



Improving Foundation Models for Few-Shot Learning via Multitask Finetuning

Zhuoyan Xu, Zhenmei Shi, Junyi Wei, Yin Li, Yingyu Liang
UW-Madison

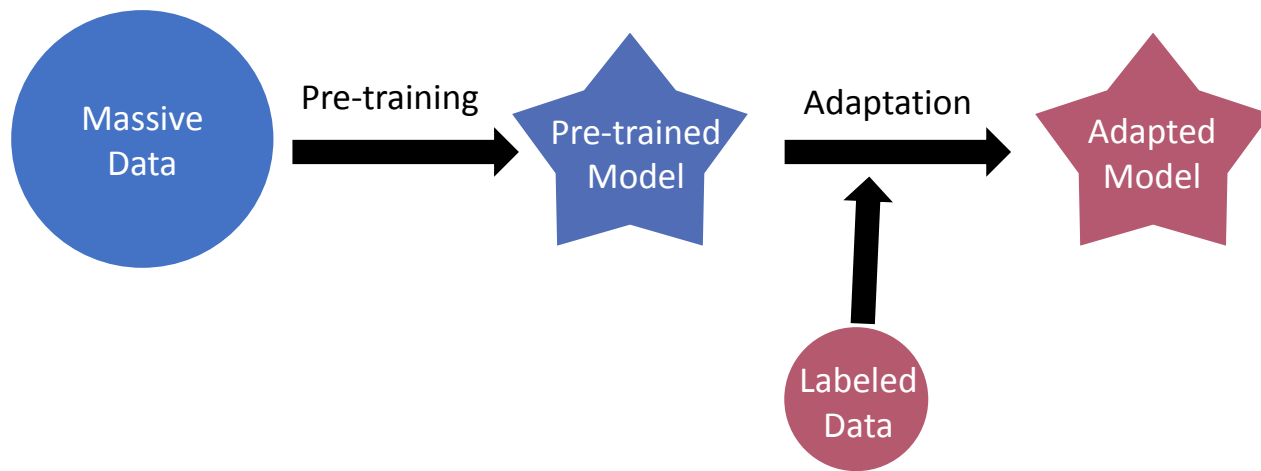
IFDS

New Paradigm: Pretraining + Adaptation

Paradigm shift: supervised learning \implies pre-training + adaptation

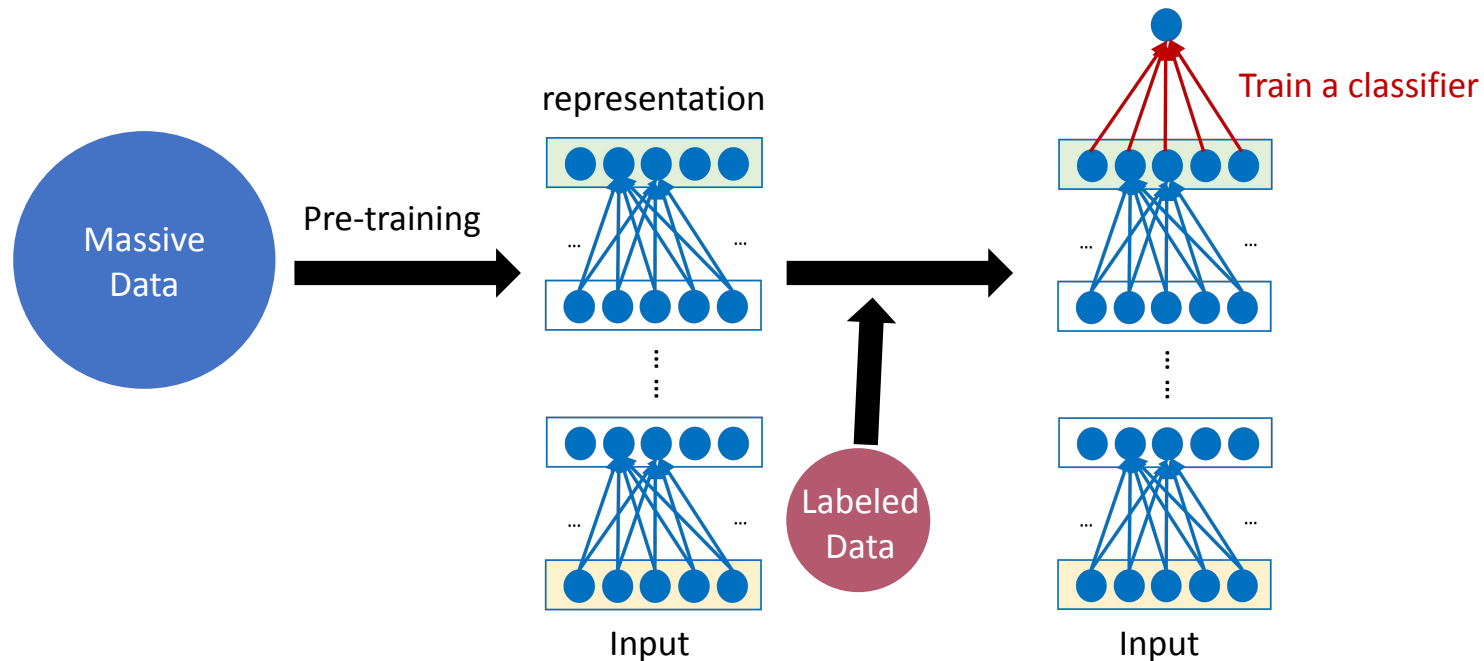
New Paradigm: Pre-trained Representations

Paradigm shift: supervised learning \implies pre-training + adaptation



New Paradigm: Pre-trained Representations

Paradigm shift: supervised learning \longrightarrow pre-training + adaptation



New Paradigm: Pre-trained Representations

Paradigm shift: supervised learning \implies pre-training + adaptation

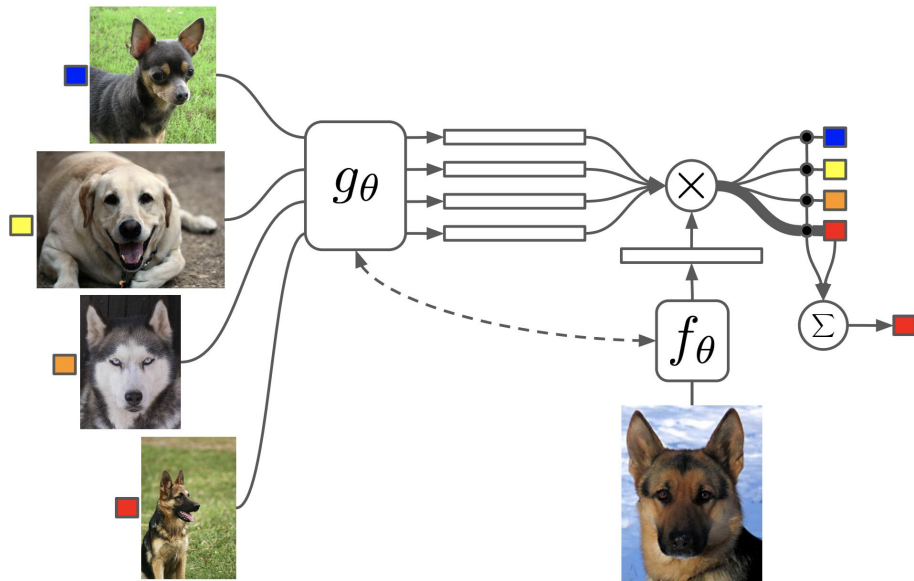


Figure 1: Matching Networks architecture

Adaptation of a pre-trained image encoder

Figures from: *Matching Networks for One Shot Learning*, 2017.

New Paradigm: Pre-trained Representations

Paradigm shift: supervised learning \implies pre-training + adaptation

Circulation revenue has increased by 5% in Finland. // Positive

Panostaja did not disclose the purchase price. // Neutral

Paying off the national debt will be extremely painful. // Negative

The company anticipated its operating profit to improve. // _____



Circulation revenue has increased by 5% in Finland. // Finance

They defeated ... in the NFC Championship Game. // Sports

Apple ... development of in-house chips. // Tech

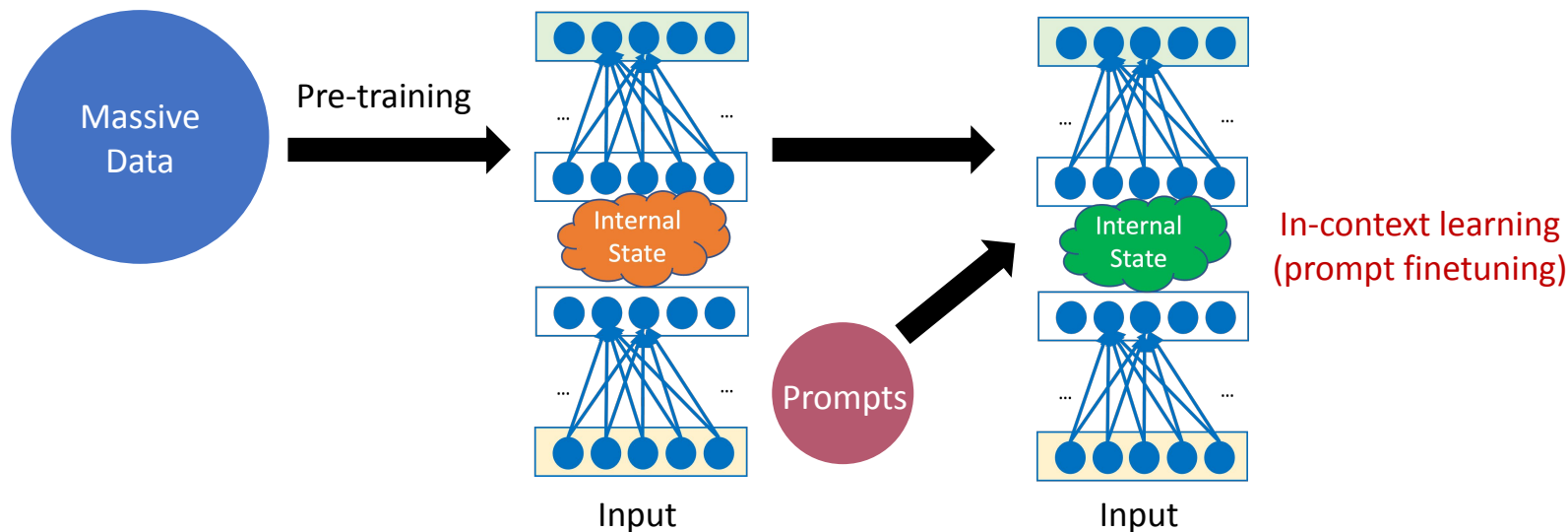
The company anticipated its operating profit to improve. // _____



Adaptation of a pre-trained language decoder

New Paradigm: Pre-trained Representations

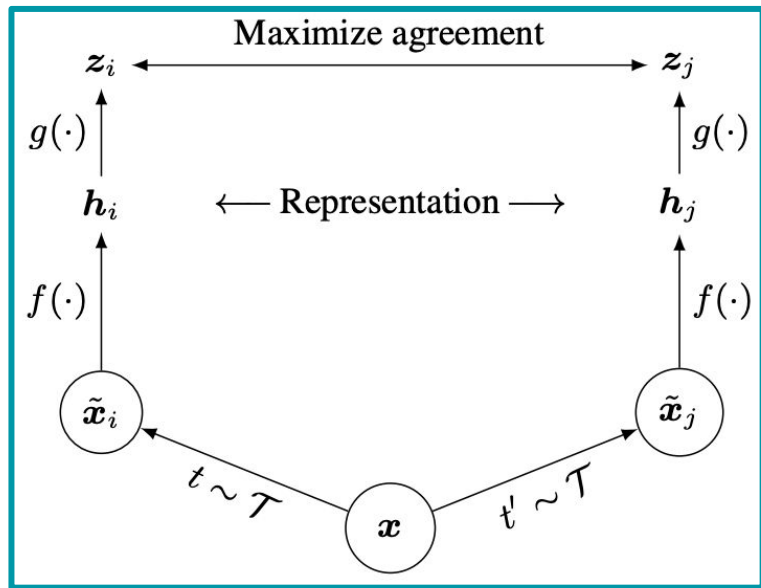
Paradigm shift: supervised learning \longrightarrow pre-training + adaptation



What does pre-training look like?

- Supervised learning
- Self-supervised learning:
 - Next sentence prediction (BERT)
 - Masked language prediction (BERT, RoBERTa)
 - Auto-regressive language modeling (GPT series)
 - Contrastive learning (SimCLR, SimCSE, CLIP, DINO)

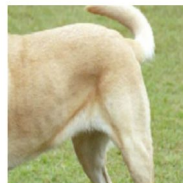
Intro - Contrastive Learning



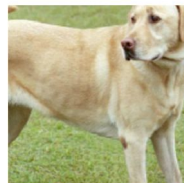
$$\ell_{i,j} = -\log \frac{\exp(\text{sim}(z_i, z_j)/\tau)}{\sum_{k=1}^{2N} \mathbb{1}_{[k \neq i]} \exp(\text{sim}(z_i, z_k)/\tau)}$$



(a) Original



(b) Crop and resize



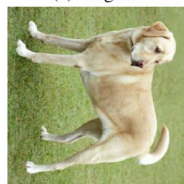
(c) Crop, resize (and flip)



(d) Color distort. (drop)



(e) Color distort. (jitter)



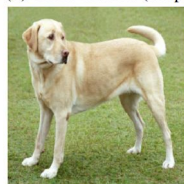
(f) Rotate $\{90^\circ, 180^\circ, 270^\circ\}$



(g) Cutout



(h) Gaussian noise



(i) Gaussian blur



(j) Sobel filtering

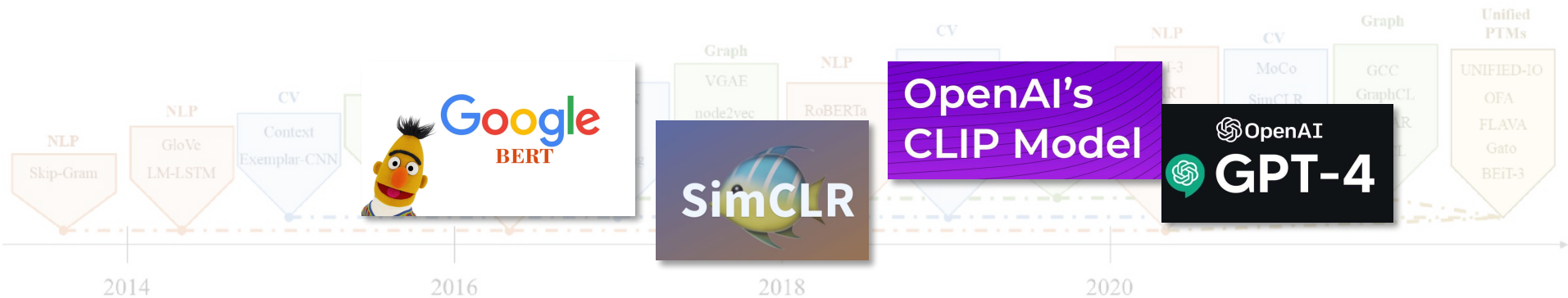
SimCLR - (Image, Image)

No need labels

Image Data Augmentation

Figures from: *A Simple Framework for Contrastive Learning of Visual Representations*, 2020

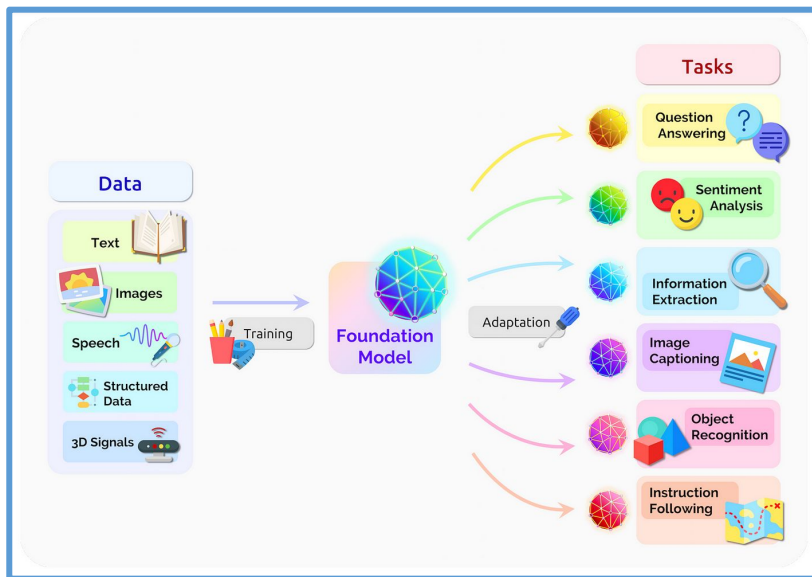
Intro - Foundation Model



The history and evolution of foundation models

Figures from: *A Comprehensive Survey on Pretrained Foundation Models: A History from BERT to ChatGPT, 2023.*

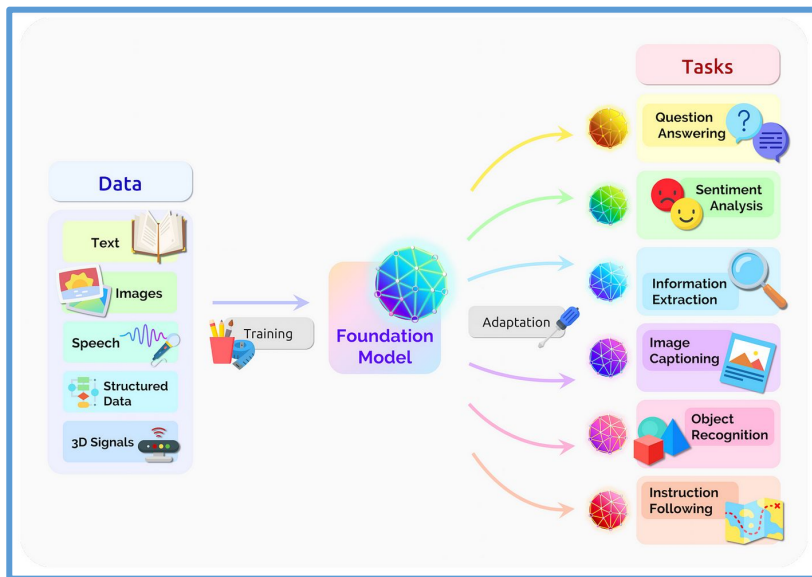
Intro - Foundation Model



Universality

Figures from: *On the opportunities and risks of foundation models, 2021.*

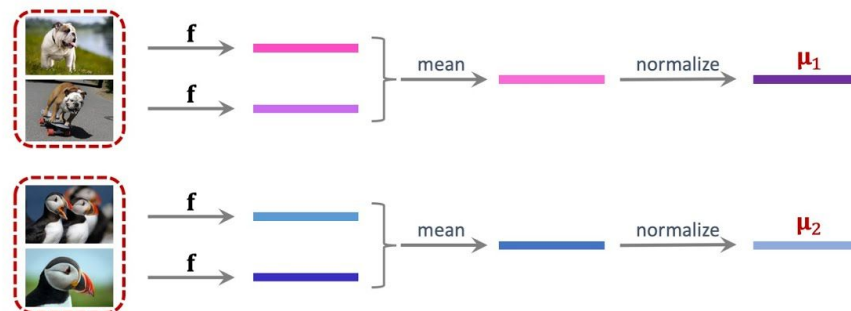
Intro - Foundation Model



Universality

Figures from: *On the opportunities and risks of foundation models, 2021.*

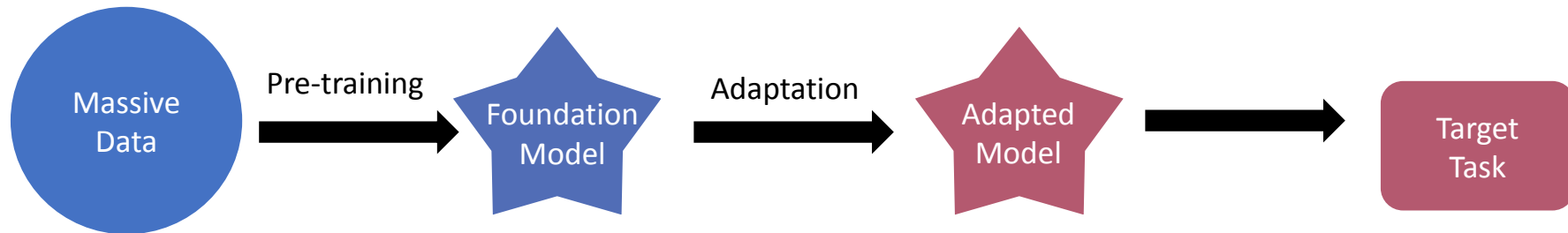
Few-Shot Learning: Pretraining + Fine Tuning



Label Efficiency

Figures from: https://www.youtube.com/watch?v=U6uFOIURcD0&ab_channel=ShusenWang, 2020

Paradigm: Pre-training + Adaptation



Pre-training

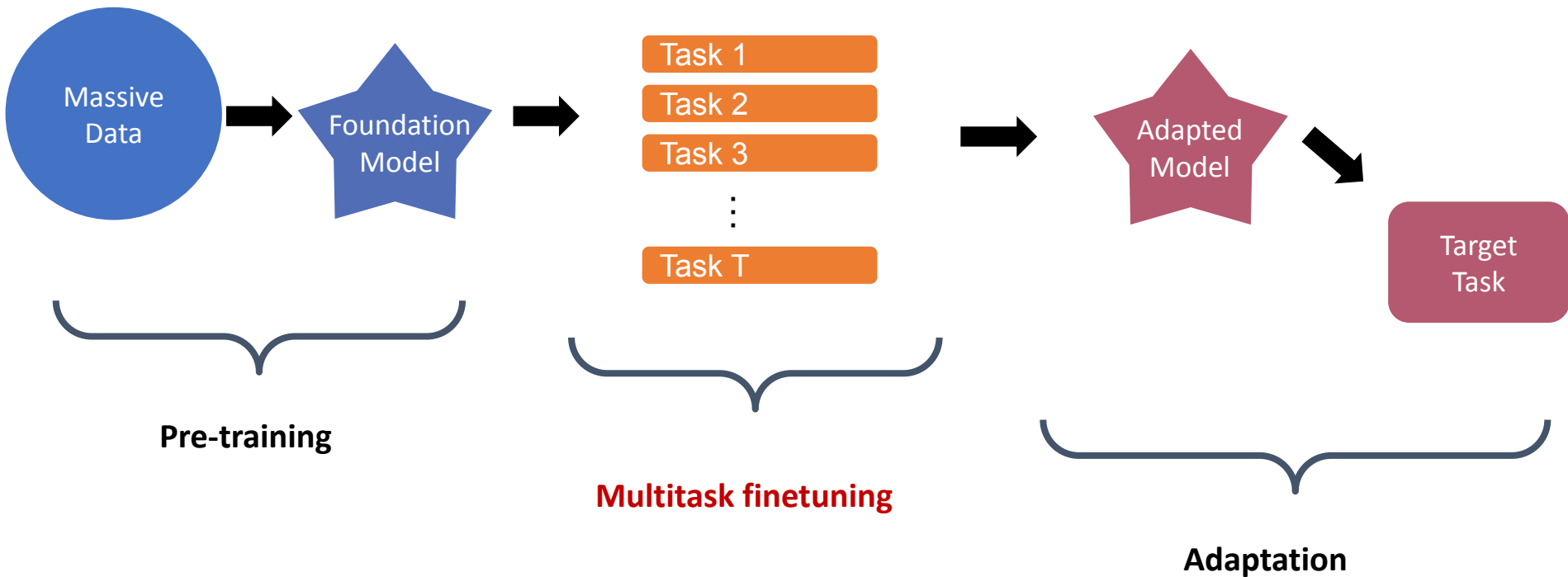


Adaptation



Q: Can we improve this?

Pre-training + Finetuning + Adaptation



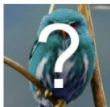
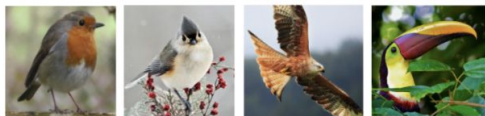
Training

Train dataset #1: "cat-bird"

cats



birds



Train dataset #2: "flower-bike"

flowers



bikes



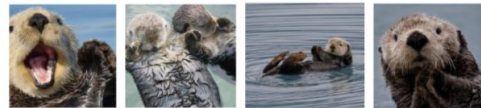
Testing

Test dataset: "dog-otter"

dogs



otters

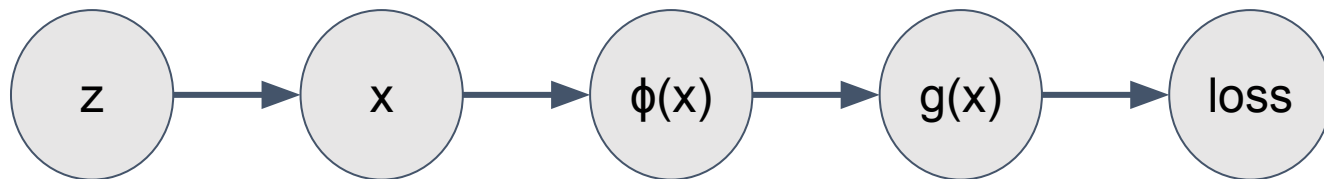


An example of 4-shot 2-class image classification

Figures from: [Meta-Learning: Learning to Learn Fast](#), 2018.

Problem Setup - Hidden representation data model

- Latent class $z \in \mathcal{C}$ over distribution $z \sim \eta$
- Task $\mathcal{T} = (z_1, \dots, z_{K+1}) \subseteq \mathcal{C}$, instance $x \sim \mathcal{D}(z)$
- $\phi \in \Phi$ hypothesis class of representation functions, e.g, ResNet, ViT
- $g(x) = W\phi(x)$ as prediction logits of latent class



Dog



$$\begin{bmatrix} \phi_1 \\ \phi_2 \\ \vdots \\ \phi_d \end{bmatrix}$$

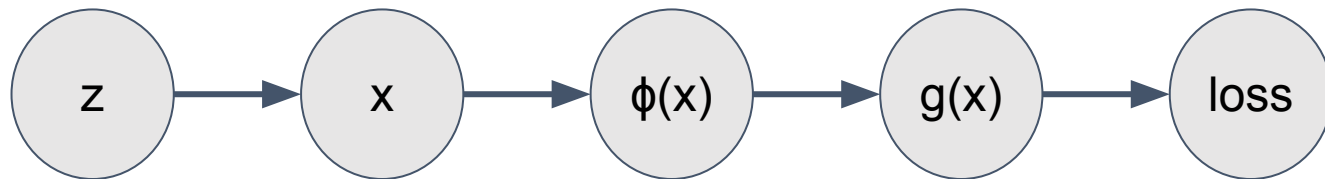
$$\begin{bmatrix} g_1 \\ g_2 \\ \vdots \\ g_{K+1} \end{bmatrix}$$

$$\ell(g(x), z) = -\log \left\{ \frac{\exp(g(\mathbf{x})_z)}{\sum_{k=1}^{K+1} \exp(g(\mathbf{x})_k)} \right\}$$

Problem Setup - Objective for a downstream task?

- Latent class $z \in \mathcal{C}$ over distribution $z \sim \eta$
- Task $\mathcal{T} = \{z_1, z_2\} \subseteq \mathcal{C}$, instance $x \sim \mathcal{D}(z)$
- $\phi \in \Phi$ hypothesis class of representation functions, e.g, ResNet, ViT
- $g(x) = W\phi(x)$ as prediction logits of latent class
- supervised loss w.r.t a task:

$$\mathcal{L}_{sup}(\mathcal{T}, \phi) := \min_W \mathbb{E}_{z \sim \mathcal{T}} \mathbb{E}_{x \sim \mathcal{D}(z)} [\ell(W\phi(x), z)]$$

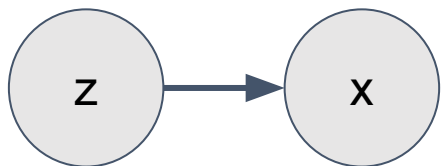


Problem Setup - Contrastive pre-training

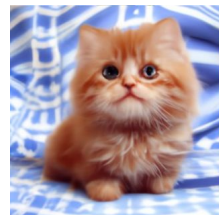
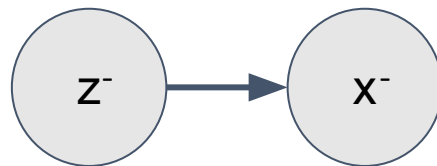
- $(z, z^-) \sim \eta^2$, $x, x^+ \sim \mathcal{D}(z)$, $x^- \sim \mathcal{D}(z^-)$, $\tau := \Pr_{(z, z^-) \sim \eta^2} \{z = z^-\}$

- Contrastive loss:

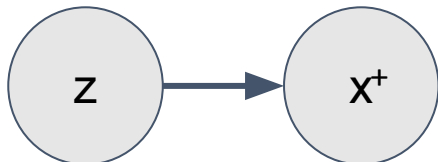
$$\mathbb{E} \left[-\log \left(\frac{e^{\phi(x)^\top \phi(x^+)}}{e^{\phi(x)^\top \phi(x^+)} + e^{\phi(x)^\top \phi(x^-)}} \right) \right]$$



positive pair



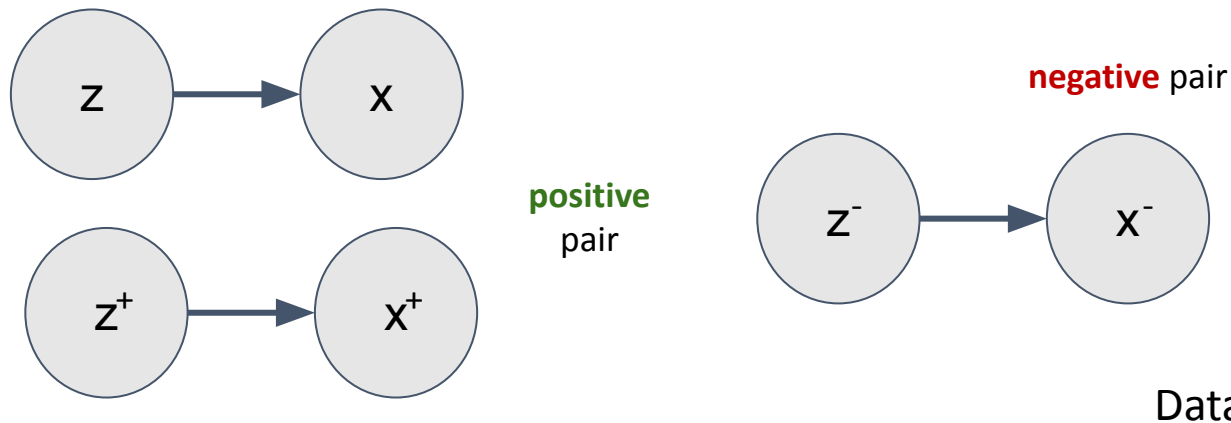
negative pair



Data Model

Problem Setup - Contrastive pre-training

- $(z, z^-) \sim \eta^2$, $x, x^+ \sim \mathcal{D}(z)$, $x^- \sim \mathcal{D}(z^-)$
- Contrastive loss:
$$\mathcal{L}_{un}(\phi) := \mathbb{E} [\ell_u (\phi(x)^\top (\phi(x^+) - \phi(x^-)))]$$
$$\widehat{\mathcal{L}}_{un}(\phi) := \frac{1}{N} \sum_{i=1}^N [\ell_u (\phi(x_i)^\top (\phi(x_i^+) - \phi(x_i^-)))]$$
- In particular: $\ell_u(v) = \log(1 + \exp(-v))$ will recover the loss in previous slide



Problem Setup - Multitask Finetuning

- Suppose in pre-training we have $\hat{\mathcal{L}}_{un}(\hat{\phi}) \leq \epsilon_0$
- Suppose we construct M tasks, each with m sample
- We further multitask finetune to get a new ϕ' by:

$$\min_{W_i \in \mathbb{R}^d, \phi \in \Phi} \frac{1}{M} \sum_{i=1}^M \frac{1}{m} \sum_{j=1}^m \ell(W_i \cdot \phi(x_j^i), z_j^i), \quad \text{s.t.} \quad \hat{\mathcal{L}}_{un}(\phi) \leq \epsilon_0$$

Intuition: Comparing to direct training, this reduce hypothesis space from Φ to $\Phi(\epsilon_0) = \left\{ \phi \in \Phi : \hat{\mathcal{L}}_{un}(\phi) \leq \epsilon_0 \right\}$

Main Result

- Suppose target task is \mathcal{T}_0
- Suppose there is ϕ^* such that supervised loss are small across all tasks
- We want to bound $\mathcal{L}_{sup}(\mathcal{T}_0, \phi) - \mathcal{L}_{sup}(\mathcal{T}_0, \phi^*)$

Theorem 1 (Contrastive pre-training loss(baseline))

Suppose in pre-training we have $\hat{\mathcal{L}}_{un}(\hat{\phi}) \leq \epsilon_0$, then:

$$\mathcal{L}_{sup}(\mathcal{T}_0, \hat{\phi}) - \mathcal{L}_{sup}(\mathcal{T}_0, \phi^*) \leq \mathcal{O}((2\epsilon_0 - \tau) - \mathcal{L}_{sup}(\phi^*))$$

Main Result

- Suppose target task is \mathcal{T}_0
- We want to bound $\mathcal{L}_{sup}(\mathcal{T}_0, \phi) - \mathcal{L}_{sup}(\mathcal{T}_0, \phi^*)$

Theorem 2 (Multitask finetuning loss(Ours))

Suppose we solve multitask finetuning optimization with empirical loss smaller than $\epsilon_1 = 2\alpha\epsilon_0$ and got ϕ' . If:

$$M \geq \Omega\left(\frac{1}{\epsilon_1} \left[\mathcal{R}_M(\Phi(\epsilon_0)) + \frac{1}{\epsilon_1} \log\left(\frac{1}{\delta}\right) \right]\right), \quad Mm \geq \Omega\left(\frac{1}{\epsilon_1} \left[\mathcal{R}_{Mm}(\Phi(\epsilon_0)) + \frac{1}{\epsilon_1} \log\left(\frac{1}{\delta}\right) \right]\right)$$

Then with prob $1 - \delta$,

$$\mathcal{L}_{sup}(\mathcal{T}_0, \phi') - \mathcal{L}_{sup}(\mathcal{T}_0, \phi^*) \leq \mathcal{O}(\alpha(2\epsilon_0 - \tau) - \mathcal{L}_{sup}(\phi^*))$$

Remark

- Comparing to pre-training + adaptation(baseline), our multitask finetuning reduce error on target task by $2(1 - \alpha)\epsilon_0$
where finetuning sample complexity is $\Theta\left(\frac{1}{\alpha\epsilon_0}\right)$
- Comparing to traditional supervised learning, self-supervised pre-training reduce error by $O\left(\frac{1}{M_m} [\mathcal{R}_{M_m}(\Phi) - \mathcal{R}_{M_m}(\Phi(\epsilon_0))]\right)$

Experiments: Few-shot Vision tasks

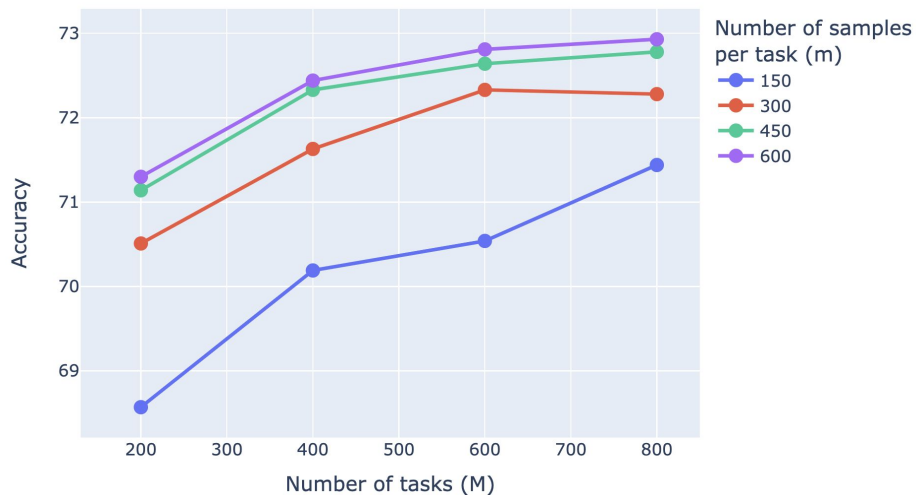
15-way accuracy (%) on *tiered-ImageNet*, 1 image per class in target task

Backbone	Direct Adaptation	Finetuning
ViT-B32	59.55 \pm 0.21	68.57 \pm 0.37
ResNet50	51.76 \pm 0.36	57.56 \pm 0.36

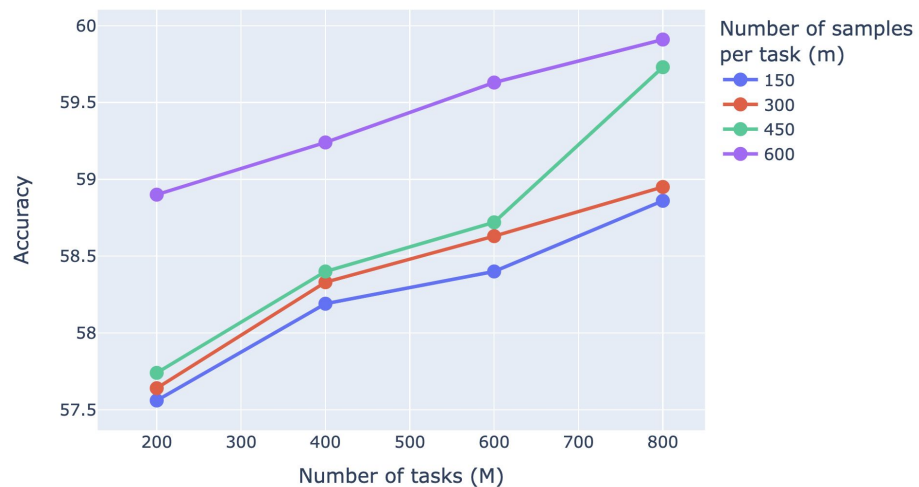
Effects of multitask finetuning

Experiments: Few-shot Vision tasks

15-way accuracy (%) on *tiered-ImageNet*, 1 image per class in target task



ViT-B32



ResNet50

Accuracy with varying number of tasks and samples

Experiments: Few-shot Language task

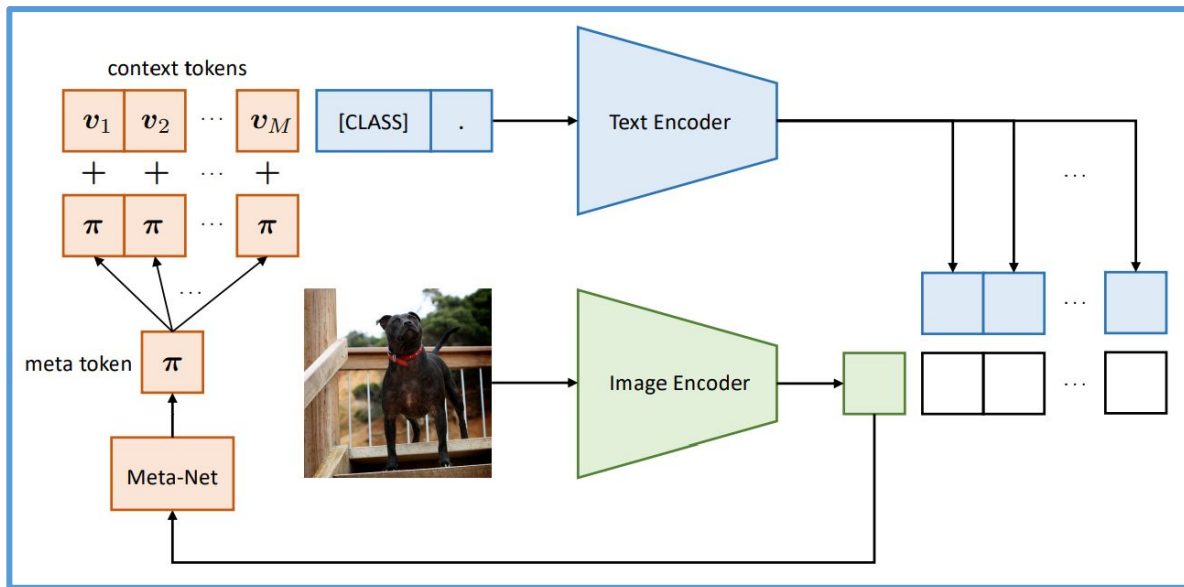
Text classification for different text dataset, with prompt-base finetuning

	SST-2 (acc)	SST-5 (acc)	MR (acc)	CR (acc)	MPQA (acc)	Subj (acc)	TREC (acc)	CoLA (Matt.)
Prompt-based zero-shot	83.6	35.0	80.8	79.5	67.6	51.4	32.0	2.0
Multitask FT zero-shot	92.9	37.2	86.5	88.8	73.9	55.3	36.8	-0.065
Prompt-based FT [†]	92.7 (0.9)	47.4 (2.5)	87.0 (1.2)	90.3 (1.0)	84.7 (2.2)	91.2 (1.1)	84.8 (5.1)	9.3 (7.3)
Multitask Prompt-based FT	92.0 (1.2)	48.5 (1.2)	86.9 (2.2)	90.5 (1.3)	86.0 (1.6)	89.9 (2.9)	83.6 (4.4)	5.1 (3.8)
+ task selection	92.6 (0.5)	47.1 (2.3)	87.2 (1.6)	91.6 (0.9)	85.2 (1.0)	90.7 (1.6)	87.6 (3.5)	3.8 (3.2)
	MNLI (acc)	MNLI-mm (acc)	SNLI (acc)	QNLI (acc)	RTE (acc)	MRPC (F1)	QQP (F1)	
Prompt-based zero-shot	50.8	51.7	49.5	50.8	51.3	61.9	49.7	
Multitask FT zero-shot	63.2	65.7	61.8	65.8	74.0	81.6	63.4	
Prompt-based FT [†]	68.3 (2.3)	70.5 (1.9)	77.2 (3.7)	64.5 (4.2)	69.1 (3.6)	74.5 (5.3)	65.5 (5.3)	
Multitask Prompt-based FT	70.9 (1.5)	73.4 (1.4)	78.7 (2.0)	71.7 (2.2)	74.0 (2.5)	79.5 (4.8)	67.9 (1.6)	
+ task selection	73.5 (1.6)	75.8 (1.5)	77.4 (1.6)	72.0 (1.6)	70.0 (1.6)	76.0 (6.8)	69.8 (1.7)	

Our main results using RoBERTa-large. †: Result in (GFC20);

Experiments: zero-shot vision language task

Conditional context optimization for CLIP model



CoCoOp

Figures from: *Conditional Prompt Learning for Vision-Language Models, 2022.*

Experiments: zero-shot vision language task

160(all)-way zero-shot accuracy (%) on *tiered-ImageNet* test split

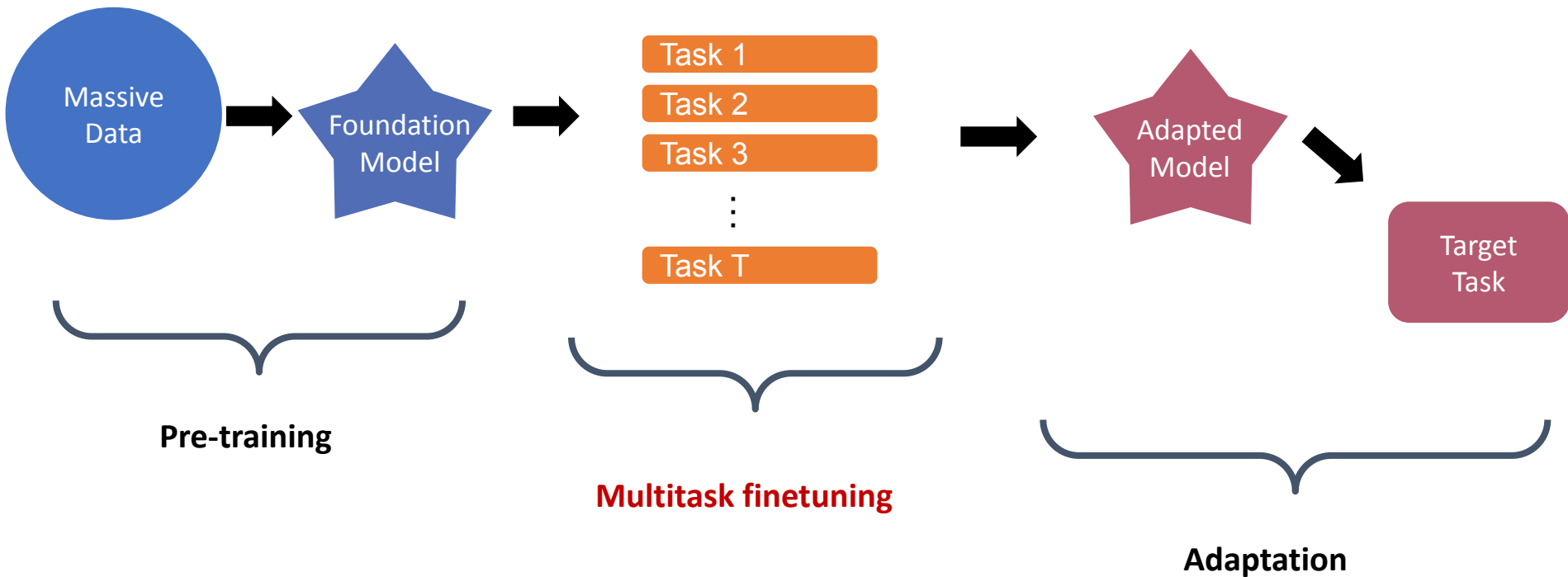
Backbone	Zero-shot	Multitask finetune
ViT-B32	69.9	71.4

Effects of multitask finetuning

Future Work

- Theoretically: How would we quantify the relationship of data between multitask and target task? Concrete and well-motivated problem instances satisfying the task diversity assumptions for instantiating the error guarantee.
- Empirically: Does task diversity provide any insights on data selection in multitask finetuning? Can we design better strategies for constructing and choosing finetuning task?

Take Home Message



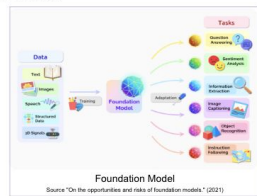
Thanks!

Appendix

Our Workshop Poster: [link](#)

Our Workshop Paper: [link](#)

Motivation

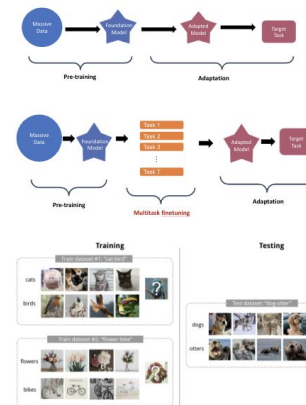


Take-Home Message

We use a paradigm that first finetunes a foundation model with multiple relevant tasks before adapting it to a target task.

Key Intuition

- Pre-training uses unlabeled and noisy data for general purpose learning, where the model learns representation rather than task-specific knowledge. Its performance on a specific task may only be adequate.
- Although the target data is limited, we have a clear understanding of the target task and its associated data.
 - We select additional data from a relevant source that covers its characteristic data.
 - We construct specific tasks for multitask finetuning to allow the model to handle the particular types of target tasks.



An example of 4-shot 2-class image classification

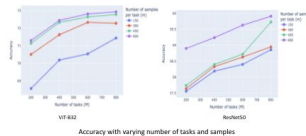
Experiments

Few-shot Vision tasks

15-way accuracy (%) on ImageNet, 3 images per class in target task

Backbone	Direct Adaptation	Finetuning
ViT-B/32	59.55 ± 0.21	68.57 ± 0.37
ResNeXt50	51.79 ± 0.36	57.56 ± 0.36

200 finetuning tasks, 150 images per task



Few-shot Language task

Test classification for different test dataset, with prompt-base finetuning

	SW1	SW4	MR	CR	MPYA	Shy	TRIC	CoLA
Direct Adaptation	83.8	76.0	86.8	79.5	67.6	51.4	52.8	59.0
Finetuning	92.8	77.2	90.5	88.1	73.9	65.3	56.8	65.6
Direct Adaptation	97.0 (91.0)	87.6 (73.0)	97.8 (72.0)	90.1 (81.8)	84.2 (72.0)	88.8 (71.1)	84.6 (71.0)	94.7 (71.0)
Multitask Prompt-based FT	98.0 (71.2)	88.0 (72.2)	95.0 (71.0)	94.0 (71.0)	89.0 (71.0)	83.0 (66.4)	71.0 (66.4)	81.0 (66.4)
Direct Adaptation	87.1 (63.0)	87.2 (61.0)	89.6 (61.0)	90.1 (61.0)	90.1 (61.0)	87.0 (61.0)	87.0 (61.0)	87.0 (61.0)
Finetuning	91.0	89.0	91.0	90.0	82.0	61.0	61.0	61.0
Direct Adaptation	85.0 (71.0)	85.0 (71.0)	85.0 (71.0)	85.0 (71.0)	85.0 (71.0)	85.0 (71.0)	85.0 (71.0)	85.0 (71.0)
Multitask Prompt-based FT	90.0 (71.0)	88.0 (71.0)	90.0 (71.0)	88.0 (71.0)	88.0 (71.0)	88.0 (71.0)	88.0 (71.0)	88.0 (71.0)
Direct Adaptation	78.0 (61.0)	78.0 (61.0)	78.0 (61.0)	78.0 (61.0)	78.0 (61.0)	78.0 (61.0)	78.0 (61.0)	78.0 (61.0)
Finetuning	84.0 (61.0)	82.0 (61.0)	84.0 (61.0)	82.0 (61.0)	82.0 (61.0)	82.0 (61.0)	82.0 (61.0)	82.0 (61.0)

Our main results using 400k-ft dataset. * Result in (2022C).

(2022C) Test, task and data. Making our related language models better for few-shot NLP tasks.

(2022C) ViT, task and data. Making our related language models better for few-shot NLP tasks.

(2022C) ViT, task and data. Making our related language models better for few-shot NLP tasks.

(2022C) ViT, task and data. Making our related language models better for few-shot NLP tasks.

(2022C) ViT, task and data. Making our related language models better for few-shot NLP tasks.

(2022C) ViT, task and data. Making our related language models better for few-shot NLP tasks.

(2022C) ViT, task and data. Making our related language models better for few-shot NLP tasks.

(2022C) ViT, task and data. Making our related language models better for few-shot NLP tasks.

(2022C) ViT, task and data. Making our related language models better for few-shot NLP tasks.

(2022C) ViT, task and data. Making our related language models better for few-shot NLP tasks.

(2022C) ViT, task and data. Making our related language models better for few-shot NLP tasks.

(2022C) ViT, task and data. Making our related language models better for few-shot NLP tasks.

(2022C) ViT, task and data. Making our related language models better for few-shot NLP tasks.

(2022C) ViT, task and data. Making our related language models better for few-shot NLP tasks.

(2022C) ViT, task and data. Making our related language models better for few-shot NLP tasks.

(2022C) ViT, task and data. Making our related language models better for few-shot NLP tasks.

(2022C) ViT, task and data. Making our related language models better for few-shot NLP tasks.

(2022C) ViT, task and data. Making our related language models better for few-shot NLP tasks.

(2022C) ViT, task and data. Making our related language models better for few-shot NLP tasks.

(2022C) ViT, task and data. Making our related language models better for few-shot NLP tasks.

(2022C) ViT, task and data. Making our related language models better for few-shot NLP tasks.

(2022C) ViT, task and data. Making our related language models better for few-shot NLP tasks.

(2022C) ViT, task and data. Making our related language models better for few-shot NLP tasks.

(2022C) ViT, task and data. Making our related language models better for few-shot NLP tasks.

(2022C) ViT, task and data. Making our related language models better for few-shot NLP tasks.

(2022C) ViT, task and data. Making our related language models better for few-shot NLP tasks.

(2022C) ViT, task and data. Making our related language models better for few-shot NLP tasks.

(2022C) ViT, task and data. Making our related language models better for few-shot NLP tasks.

(2022C) ViT, task and data. Making our related language models better for few-shot NLP tasks.

(2022C) ViT, task and data. Making our related language models better for few-shot NLP tasks.

(2022C) ViT, task and data. Making our related language models better for few-shot NLP tasks.

Theoretical Analysis

Contrastive Learning

$$\text{objective function: } \mathcal{L}_{\text{con}}(\phi) := \mathbb{E} \left[-\log \left(\frac{e^{\phi(x^+) \cdot \phi(x^+)}}{e^{\phi(x^+) \cdot \phi(x^+)} + e^{\phi(x^+) \cdot \phi(x^-)}} \right) \right]$$

Supervised loss respect to a task T . W is a linear classifier.

$$\mathcal{L}_{\text{sup}}(T, \phi) := \min_{W, x, z} \mathbb{E} [l(W\phi(x), z)]$$

Multitask finetuning

Suppose we construct M tasks, each with m sample

$$\min_{W \in \mathbb{R}^{k \times d}, \phi \in \mathcal{M}} \frac{1}{M} \sum_{i=1}^M \frac{1}{m} \sum_{j=1}^m \ell(W_i \cdot \phi(x_j^i), z_j^i), \quad \text{s.t. } \mathcal{L}_{\text{con}}(\phi) \leq \epsilon_0$$

Hidden Representation Data Model

- First sampling the latent class, and then sampling input.
- In contrastive pre-training, positive pair sampling from the same latent class.
- A task T contains a subset of latent classes.

Proposition of target task error (Informal)

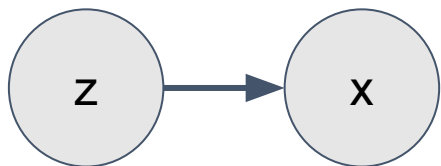
Suppose in pre-training we have target task error bounded by ϵ with high probability, our multitask finetuning reduce error on target task to $\alpha\epsilon$, where finetuning sample complexity is $\mathcal{O}(1/\alpha\epsilon)$.

Problem Setup - Contrastive pre-training

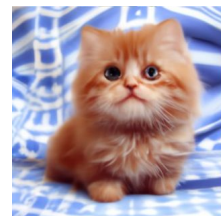
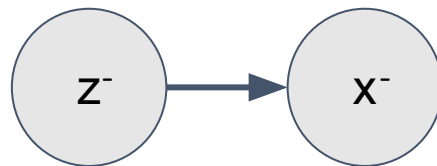
- $(z, z^-) \sim \eta^2, x, x^+ \sim \mathcal{D}(z), x^- \sim \mathcal{D}(z^-)$

- Contrastive loss:

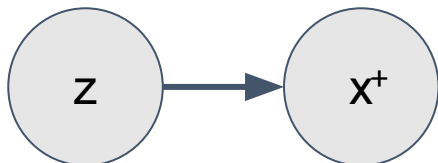
$$\mathbb{E} \left[-\log \left(\frac{e^{\phi(x)^\top \phi(x^+)}}{e^{\phi(x)^\top \phi(x^+)} + e^{\phi(x)^\top \phi(x^-)}} \right) \right]$$



positive pair



negative pair



Data Model

Main Result

- Suppose target task is \mathcal{T}_0
- We want to bound $\mathcal{L}_{sup}(\mathcal{T}_0, \phi)$
- let ζ denote the conditional distribution of $(z_1, z_2) \sim \eta^2$ conditioned on $z_1 \neq z_2$

Definition 1 (Averaged representation difference)

$$\bar{d}_\zeta(\phi, \tilde{\phi}) := \mathbb{E}_{\mathcal{T} \sim \zeta} \left[\mathcal{L}_{sup}(\mathcal{T}, \phi) - \mathcal{L}_{sup}(\mathcal{T}, \tilde{\phi}) \right] = \mathcal{L}_{sup}(\phi) - \mathcal{L}_{sup}(\tilde{\phi})$$

Definition 2 (worst-case representation difference)

$$d_{\mathcal{C}_0}(\phi, \tilde{\phi}) := \sup_{\mathcal{T}_0 \subseteq \mathcal{C}_0} \left[\mathcal{L}_{sup}(\mathcal{T}_0, \phi) - \mathcal{L}_{sup}(\mathcal{T}_0, \tilde{\phi}) \right]$$

(ν, ϵ) -diversity: For any $\phi, \tilde{\phi} \in \Phi$, $d_{\mathcal{C}_0}(\phi, \tilde{\phi}) \leq \bar{d}_\zeta(\phi, \tilde{\phi})/\nu + \epsilon$

Main Result

- Suppose target task is \mathcal{T}_0
- let ζ denote the conditional distribution of $(z_1, z_2) \sim \eta^2$ conditioned on $z_1 \neq z_2$
- (ν, ϵ) -diversity: For any $\phi, \tilde{\phi} \in \Phi$, $d_{\mathcal{C}_0}(\phi, \tilde{\phi}) \leq \bar{d}_{\zeta}(\phi, \tilde{\phi})/\nu + \epsilon$
- Suppose there is ϕ^* such that supervised loss are small across all tasks

Theorem 1 (Contrastive pre-training loss(baseline))

Suppose in pre-training we have $\hat{\mathcal{L}}_{un}(\hat{\phi}) \leq \epsilon_0$, then:

$$\mathcal{L}_{sup}(\mathcal{T}_0, \hat{\phi}) - \mathcal{L}_{sup}(\mathcal{T}_0, \phi^*) \leq \frac{1}{\nu} \left[\frac{1}{1 - \tau} (2\epsilon_0 - \tau) - \mathcal{L}_{sup}(\phi^*) \right] + \epsilon.$$

Main Result

- Suppose target task is \mathcal{T}_0
- let ζ denote the conditional distribution of $(z_1, z_2) \sim \eta^2$ conditioned on $z_1 \neq z_2$
- (ν, ϵ) -diversity: For any $\phi, \tilde{\phi} \in \Phi$, $d_{\mathcal{L}_0}(\phi, \tilde{\phi}) \leq \bar{d}_{\zeta}(\phi, \tilde{\phi})/\nu + \epsilon$

Theorem 2 (Multitask finetuning loss(Ours))

Suppose we solve multitask finetuning optimization with empirical loss smaller than $\epsilon_1 = \frac{\alpha}{3} \frac{1}{1-\tau} (2\epsilon_0 - \tau)$ and got ϕ' . If:

$$M \geq \Omega \left(\frac{1}{\epsilon_1} \left[\mathcal{R}_M(\Phi(\epsilon_0)) + \frac{1}{\epsilon_1} \log \left(\frac{1}{\delta} \right) \right] \right), \quad Mm \geq \Omega \left(\frac{1}{\epsilon_1} \left[\mathcal{R}_{Mm}(\Phi(\epsilon_0)) + \frac{1}{\epsilon_1} \log \left(\frac{1}{\delta} \right) \right] \right)$$

Then with prob $1 - \delta$,

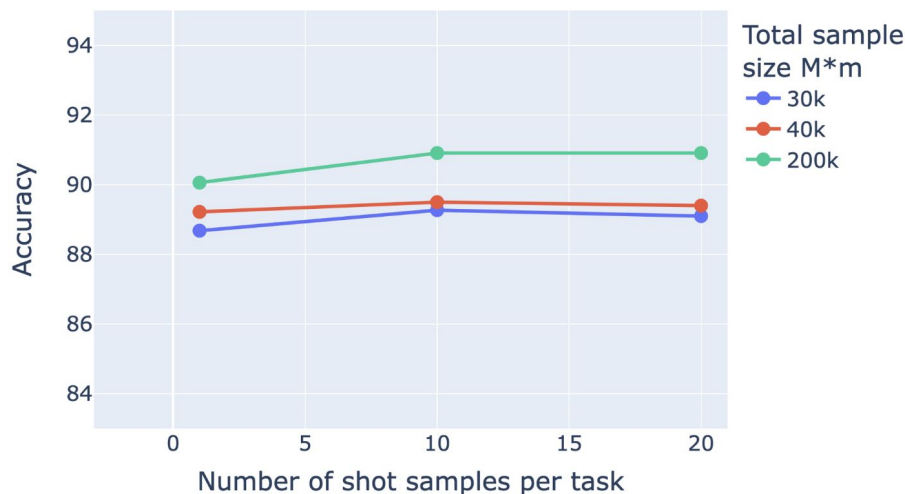
$$\mathcal{L}_{sup}(\mathcal{T}_0, \phi') - \mathcal{L}_{sup}(\mathcal{T}_0, \phi^*) \leq \frac{1}{\nu} \left[\alpha \frac{1}{1-\tau} (2\epsilon_0 - \tau) - \mathcal{L}_{sup}(\phi^*) \right] + \epsilon$$

Remark

- Comparing to pre-training + adaptation(baseline), our multitask finetuning reduce error on target task by $\frac{1}{\nu} \left[(1 - \alpha) \frac{1}{1 - \tau} (2\epsilon_0 - \tau) \right]$
where finetuning sample complexity is $\Theta \left(\frac{1}{\alpha \epsilon_0} \right)$
- Comparing to traditional supervised learning, self-supervised pre-training reduce error by $O \left(\frac{1}{M_m} [\mathcal{R}_{M_m}(\Phi) - \mathcal{R}_{M_m}(\Phi(\epsilon_0))] \right)$

Experiments: Few-shot Vision tasks

5-way accuracy (%) on *mini-ImageNet*, 1/10/20 image per class in target task



ViT-B32

Accuracy with varying number shot images