MAP: Model Merging with Amortized Pareto Fronts Using Limited Computation

Lu Li[1]*, Tianyu Zhang[2,3]*, Zhiqi Bu[4]*, Suyuchen Wang[5], Huan He[1],
Jie Fu[6], Jiang Bian[7], Yonghui Wu[7], Yong Chen[1], Yoshua Bengio[2]

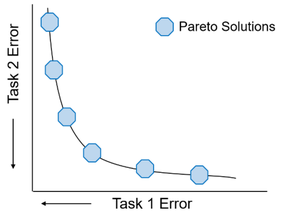
[1] University of Pennsylvania, [2] MILA, [3] ServiceNow, [4] Amazon AGI, [5] Université de Montréal, [6] HKUST, [7] University of Florida *Equal contribution

Motivation

- **Task-vector based model merging** combines multiple single-task models into a single merged model that is capable of multi-task learning, without the need for sharing training data or any further training.
- The scaling coefficient for each task determines the relative performance of the merged model on that task

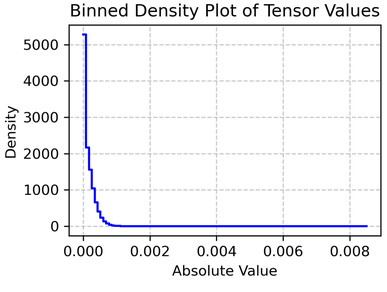
$$\boldsymbol{\theta}_m = \boldsymbol{\theta}_{pre} + \sum_i \boldsymbol{c}_i \boldsymbol{v}_i$$

- However, practitioners might have different preferences for the tasks, leading to trade-offs in how to merge the models. A set of Pareto optimal solutions is preferable.
- However, computing a Pareto front using conventional method involves many inferences and time-consuming



Key observation

- Fine-tuned models tend to converge near the pre-trained model in parameter space.

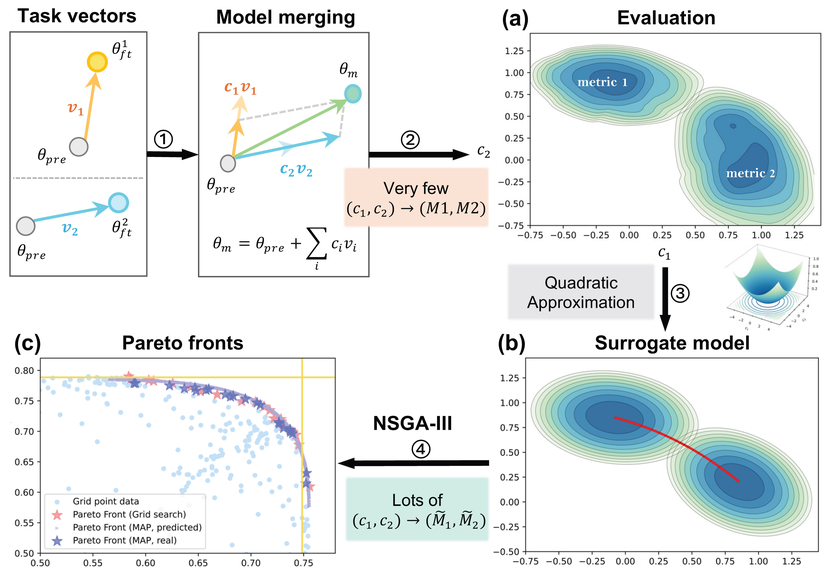


This motivates us to use second-order Taylor expansion to approximate the evaluation metrics for each task given a merged model

Metric	SUN397	Cars	DTD	SVHN
$\|\|\boldsymbol{\theta}_{pre}\|\|_1$	1,270,487	1,270,487	1,270,487	1,270,487
$\|\|\mathbf{v}_n\|\|_1$	21,055	20,127	13,621	19,349
$\|\|\mathbf{v}_n\|\|_1/\|\|\boldsymbol{\theta}_{pre}\|\|_1(\%)$	1.66%	1.58%	1.07%	1.52%

Method

- We propose **MAP**, **a computationally efficient method to identify Pareto fronts**, allowing trade-offs **without requiring additional training.**



- Let $M_i(\boldsymbol{\theta}_m(\boldsymbol{c})) = M_i(\boldsymbol{\theta}_{pre} + \sum_i \boldsymbol{c}_i \boldsymbol{v}_i)$ denote the evaluation metrics on task i
- $M_i(c) = M_i(\boldsymbol{\theta}_m(\mathbf{c}))$

$$= M_i(\boldsymbol{\theta}_{pre}) + \nabla M_i(\boldsymbol{\theta}_{pre})^\top (\boldsymbol{\theta}_m(\mathbf{c}) - \boldsymbol{\theta}_{pre}) + \frac{1}{2}(\boldsymbol{\theta}_m(\mathbf{c}) - \boldsymbol{\theta}_{pre})^\top \mathbf{H}_n(\boldsymbol{\theta}_{pre})(\boldsymbol{\theta}_m(\mathbf{c}) - \boldsymbol{\theta}_{pre}) + R_i(\boldsymbol{\theta}_m(\mathbf{c}) - \boldsymbol{\theta}_{pre})$$

2nd order Taylor expansion around $\boldsymbol{\theta}_{pre}$

$$\approx \underbrace{M_i(\boldsymbol{\theta}_{pre})}_{\in \mathbb{R}} + \underbrace{\nabla M_i(\boldsymbol{\theta}_{pre})^\top}_{\in \mathbb{R}^{1\times p}} \underbrace{\mathbf{V}}_{\in \mathbb{R}^{p\times N}} \underbrace{\mathbf{c}}_{\in \mathbb{R}^{N\times 1}} + \frac{1}{2} \underbrace{\mathbf{c}^\top}_{\in \mathbb{R}^{1\times N}} \underbrace{\mathbf{V}^\top}_{\in \mathbb{R}^{N\times p}} \underbrace{\mathbf{H}_i(\boldsymbol{\theta}_{pre})}_{\in \mathbb{R}^{p\times p}} \underbrace{\mathbf{V}}_{\in \mathbb{R}^{p\times N}} \underbrace{\mathbf{c}}_{\in \mathbb{R}^{N\times 1}}$$

$$\widetilde{M}_i(\boldsymbol{c}; \boldsymbol{A}_i, \boldsymbol{b}_i, \boldsymbol{e}_i) \equiv e_i + \boldsymbol{b}_i^\top \boldsymbol{c} + \frac{1}{2}\boldsymbol{c}^\top \boldsymbol{A}_i \boldsymbol{c}$$

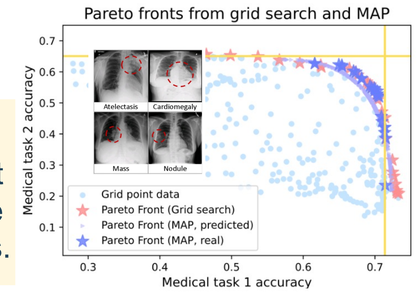
- Because of symmetry of A, the number of variables need to estimate is $\frac{(N+1)(N+2)}{2}$
- Since we already have the $\boldsymbol{c}$ and $M_i(\boldsymbol{c})$, it can be formulated as a linear regression, with

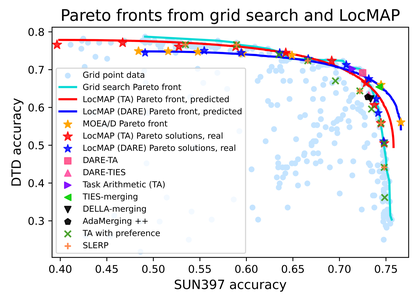
$$\mathrm{X} = \left(\boldsymbol{c}_1^2, \boldsymbol{c}_2^2, \ldots, \boldsymbol{c}_T^2, \boldsymbol{c}_1\boldsymbol{c}_2, \boldsymbol{c}_1\boldsymbol{c}_3, \ldots, \boldsymbol{c}_{T-1}\boldsymbol{c}_T, \boldsymbol{c}_1, \boldsymbol{c}_2, \ldots, \boldsymbol{c}_T, \mathbf{1}\right) \text{ and } \mathrm{y} = \boldsymbol{M}_i(\boldsymbol{\theta}_m(\boldsymbol{c}))$$

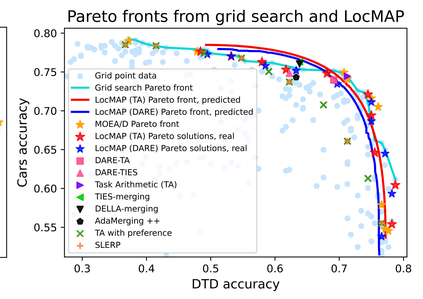
- **Closed form solution**: $(\mathrm{X}^\top\mathrm{X})^{-1}\mathrm{X}^\top\mathrm{y}$

Results

MAP is able to find diverse Pareto front, not captured by other single merged model baselines.







For higher dimensions, where we cannot visualize the Pareto front, the **win rate** is used to measure how often MAP outperforms other methods in terms of Pareto front solutions across tasks.

$$\text{Win Rate} = \frac{1}{K^2 N} \sum_{i=1}^{K} \sum_{j=1}^{K} \sum_{n=1}^{N} \mathbb{I}\left[M_n(\theta(\mathbf{c}_i^{\text{LocMAP}})) > M_n(\theta(\mathbf{c}_j^{\text{baseline}}))\right]$$

N	# c (direct search)	# c per dim	# c (MAP)	Win rate (MAP)	R^2 (MAP)
2	200	14.14	30	49.81% (±0.30)	0.953 (±0.018)
3	300	6.69	50	46.90% (±0.71)	0.980 (±0.003)

N	# c (direct search)	# c per dim	# c (MAP)	Win rate (MAP)	R^2 (MAP)
4	300	4.16	60	50.67% (±2.44)	0.984 (±0.004)
5	500	3.47	85	53.00% (±1.88)	0.941 (±0.019)
6	500	2.82	100	60.71% (±1.34)	0.941 (±0.030)
7	1000	2.68	140	63.42% (±1.91)	0.891 (±0.024)
8	1000	2.37	250	65.58% (±0.94)	0.868 (±0.028)

More Info

Paper

Code