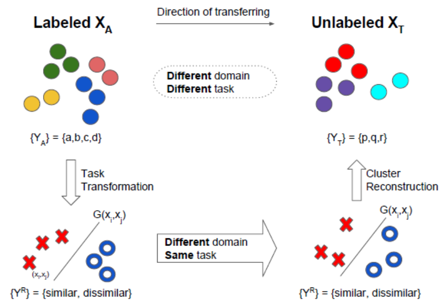
Out-of-Distribution Robustness when Finetuning Foundation Models

Zsolt Kira
Associate Professor
School of Interactive Computing
Georgia Tech

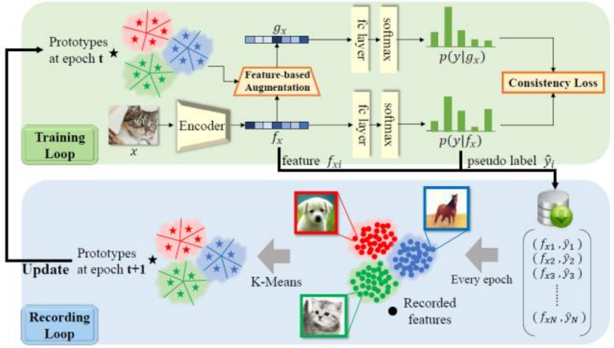
Outline

- Background & Motivation – Rise of Foundation Models
- Robust Finetuning of Foundation Models
- Generalizing to Vision-Language/Multi-modal Models
- Conclusions

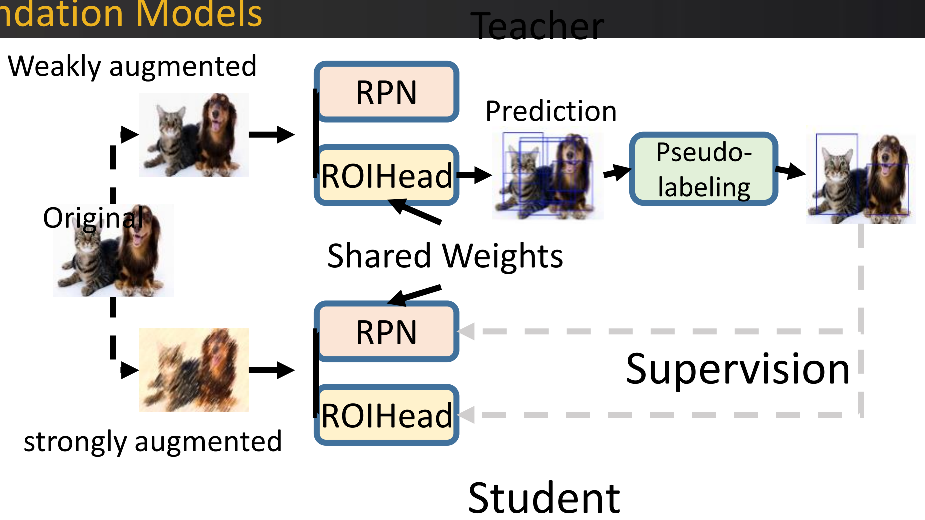
2018-2022



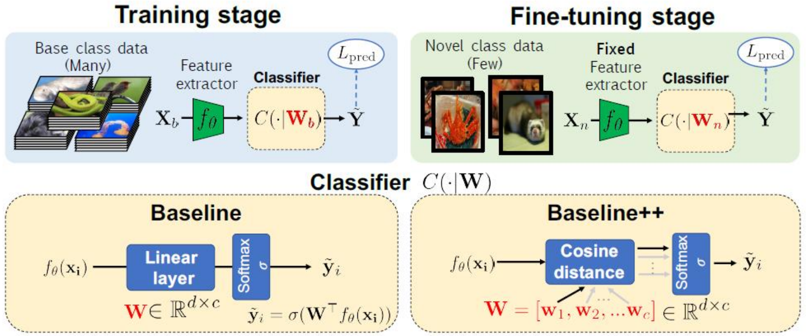
Pairwise Similarity for Cross-Task Object Discovery
[ICLRW 2016, ICLR 2018, 2019]



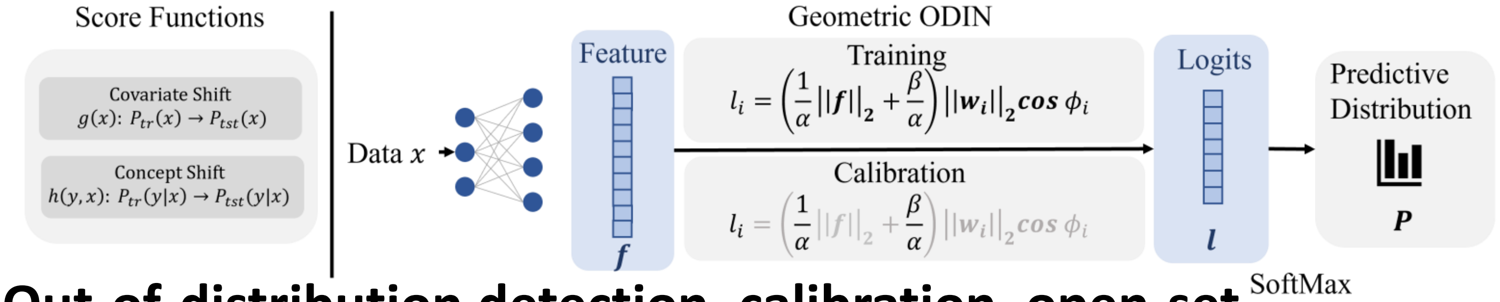
Complex Data Augmentation Domain Generalization/SSL
[ECCV 2020]



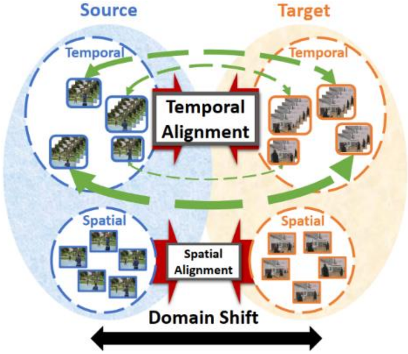
Unbiased Teacher for Anchor-based, Anchor-Free Open-Set Object Detection
[ICLR 2020, CVPR 2022, ECCV 2022, w/ Meta]



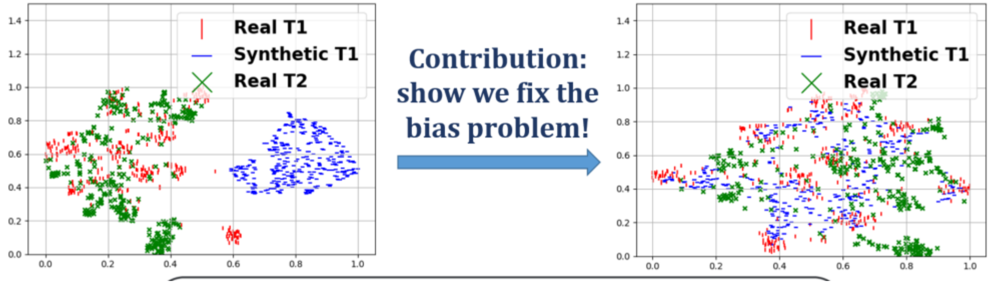
Closer Look @ Few-Shot (w/ VT)
[ICLR 2019]



Out-of-distribution detection, calibration, open-set
[CVPR 2020, NeurIPS 2021]

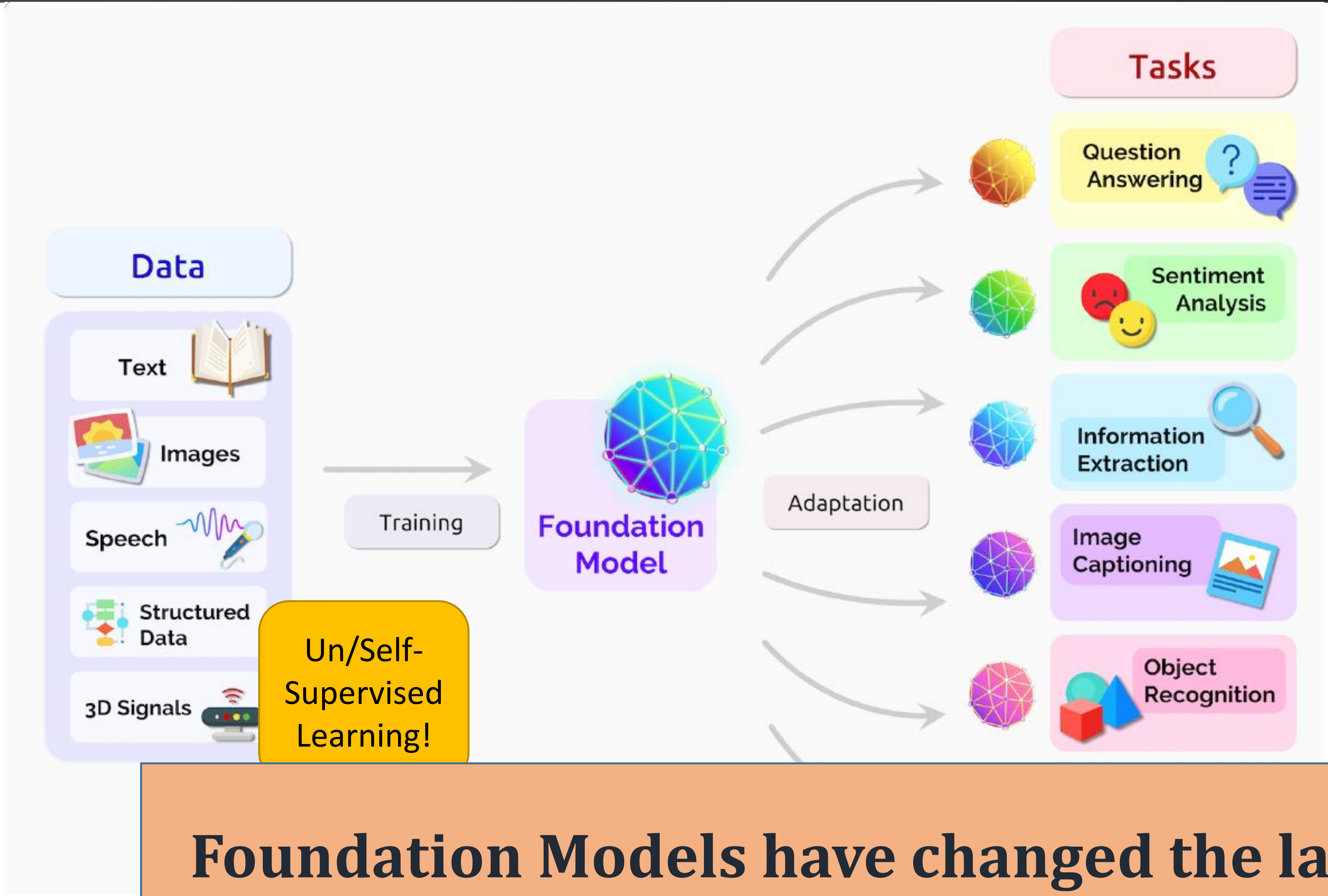


Video Domain adaptation
[CVPR 2019, ICCV 2019]



Continual Learning
[ICCV 2021, Nature 2022]

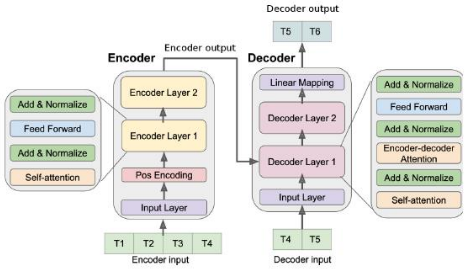
https://www.gwern.net/newsletter/2020/05



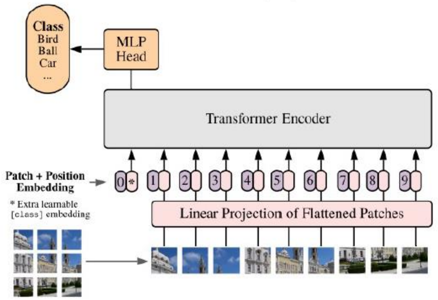
Un/Self-Supervised Learning!

Foundation Models have changed the landscape

The past ~2 years

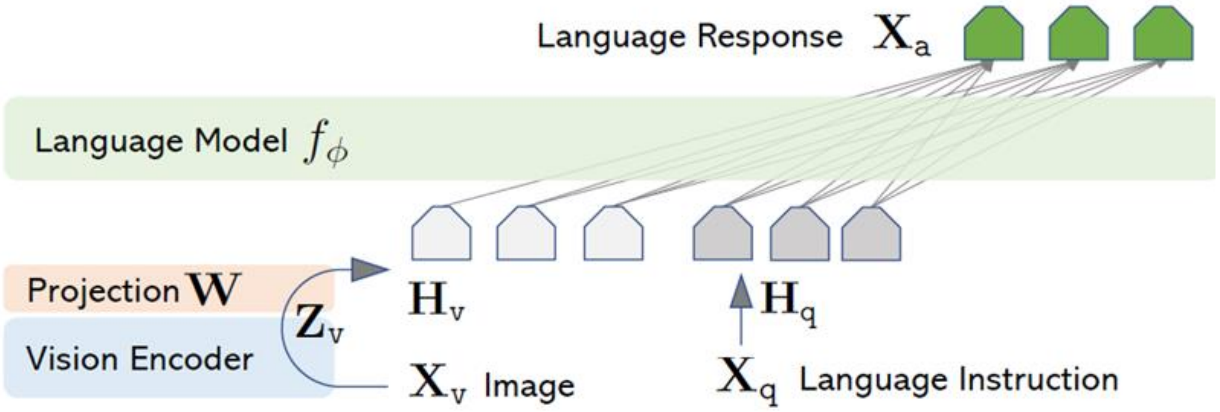


Transformer Models originally designed for NLP



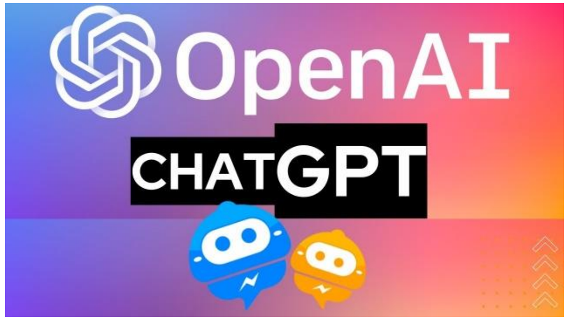
Almost identical model (Visual Transformers) can be applied to Computer Vision tasks

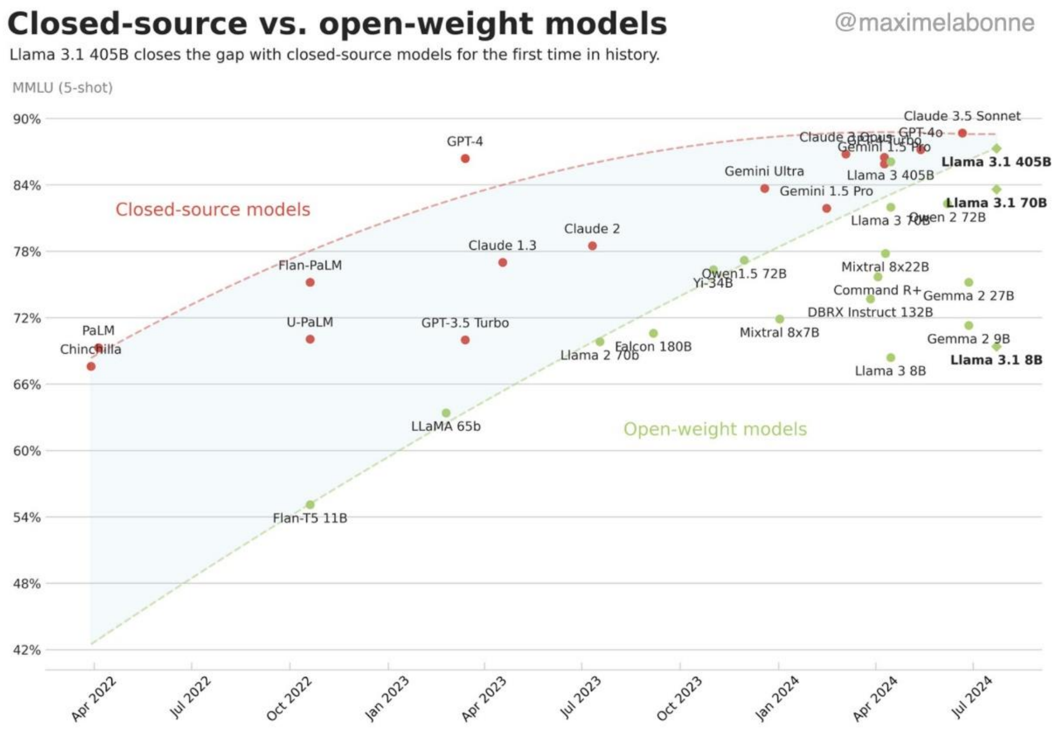
Unification with Transformers



Vision-Language Models

GitHub Copilot





https://x.com/MikelEcheve/

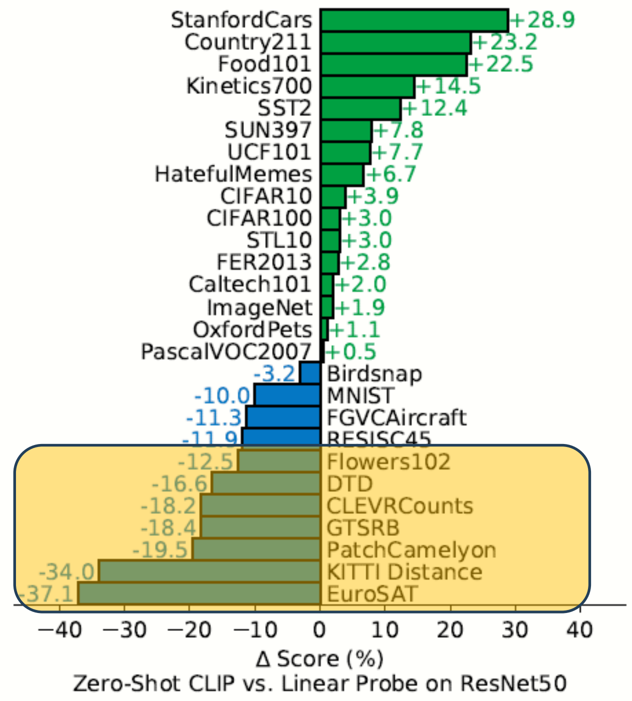
Is Generalization Solved? Are We Done?

- Positive View:
  - Bypass distribution shift!
  - Train on as much “in-distribution data” as possible
  - Nothing is OOD any more

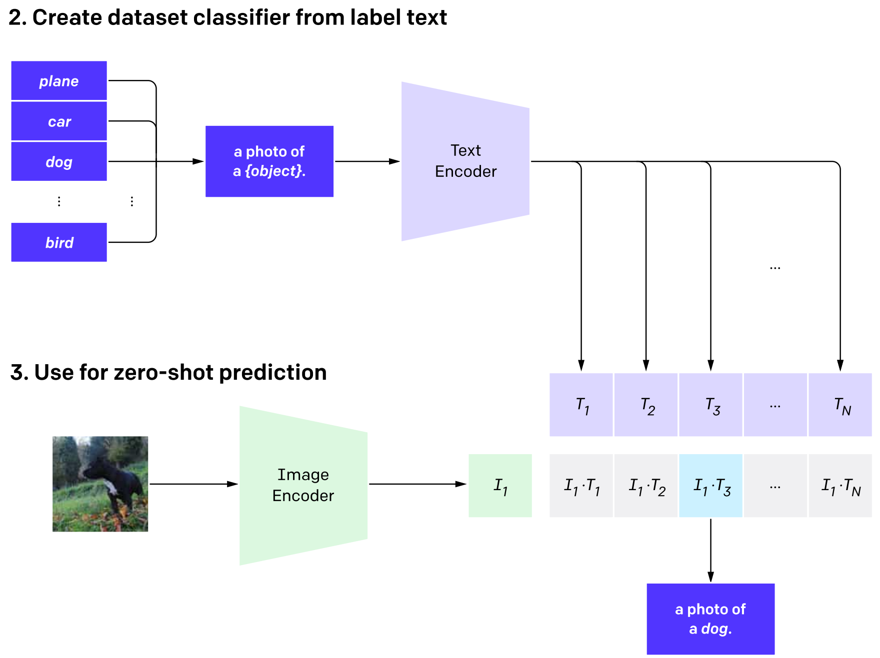
?

Is Generalization Solved? Are We Done?

- Positive View:
  - Bypass distribution shift!
  - Train on as much "in-distribution data" as possible
  - Nothing is OOD any more



[Radford et al., Learning Transferable Visual Models From Natural Language Supervision]



Is Generalization Solved? Are We Done?

- Skeptical View:
  - This is a “brute-force” approach – is it really scalable?
  - Lots of “sub-distributions” without sufficient statistical support.
    - This could be the data you care about!
  - Practically, clearly still under-performs and biased
    - US-centric, not “in-the-wild” distributions, etc.
    - How much do we need to soak up “literally all” the distributions we care about?
    - Generalist **vision** models still resist
  - **Something we might want to do:** Finetune to our data!

How to Improve Robustness?

	In-Distribution	Out-of-Distribution			
	IN	IN-V2	IN-Adversarial	IN-Rendition	IN-Sketch
CLIP Zero-Shot	67.68	61.41	**30.60**	**56.77**	**45.33**
Vanilla FT	**83.66**	**73.82**	21.40	43.06	45.22

Zero-Shot and fine-tuned classification accuracy of CLIP ViT-B on ImageNet (IN) and its variants.
The fine-tuning dataset is ImageNet.

Unconstrained optimization only encourages *fitting* to the new data

$$\min_{\boldsymbol{W}|(x,y)\in\mathcal{D}_{train}} \mathcal{L}(x, y; \boldsymbol{W})$$

Wortsman, Mitchell, et al. "Robust fine-tuning of zero-shot models." CVPR 2022.

Pre-trained Robustness

- Pre-trained models do have great generalization capability
  - Some OOD-detection and robustness capabilities

- **Question:** How do we preserve this during finetuning?

Radford, Alec, et al. "Learning transferable visual models from natural language supervision." ICML, 2021.

Preservation of Pre-trained Robustness

- L2-SP
  - Imposes L2 regularization on the difference between the fine-tuned model and the pre-trained model. $L(\theta) = \tilde{L}(\theta) + \frac{\lambda}{2}||\theta - \theta_0||_2^2$
- WiSE-FT
  - Linearly interpolate between a fine-tuned model and its pre-trained initialization.
  - Works very well for vision-language models

Hypothesis: unconstrained optimization to target leads to worse robustness.

Xuhong, L. I., Yves Grandvalet, and Franck Davoine. "Explicit inductive bias for transfer learning with convolutional networks." ICML, 2018.
Wortsman, Mitchell, et al. "Robust fine-tuning of zero-shot models." *CVPR* 2022.

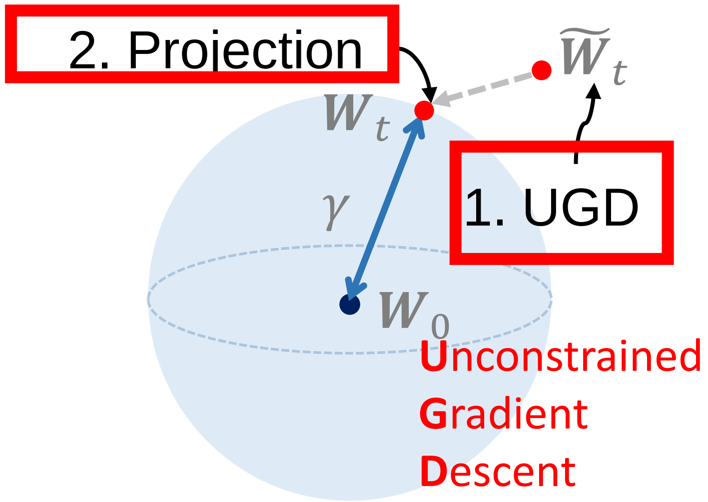
Projected Gradient Method

$$\min_{\boldsymbol{W}|(x,y)\in\mathcal{D}_{train}} \mathcal{L}(x,y;\boldsymbol{W})\ s.t.\ ||\boldsymbol{W}-\boldsymbol{W}_0|| \leq \gamma$$

- Projected Gradient Descent

$$\widetilde{\boldsymbol{W}}_t = SGD(x,y|\boldsymbol{W}_{t-1})$$

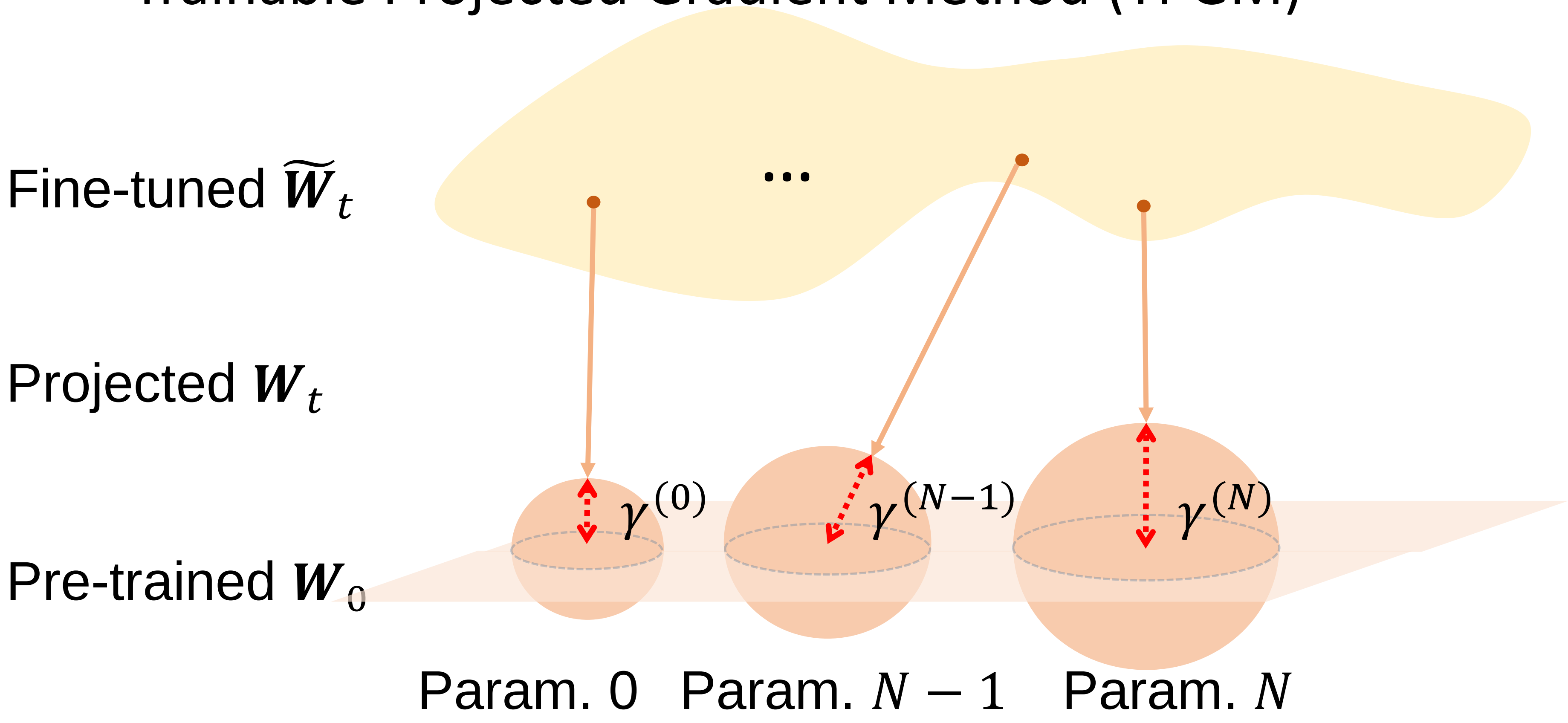
$$\boldsymbol{W}_t = \Pi(\widetilde{\boldsymbol{W}}_t, \boldsymbol{W}_0; \gamma)$$



Π defines a (**differentiable**) *projection function* and γ is the projection radius

Trainable Projected Gradient Method

- Trainable Projected Gradient Method (TPGM)

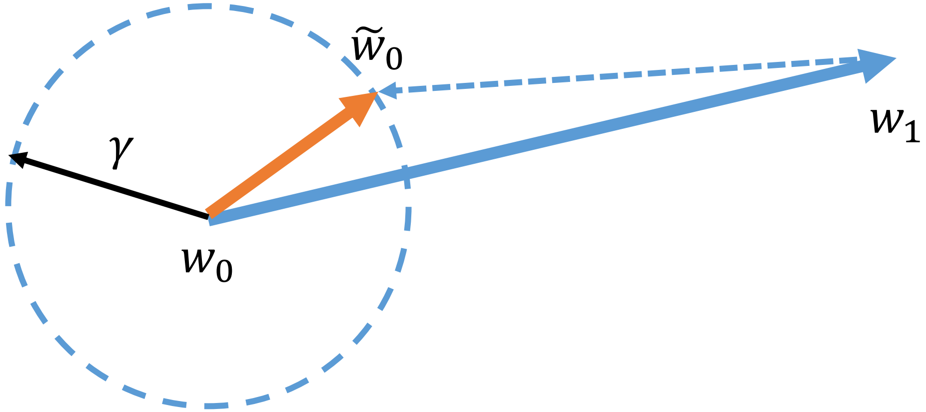


- Open Questions
  - *Which* layers to fine-tune?
  - *How much* to fine-tune?
  - Not feasible to specify a different constraint for each layer .

Tian, Junjiao, et al. "Trainable projected gradient method for robust fine-tuning." ***CVPR 2023.***

Our Prior Work: TPGM and FTP

TPGM and FTP use *outer loop bi-level optimization* for robust training



$$\underbrace{\min_{\lambda,\gamma|(x,y)\in\mathcal{D}_{val}}}_{\text{Step 2}} \underbrace{\min_{\theta|(x,y)\in\mathcal{D}_{tr}}}_{\text{Step 1}} \mathcal{L}(x,y;\theta,\lambda,\gamma) \quad \text{s.t.} \quad \underbrace{\|\theta-\theta_0\|_* \le \gamma}_{\text{Step 3}}$$

Algorithm 1: TPGM

Data: $\mathcal{D}_{tr}, \mathcal{D}_{val}$

Result: θ

Initialize $\theta_0^* = \theta_0, \gamma_0 = \epsilon$

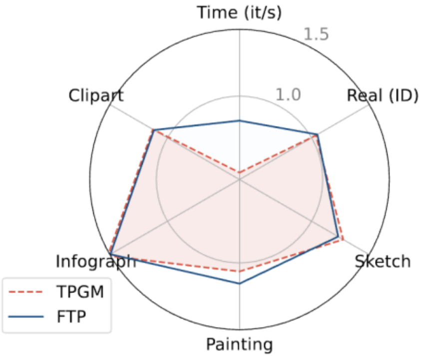
for $t = \{0, ..., T-1\}$ **do**

Step 1 $\quad \theta_{t+1} = \arg\min_\theta \mathcal{L}(x, y; \theta_t^*) \quad x, y \in \mathcal{D}_{tr}$

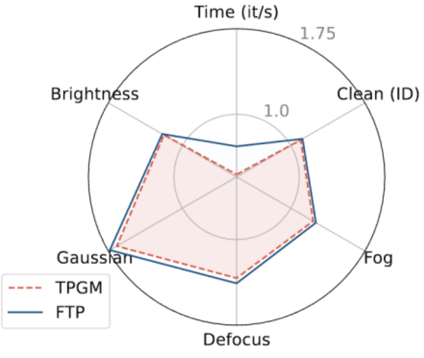
Step 2 $\quad \gamma_{t+1} = \text{ProjectTune}(\mathcal{D}_{val}, \theta_0, \theta_{t+1}, \gamma_t)$

Step 3 $\quad \theta_{t+1}^* = \Pi(\theta_0, \theta_{t+1}, \gamma_{t+1})$

$$\Pi_{l2}(\theta_0, \theta_t, \gamma) : \tilde{\theta} = \theta_0 + \frac{1}{\max\left(1, \frac{\|\theta_t - \theta_0\|_2}{\gamma}\right)}(\theta_t - \theta_0).$$



(b) Image Classification



(c) Semantic Segmentation

Junjiao Tian
Robotics Ph.D.

Tian et al., CVPR 2023 / NeurIPS 2023

Can we simplify this to reduce complexity/computation?

Selective Projection Decay

Learning the New Without Forgetting the Old Even More Efficiently

Junjiao Tian

Robotics Ph.D.

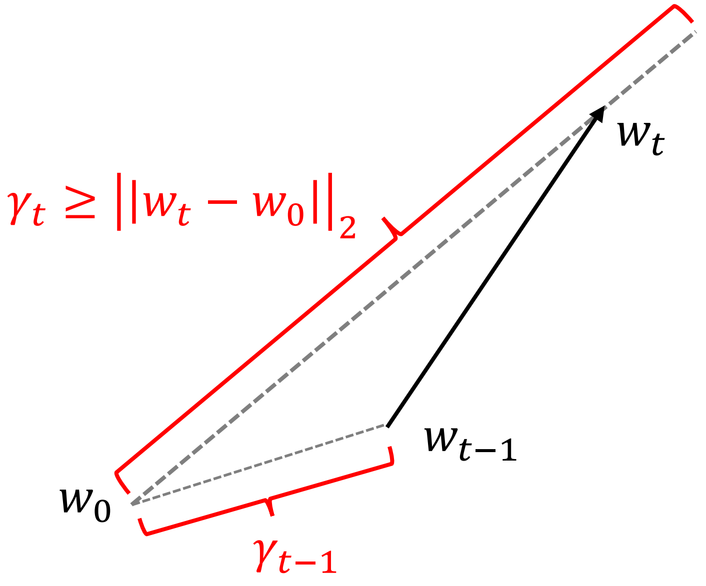
Chengyue Huang

ML Ph.D.

Tian, Junjiao, Chengyue Huang, and Zsolt Kira. "Selective Projection Decay for Robust Fine-Tuning", **NeurIPS *2024***.

Observations

- TPGM/FTP **grows** and **shrinks** the projection radius.
  - When the radius grows, it often provides no regularization (no projection).
  - The regularization effect mainly comes from the shrinkage of the projection radius.



γ: constraints
w_0: Initialization

Tian, Junjiao, Chengyue Huang, and Zsolt Kira. "Selective Projection Decay for Robust Fine-Tuning", **NeurIPS *2024***.

Hypothesis

- No need to explicitly maintain a set of projection radii.
- No need to know when to grow.

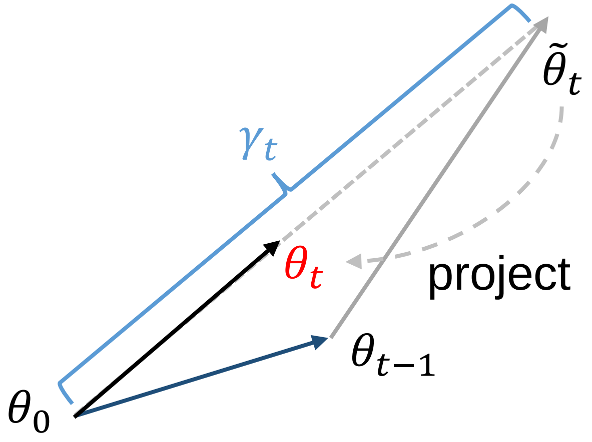
- Just need to know when to shrink/apply regularization.
  - Do this per layer/iteration
  - **When:** Alignment between gradient and direction to original weights
  - **How much:** $\gamma_t = ||w_t - w_0||_2$

Tian, Junjiao, Chengyue Huang, and Zsolt Kira. "Selective Projection Decay for Robust Fine-Tuning", **NeurIPS** ***2024***.

Selective Projection Decay (SPD)

Selecting criterion

- L2-SP: $L(\theta) = \tilde{L}(\theta) + \frac{\lambda}{2}||\theta - \theta_0||_2^2$
- Hyper-optimize λ: $\nabla\lambda = \frac{\partial f(\theta_t)}{\partial \lambda} = \frac{\partial f(\theta_t)}{\partial \theta}^T \frac{\theta_t}{\partial \lambda} = \alpha * -g_{t+1}^T(\theta_t - \theta_0)$
  - This was the gradient calculation in Fast Trainable Projection $\nabla\gamma \propto g_t^T(\theta_{t-1} - \theta_0)$
- Selection condition: $c_t = c_{t-1} - g_t^T(\theta_{t-1} - \theta_0) < 0$

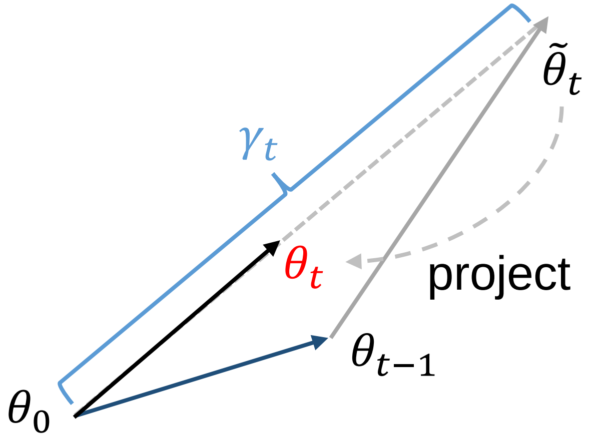


γ_t: constraints
θ_0: initialization
$\tilde{\theta}_t$: unconstrained update

Selective Projection Decay (SPD)

Selecting criterion

- L2-SP: $L(\theta) = \tilde{L}(\theta) + \frac{\lambda}{2}||\theta - \theta_0||_2^2$
- Hyper-optimize λ: $\nabla\lambda = \frac{\partial f(\theta_t)}{\partial \lambda} = \frac{\partial f(\theta_t)^T}{\partial \theta}\frac{\theta_t}{\partial \lambda} = \alpha * -g_{t+1}^T(\theta_t - \theta_0)$
- Selection condition: $c_t = c_{t-1} - g_t^T(\theta_{t-1} - \theta_0) < 0$



γ_t: constraints
θ_0: initialization
$\tilde{\theta}_t$: unconstrained update

Projection coefficient

- L2-SP is a projection: $\theta_p = \theta_t - \left(1 - \frac{\gamma}{\max\{\gamma, ||\theta_t - \theta_0||_2\}}\right) * (\theta_t - \theta_0)$
- Deviation: $\gamma_t = ||\theta_t - \theta_0||_2$
- Deviation ratio: $r_t = \frac{\max\{0, \gamma_t - \gamma_{t-1}\}}{\gamma_t}$
- $\theta_t \leftarrow \theta_t - \lambda \frac{\max\{0, \gamma_t - \gamma_{t-1}\}}{\gamma_t}(\theta_t - \theta_0)$

Selective Projection Decay

Algorithm 1: Adam with L2-Regularization

Initialize $m_0 \leftarrow 0$, $v_0 \leftarrow 0$, $t \leftarrow 0$
While θ_t not converged
$t \leftarrow t+1$
$g_t \leftarrow \nabla_\theta f_t(\theta_{t-1})$
$m_t \leftarrow \beta_1 m_{t-1} + (1-\beta_1) g_t$
$v_t \leftarrow \beta_2 v_{t-1} + (1-\beta_2) g_t^2$
Bias Correction
$\widehat{m_t} \leftarrow \frac{m_t}{1-\beta_1^t}$, $\widehat{v_t} \leftarrow \frac{v_t}{1-\beta_2^t}$
Update
$\theta_t \leftarrow \theta_{t-1} - \frac{\alpha \widehat{m_t}}{\sqrt{\widehat{v_t}}+\epsilon}$
$\theta_t \leftarrow \theta_t - \lambda \underline{\alpha} (\theta_t - \theta_0)$

Learning rate

Algorithm 2: Adam with Selective L2-Reg.

Initialize $m_0 \leftarrow 0$, $v_0 \leftarrow 0$, $t \leftarrow 0$, $c_0 \leftarrow 0$
While θ_t not converged
$t \leftarrow t+1$
$g_t \leftarrow \nabla_\theta f_t(\theta_{t-1})$
$m_t \leftarrow \beta_1 m_{t-1} + (1-\beta_1) g_t$
$v_t \leftarrow \beta_2 v_{t-1} + (1-\beta_2) g_t^2$
Bias Correction
$\widehat{m_t} \leftarrow \frac{m_t}{1-\beta_1^t}$, $\widehat{v_t} \leftarrow \frac{v_t}{1-\beta_2^t}$
Update
$\theta_t \leftarrow \theta_{t-1} - \frac{\alpha \widehat{m_t}}{\sqrt{\widehat{v_t}}+\epsilon}$
$\boxed{c_t = c_{t-1} - g_t^\top (\theta_{t-1} - \theta_0)}$
If $c_t < 0$:
$\theta_t \leftarrow \theta_t - \lambda \underline{r_t} (\theta_t - \theta_0)$

1, Condition

2, Deviation Ratio

20

Algorithm 1: Adam with L2-SP

Initialize $m_0 \leftarrow 0$, $v_0 \leftarrow 0$, $t \leftarrow 0$
While θ_t not converged
$t \leftarrow t+1$
$g_t \leftarrow \nabla_\theta f_t(\theta_{t-1})$
$m_t \leftarrow \beta_1 m_{t-1} + (1-\beta_1) g_t$
$v_t \leftarrow \beta_2 v_{t-1} + (1-\beta_2) g_t^2$
Bias Correction
$\widehat{m_t} \leftarrow \frac{m_t}{1-\beta_1^t}$, $\widehat{v_t} \leftarrow \frac{v_t}{1-\beta_2^t}$
Update
$\theta_t \leftarrow \theta_{t-1} - \frac{\alpha \widehat{m_t}}{\sqrt{\widehat{v_t}}+\epsilon}$
$\theta_t \leftarrow \theta_t - \lambda\alpha(\theta_t - \theta_0)$

Algorithm 2: Adam with SPD

Initialize $m_0 \leftarrow 0$, $v_0 \leftarrow 0$, $t \leftarrow 0$,$c_0 \leftarrow 0$
While θ_t not converged
$t \leftarrow t+1$
$g_t \leftarrow \nabla_\theta f_t(\theta_{t-1})$
$m_t \leftarrow \beta_1 m_{t-1} + (1-\beta_1) g_t$
$v_t \leftarrow \beta_2 v_{t-1} + (1-\beta_2) g_t^2$
Bias Correction
$\widehat{m_t} \leftarrow \frac{m_t}{1-\beta_1^t}$, $\widehat{v_t} \leftarrow \frac{v_t}{1-\beta_2^t}$
Update
$\theta_t \leftarrow \theta_{t-1} - \frac{\alpha \widehat{m_t}}{\sqrt{\widehat{v_t}}+\epsilon}$
$c_t = c_{t-1} - g_t^\intercal(\theta_{t-1} - \theta_0)$
If $c_t < 0$:
$\theta_t \leftarrow \theta_t - \lambda r_t(\theta_t - \theta_0)$

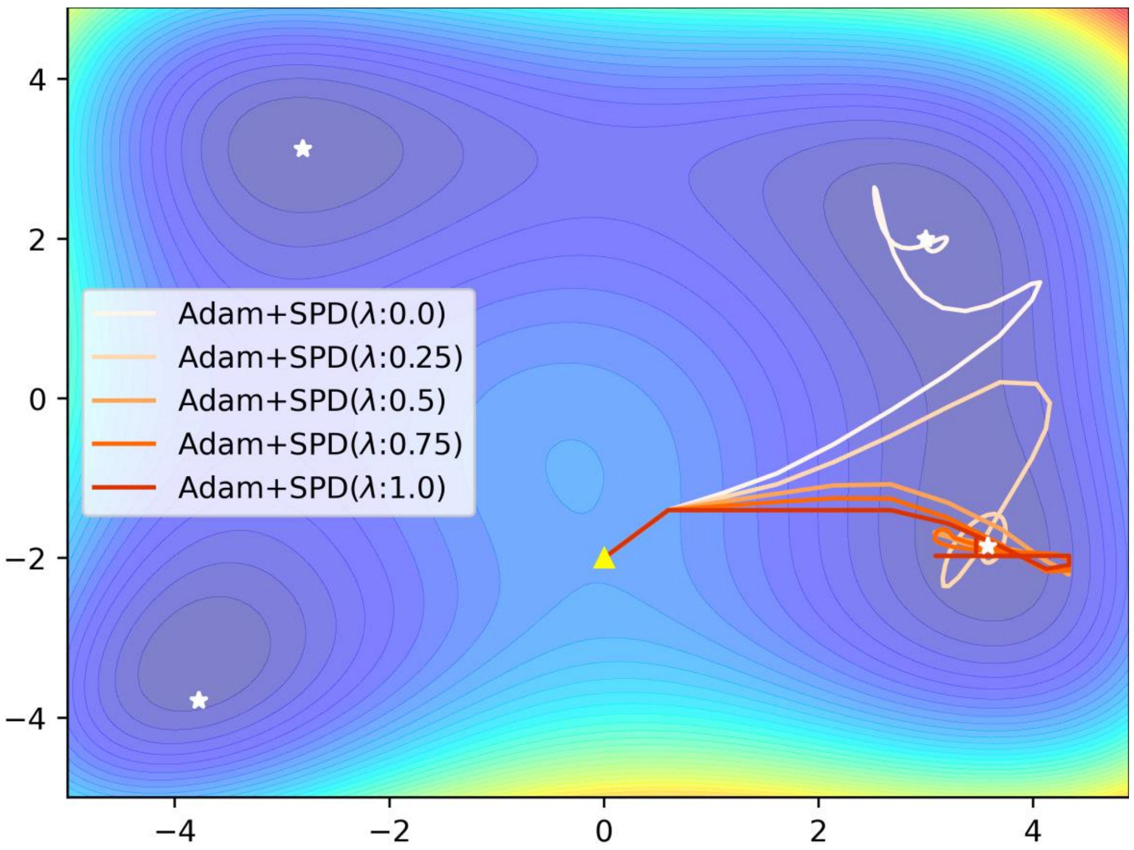
More intuitive hyper-parameter (λ) tuning

- No regularization ($\lambda = 0$): the projection radius is 1.
- Weak regularization ($1 \geq \lambda > 0$): the projection radius lies between $||\theta_t - \theta_0||_2$ and $||\theta_{t-1} - \theta_0||_2$. Within this range, layers will expand.
- Strong regularization ($\lambda > 1$): the projection radius lies between 0 and $||\theta_{t-1} - \theta_0||_2$. In this range, it's possible that regularized layers can contract.

Interpretation

- The condition measures the ***alignment*** between the current gradient direction g_t and the overall heading $(\theta_{t-1} - \theta_0)$.
  - Prioritizes consistent update directions

- Toy example
  - Adam + SPD **panelizes vertical traversal** and converges to the global minimum closer to the initialization.



Optimization on Himmelblau's function (4 identical global minima) using Adam with SPD.

Sensitivity to Hyper-parameter (λ) tuning

Hyper-Parameter λ	1e-1	1e-2	6e-3	3e-3	1e-3	6e-4	3e-4	1e-4	1e-5	1e-6	1e-7	0.0
Deviation	0.03	0.14	0.18	0.24	0.34	0.39	0.46	0.53	0.58	0.58	0.58	0.59
OOD	14.90	37.20	39.43	40.52	41.13	41.76	40.52	41.26	41.35	41.73	40.62	41.34
ID	27.25	69.74	73.76	76.62	78.90	79.30	79.30	79.84	79.80	79.95	79.80	79.91

(a) L2-SP hyper-parameter (λ) sweep. Stronger regularizations (larger values) decrease deviation; however, they do not improve OOD performance and even deteriorate ID performance.

Hyper-Parameter λ	2.1	1.9	1.7	1.5	1.3	1.1	0.9	0.7	0.5	0.3	0.1	0.0
Deviation	0.31	0.32	0.33	0.34	0.36	0.36	0.42	0.44	0.48	0.51	0.54	0.59
OOD	45.67	45.77	45.23	45.27	44.81	43.99	44.18	42.73	41.84	42.43	41.20	41.34
ID	81.21	80.76	81.25	80.67	81.11	79.89	79.57	80.00	79.92	80.26	80.00	79.91

(b) Adam-SPD hyper-parameter (λ) sweep. Stronger regularizations (larger values) decrease deviation, simultaneously improving OOD performance. The ID performance is not impacted significantly.

Comparisons between L2-SP and Adam-SPD on DomainNet

- ID dataset: {clipart}, OOD datasets: {real, sketch, quickdraw, painting}
- Selective regularization can effectively restrain model's deviation ($||\theta_t - \theta_0||_2$) and improve OOD robustness without significantly impacting ID robustness.

Experiments

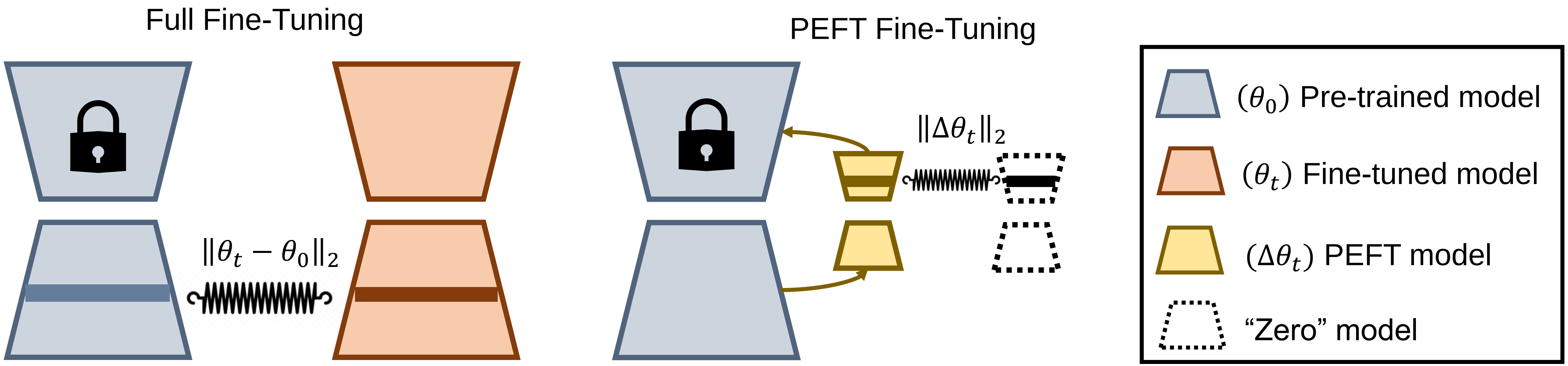
- Selective regularization is on par with predecessors and outperforms other methods.

Table 3: ImageNet Fine-Tuning Result using CLIP ViT-Base. SPD outperforms more complicated algorithms and beats L2-SP by 8.8% by selectively imposing regularization.

	ID	OOD				Statistics	
	Im	Im-V2	Im-Adversarial	Im-Rendition	Im-Sketch	OOD Avg.	Avg.
Zero-Shot	67.68	61.41	30.60	56.77	45.53	48.58	52.40
vanilla FT	83.66	73.82	21.40	43.06	45.52	46.98	54.29
Linear Prob.	78.25	67.68	**26.54**	52.57	48.26	48.76	54.66
LP-FT [19]	82.99	72.96	21.08	44.65	47.56	46.56	53.85
L2-SP [13]	83.44	73.2	20.55	43.89	46.60	46.06	53.54
FTP [11]	84.19	74.64	26.50	47.23	50.23	49.65	56.56
Adam-SPD	**84.21**	**74.83**	25.42	**49.09**	**51.18**	**50.13**	**56.95**

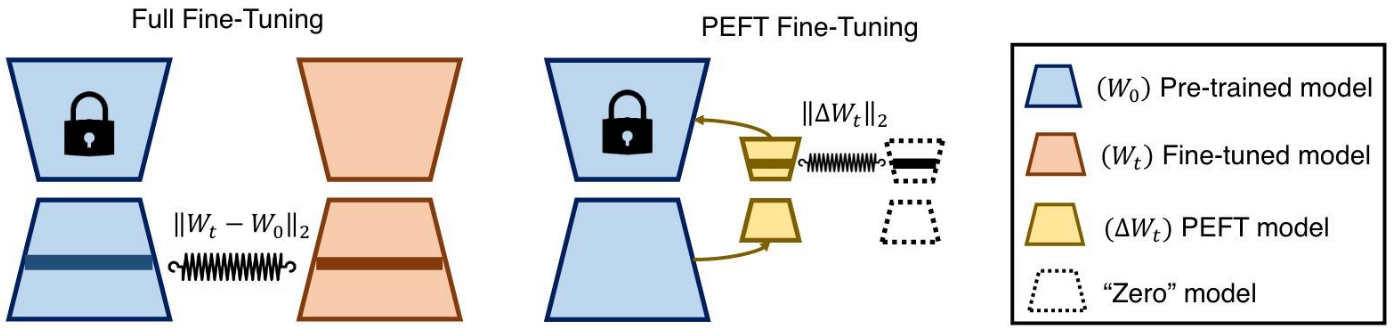
Compatible with Parameter-Efficient Fine-Tuning

- Our method reduces to selective weight decay when working with Parameter Efficient Fine-Tuning (PEFT) methods.



25

LLaMA PEFT Fine-Tuning Experiments



PEFT	LLM	Optimizer	BoolQ	PIQA	SIQA	HellaSwag	WinoGrande	ARC-e	ARC-c	OBQA	Avg.
Series	LLaMA_{7B}	AdamW	63.0	79.2	76.3	67.9	75.7	74.5	57.1	72.4	70.8
		Adam-SPD (1.0)	**68.3**	**80.4**	**77.4**	**81.6**	**79.7**	**79.4**	**63.5**	**78.4**	**76.1**
Parallel	LLaMA_{7B}	AdamW	67.9	76.4	**78.8**	69.8	78.9	73.7	57.3	75.2	72.3
		Adam-SPD (1.0)	**68.8**	**80.9**	78.3	**82.0**	**80.8**	**80.0**	**63.1**	**78.0**	**76.5**
LoRA	LLaMA_{7B}	AdamW	68.9	80.7	77.4	78.1	78.8	77.8	61.3	74.8	74.7
		Adam-SPD (0.7)	**69.1**	**82.8**	**78.9**	**84.8**	**80.7**	**80.9**	**65.8**	**79.2**	**77.8**
LoRA	LLaMA_{13B}	AdamW	72.1	83.5	80.5	80.5	**83.7**	82.8	68.3	82.4	80.5
		Adam-SPD (1.2)	**72.9**	**85.6**	**80.7**	**92.0**	**83.7**	**85.6**	**71.6**	**85.6**	**82.2**

Compatibility with PEFT methods

- SPD regularizes $||\theta_t - \theta_0||_2$ for full fine-tuning and $||\Delta\theta_t||_2$ for PEFT fine-tuning
- SPD can also improve the performance of PEFT methods (e.g. LoRA, series adapters, parallel adapters)

What about Vision-Language Models (VLMs)?

- Robustness and distribution shift is much more complicated!

- Many types of shift possible

- **Distribution Shifts to Images**
  - IV-VQA
  - CV-VQA
- **Distribution Shifts to Questions**
  - VQA-Rephrasings
  - VQA-LOL
- **Distribution Shifts to Answers**
  - VQA-CP
- **Distribution Shifts to Multi-modalities.**
  - VQA-GEN
  - VQA-CE
  - VQA-VS Adversarial Distribution Shifts
  - AVQA
- **Adversarial**
  - **AdVQA**
- **Far OOD:** TextVQA, VizWiz, OK-VQAv2

Visual Question Answering (VQA) Fine-Tuning Experiments

	ID	Near OOD							Far OOD		
		Vision		Question	Answer	Multimodal	Adversarial				
	VQAv2	IV-VQA	CV-VQA	VQA-Rephrasings	VQA-CP v2	VQA-CE	AdVQA		TextVQA	VizWiz	OK-VQA
Zero-Shot	54.42	63.95	44.72	50.10	54.29	30.68	30.46		14.86	16.84	28.60
Vanilla FT(LoRA)	86.29	94.43	**69.36**	78.90	86.21	71.73	49.82		42.08	22.92	48.30
Linear Prob.	78.24	87.83	63.87	69.61	78.48	61.66	42.90		29.61	18.80	42.27
LP-FT(LoRA)	85.97	93.30	65.93	76.49	86.16	72.73	45.68		31.41	19.01	43.27
WiSE-FT(LoRA)	71.36	85.06	64.55	66.42	70.89	48.74	43.95		36.98	22.41	42.35
Adam-SPD(LoRA)	**87.39**	**95.25**	68.85	**79.48**	**87.27**	**73.52**	**50.90**		**43.56**	**23.05**	**50.11**

New setting: robust fine-tuning for VQA

- ID dataset: VQAv2
- OOD datasets
  - Distribution shifts to images: IV-VQA, CV-VQA
  - Distribution shifts to questions: VQA-Rephrasings
  - Distribution shifts to multi-modalities: VQA-CE
  - Adversarial distribution shifts: AdVQA
  - Far OODs: TextVQA, VizWiz, OK-VQAv2

SPD shows competitiveness across ID, near OOD, and far OOD datasets on multimodal tasks.

Finetuning and Forgetting are common!

We anticipate a number of places for this to be useful!

- Training vision-language-action models for robotics!
    - Some can afford to co-finetune with VQA, etc. but difficult!
- Finetuning to large open-vocabulary corpora (e.g. Wikipedia)
- Multi-task finetuning from pre-trained model

Conclusions

- Distribution shift is ***still*** a problem
  - Private, in-the-wild data

- One approach: Finetune!
  - Question: How to do so robustly? **Per-layer/iteration constraint of gradient update**
  - Not the only choice: Retrieval/RAG, etc.

- Lots of other "distributions" of data!
  - Reasoning, planning, etc.
  - Current approach (o1): Show it the distribution
  - Other approaches?

Acknowledgement and Questions

Junjiao Tian

Robotics Ph.D.

Chengyue Huang

ML Ph.D.

Shivang Chopra

M.S. Student

Brisa Maneech-otesuwan

Undergrad Student

SAMSUNG RESEARCH AMERICA

MIT-IBM Watson AI Lab

Meta

DARPA

NSF

Office of Naval Research Science & Technology

[1] J. Tian, X. Dai, C.Y. Ma, Z. He, Y.C. Liu, and Z. Kira "Trainable Projected Gradient Method for Robust Fine-tuning", CVPR 2023.

[2] J. Tian, Y. Liu, J. Smith, Z. Kira, "Fast Trainable Projection for Robust Fine-tuning", NeurIPS 2023.

[3] J. Tian, C. Huang, and Z. Kira "Rethinking Weight Decay for Robust Fine-Tuning of Foundation Models", NeurIPS 2024.