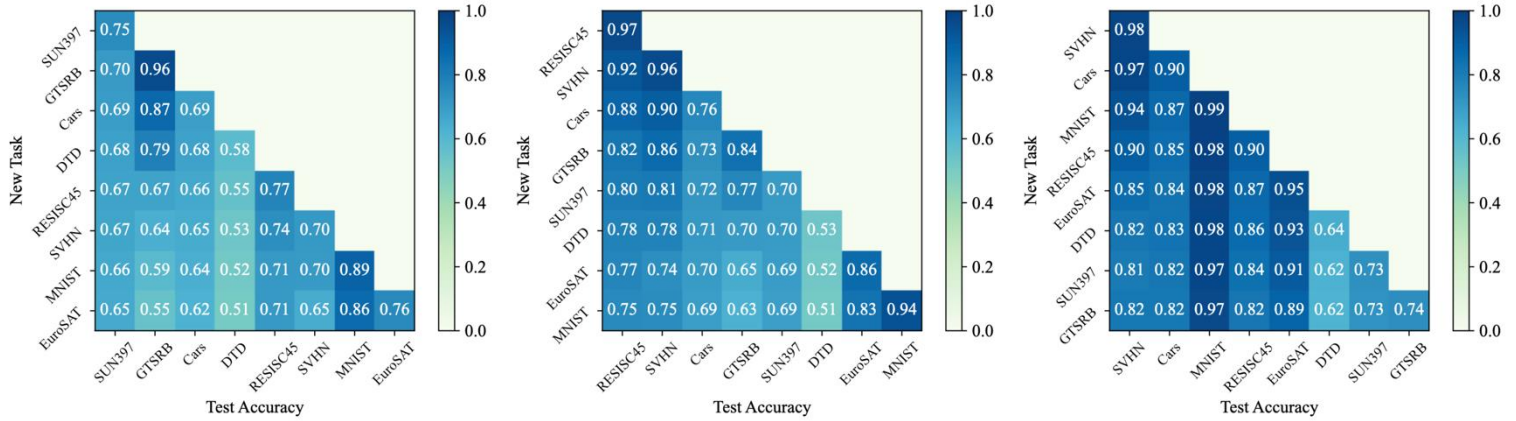
	Method	ViT-B/32			ViT-B/16			ViT-L/14		
		8 tasks	14 tasks	20 tasks	8 tasks	14 tasks	20 tasks	8 tasks	14 tasks	20 tasks
	Pre-Trained	48.1	56.9	55.6	55.4	62.0	59.8	64.9	69.1	65.6
	Fine-Tuned	90.4	89.3	89.8	92.4	91.3	91.6	94.3	93.4	93.5
	C. Fine-Tuned	79.8	67.4	62.6	82.9	72.2	68.2	90.0	70.9	77.7
ACC↑	Average (SWA)	66.3±0.0	65.4±0.0	61.1±0.0	72.3±0.0	69.7±0.0	64.8±0.0	80.0±0.0	77.5±0.0	71.1±0.0
	C. Task Arithmetic	67.5±0.0	66.5±0.0	60.6±0.0	77.1±0.0	70.9±0.0	64.2±0.0	82.1±0.0	77.9±0.0	70.3±0.0
	C. Ties-Merging	49.0±10.2	66.2±0.6	59.9±0.7	66.8±3.7	70.5±0.8	63.0±1.6	64.3±7.0	78.0±0.6	68.3±0.9
	OPCM (Ours)	**75.5**±0.5	**71.9**±0.3	**65.7**±0.2	**81.8**±0.3	**77.1**±0.5	**70.3**±0.2	**87.0**±0.4	**83.5**±0.2	**76.0**±0.2
BWT↑	Average (SWA)	-11.5±2.2	-8.0±1.3	-7.1±2.1	-9.7±1.5	-7.1±1.4	-7.3±1.7	-7.3±1.4	-5.8±1.0	-6.4±1.5
	C. Task Arithmetic	-9.6±1.5	-1.3±0.3	-3.4±0.4	**-4.2**±1.0	-1.3±0.4	-3.6±0.4	-7.1±0.8	-1.8±0.3	-3.3±0.3
	C. Ties-Merging	-15.3±8.0	**1.9**±0.6	**-1.5**±0.7	-5.5±0.4	**1.4**±0.7	**-1.5**±1.2	-13.0±5.7	**1.1**±0.4	**-2.9**±1.0
	OPCM (Ours)	**-6.3**±1.1	-6.0±1.0	-7.8±1.5	-4.8±0.7	-5.1±1.4	-6.3±2.2	**-2.6**±1.0	-4.3±0.7	-6.5±1.8



*A.Tang, et al. "*Merging Models on the Fly Without Retraining: A Sequential Approach to Scalable Continual Model Merging" NeurIPS 2025.

OPCM Experiments

Experimental details on eight image classification tasks



(a) ViT-B/32 on 8 tasks (b) ViT-B/16 on 8 tasks (c) ViT-L/14 on 8 tasks

Method	SUN397	Cars	RESISC45	EuroSAT	SVHN	GTSRB	MNIST	DTD	Avg.
Conventional Model Merging									
Fisher Merging	66.7	64.0	72.2	91.6	69.0	64.3	83.5	53.7	70.6
RegMean	67.8	68.9	82.5	94.4	90.6	79.2	97.6	63.2	80.5
Task Arithmetic	57.1	55.7	64.9	76.7	77.9	68.5	96.1	47.2	68.0
Ties-Merging	67.1	64.2	74.1	76.8	77.7	69.4	94.1	54.0	72.2
Task-wise AdaMerging	58.6	56.9	69.8	82.4	70.3	58.9	97.2	55.3	68.7
Layer-wise AdaMerging	67.9	71.3	83.5	92.7	87.4	92.9	98.2	67.0	82.6
Continual Model Merging									
C. Layer-wise AdaMerging	62.2	64.6	71.1	70.6	91.2	94.5	98.4	35.3	73.5
Ours (10 runs)	66.2±0.5	63.5±0.5	78.0±2.2	86.5±1.9	82.4±1.3	74.3±1.4	96.9±0.7	56.4±1.3	75.5±0.5

A.Tang, et al. "Merging Models on the Fly Without Retraining: A Sequential Approach to Scalable Continual Model Merging" NeurIPS 2025.

Conclusions

Pros and Cons of model fusion, limitations, and future directions

Pros

- Training-data free, data-efficiency, scalable, fast, cheap,
- even GPU-free
- Accelerating training process, fast performance recovery
- Improved performance, and robustness.

Cons & Limitations

- Suboptimal performance compared to task-specific models that have been well-trained for that particular vertical tasks.
- Gradient based methods need to load all models, resulting in high memory usage

Future Directions

- Fuse LLM, MLLM to reduce the GPU requirements.
- Task-wise heterogenous model fusion
- Trustworthy model fusion: robust FL
- Theoretical guarantees for model fusion



Reference

- Anke Tang, et al. Parameter-Efficient Multi-Task Model Fusion with Partial Linearization, ICLR, 2024.
- Enneng Yang, et al. AdaMerging: Adaptive Model Merging for Multi-Task Learning, ICLR, 2024.
- Shwai He, et al. Merging Experts into One: Improving Computational Efficiency of Mixture of Experts, EMNLP Main, 2023.
- Enneng Yang, et al. Representation Surgery for Multi-Task Model Merging. ICML, 2024.
- Anke Tang, et al. Merging Multi-Task Models via Weight-Ensembling Mixture of Experts. ICML, 2024.
- Yongxian Wei, et al. Unifying Multimodal Large Language Model Capabilities and Modalities via Model Merging. ICLR 2026.
- Peng Wang, et al. SeWA: Selective Weight Average via Probabilistic Masking. ArXiv 2025.
- Anke Tang, et al. Merging Models on the Fly Without Retraining: A Sequential Approach to Scalable Continual Model Merging. ArXiv 2025.
- Anke Tang, et al. Concrete Subspace Learning based Interference Elimination for Multi-task Model Fusion. arXiv:2312.06173 (2023).
- Hongling Zheng, et al. Learn from model beyond fine-tuning: A survey. Nature Machine Intelligence, 2025.
- Anke Tang, et al. SMILE: Zero-Shot Sparse Mixture of Low-Rank Experts Construction From Pre-Trained Foundation Models. TPAMI 2025.
- Anke Tang, et al. Data-Adaptive Weight-Ensembling for Multi-Task Model Fusion, IJCV 2025


THANK YOU