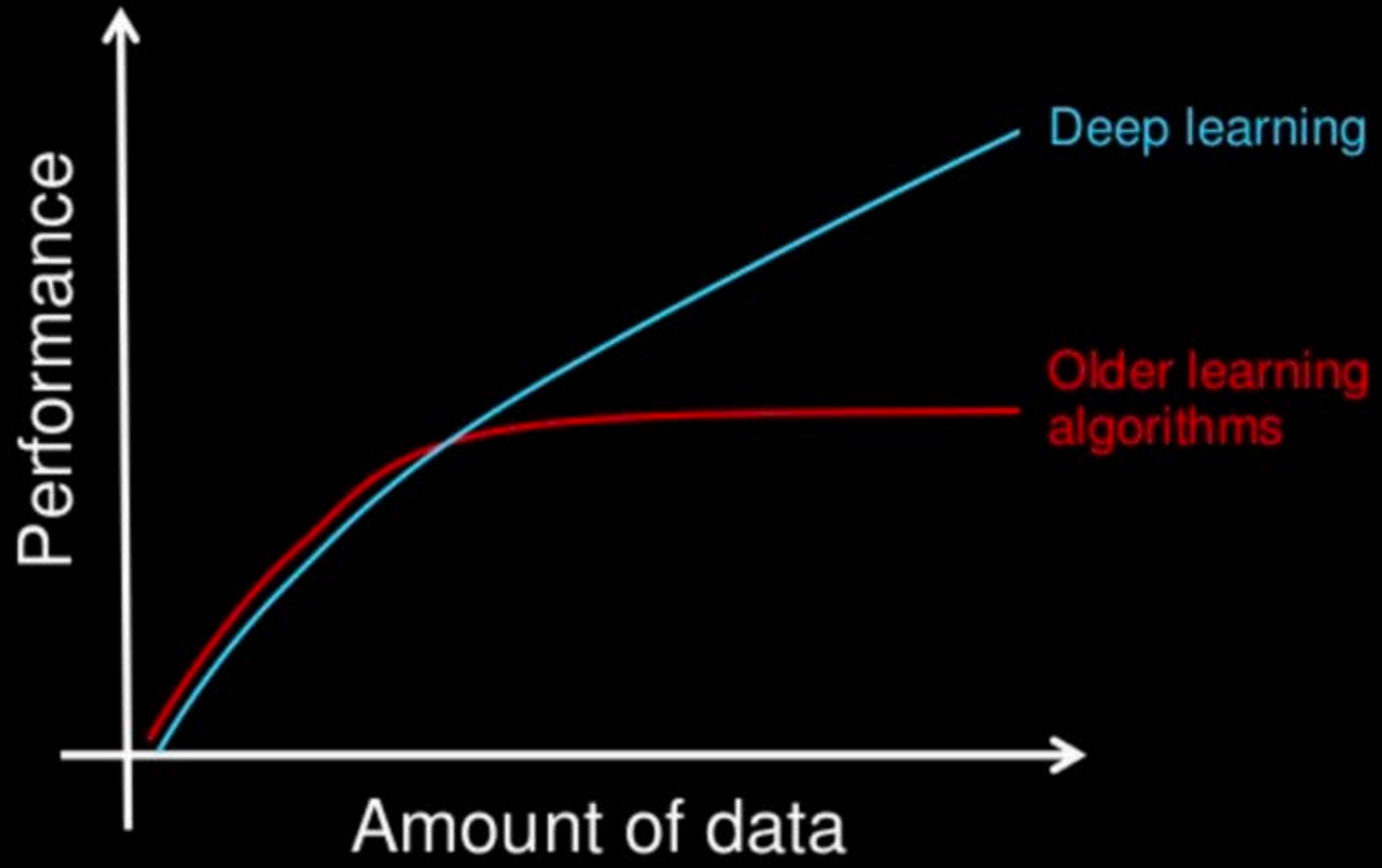
Lecture 18: Transfer Learning I

Speaker: Sara Beery

Transfer Learning I

- Learning with little data
- Transferring knowledge about the mapping
  - Finetuning
  - Domain adaptation
- Transferring knowledge about the outputs
  - Knowledge distillation
- Foundation models
  - Prompting

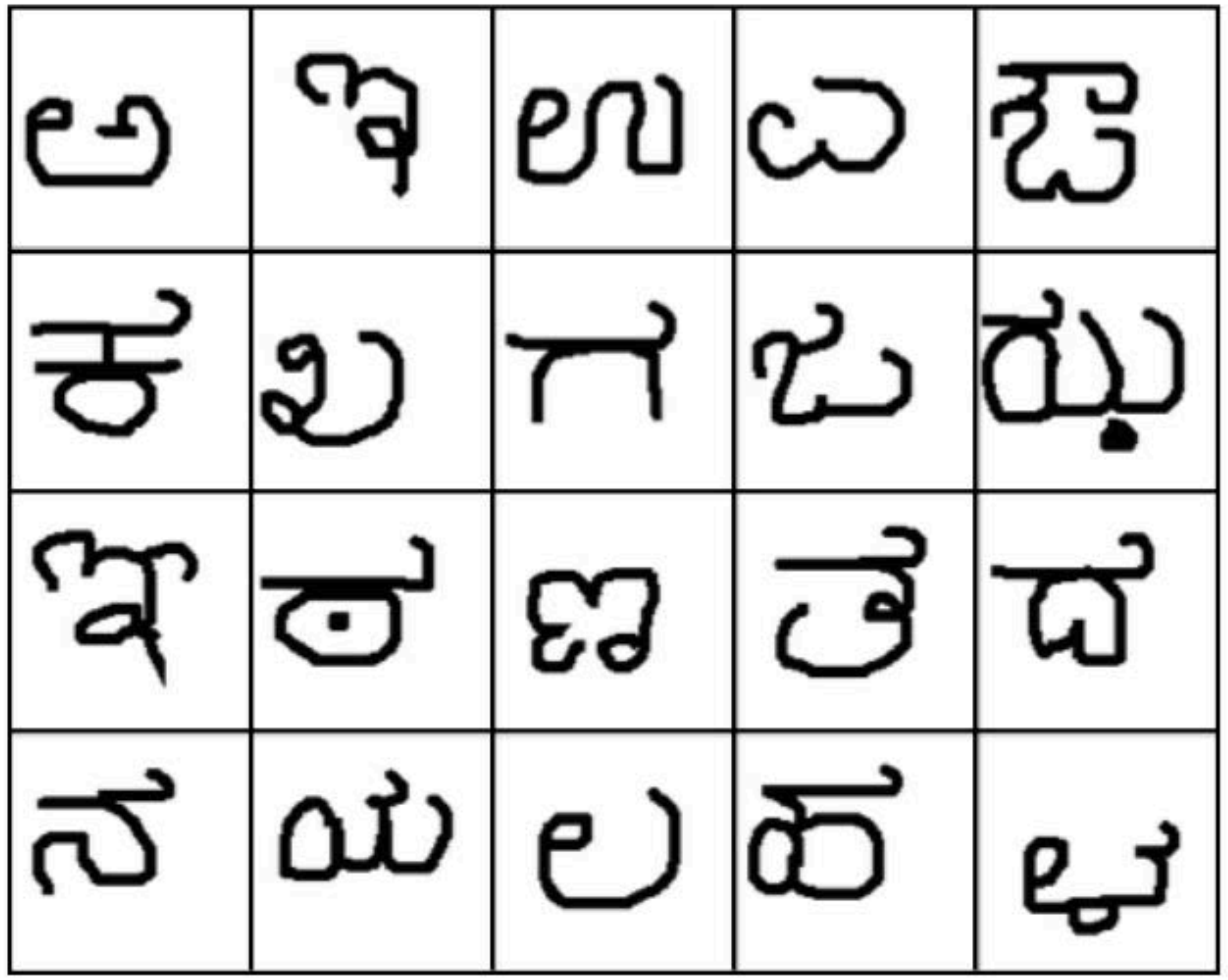
Why deep learning



How do data science techniques scale with amount of data?

[Slide credit: Andrew Ng]

Few-shot Learning



This is a "dax".

Which of the below symbols are also daxes?

[Lake, Salakhutdinov, Tenenbaum, 2015]

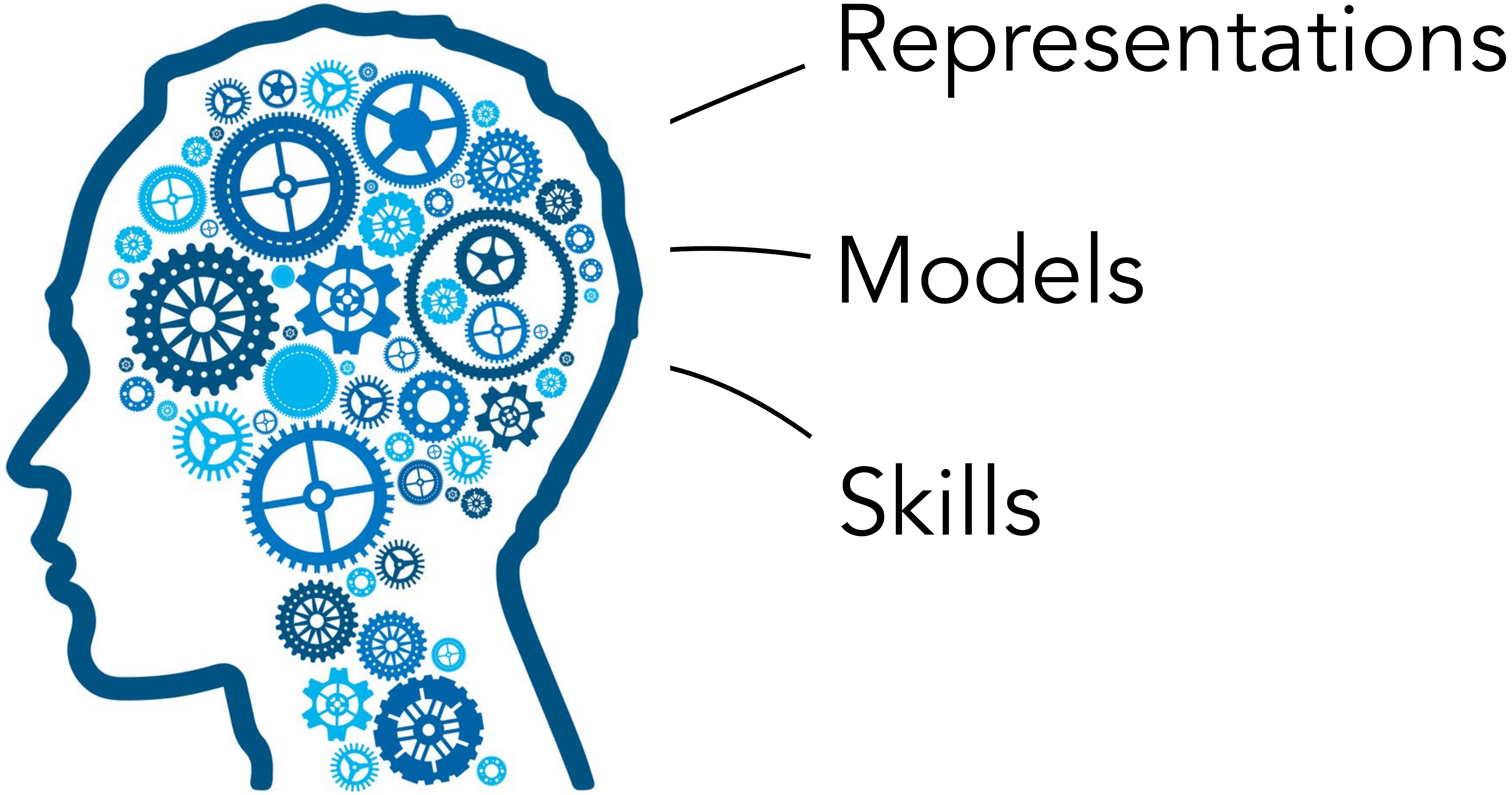
Few-shot Learning

Which of these is an example of the same concept as the item in the box?

[Lake, Salakhutdinov, Tenenbaum, 2015]

"Deep learning"

Human learning



How can we give deep nets prior knowledge?

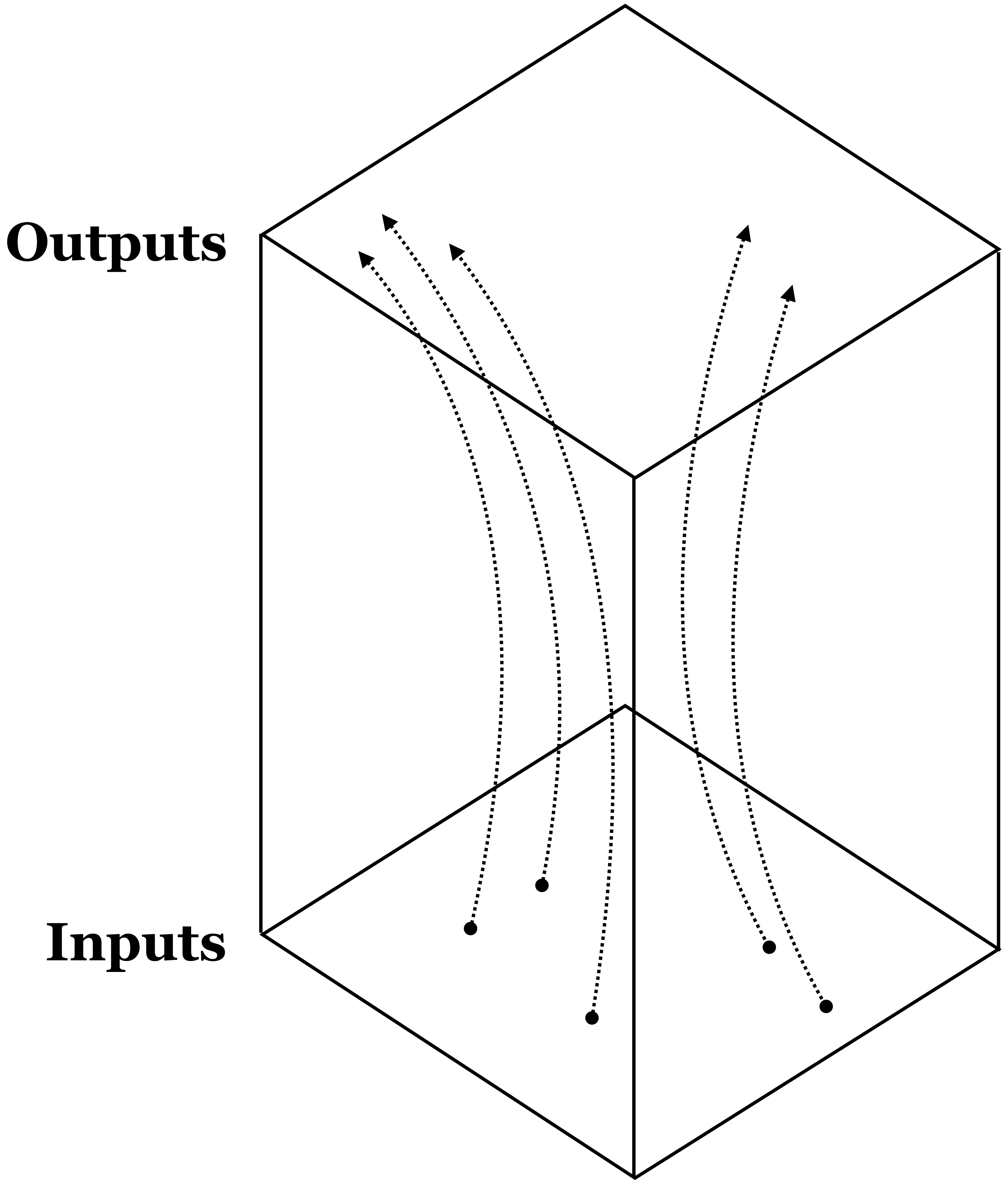
How can we give deep nets prior knowledge?

Just "pretrain" them on prior tasks!

Then transfer knowledge from the pretrained nets to solve novel tasks.

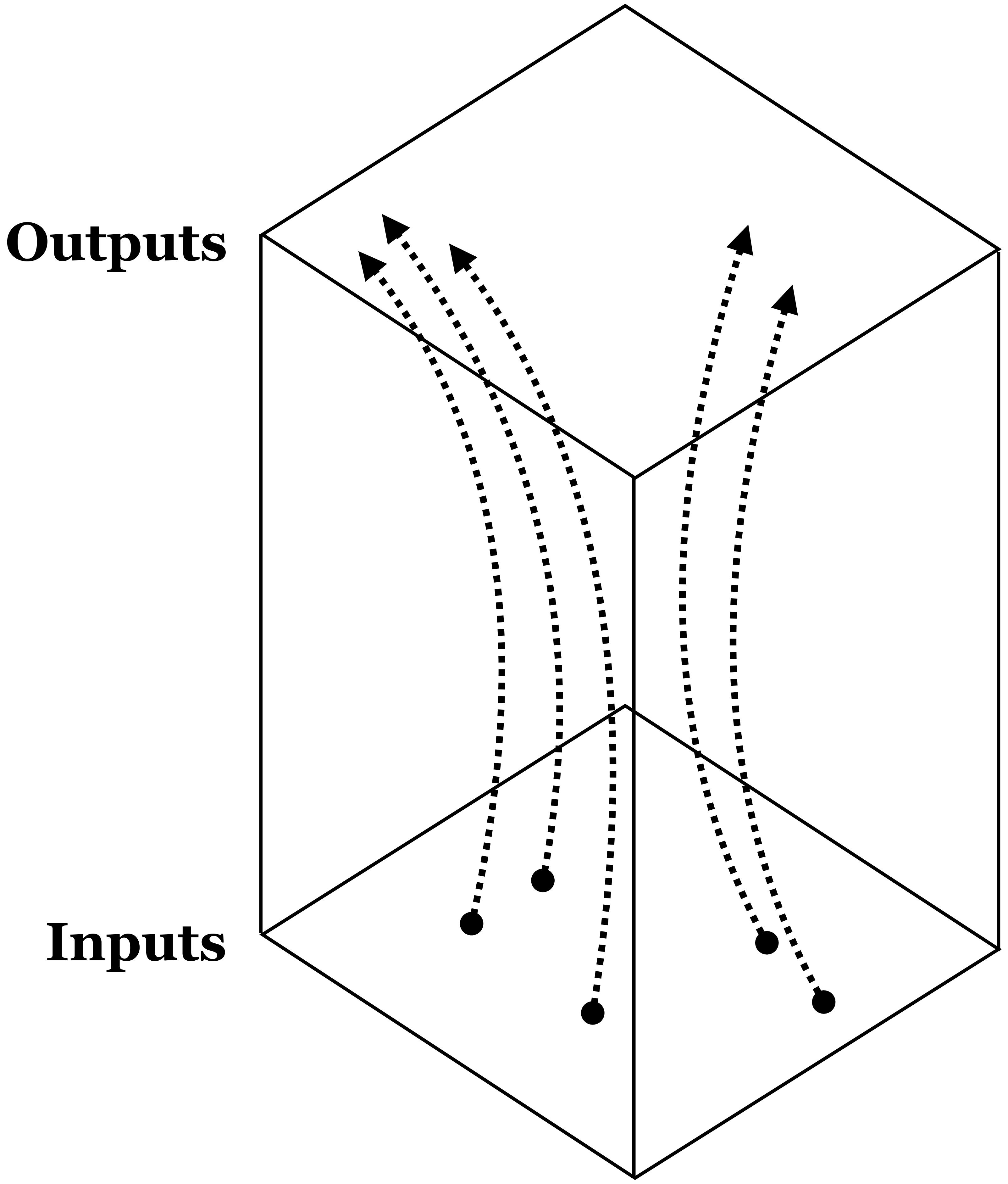
One view: the ***point*** of deep learning is to enable problem solving with little data

1. Knowledge about the mapping
2. Knowledge about the outputs
3. Knowledge about the inputs



Transfer Learning

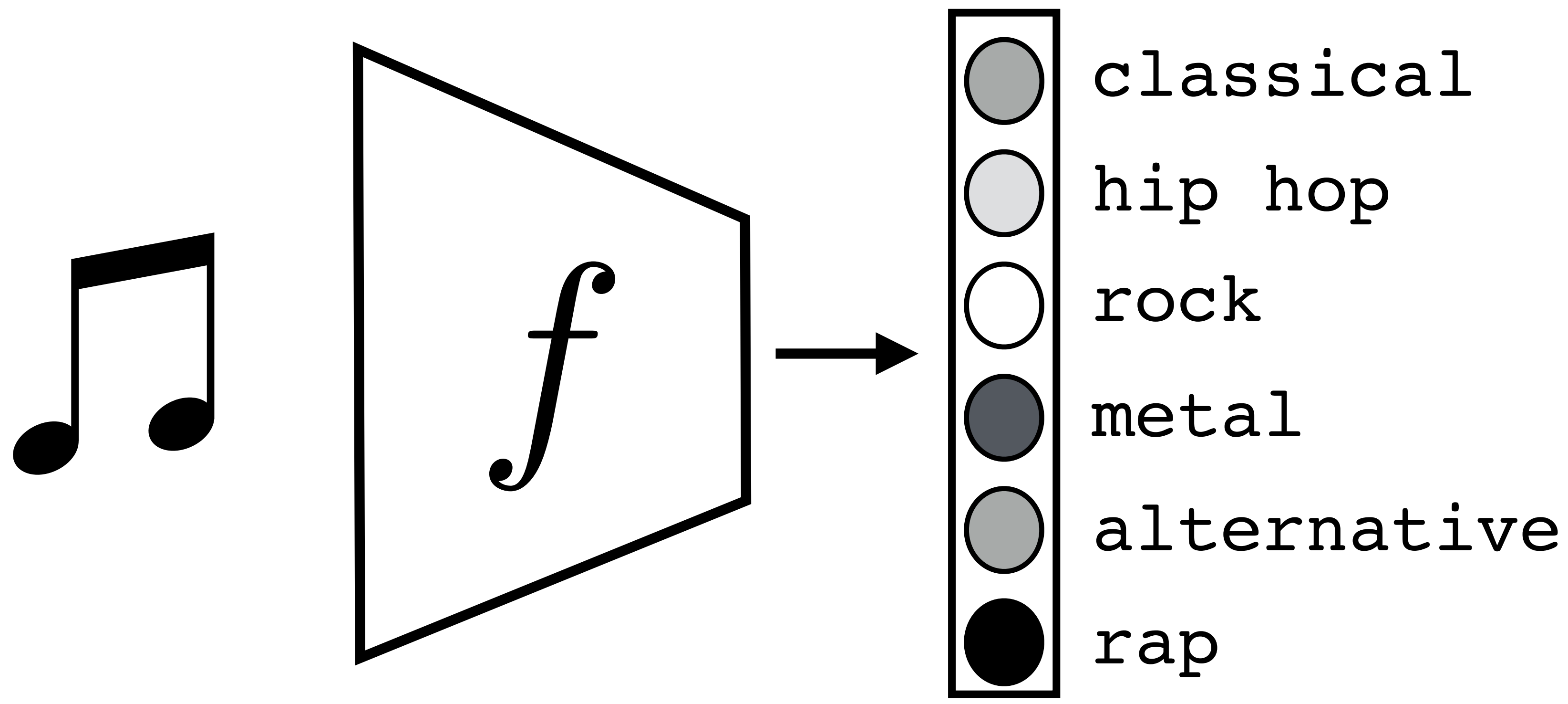
1. **Knowledge about the mapping**
2. Knowledge about the outputs
3. Knowledge about the inputs



Transfer Learning

Pretraining

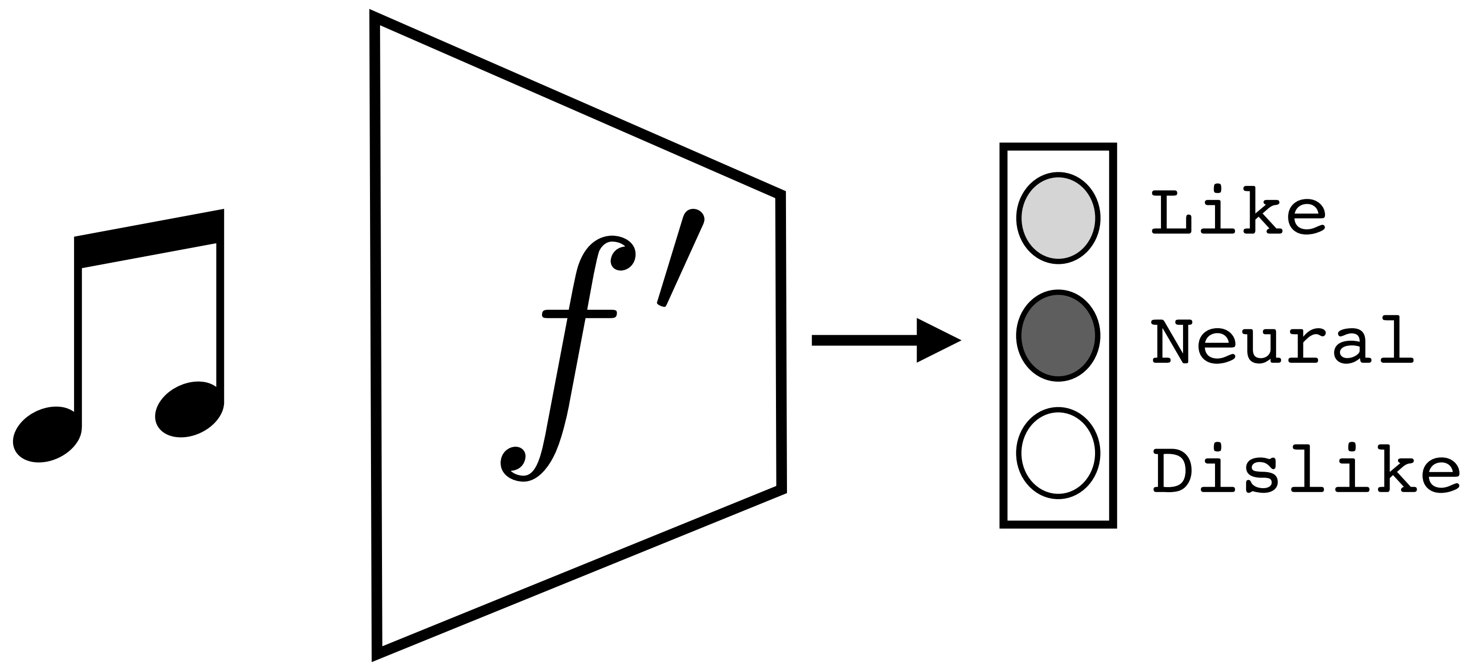
Genre recognition



A lot of data

Adapting

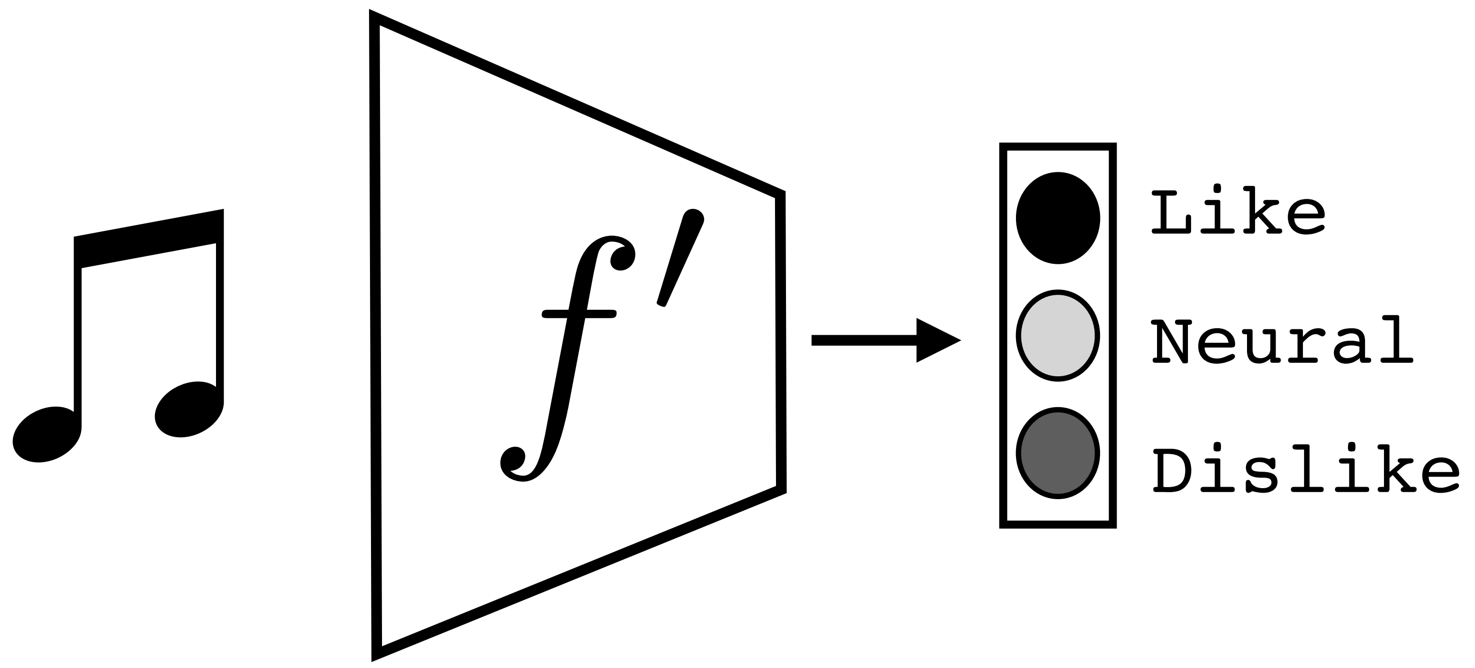
Preference prediction



A little data

Testing

Preference prediction



Finetuning

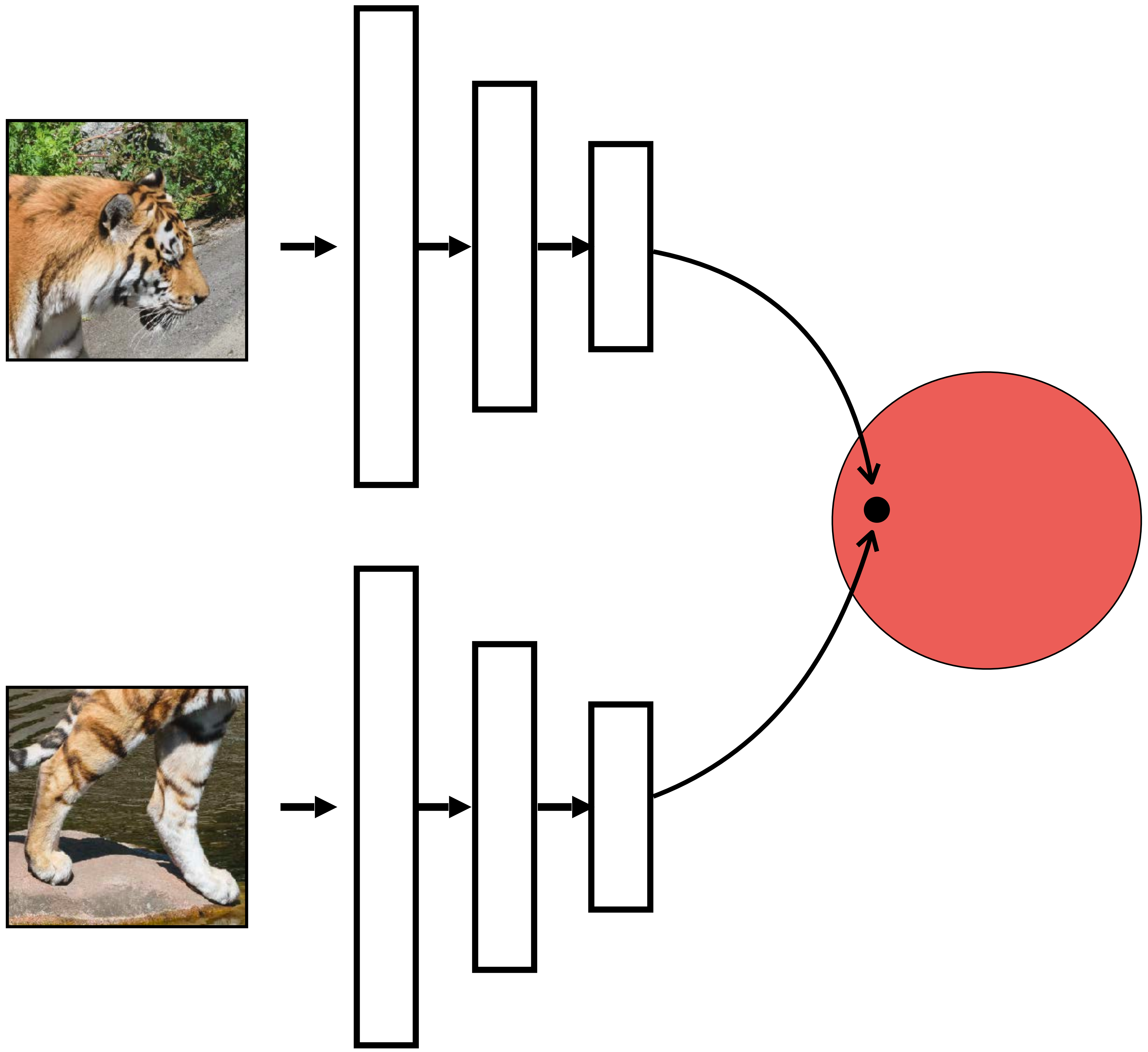
- Pretrain a network on task A, resulting in parameters **W** and **b**
- Initialize a second network with some or all of **W** and **b**
- Train the second network on task B, resulting in parameters **W'** and **b'**

The "learned representation" is the *encoder* (its weights and biases) so that's what we transfer

Contrastive pre-training

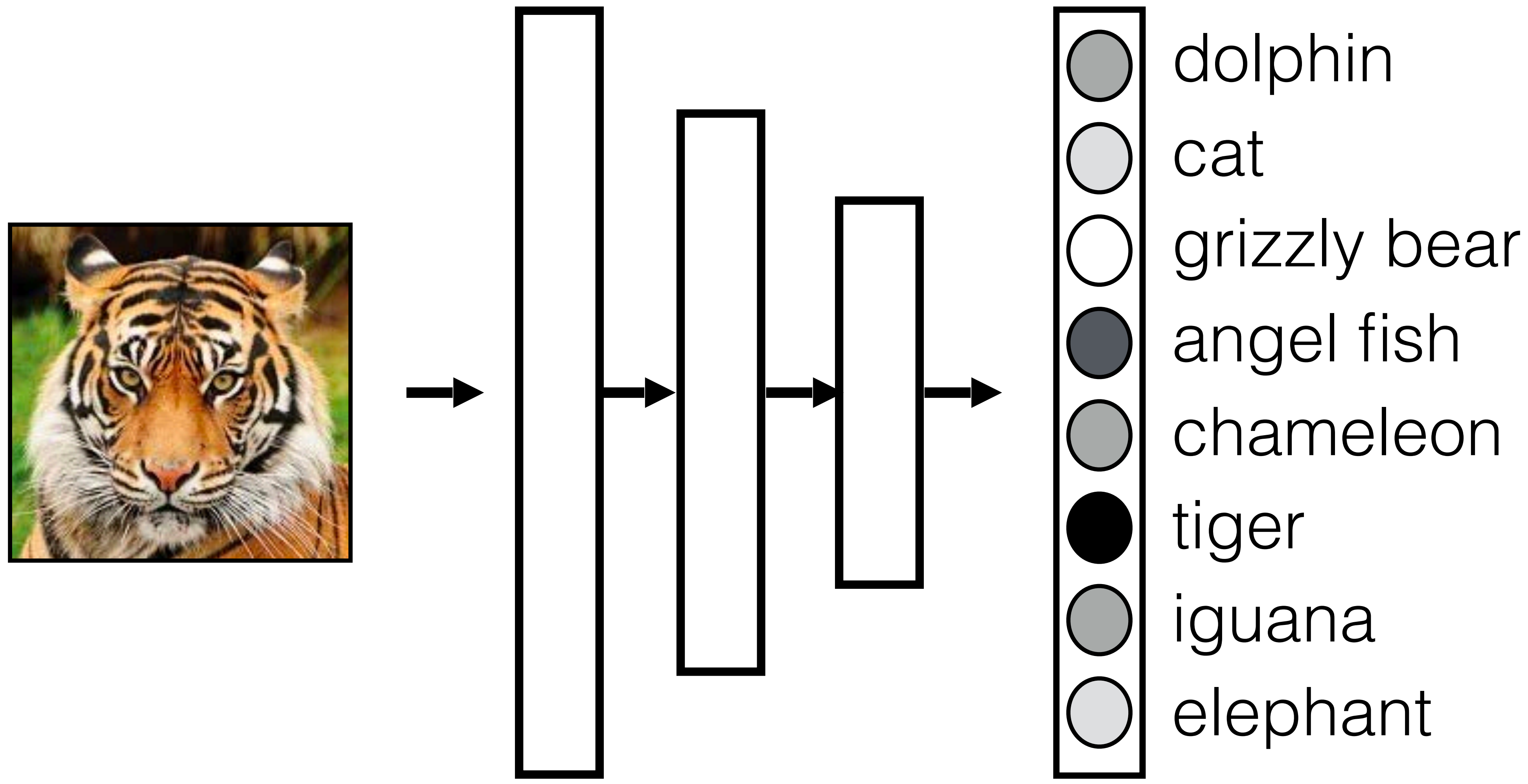
Pretraining

Self-supervised contrastive learning



Testing

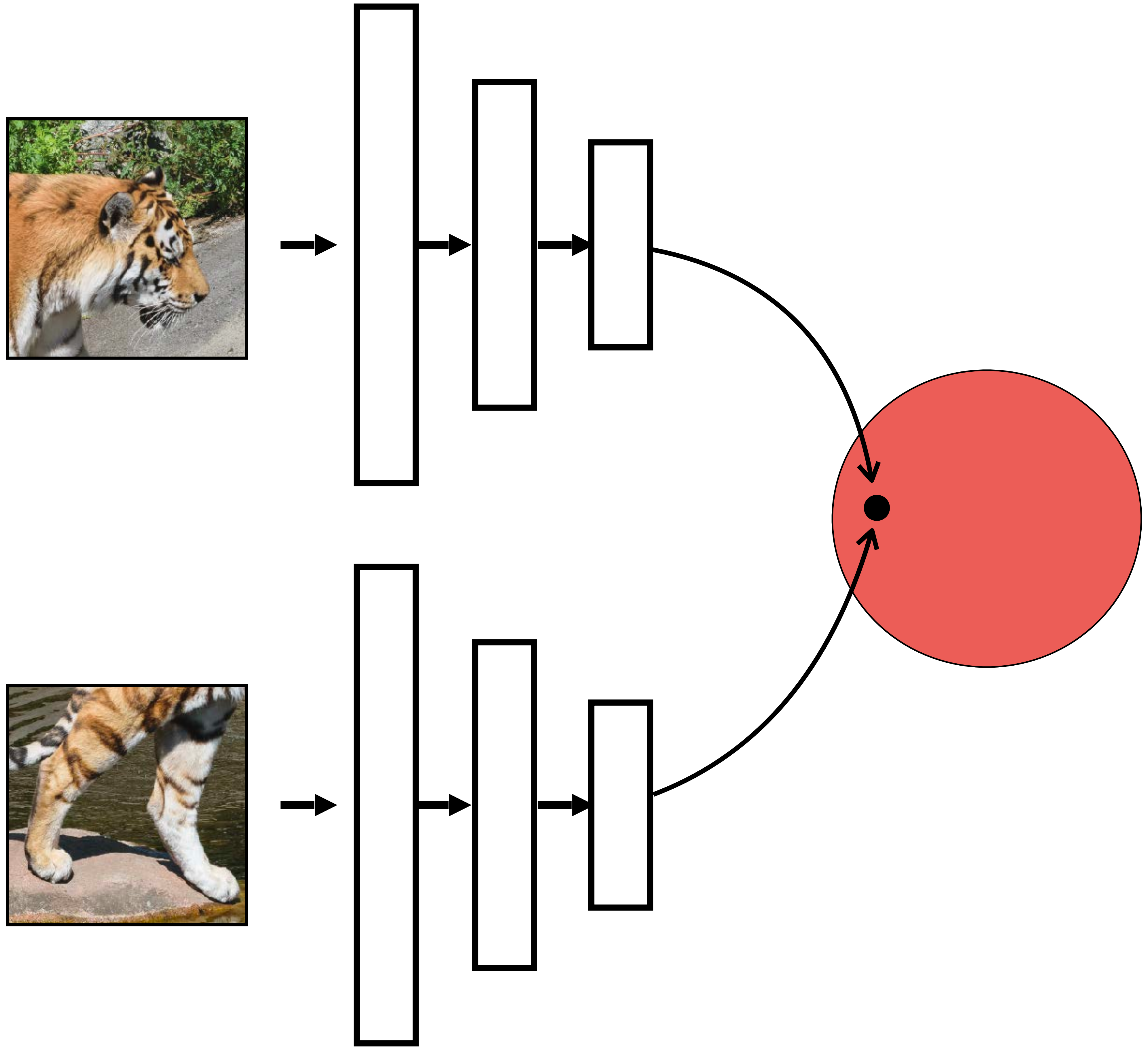
New recognition task



Contrastive pre-training

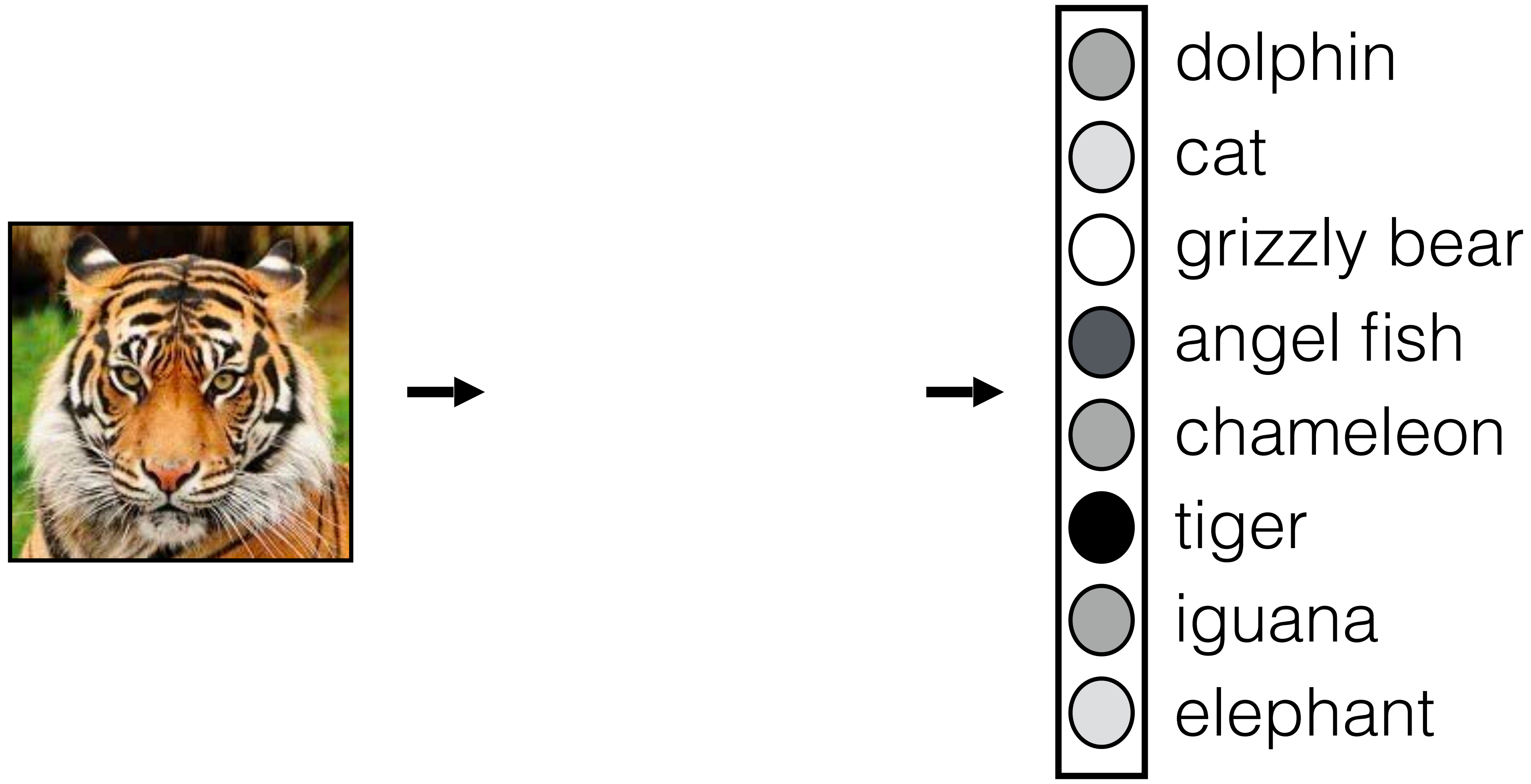
Pretraining

Self-supervised contrastive learning



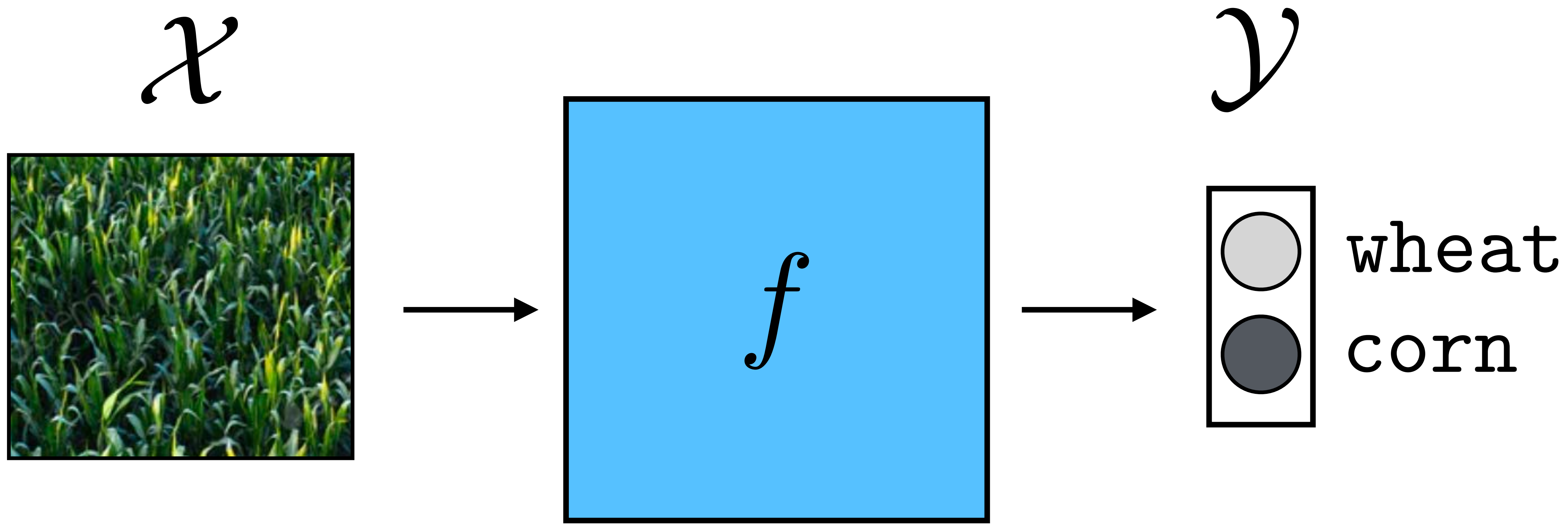
Finetuning

New recognition task

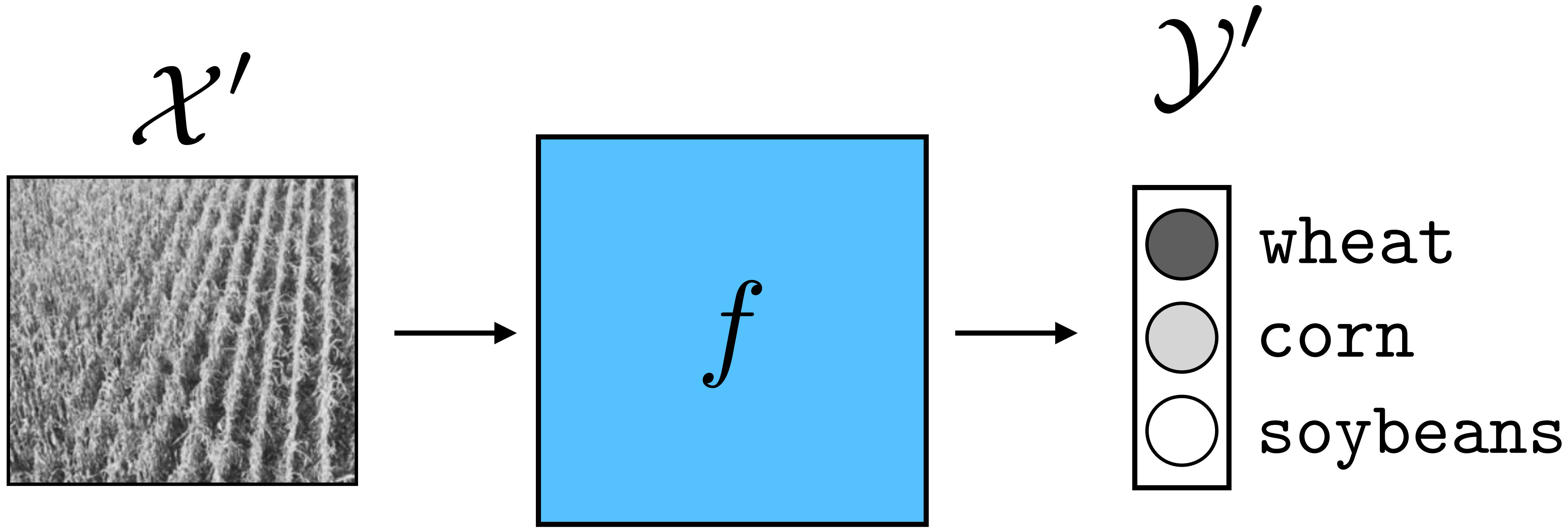


What if the input/output dimensions don't match?

Pretraining

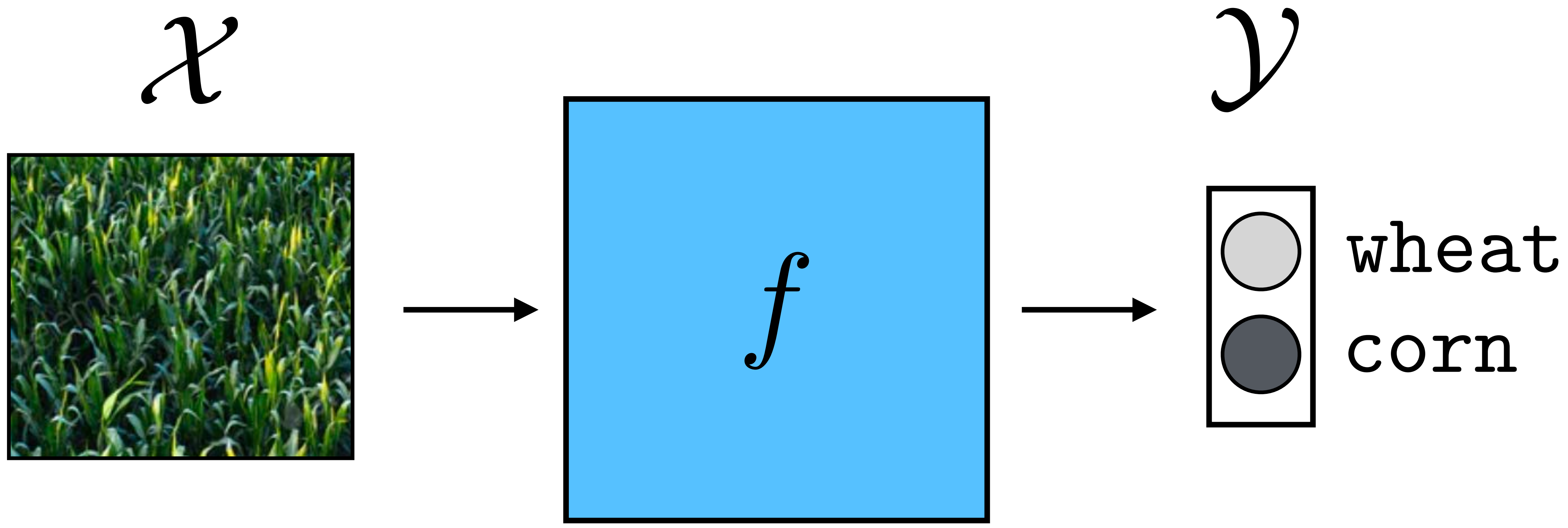


Finetuning

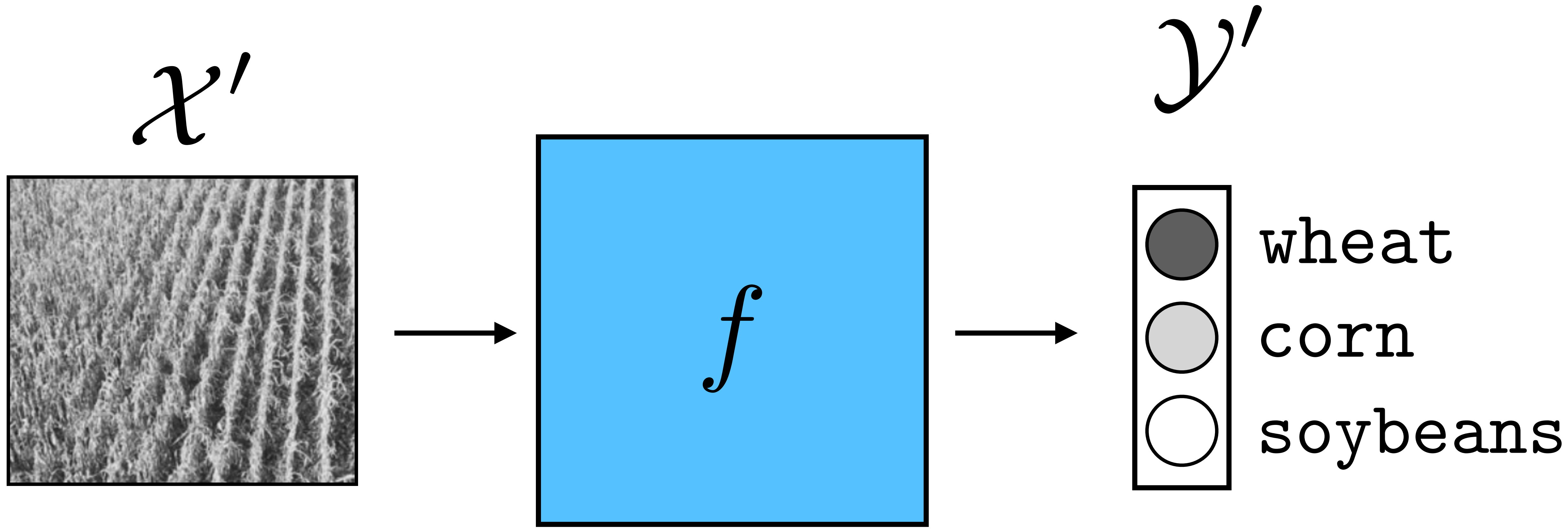


What if the input/output dimensions don't match?

Pretraining

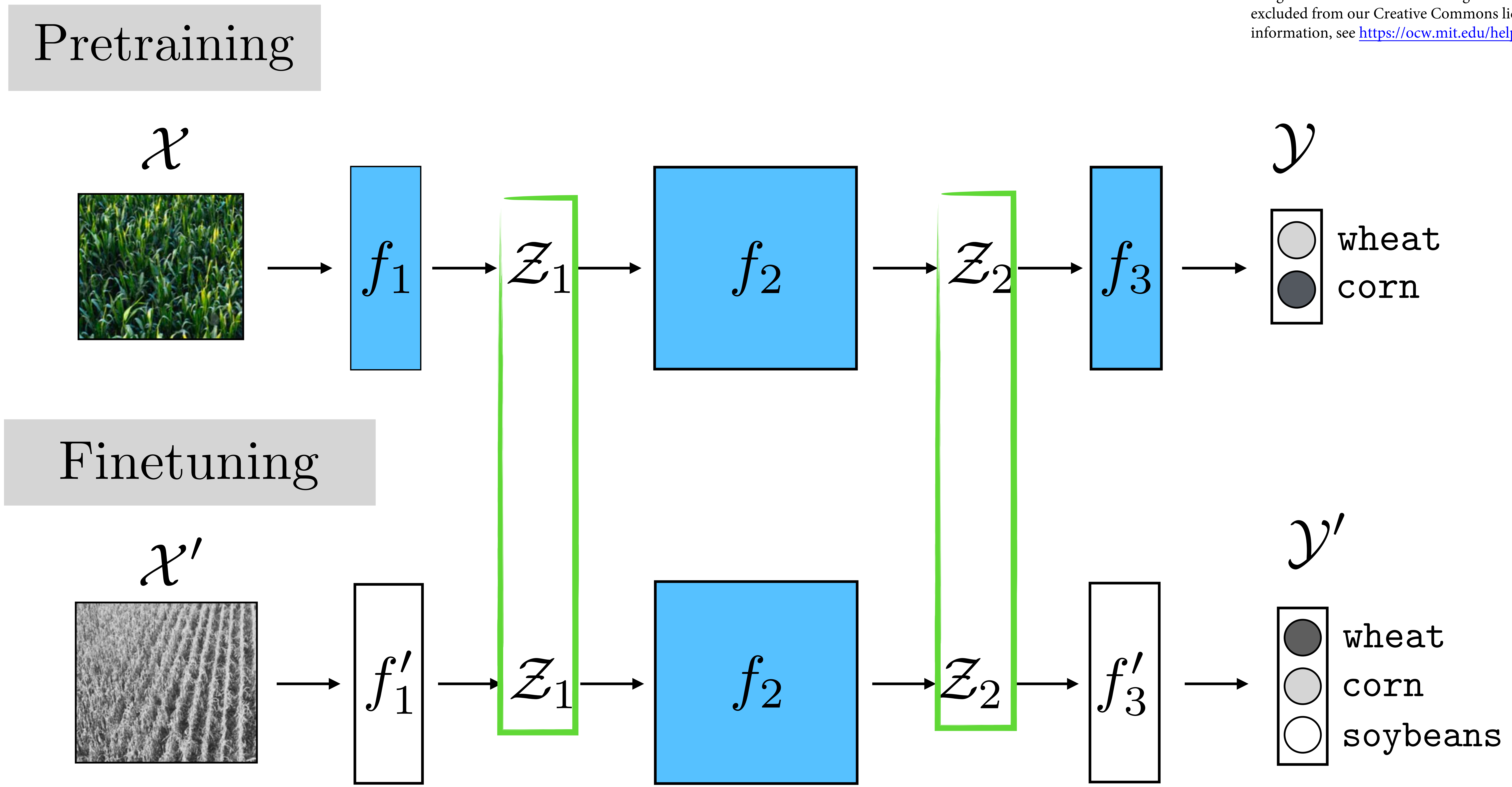


Finetuning



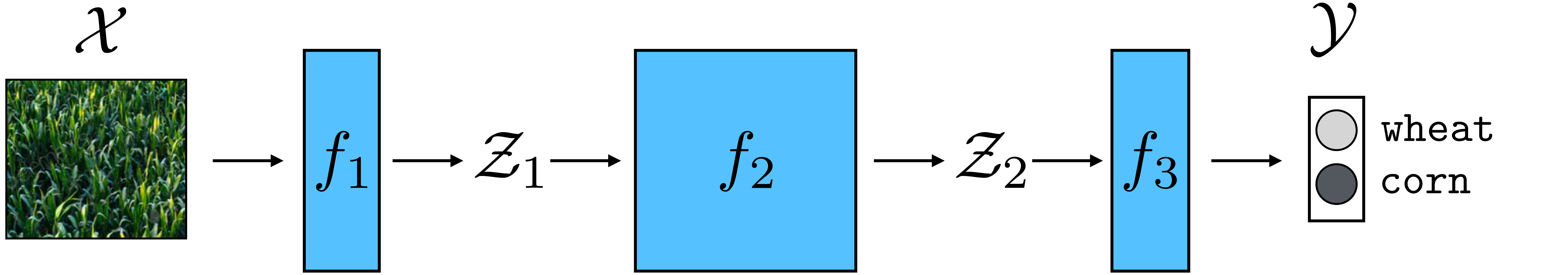
$$\boxed{\begin{array}{c}\mathcal{X}' \neq \mathcal{X} \\ \mathcal{Y}' \neq \mathcal{Y}\end{array}}$$

What if the input/output dimensions don't match?

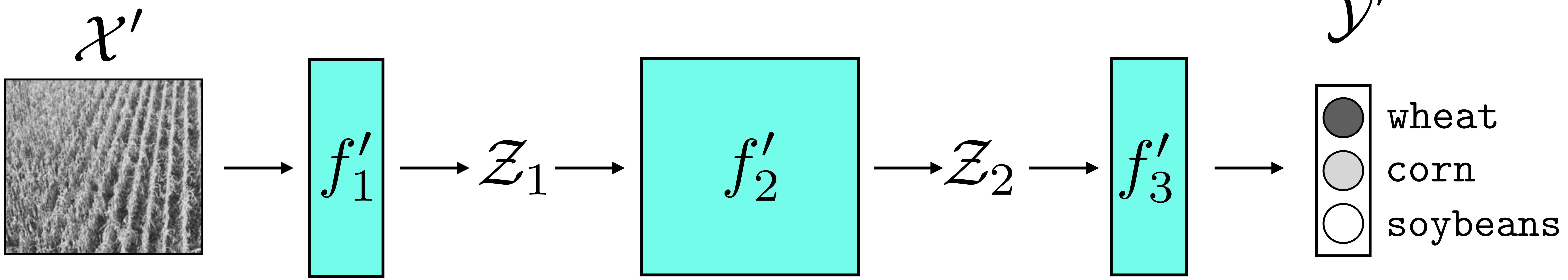


What if the input/output dimensions don't match?

Pretraining



Finetuning

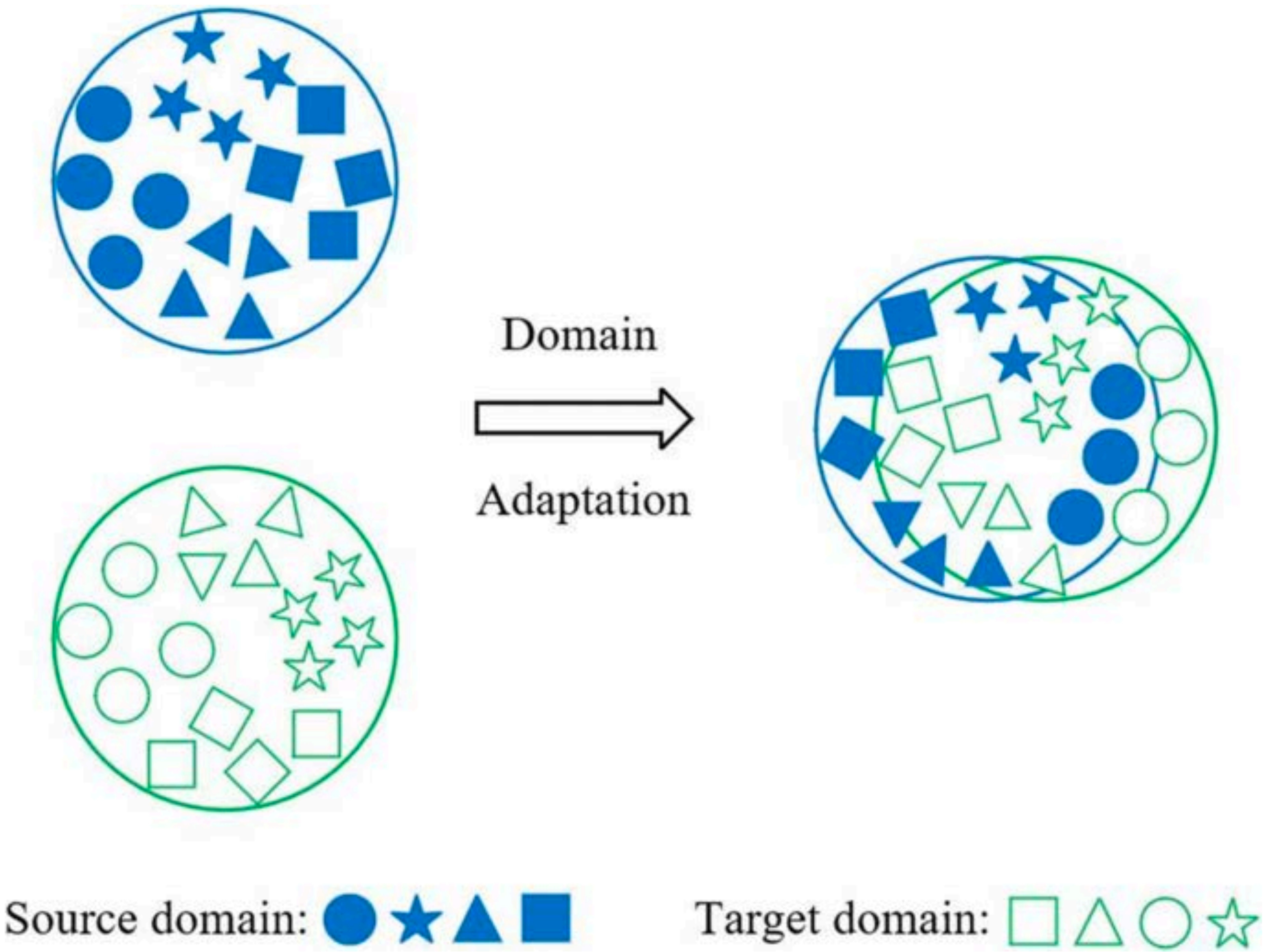


How to avoid forgetting the general representation?

Avoiding overfitting on a small amount of data for a new task while still enabling specialization can be a balancing act. A few methods:

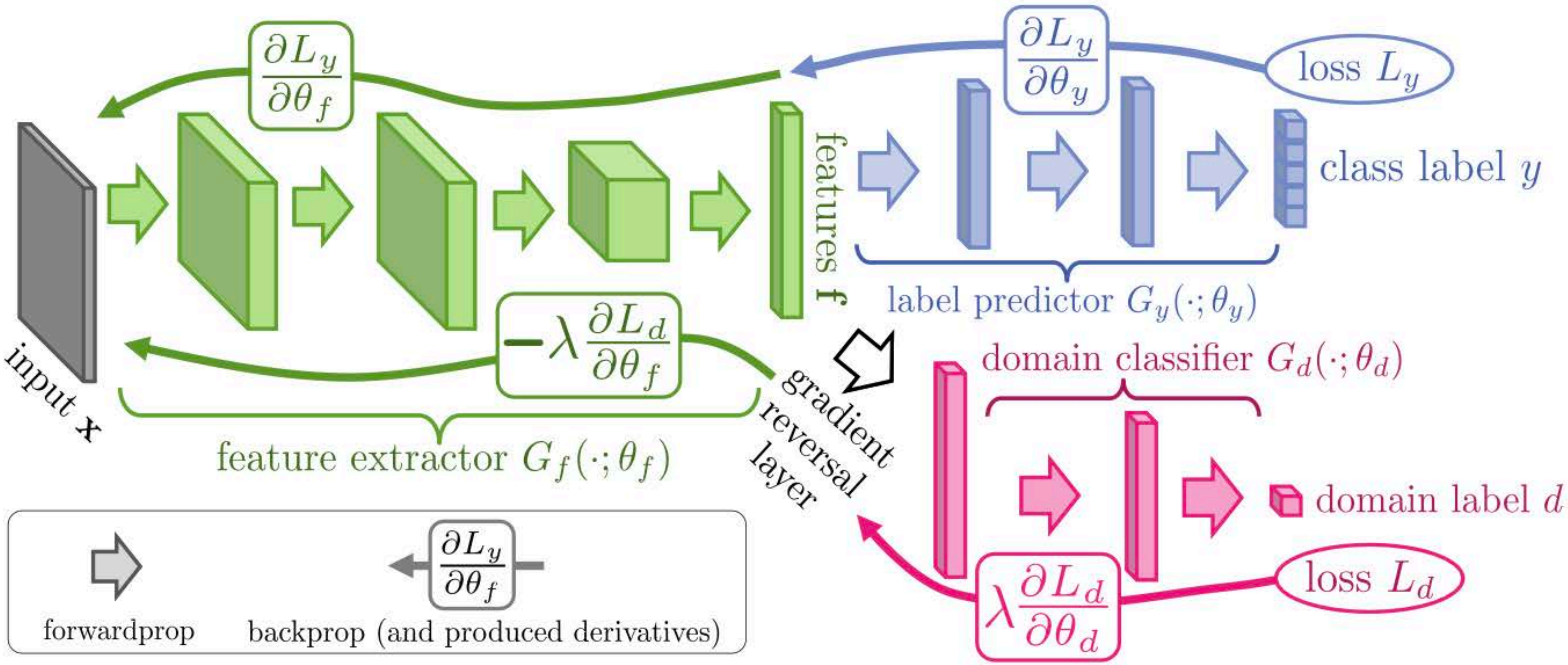
- Freeze most of the network, train only the final layers
- Use a small learning rate
- Early stopping
- Continue to train on the original data as well

All assume that you have a good, representative validation set for your task (not always possible when you may see distribution shift)



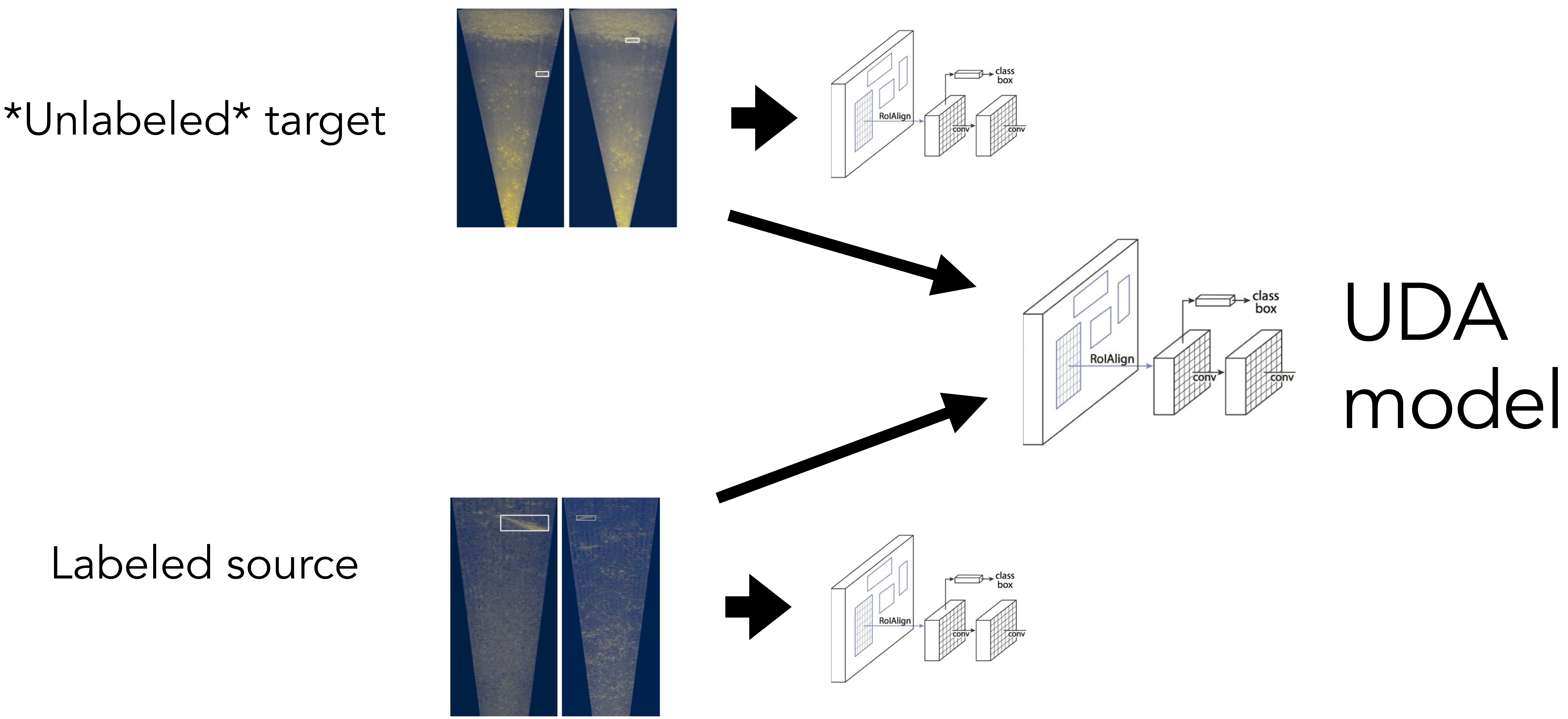
https://link.springer.com/article/10.1007/s10489-022-03709-8

Domain Adversarial Neural Networks (DANN)

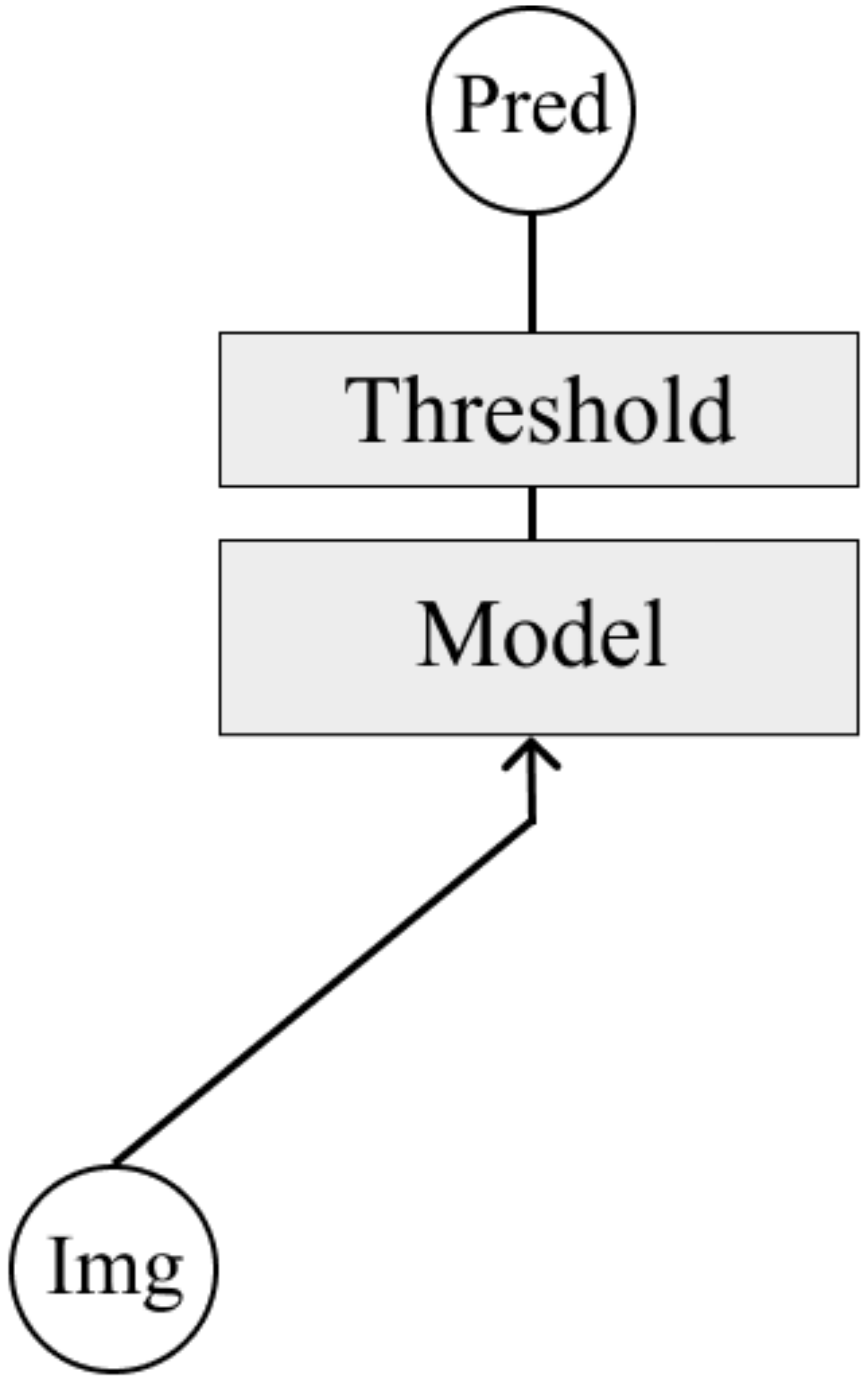


https://arxiv.org/abs/1505.07818

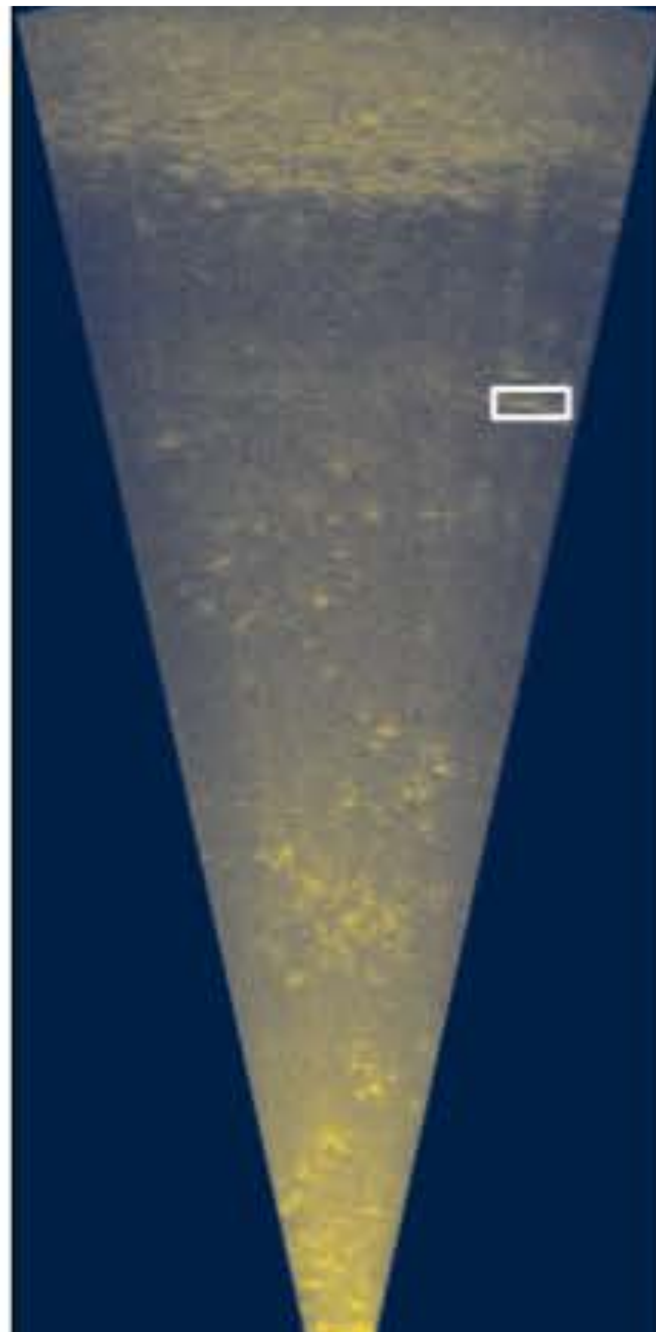
Unsupervised Domain Adaptation (UDA)

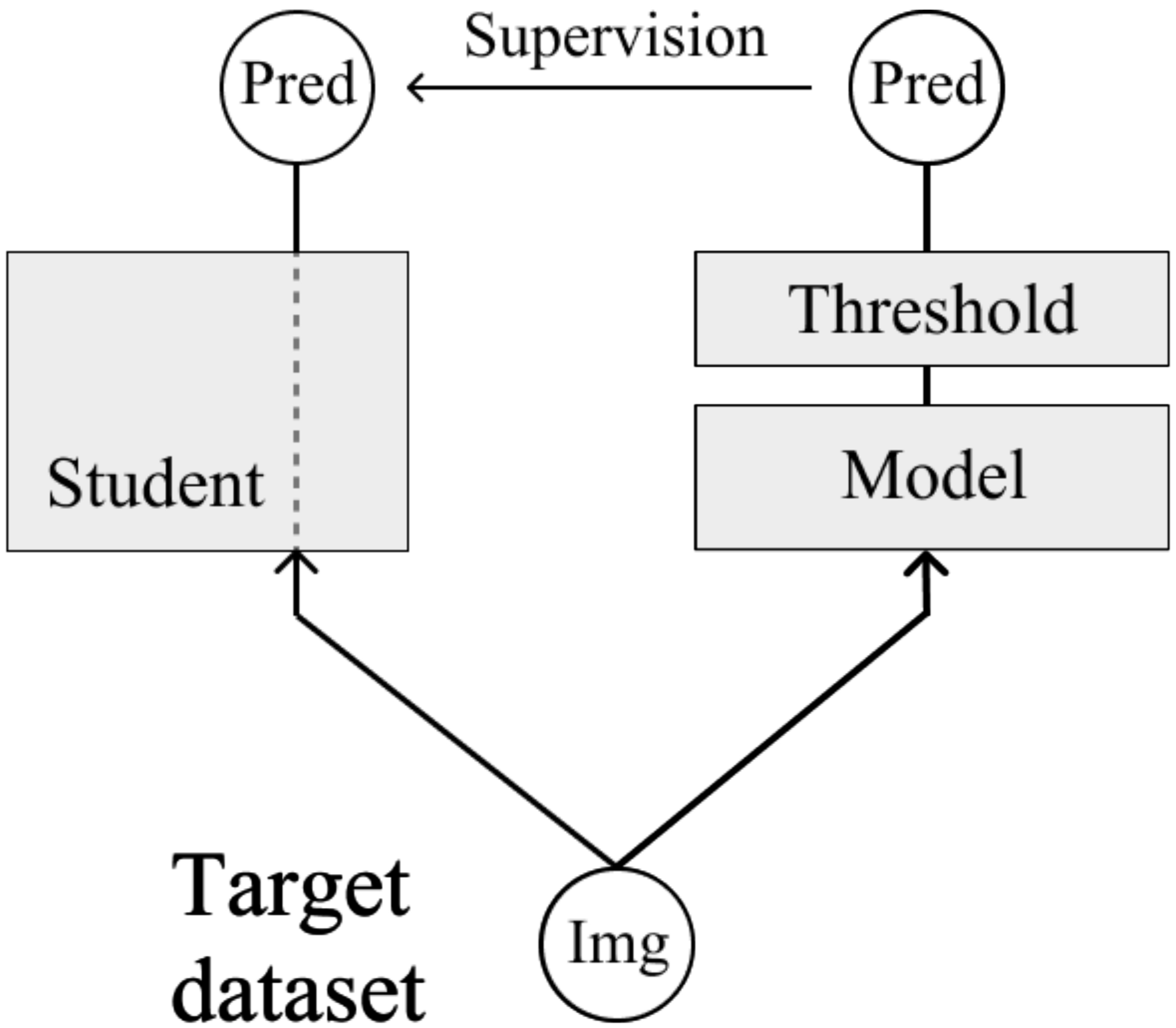


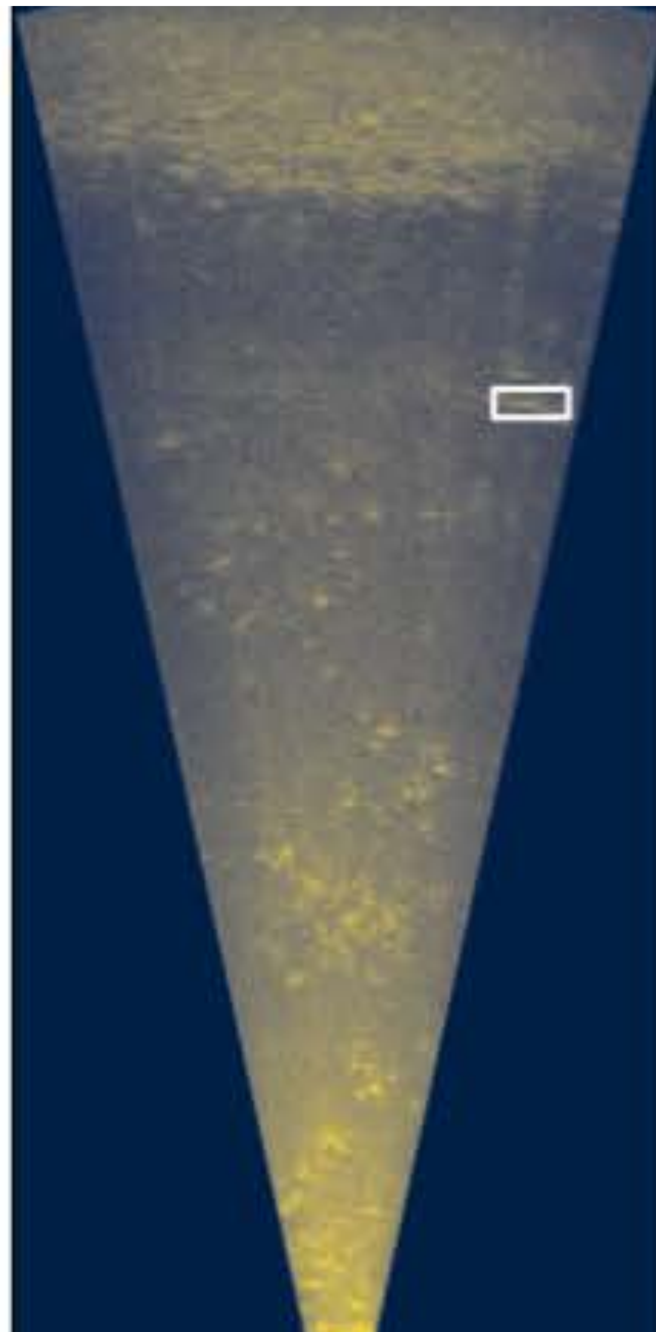
21

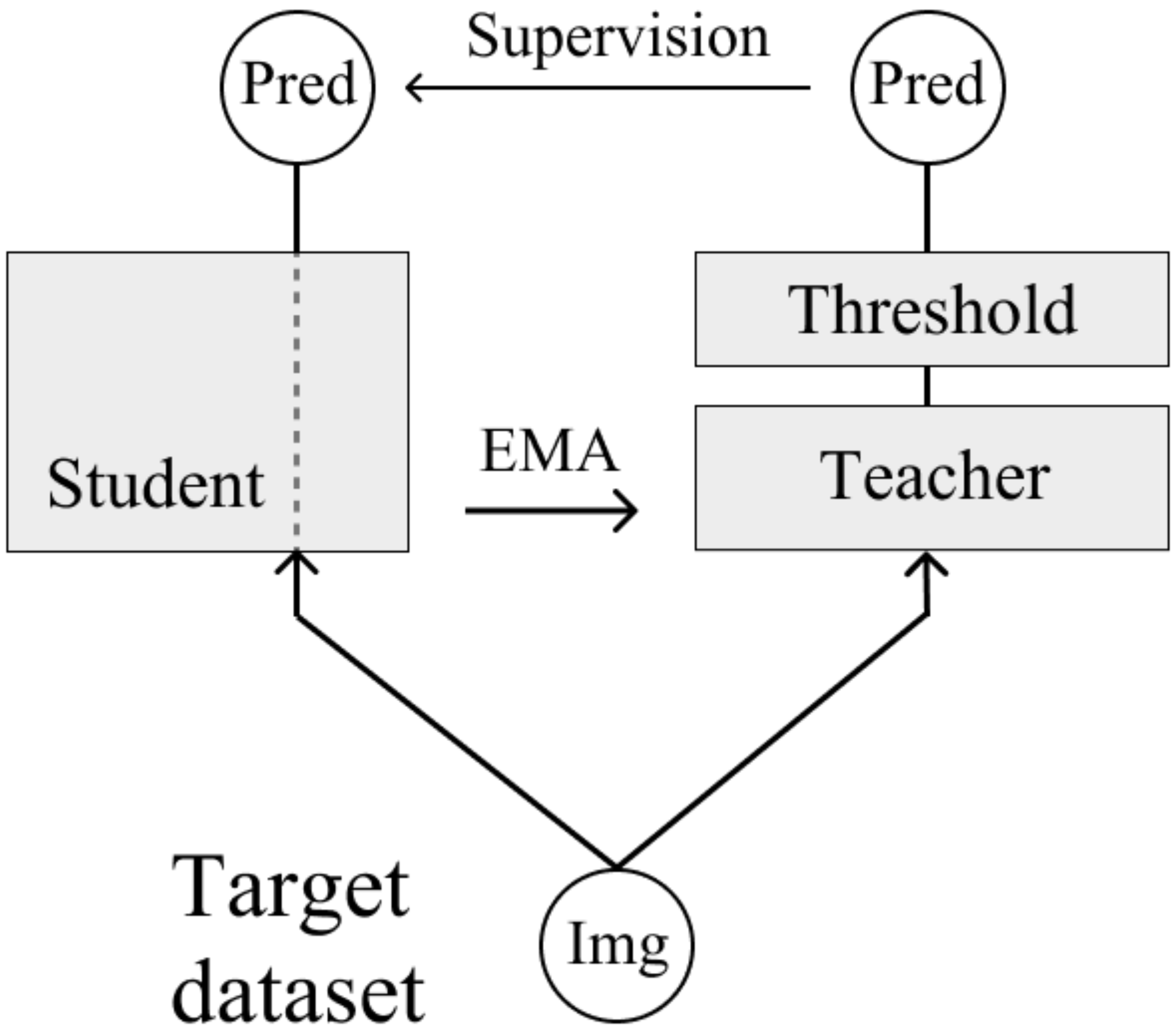


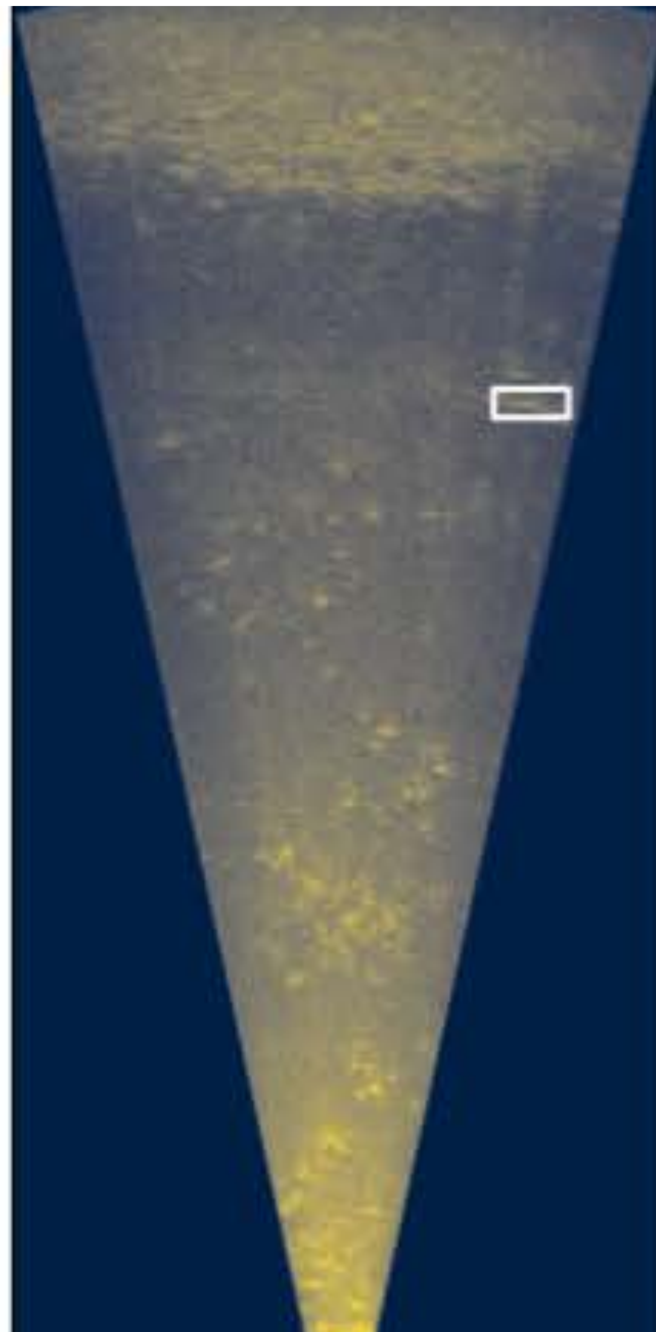
Target
dataset

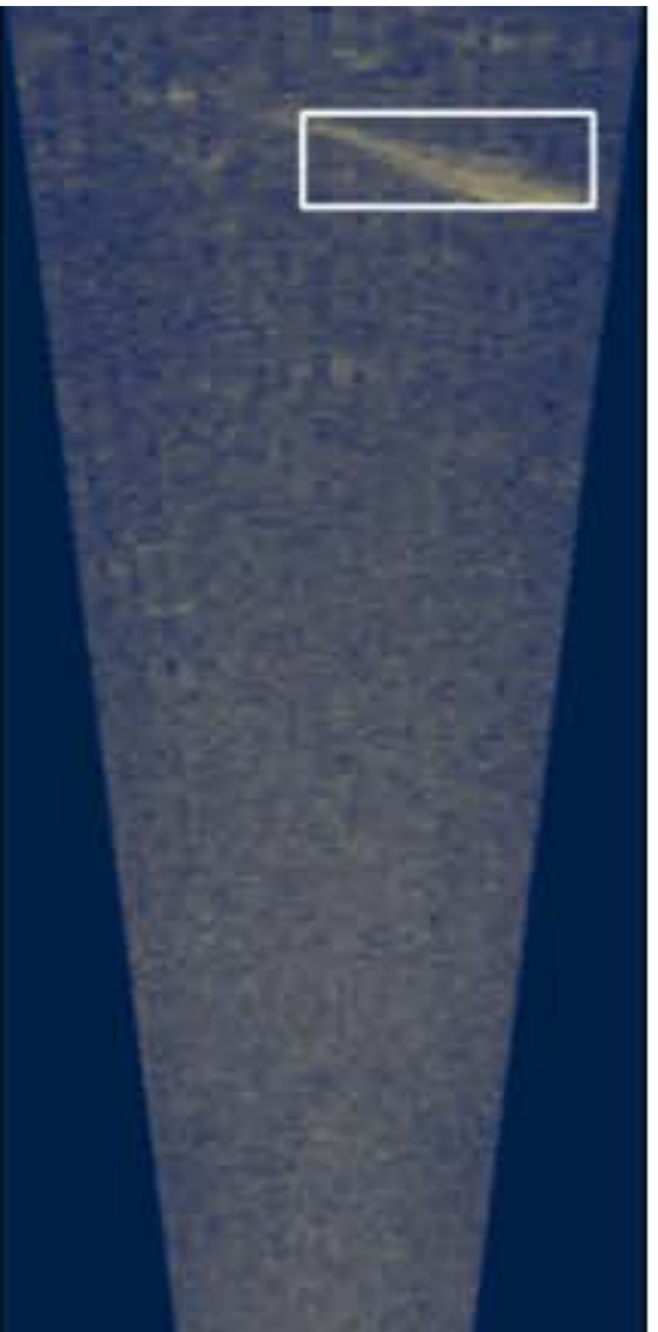


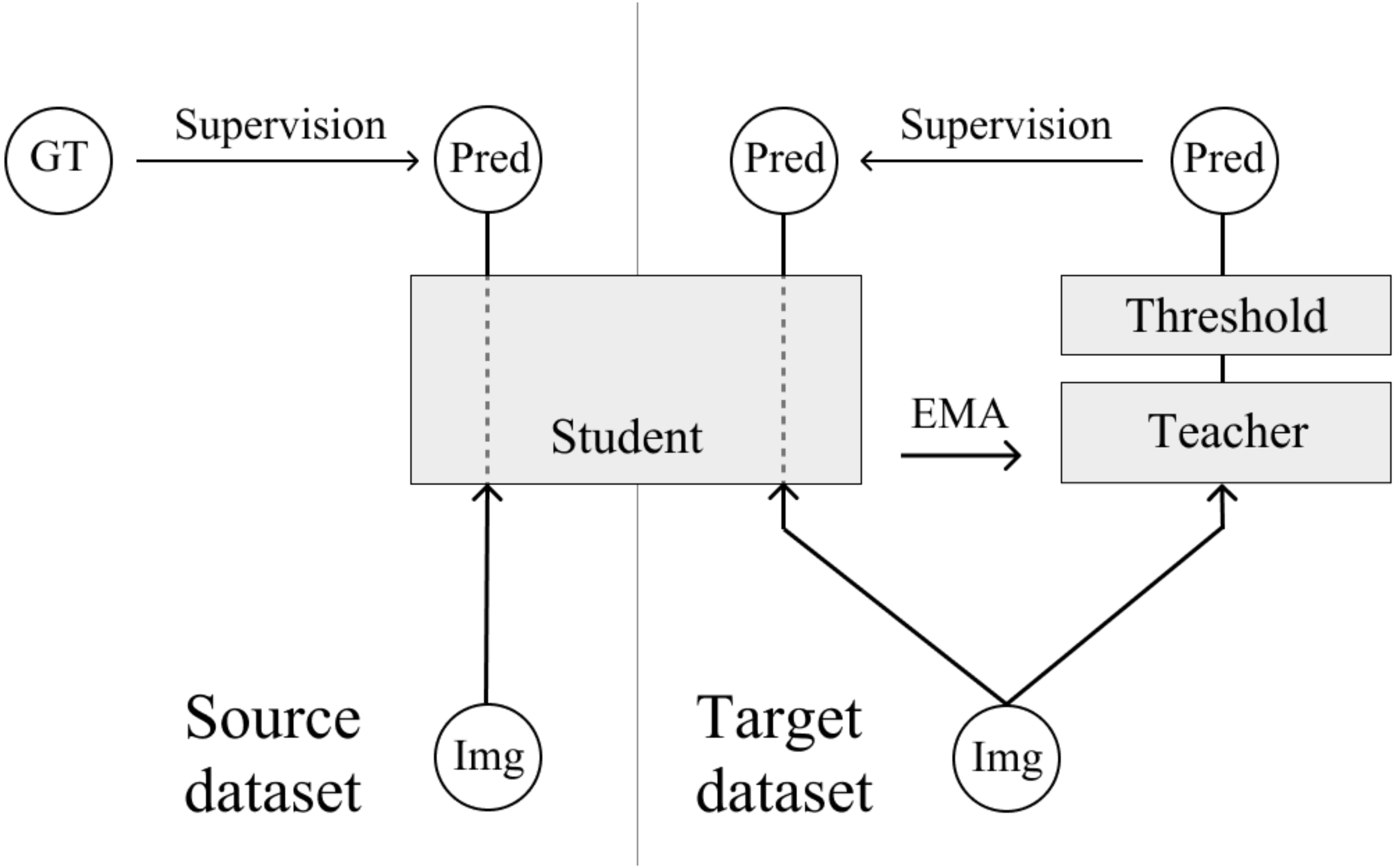

Supervision
Pred
Pred
Student
Threshold
Model
Target
dataset
Img



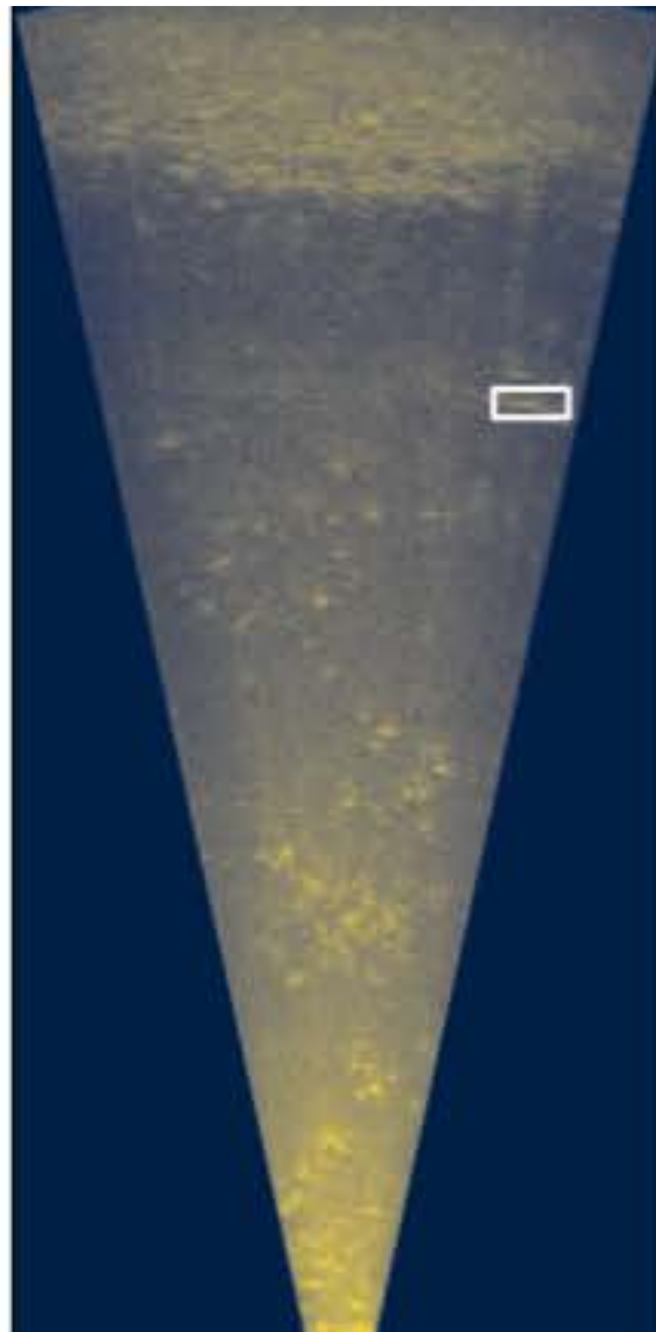

Supervision
Pred
Pred
Threshold
Student
EMA
Teacher
Target dataset
Img

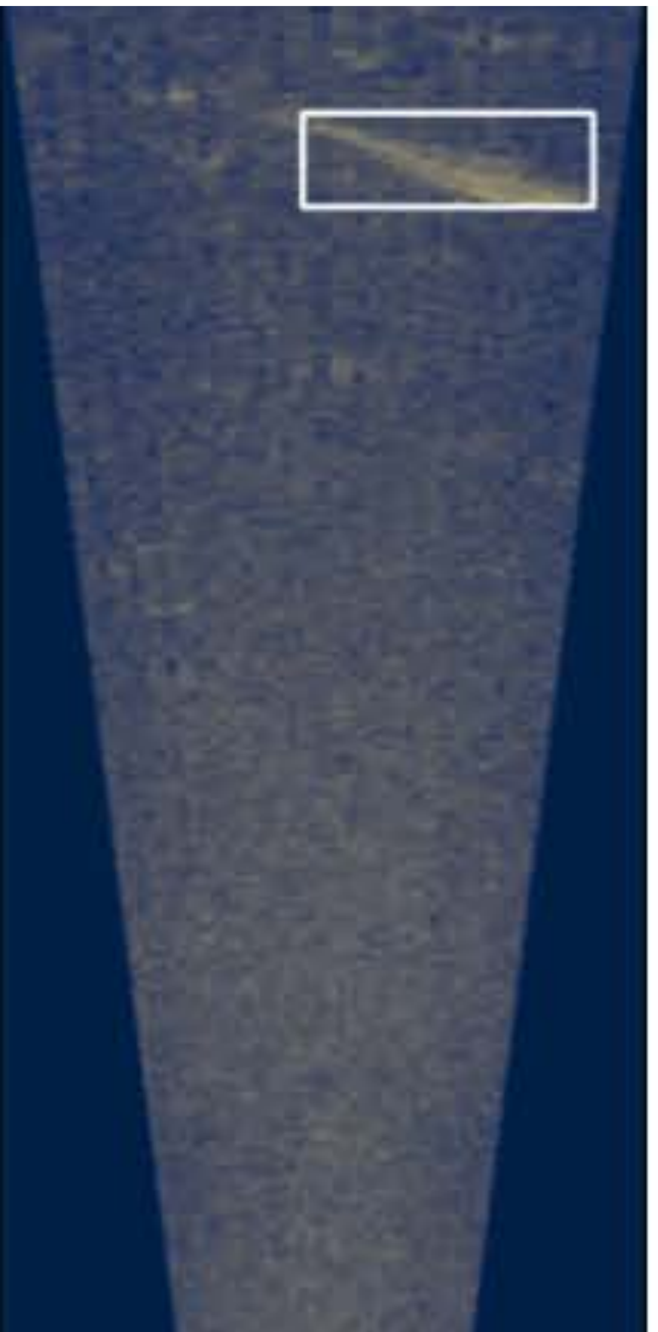


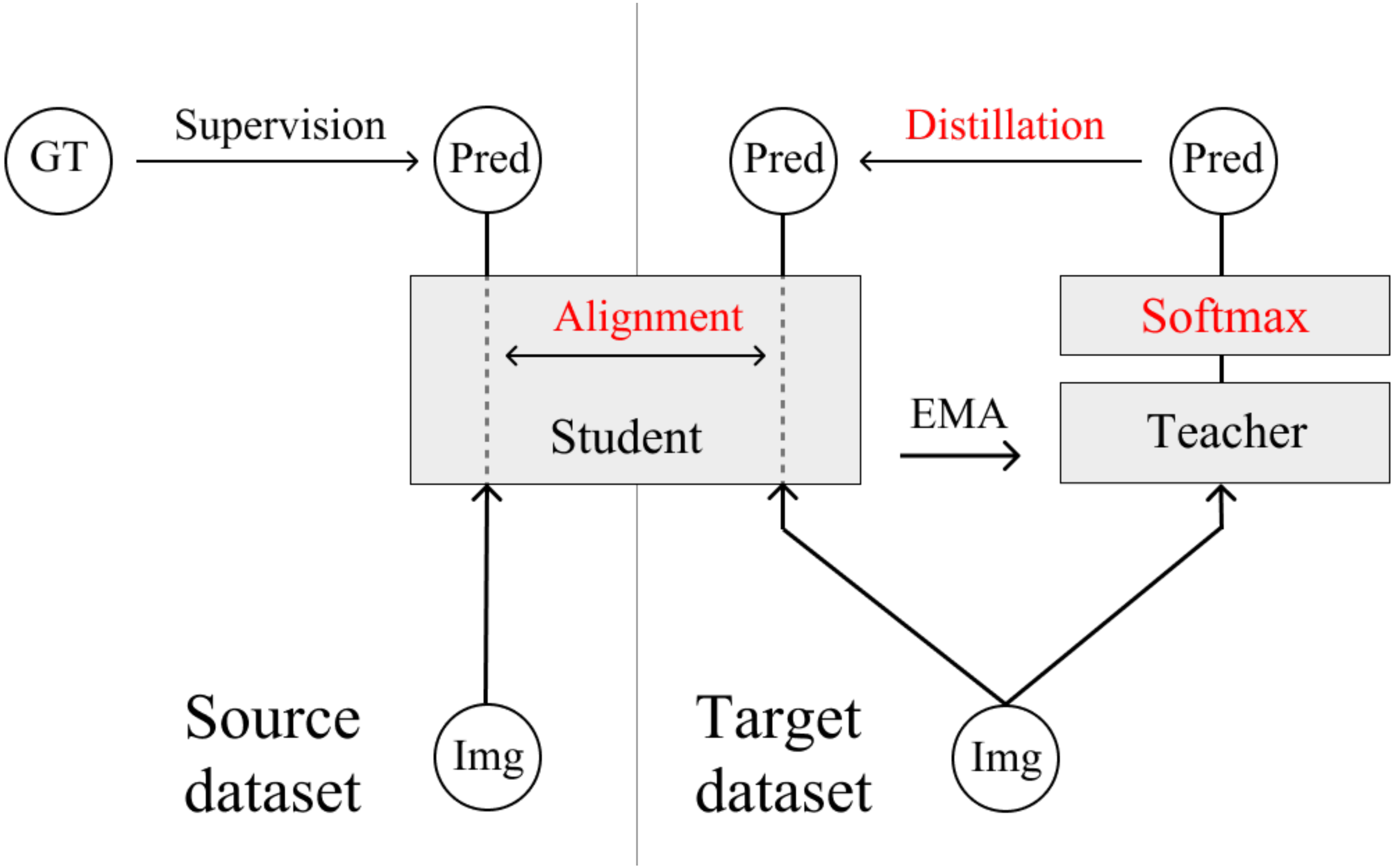


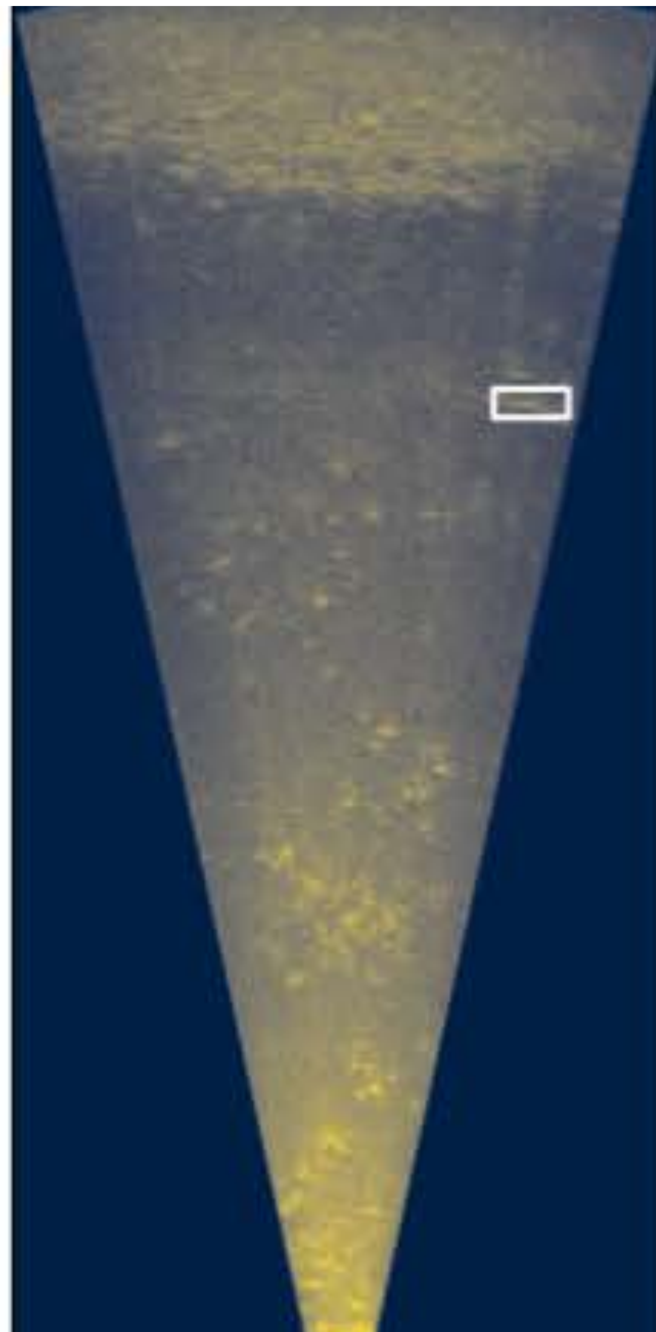


GT
Supervision
Pred
Pred
Supervision
Pred
Threshold
Student
EMA
Teacher
Source dataset
Img
Target dataset
Img

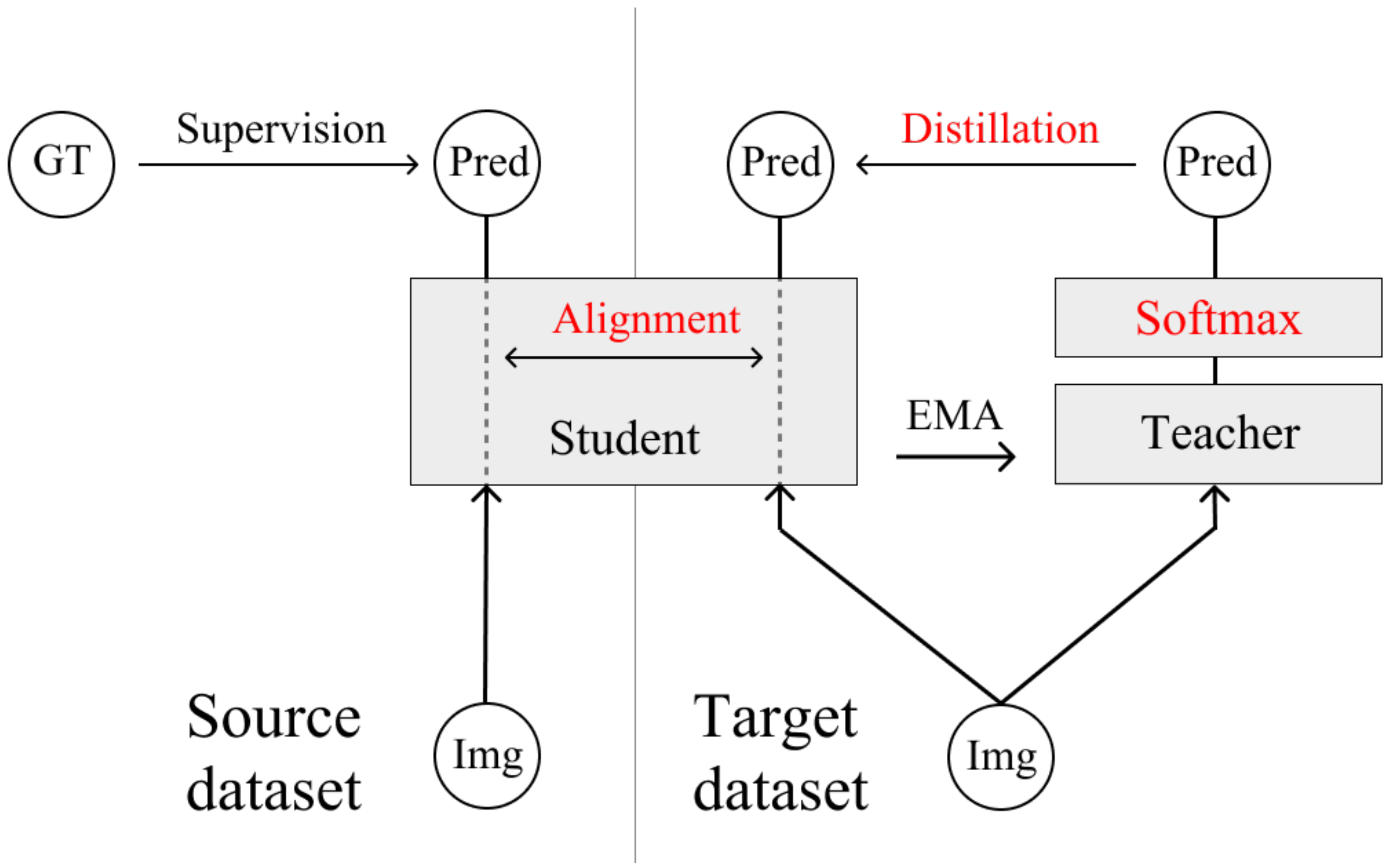


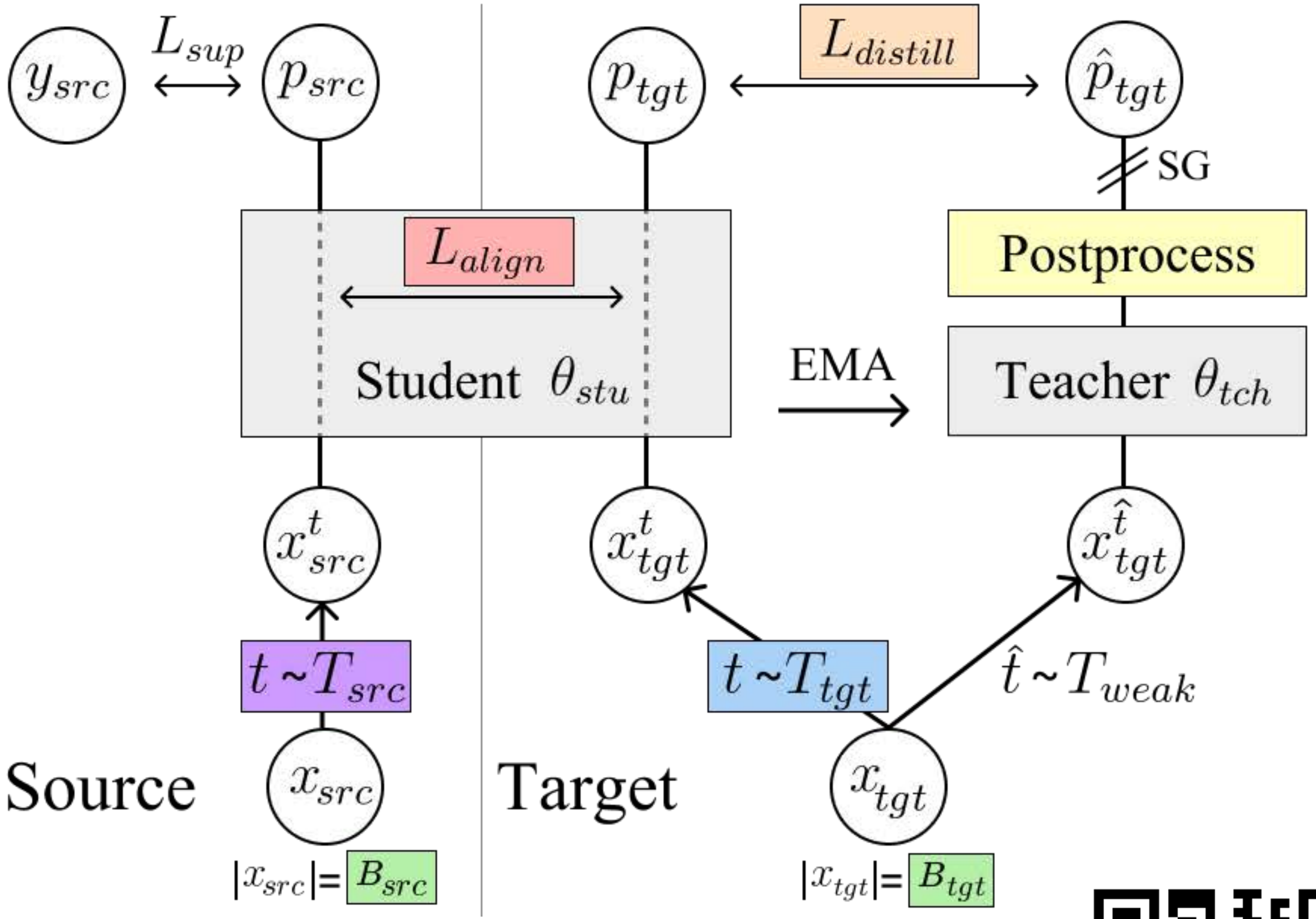



GT
Supervision
Pred
Pred
Distillation
Pred
Alignment
Student
Softmax
EMA
Teacher
Source dataset
Img
Target dataset
Img

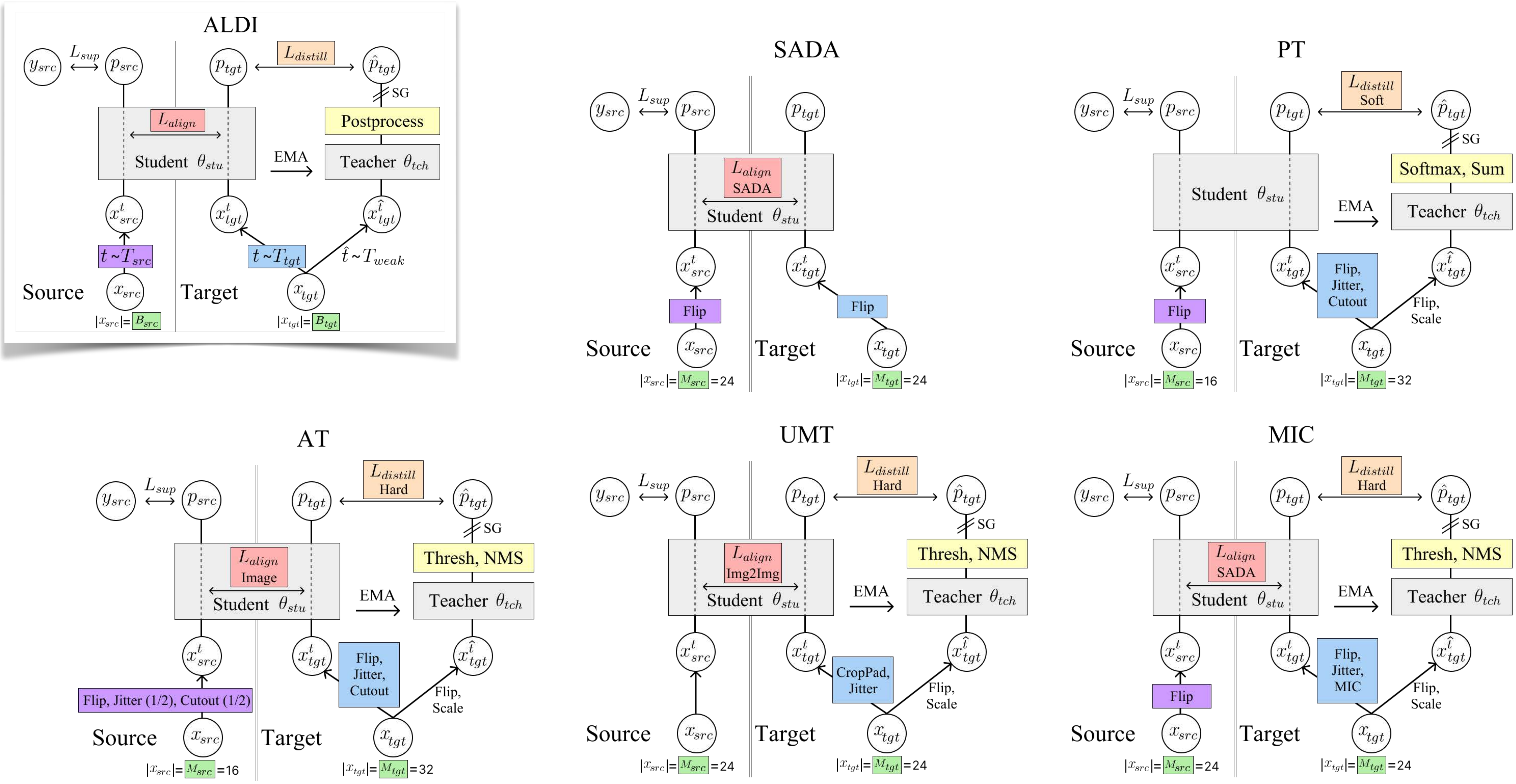


Align and Distill (ALDI)

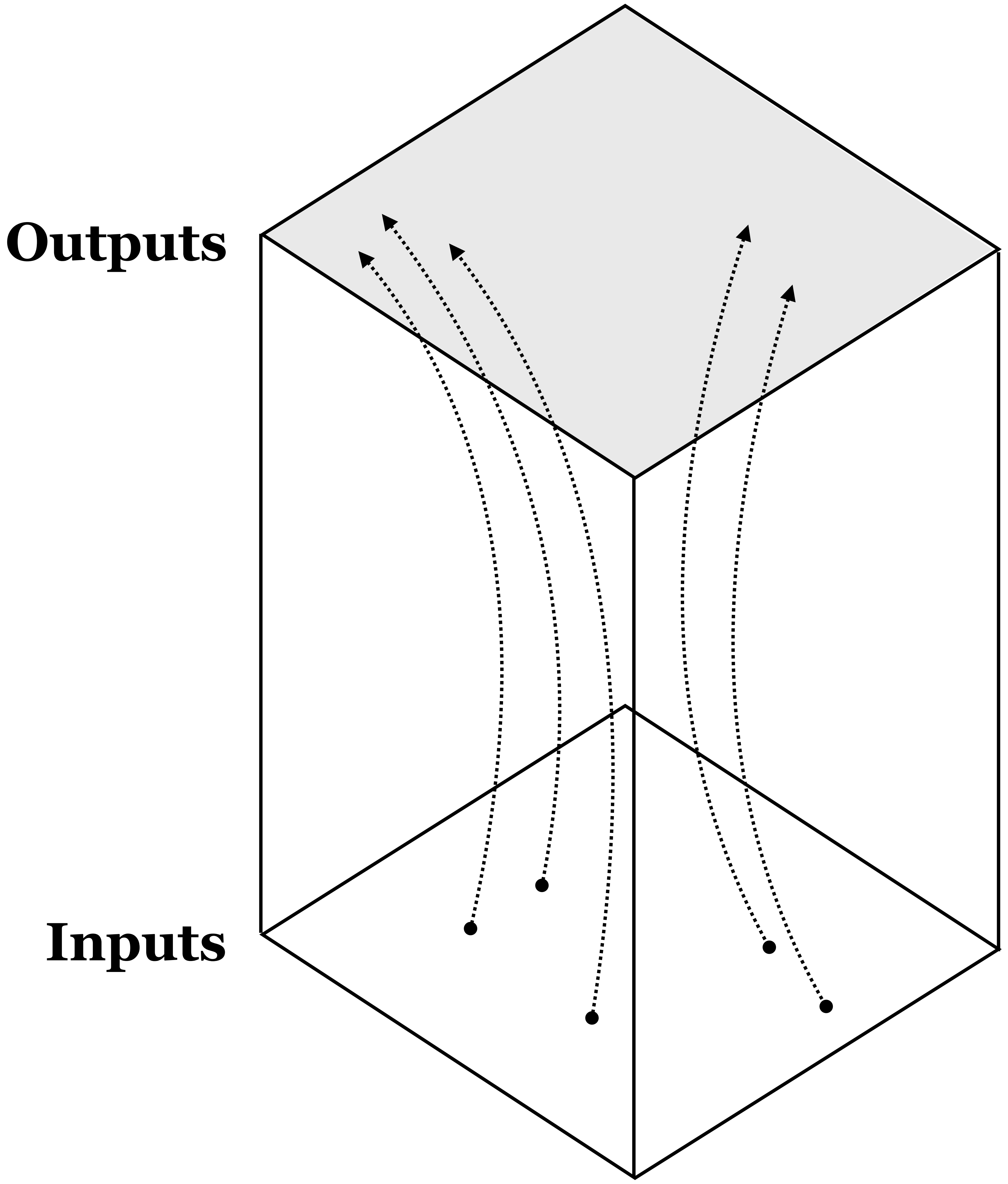




aldi-daod.github.io

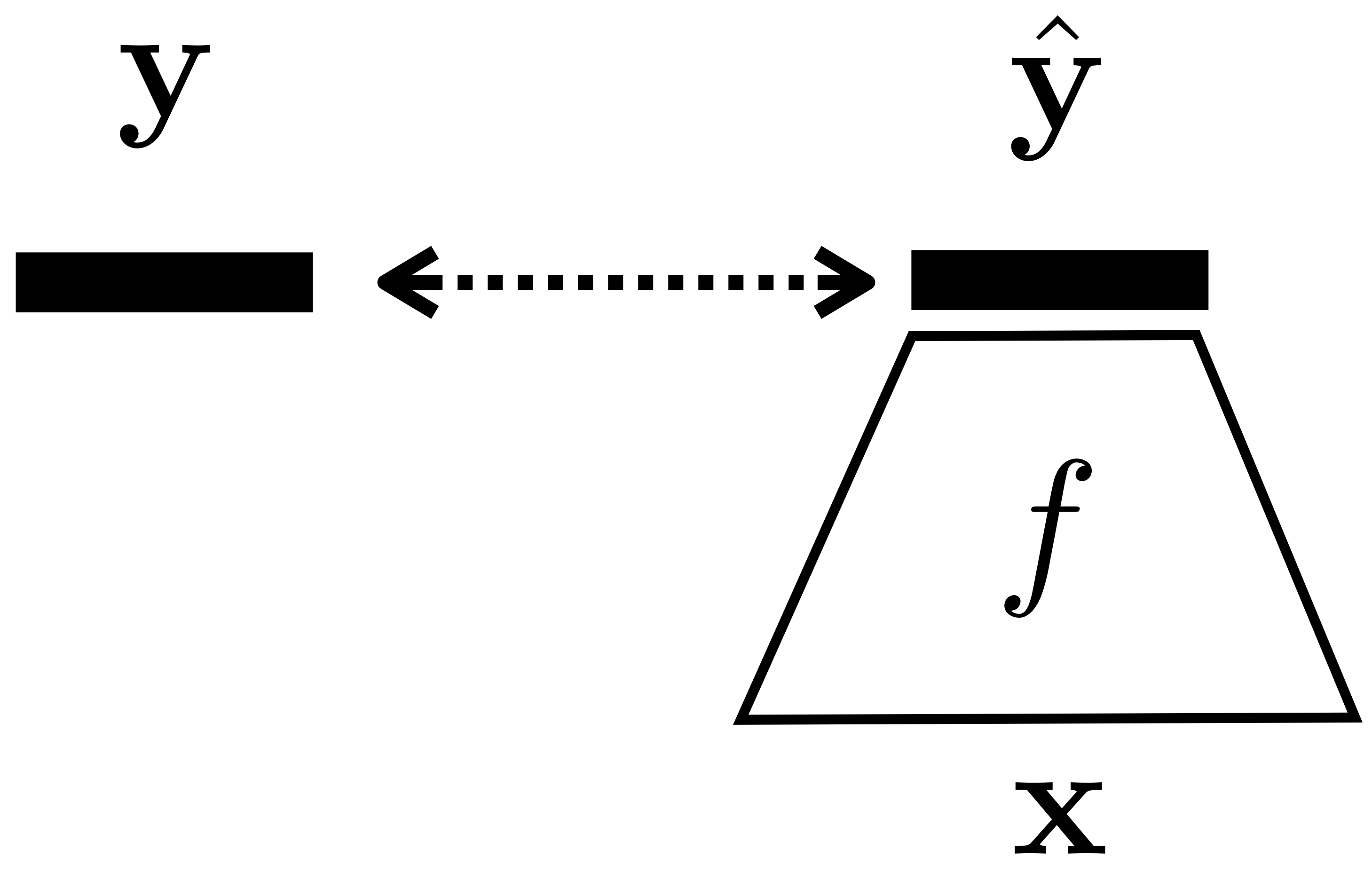

ALDI
SADA
PT
AT
UMT
MIC
L_{sup}
$L_{distill}$
L_{align}
Student θ_{stu}
Teacher θ_{tch}
EMA
SG
Postprocess
$t \sim T_{src}$
$t \sim T_{tgt}$
$\hat{t} \sim T_{weak}$
Source
Target
$|x_{src}|=B_{src}$
$|x_{tgt}|=B_{tgt}$
SADA
Flip
$|x_{src}|=M_{src}=24$
$|x_{tgt}|=M_{tgt}=24$
Soft
Softmax, Sum
Flip, Jitter, Cutout
Flip, Scale
$|x_{src}|=M_{src}=16$
$|x_{tgt}|=M_{tgt}=32$
Hard
Image
Thresh, NMS
Flip, Jitter (1/2), Cutout (1/2)
Img2Img
CropPad, Jitter
Flip, Jitter, MIC

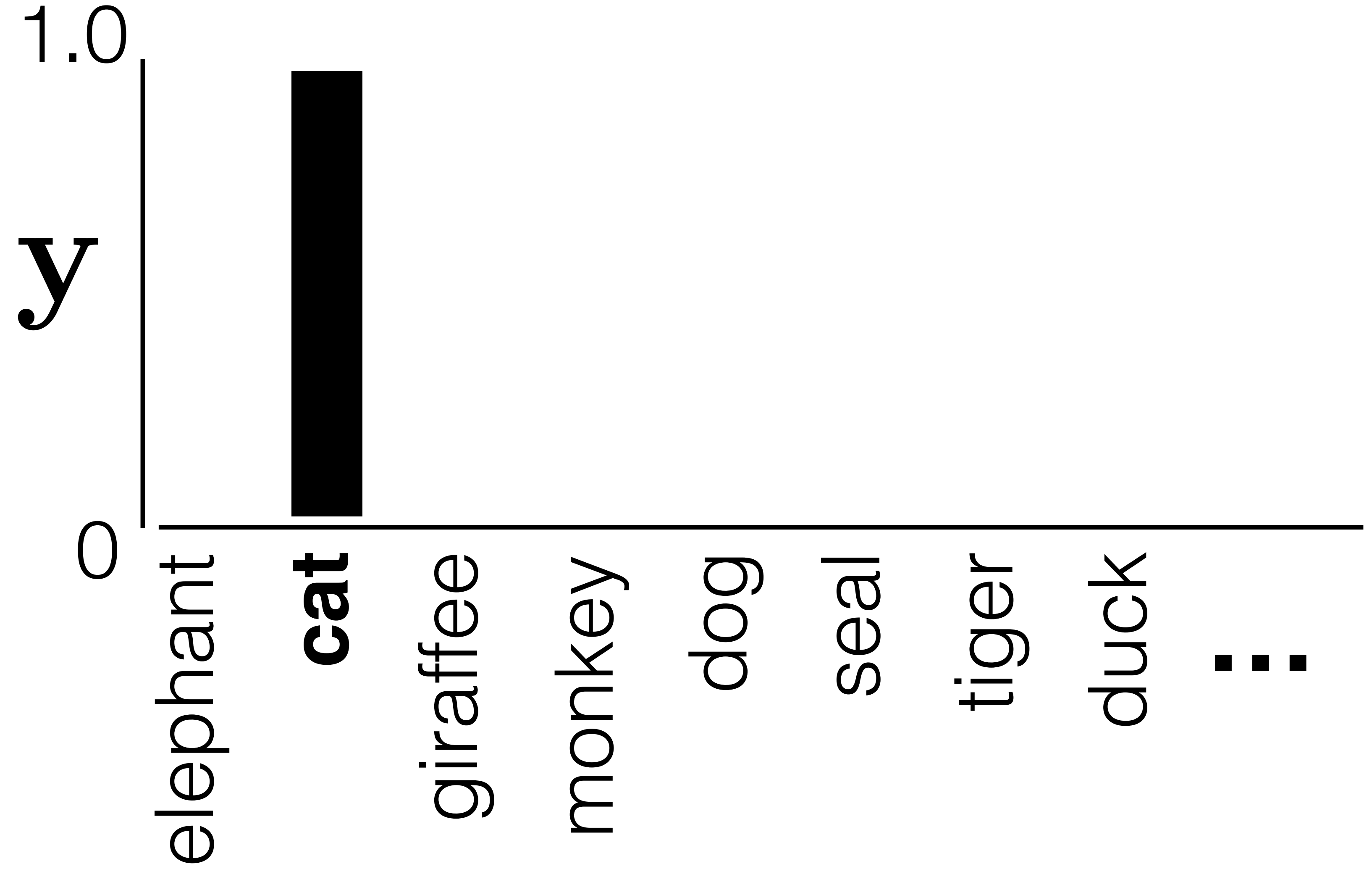
1. Knowledge about the mapping
2. **Knowledge about the outputs**
3. Knowledge about the inputs



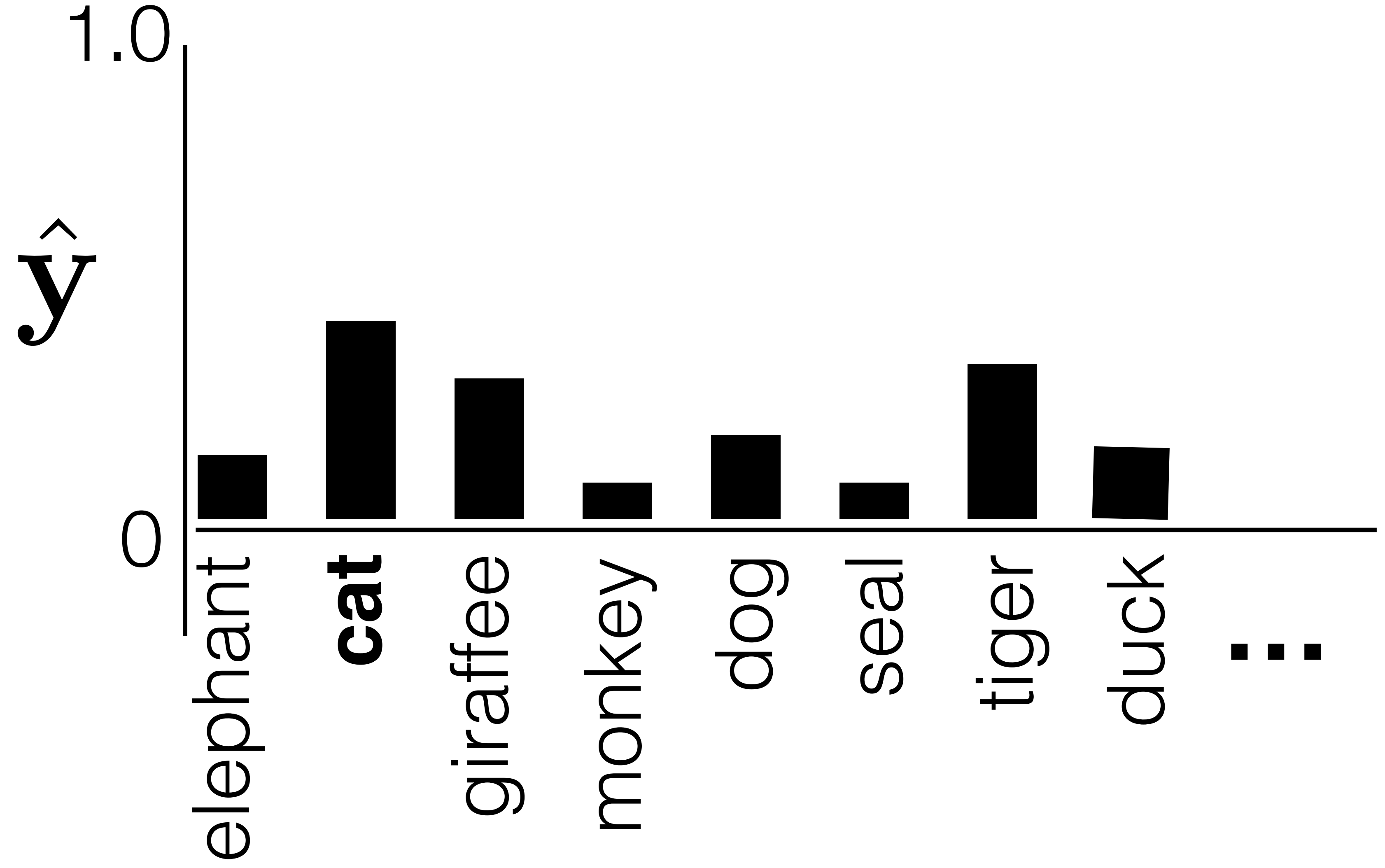
Transfer Learning

Recall: training a classifier with cross-entropy



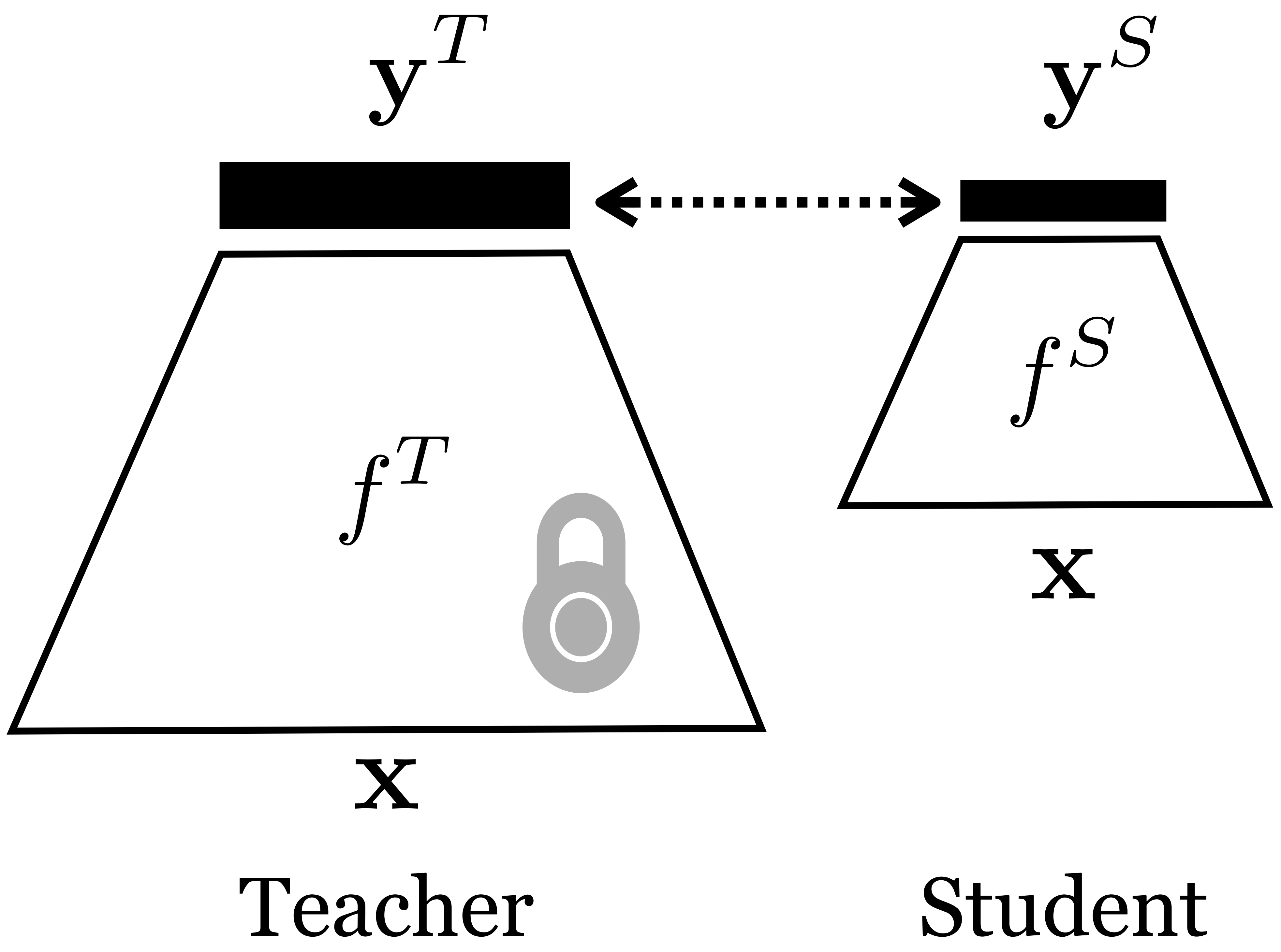


“Teacher” “Student”

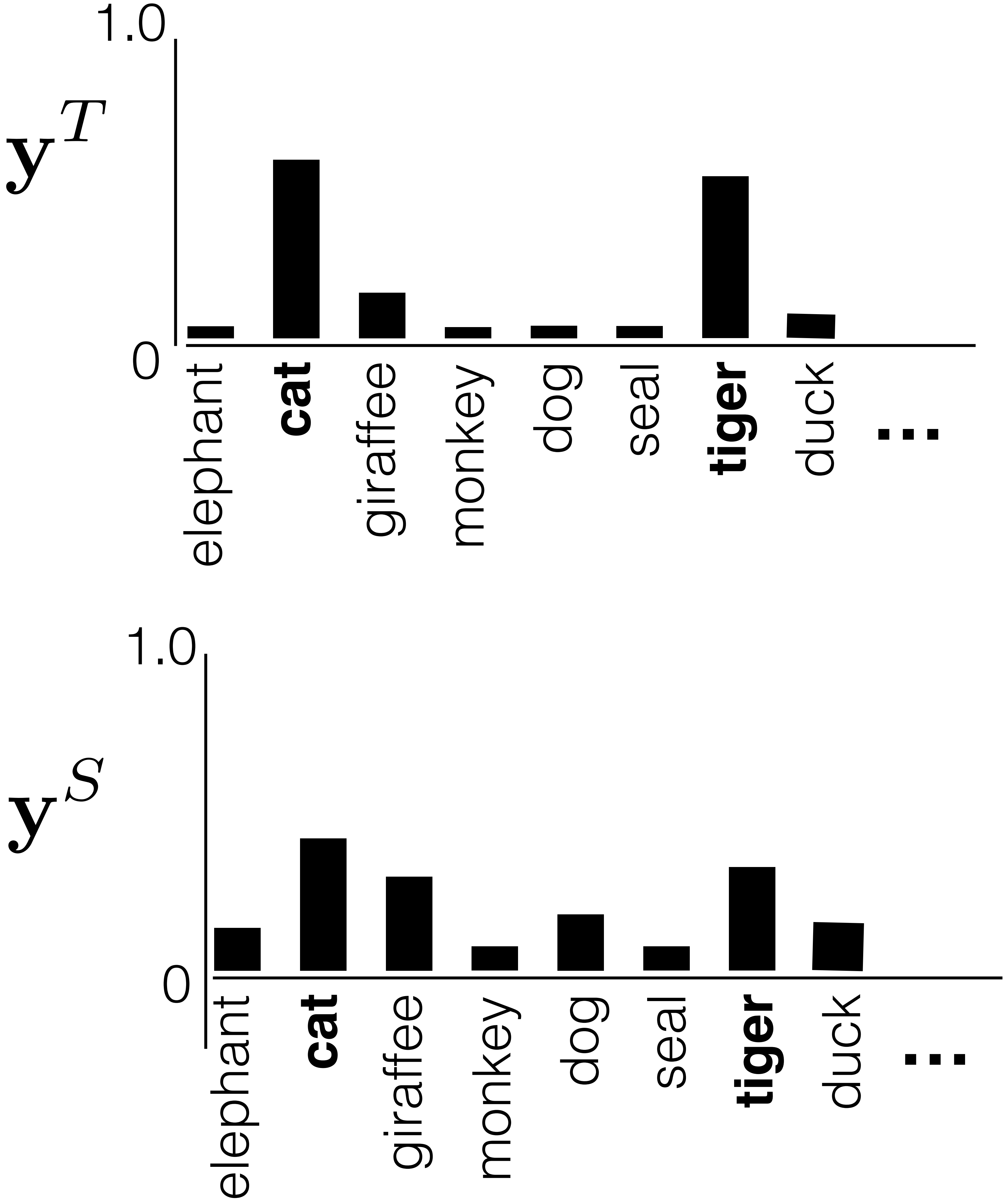


$$L(\hat{\mathbf{y}}, \mathbf{y}) = -\sum_i \mathbf{y}_i \log \hat{\mathbf{y}}_i$$

Knowledge Distillation [Hinton, Vinyals, Dean, 2015]



$$L(\mathbf{y}^S, \mathbf{y}^T) = -\sum_i \mathbf{y}_i^T \log \mathbf{y}_i^S$$



Knowledge Distillation [Hinton, Vinyals, Dean, 2015]

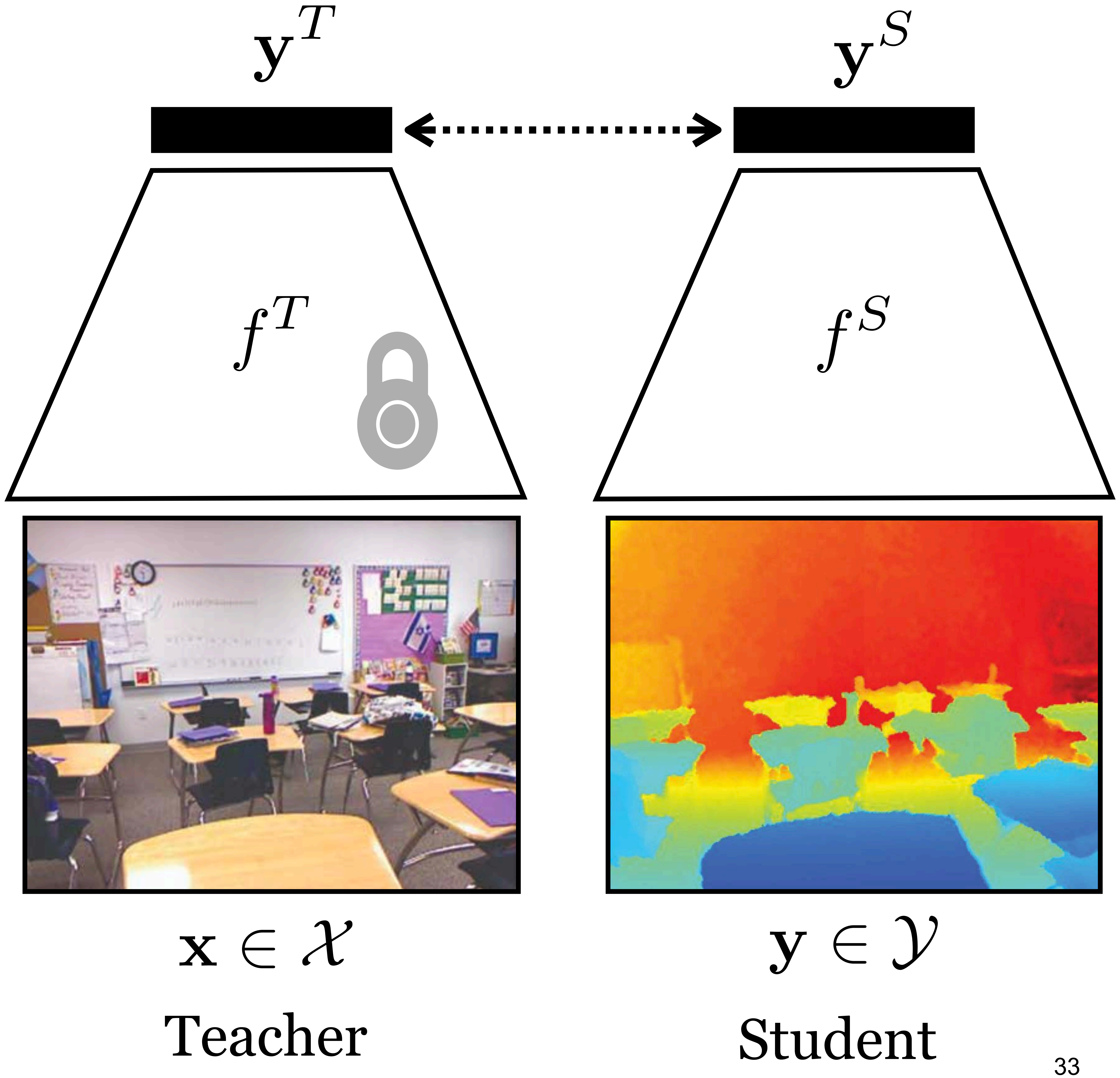
- The "label" you get for each input is more informative than just the ground truth class
- The "label" (**soft target**) says not just "this is a dog" but "this is a dog that looks a bit like a cat and very different from a bear" (so maybe it's a small dog).

Hinton paper shows lower train accuracy but higher test accuracy on speech recognition with soft targets

System & training set	Train Frame Accuracy	Test Frame Accuracy
Baseline (100% of training set)	63.4%	58.9%
Baseline (3% of training set)	67.3%	44.5%
Soft Targets (3% of training set)	65.4%	57.0%

Table 5: Soft targets allow a new model to generalize well from only 3% of the training set. The soft targets are obtained by training on the full training set.

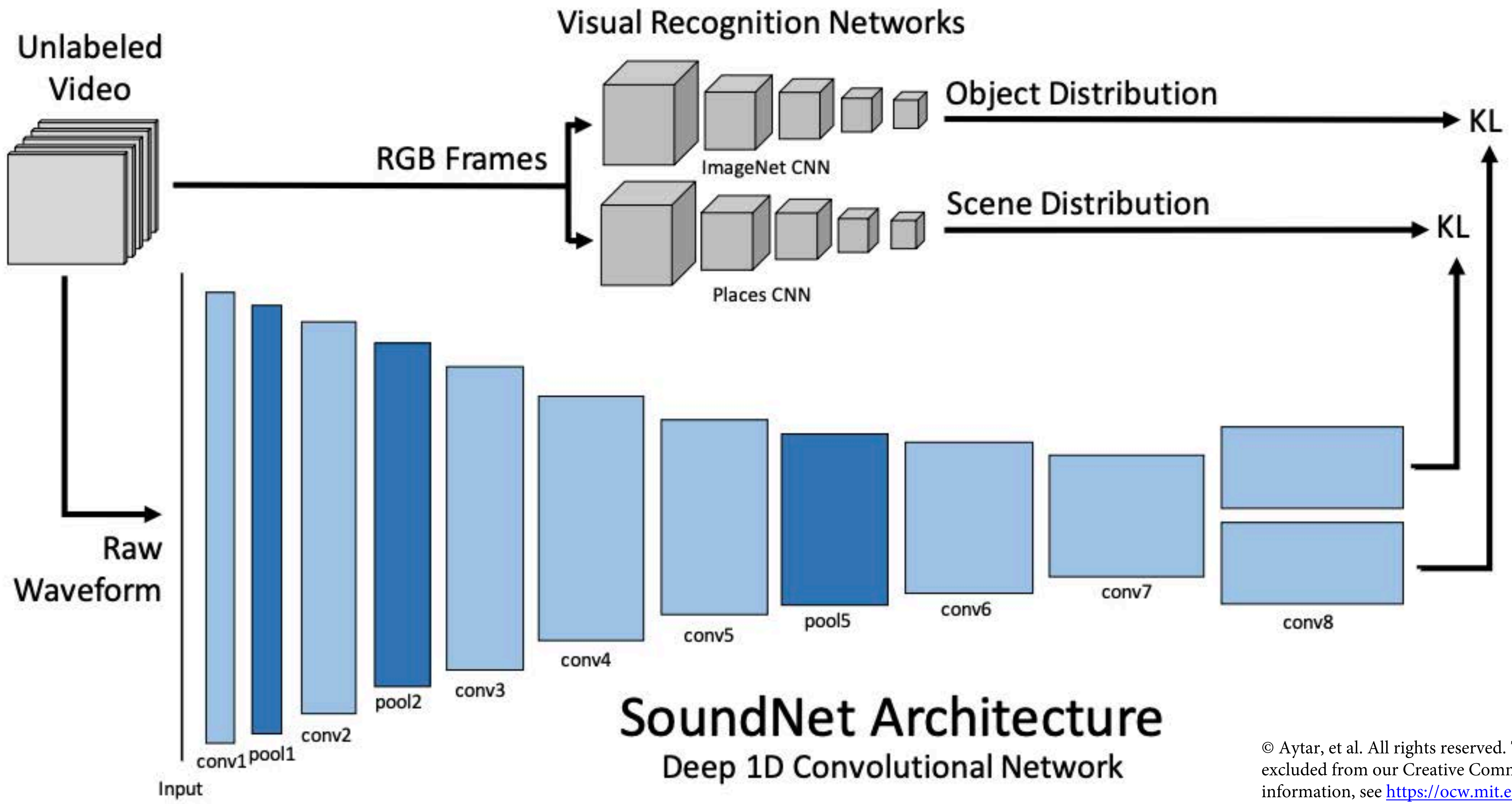
Cross-Modal Distillation [Gupta, Hoffman, Malik, 2015]



Useful if you have labeled data for one domain but not the other.

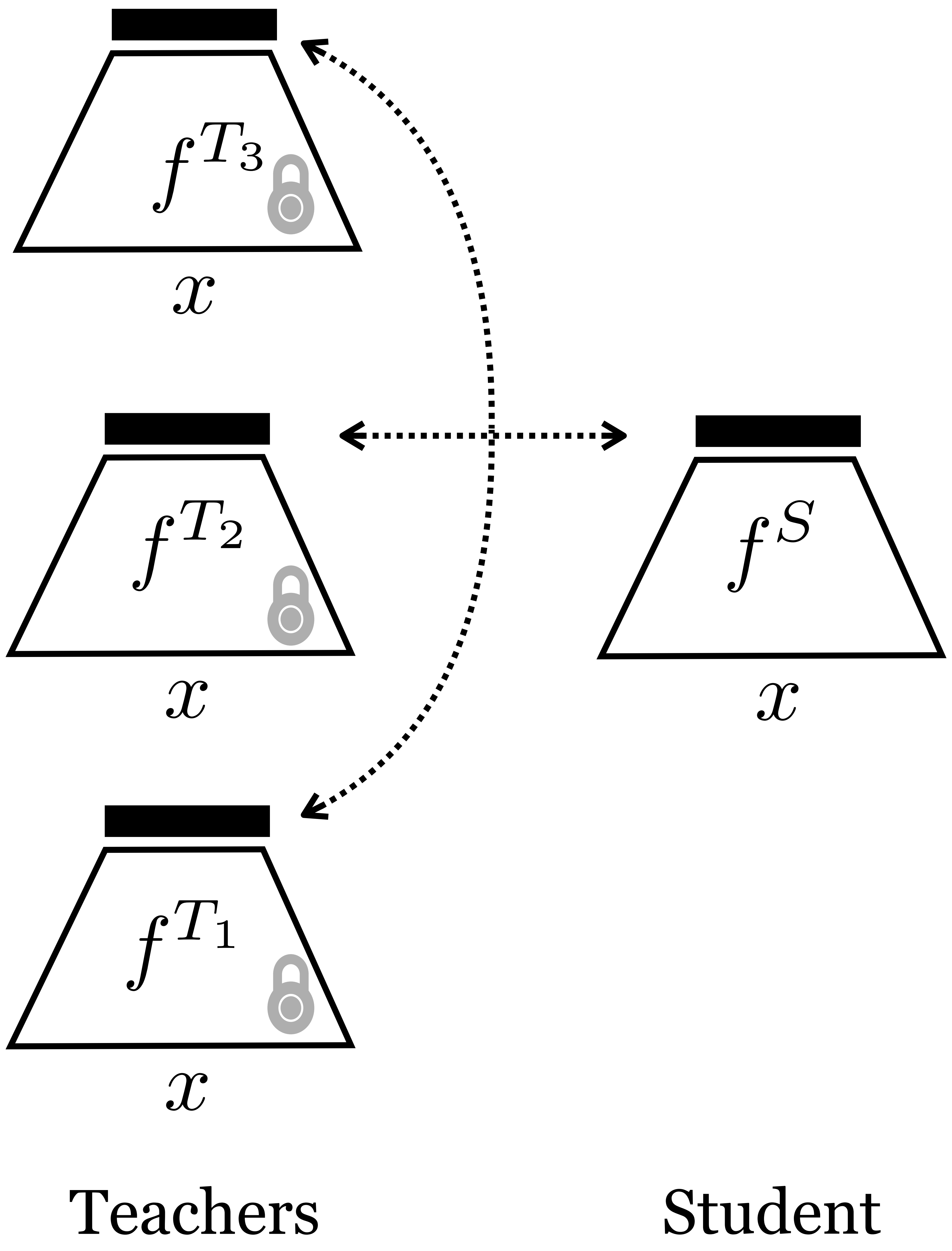
Cross-Modal Distillation: SoundNet

[Aytar*, Vondrick*, Torralba, 2016]

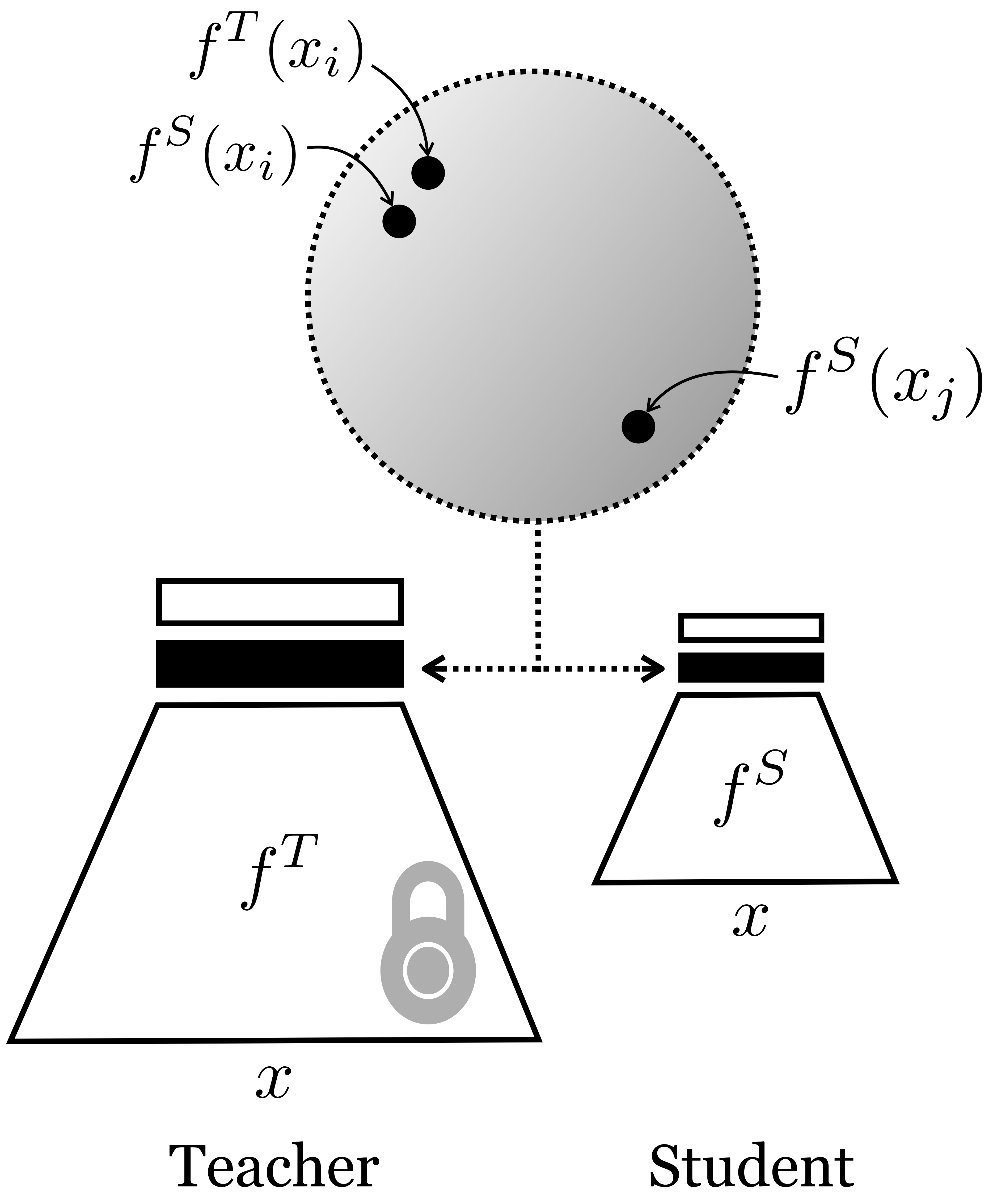


Distilling an ensemble

- Ensembling model outputs creates more accurate estimates.
- Why? Each model is imperfect in different random ways. Errors cancel out, truth is shared.
- Distill ensemble into a single model for fast inference.

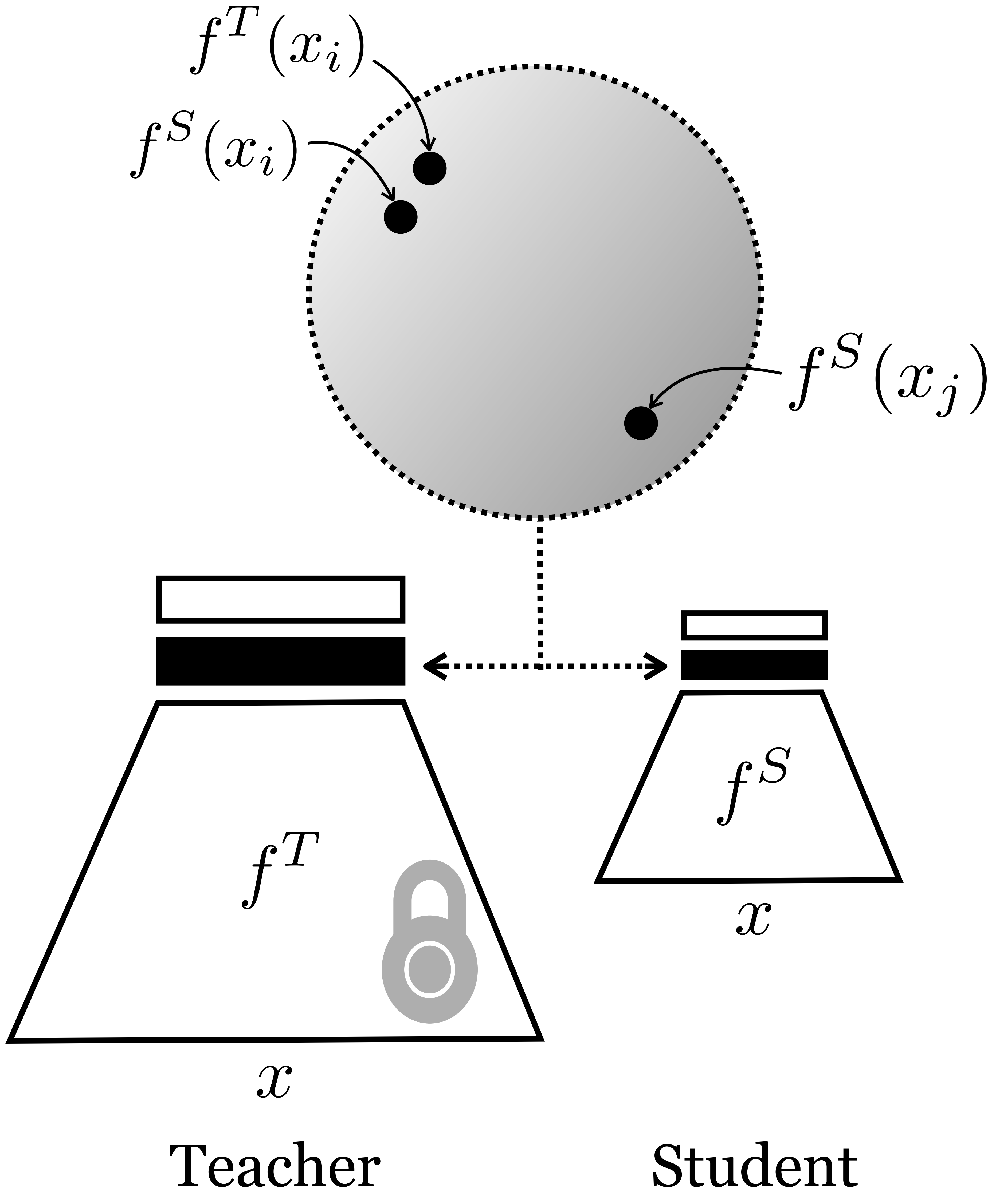


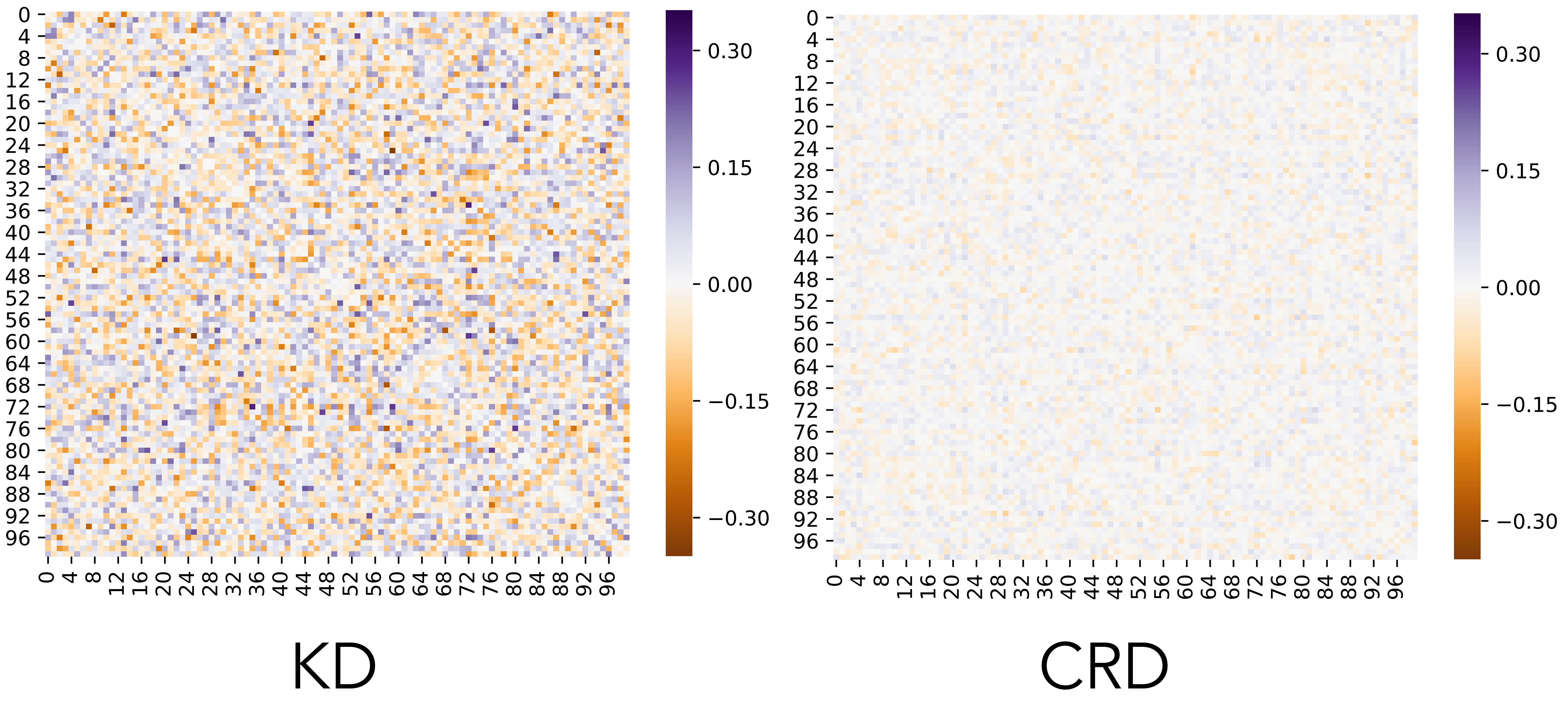
Contrastive Representation Distillation [Tian, Krishnan, Isola, 2020]



- Contrastive learning supervises embeddings to be invariant to some viewing transformation.
- The "viewing transformation" can be "teacher's view" (big net) vs "students view" (small net).
- Then the student will output embeddings that are the same as the teacher's.

Contrastive Representation Distillation [Tian, Krishnan, Isola, 2020]





Correlation coefficients between class output logits — Student's minus Teacher's (CIFAR100: plots are 100 classes by 100 classes)

Compression vs. Distillation

- Both give us smaller models with similar accuracy to large ones
- Distillation uses SGD to train a smaller model to match characteristics of a larger one
- Compression uses ***pruning*** or ***quantization*** to reduce the computational cost of a large model

Algorithm 1 Prune $k \leq d_{\text{col}}$ weights from row $\mathbf{w}$ with inverse Hessian $\mathbf{H}^{-1} = (2\mathbf{X}\mathbf{X}^\top)^{-1}$ according to OBS in $O(k \cdot d_{\text{col}}^2)$ time.

$M = \{1, \ldots, d_{\text{col}}\}$
for $i = 1, \ldots, k$ **do**
  $p \leftarrow \text{argmin}_{p \in M} \frac{1}{[\mathbf{H}^{-1}]_{pp}} \cdot w_p^2$
  $\mathbf{w} \leftarrow \mathbf{w} - \mathbf{H}^{-1}_{:,p} \frac{1}{[\mathbf{H}^{-1}]_{pp}} \cdot w_p$
  $\mathbf{H}^{-1} \leftarrow \mathbf{H}^{-1} - \frac{1}{[\mathbf{H}^{-1}]_{pp}} \mathbf{H}^{-1}_{:,p} \mathbf{H}^{-1}_{p,:}$
  $M \leftarrow M - \{p\}$
end for

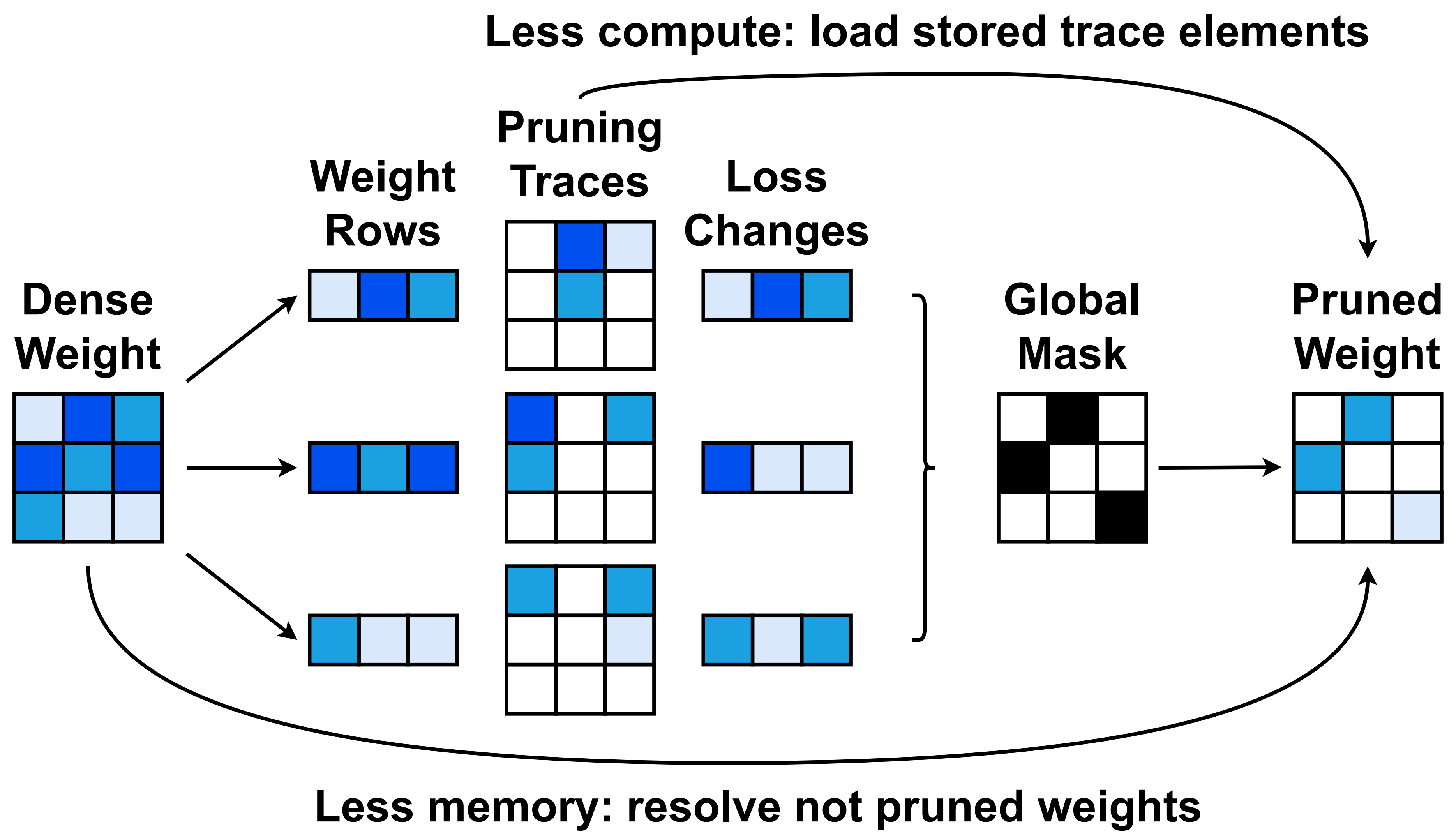
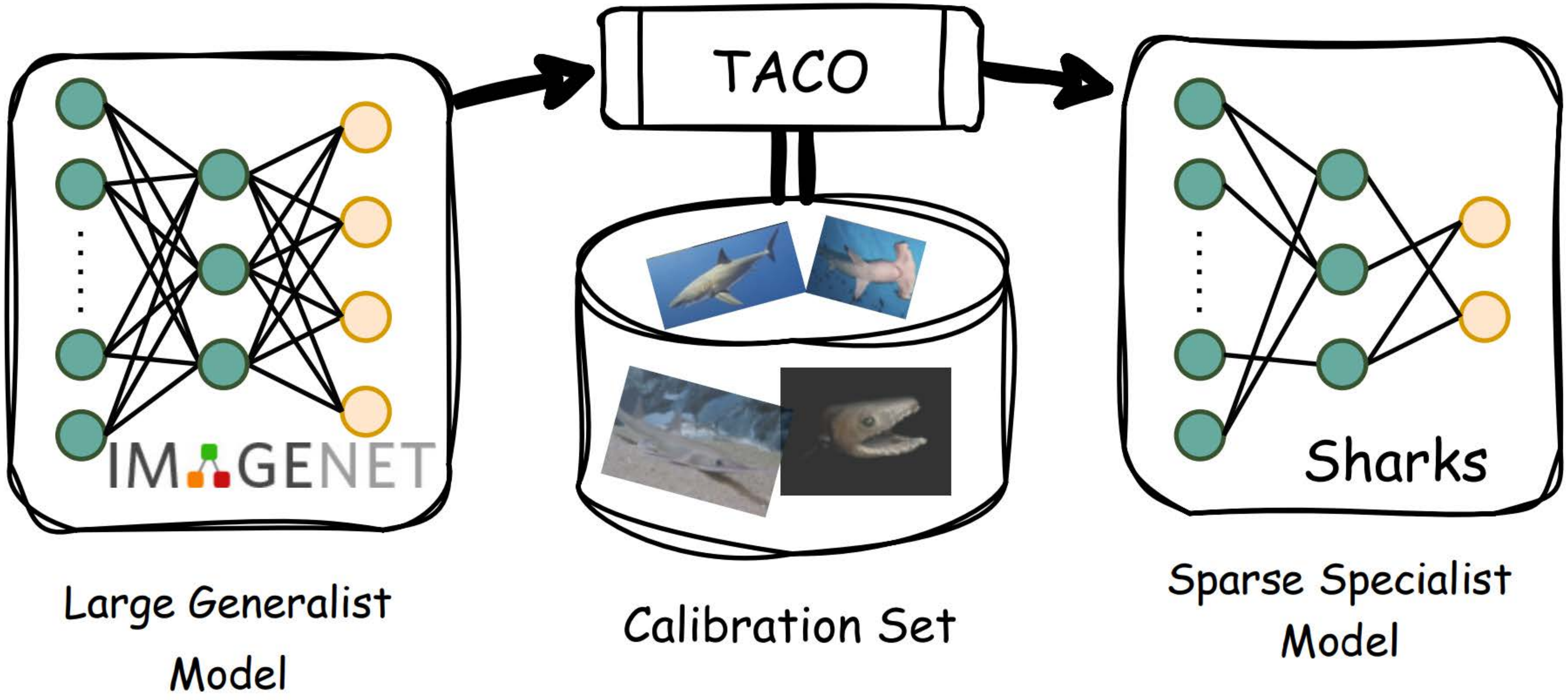


Figure 1: Efficient global OBS using the row-wise results.

Optimal Brain Compression: A Framework for Accurate Post-Training Quantization and Pruning, Frantar et al., 2023

We almost never need a truly generalist model in practice. Specialization can enable much higher rates of compression with the same accuracy.



Vision Models Can Be Efficiently Specialized via Few-Shot Task-Aware Compression, Kuznedelev et al., 2023

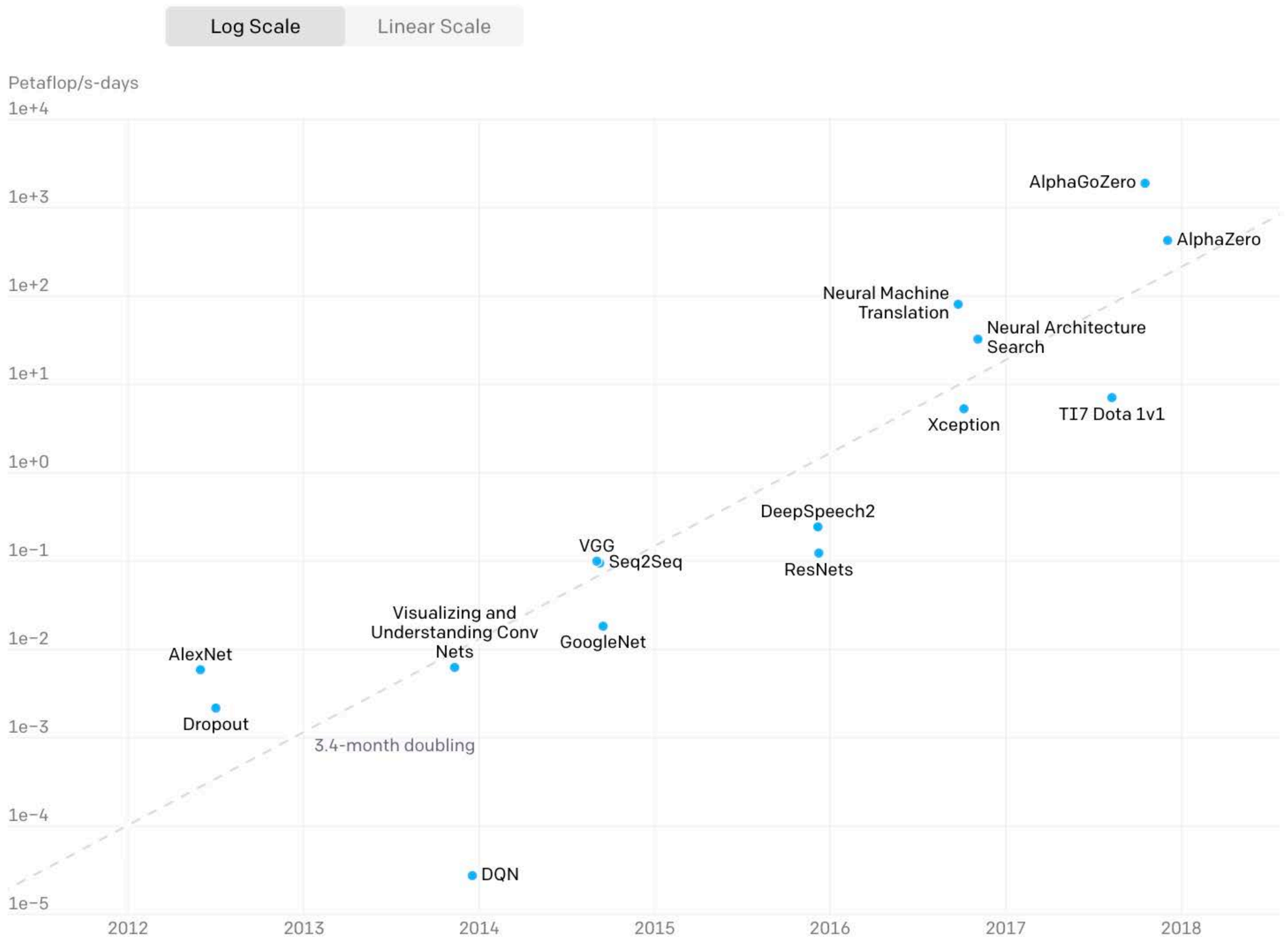
Foundation models

[Bommasani et al. 2021] https://arxiv.org/pdf/2108.07258.pdf

"If I have seen further it is by standing on the shoulders of Giants"
— Newton

[*Blind Orion Searching for the Rising Sun* by Nicolas Poussin, 1658]

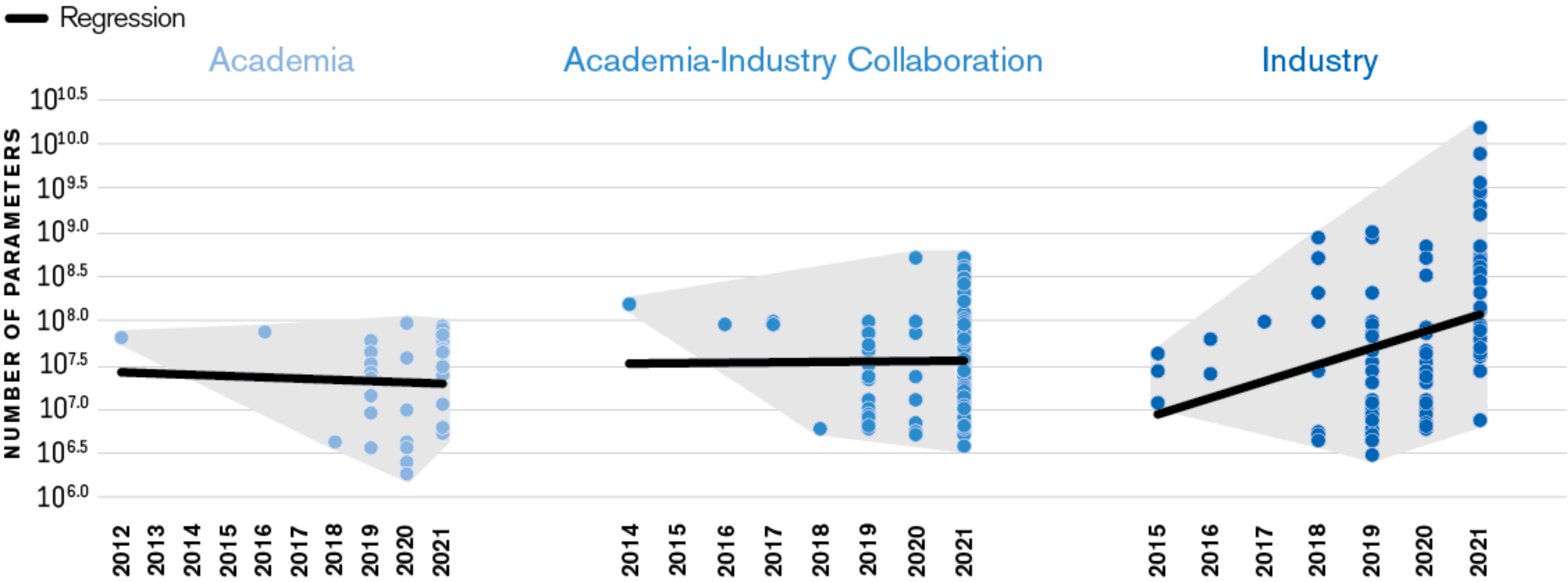
AlexNet to AlphaGo Zero: A 300,000x Increase in Compute



The total amount of compute, in petaflop/s-days,[2] used to train selected results that are relatively well known, used a lot of compute for their time, and gave enough information to estimate the compute used.

https://openai.com/blog/ai-and-compute/

D. Computing Power Usage Over Time by:



Ahmed and Thompson (2022)

Climate cost

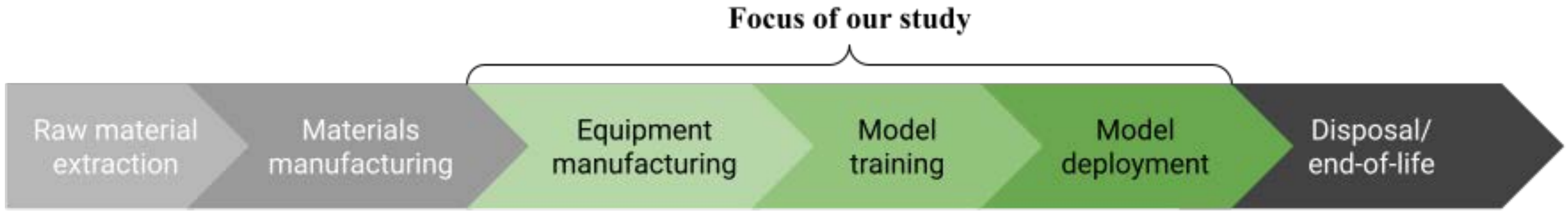


Figure 1: While the LCA approach encompasses all the stages of the product life cycle (from raw material extraction to disposal), we focus on those in green, which range from equipment manufacturing to model deployment.

Model name	**Number of parameters**	**Datacenter PUE**	**Carbon intensity of grid used**	**Power consumption**	**CO_2eq emissions**	**CO_2eq emissions × PUE**
GPT-3	175B	1.1	429 gCO_2eq/kWh	1,287 MWh	*502 tonnes*	552 tonnes
Gopher	280B	1.08	330 gCO_2eq/kWh	*1,066 MWh*	*352 tonnes*	380 tonnes
OPT	175B	*1.09* [2]	*231gCO_2eq/kWh*	*324 MWh*	70 tonnes	*76.3 tonnes* [3]
BLOOM	176B	1.2	57 gCO_2eq/kWh	433 MWh	25 tonnes	30 tonnes

Table 4: Comparison of carbon emissions between BLOOM and similar LLMs. Numbers in *italics* have been inferred based on data provided in the papers describing the models.

Estimating the Carbon Footprint of BLOOM, a 176B Parameter Language Model, Luccioni et al., 2023

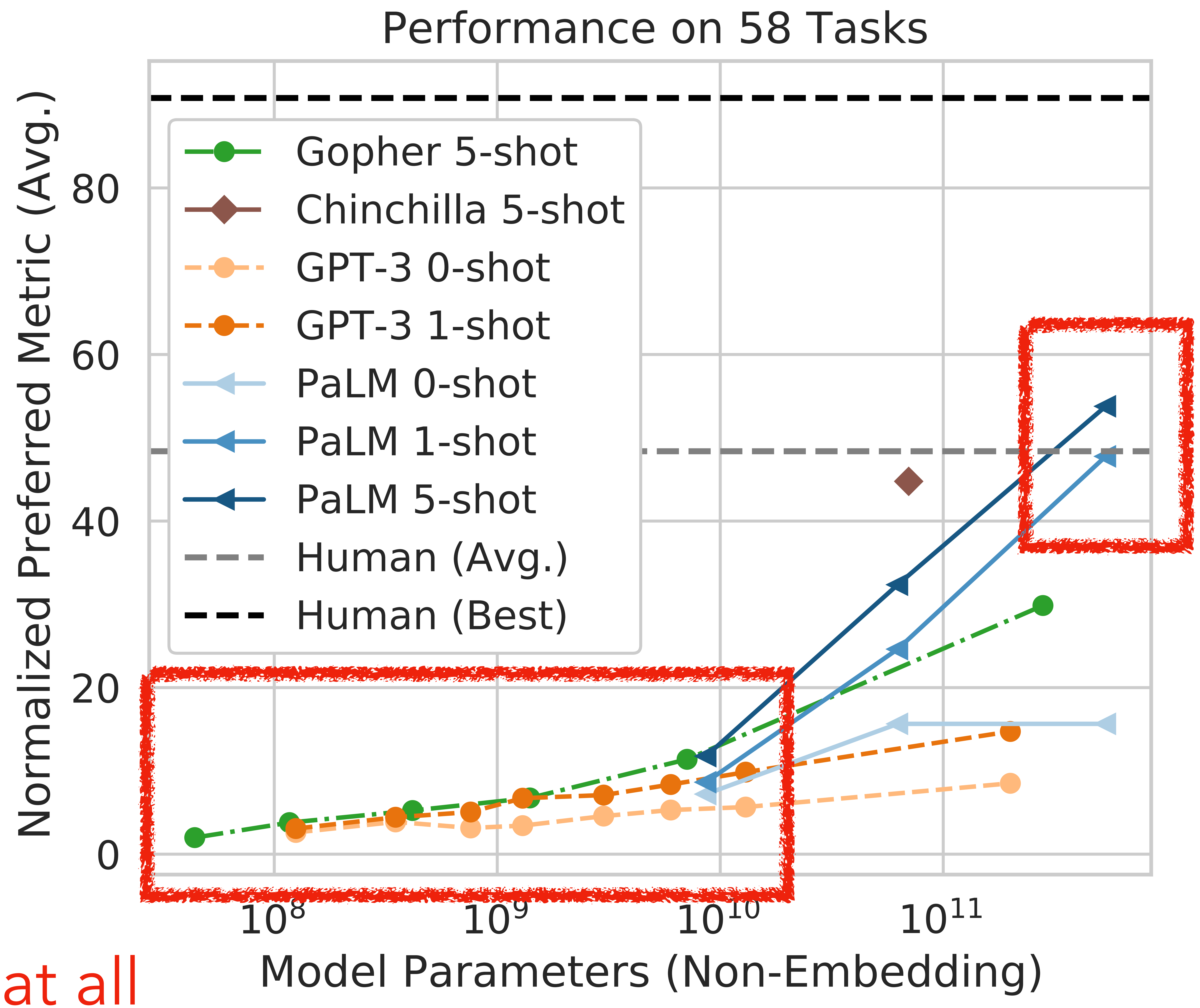
Making AI Less "Thirsty": Uncovering and Addressing the Secret Water Footprint of AI Models, Li et al., 2023

What does it take to train one of these models?

- Dalle-2: **650m training examples** (semi-curated image-text pairs)
- GPT-3: ~**$4.6m to train *once*** (https://lambdalabs.com/blog/demystifying-gpt-3/)
- PaLM: **67 authors**

And these numbers are increasing.

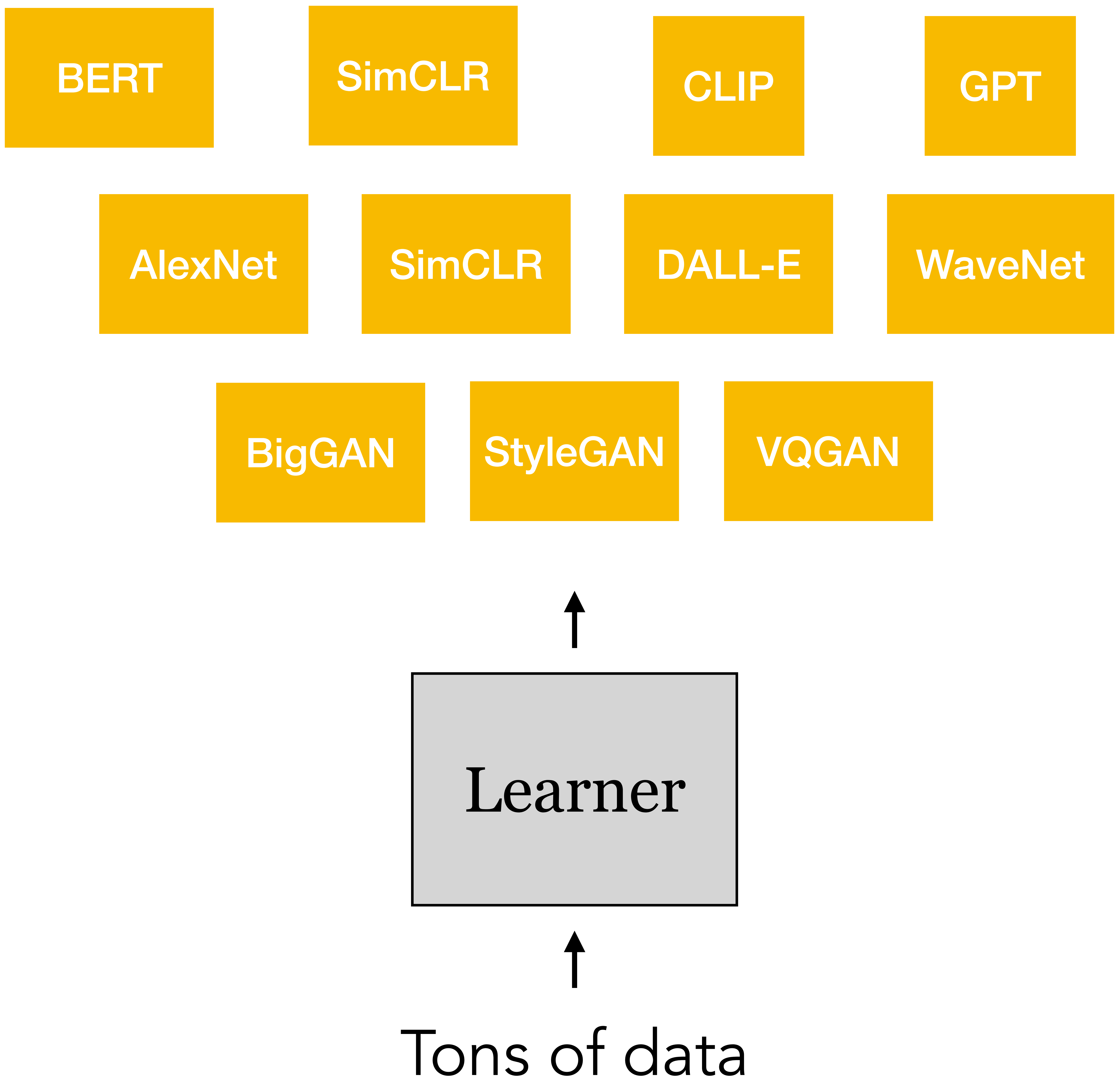
But these numbers are small if we think of the result as foundational infrastructure; compare to the cost of building the interstate highway system, or the cost of developing the Mac OS X operating system.



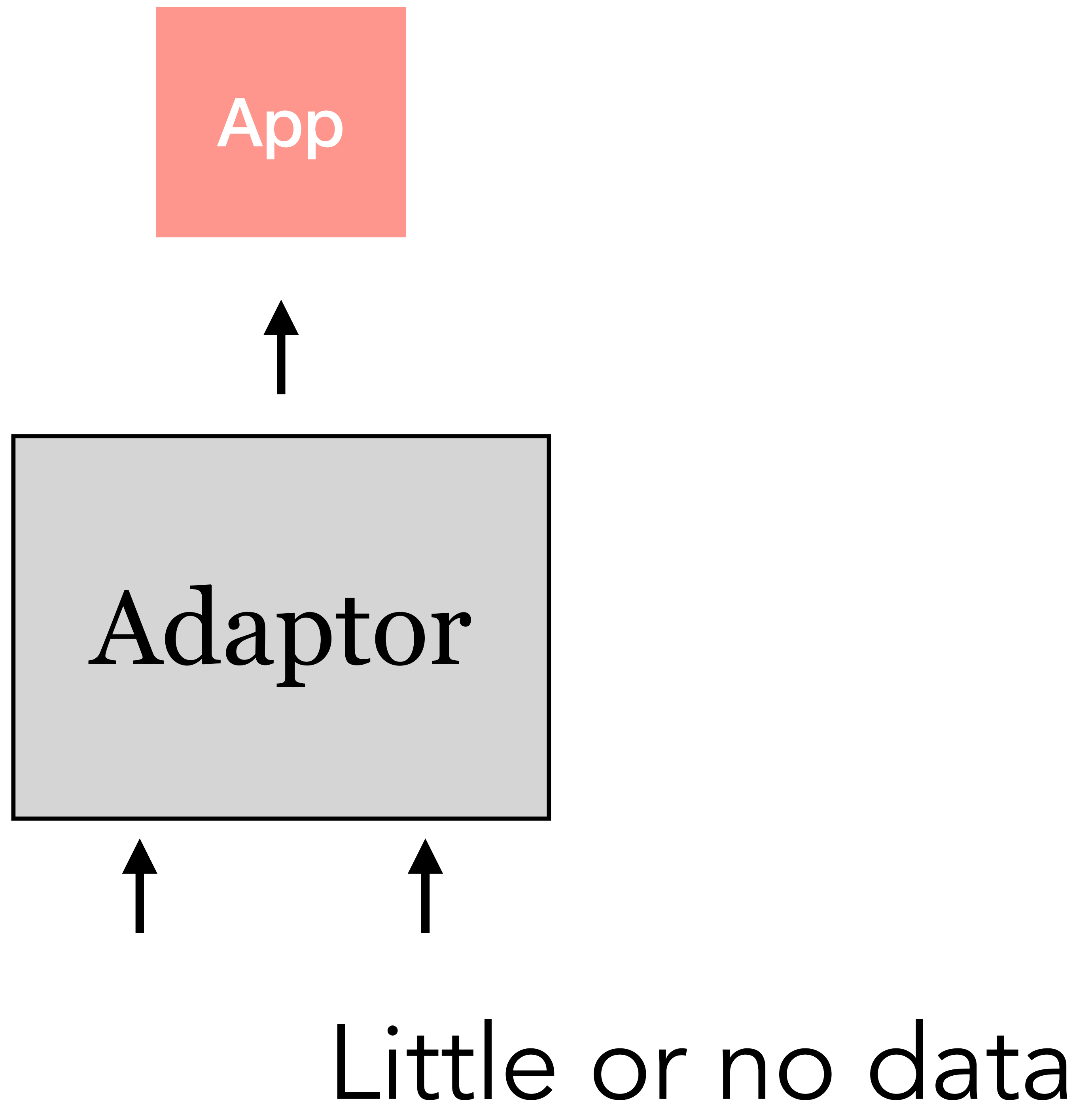
Working as well as an average human

Not working at all

Learn foundation models



Use/adapt foundations to solve new problems



New paradigm: "Deep Learning" with *No* Data

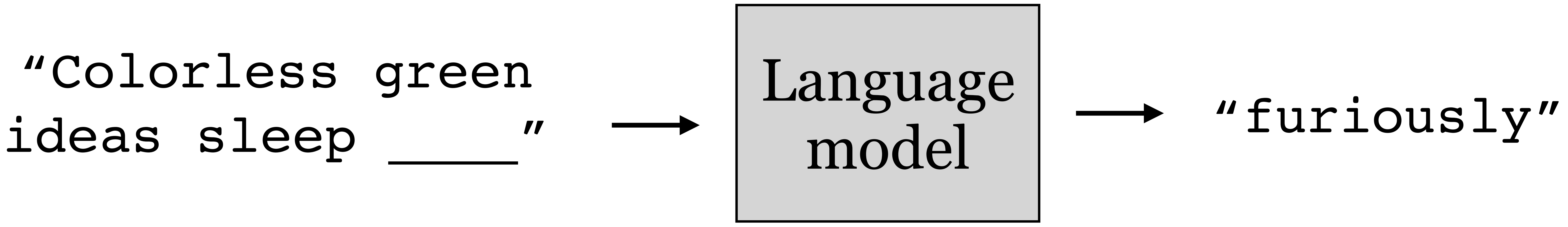
"Just ask"

(of course, this isn't really machine *learning* anymore)

https://evjang.com/2021/10/23/generalization.html

GPT-3 [Brown et al., 2020] https://arxiv.org/pdf/2005.14165.pdf

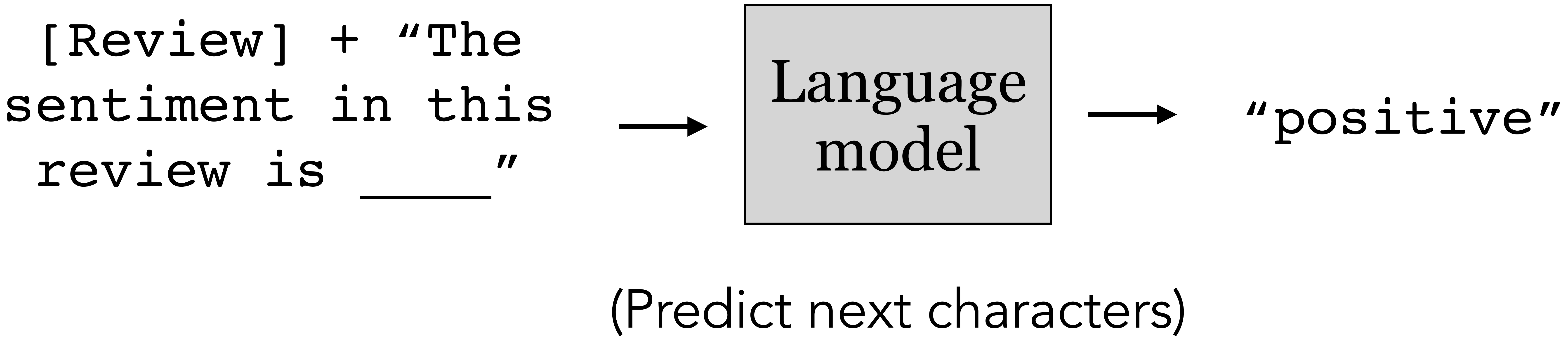
1. **Learner**:
Generative
Pretraining



(Predict next characters)

GPT-3 [Brown et al., 2020] https://arxiv.org/pdf/2005.14165.pdf

1. **Adaptor**: Prompt engineering



New capabilities by just asking

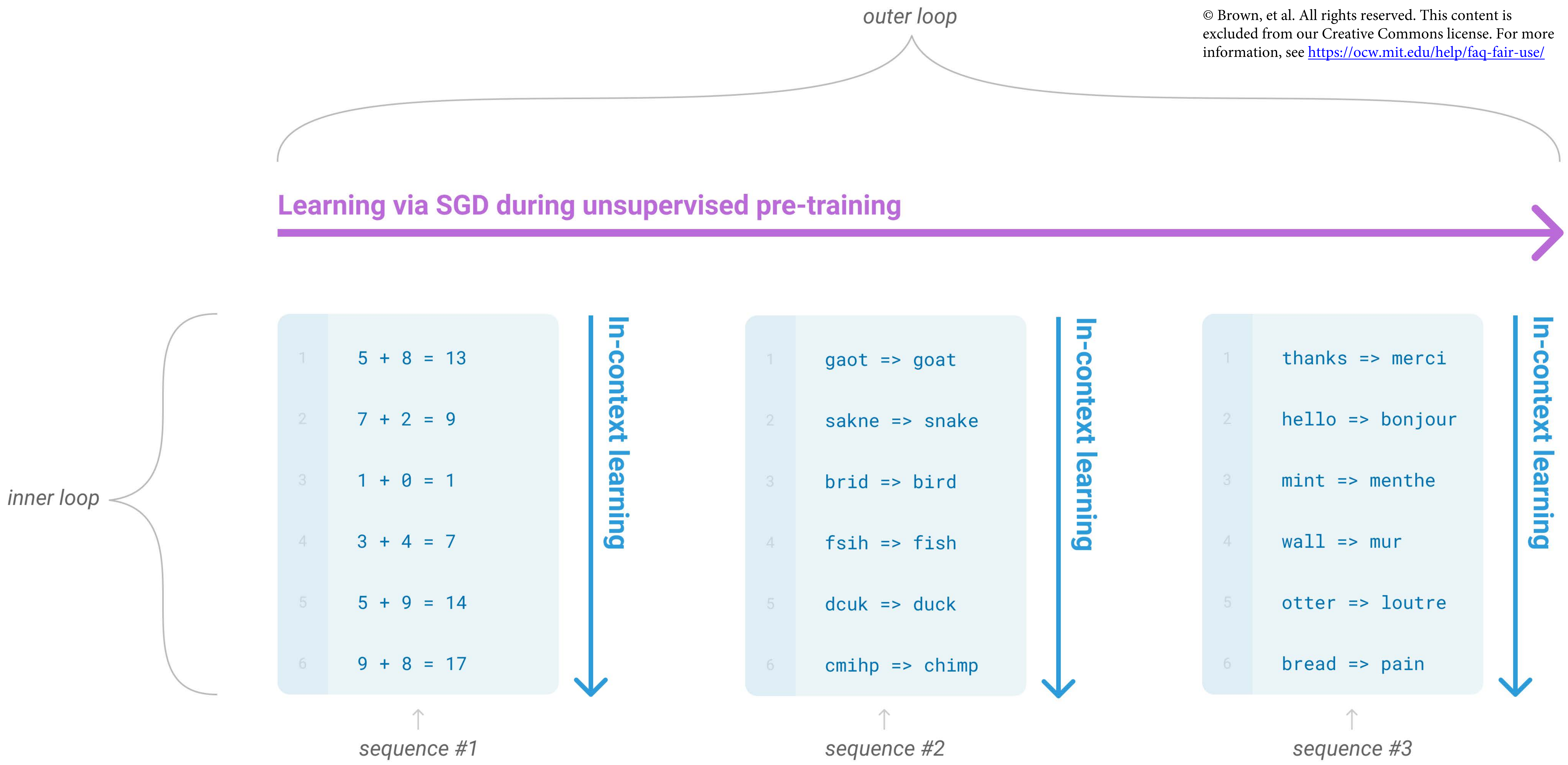
Top ten dishes at Mulan, Cambridge, MA:

New capabilities by just asking

List of prime numbers:

1. 2
2. 3
3. 5
4. 7
5. 11
6. 13

Few-shot learning ability



Few-shot learning ability: extrapolation

1. 🐘
2. 🐘 🐘
3. 🐘 🐘 🐘
4. 🐘 🐘 🐘 🐘
5. 🐘 🐘 🐘 🐘 🐘
6. 🐘 🐘 🐘 🐘 🐘 🐘
7. 🐘 🐘 🐘 🐘 🐘 🐘 🐘
8. 🐘 🐘 🐘 🐘 🐘 🐘 🐘 🐘
9. 🐘 🐘 🐘 🐘 🐘 🐘 🐘 🐘 🐘
10. 🐘 🐘 🐘 🐘 🐘 🐘 🐘 🐘 🐘 🐘
11. 🐘 🐘 🐘 🐘 🐘 🐘 🐘 🐘 🐘 🐘 🐘

1. 🐘
2. 🐘 🍎
3. 🐘 🍎 🐘
4. 🐘 🍎 🐘 🍎
5. 🐘 🍎 🐘 🍎 🐘
6. 🐘 🍎 🐘 🍎 🐘 🍎
7. 🐘 🍎 🐘 🍎 🐘 🍎 🐘

GPT-3 [Brown et al., 2020] https://arxiv.org/pdf/2005.14165.pdf

English: hello

Spanish: hola

English: this tastes really good

Spanish: esto sabe muy bien

English: where is the library?

Spanish: dónde está la biblioteca?

Prompting

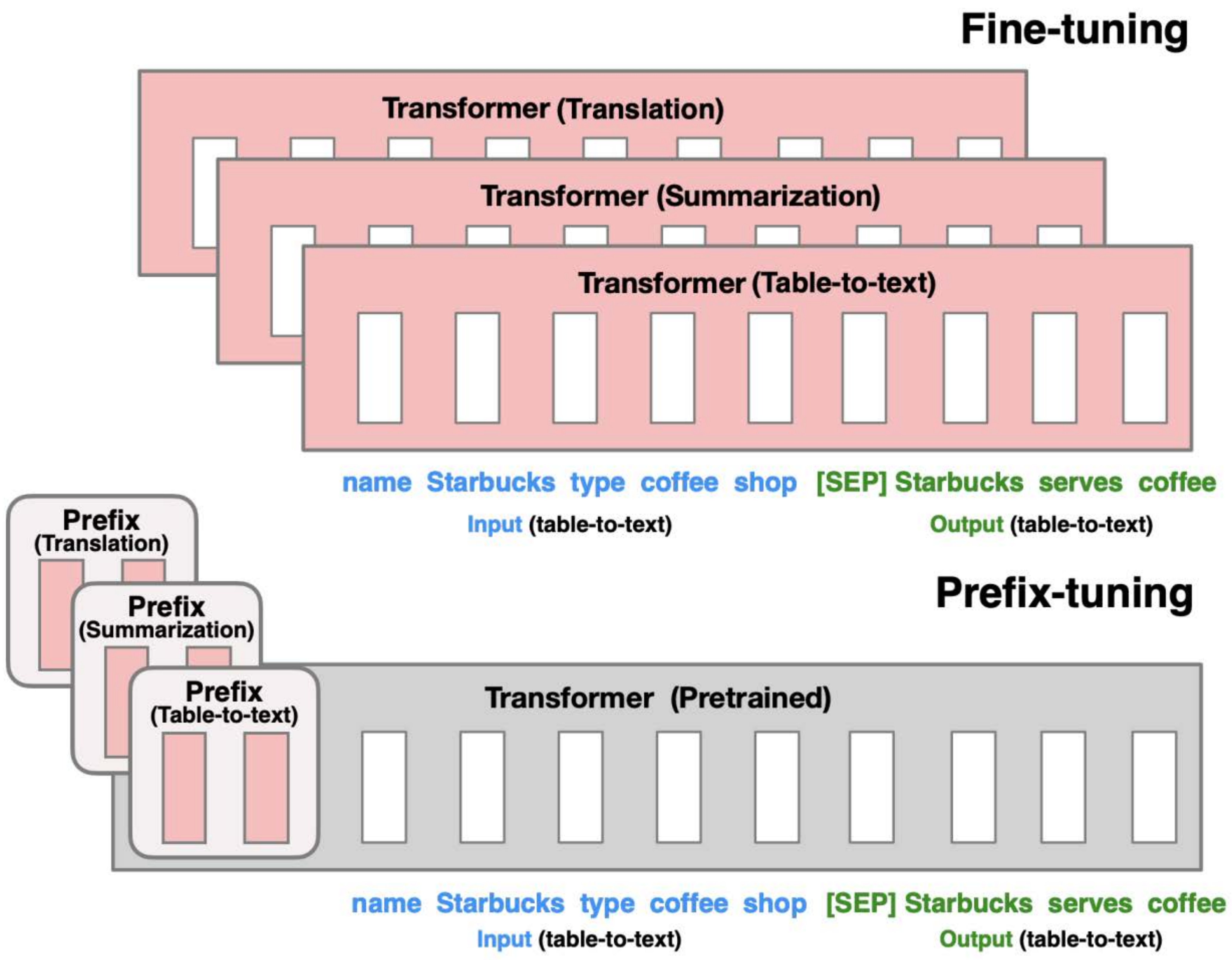
Slide credit: Hyojin Bahng

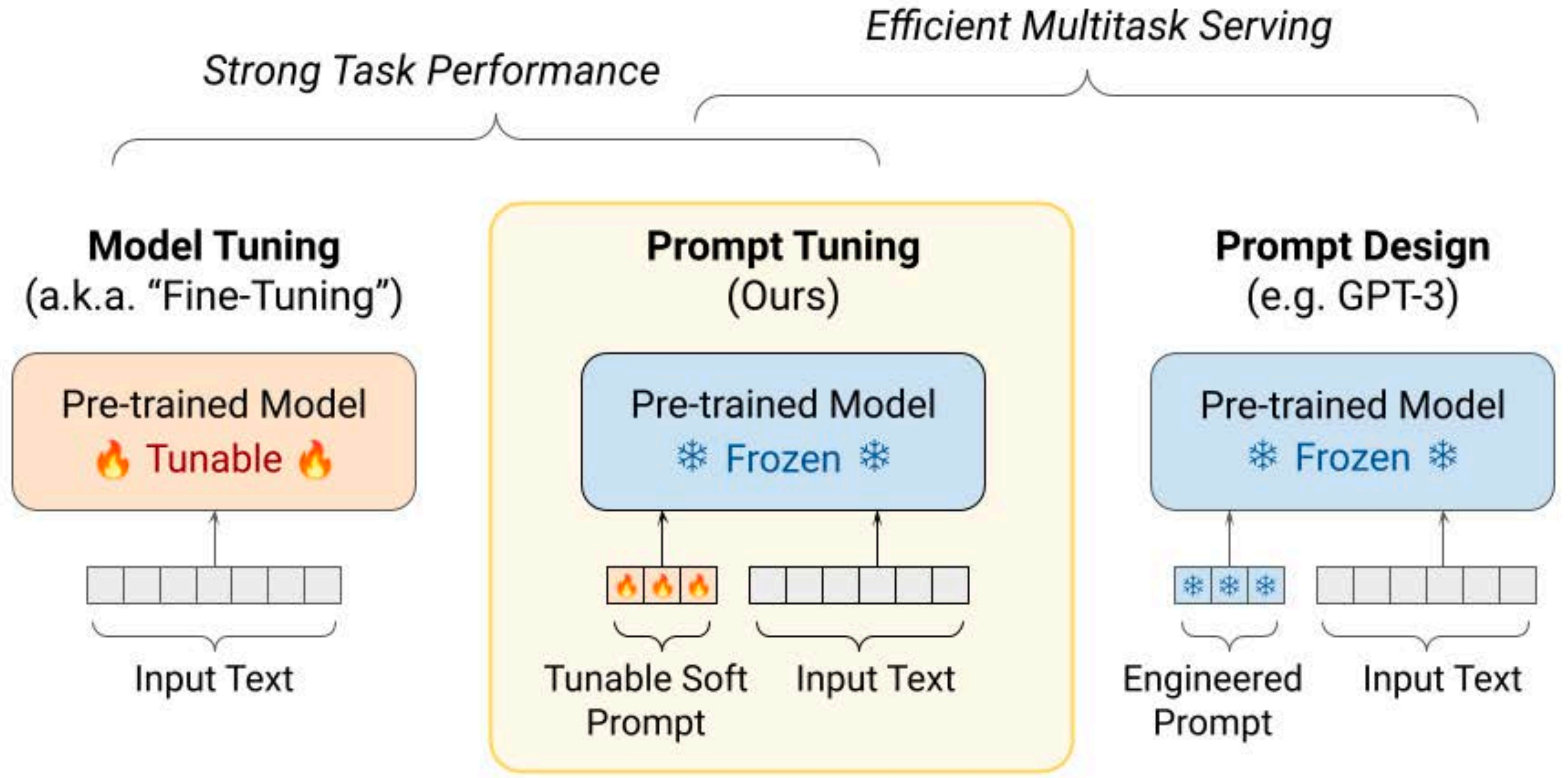
Hand-crafting good prompts is tricky…

Template	Label words	Accuracy
SST-2 (positive/negative)		mean (std)
$<S_1>$ It was `[MASK]` .	great/terrible	**92.7 (0.9)**
$<S_1>$ It was `[MASK]` .	good/bad	92.5 (1.0)
$<S_1>$ It was `[MASK]` .	cat/dog	91.5 (1.4)
$<S_1>$ It was `[MASK]` .	dog/cat	86.2 (5.4)
$<S_1>$ It was `[MASK]` .	terrible/great	83.2 (6.9)
Fine-tuning	-	81.4 (3.8)
SNLI (entailment/neutral/contradiction)		mean (std)
$<S_1>$? `[MASK]` , $<S_2>$	Yes/Maybe/No	**77.2 (3.7)**
$<S_1>$. `[MASK]` , $<S_2>$	Yes/Maybe/No	76.2 (3.3)
$<S_1>$? `[MASK]` $<S_2>$	Yes/Maybe/No	74.9 (3.0)
$<S_1>$ $<S_2>$ `[MASK]`	Yes/Maybe/No	65.8 (2.4)
$<S_2>$? `[MASK]` , $<S_1>$	Yes/Maybe/No	62.9 (4.1)
$<S_1>$? `[MASK]` , $<S_2>$	Maybe/No/Yes	60.6 (4.8)
Fine-tuning	-	48.4 (4.8)

Soft Prompts (Prompt Tuning)

- Optimizes a small continuous task-specific vector while having language model parameters frozen

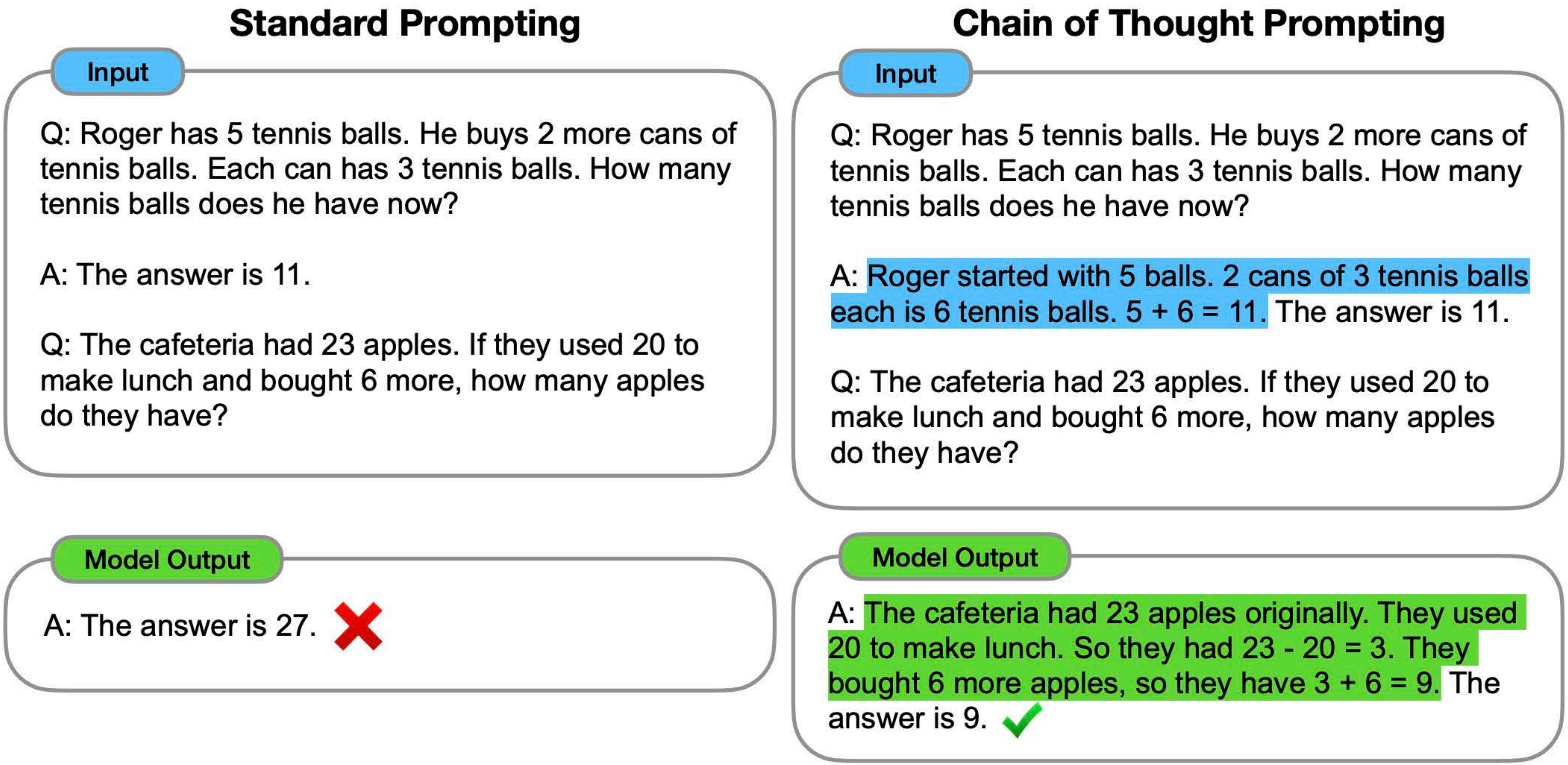




Prefix-tuning: Optimizing continuous prompts for generation. ACL 2021.
The Power of Scale for Parameter-Efficient Prompt Tuning. EMNLP 2021.

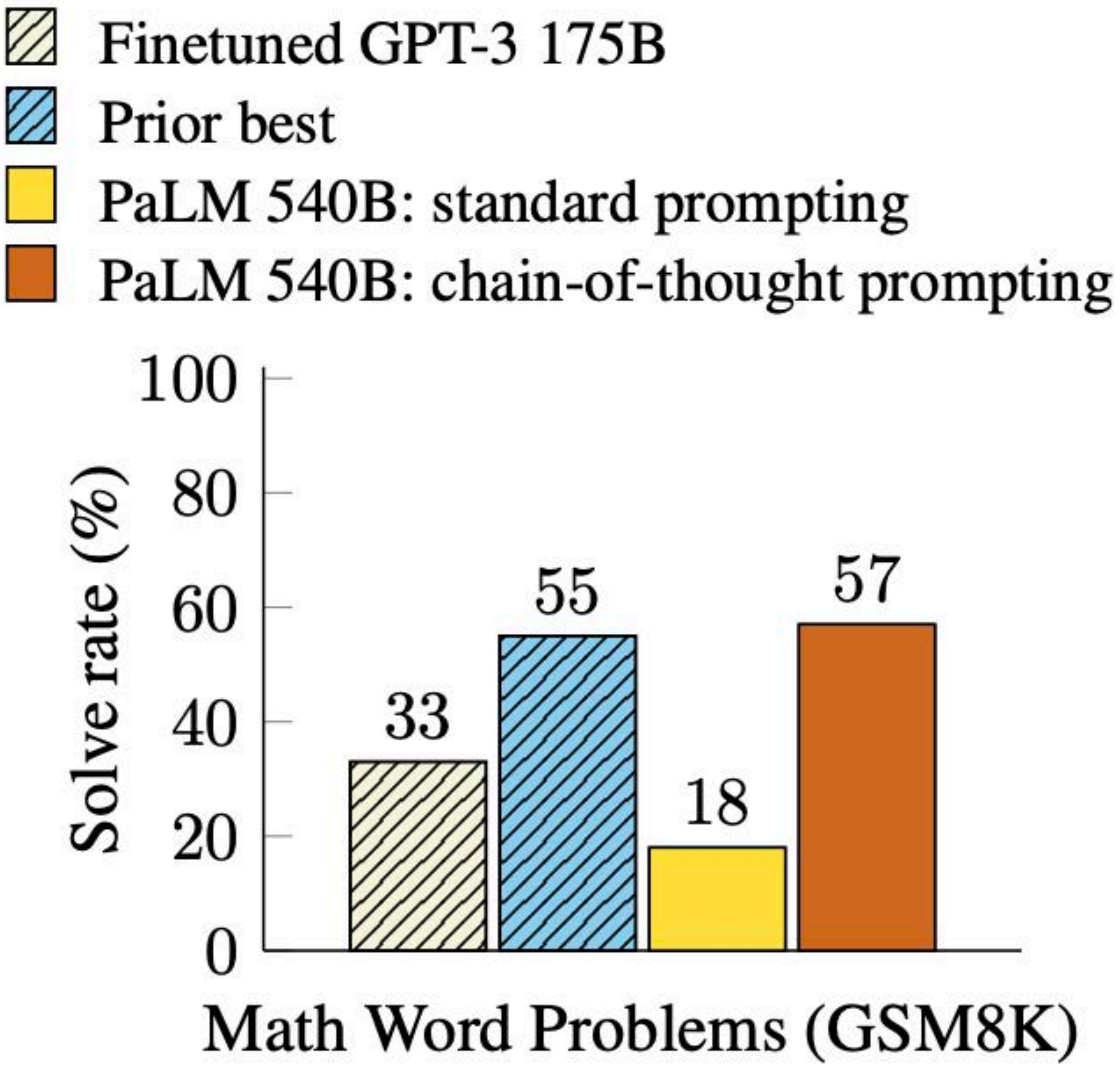
Chain-of-thought prompting

[Wei, Wang, Schuurmans et al. 2022]



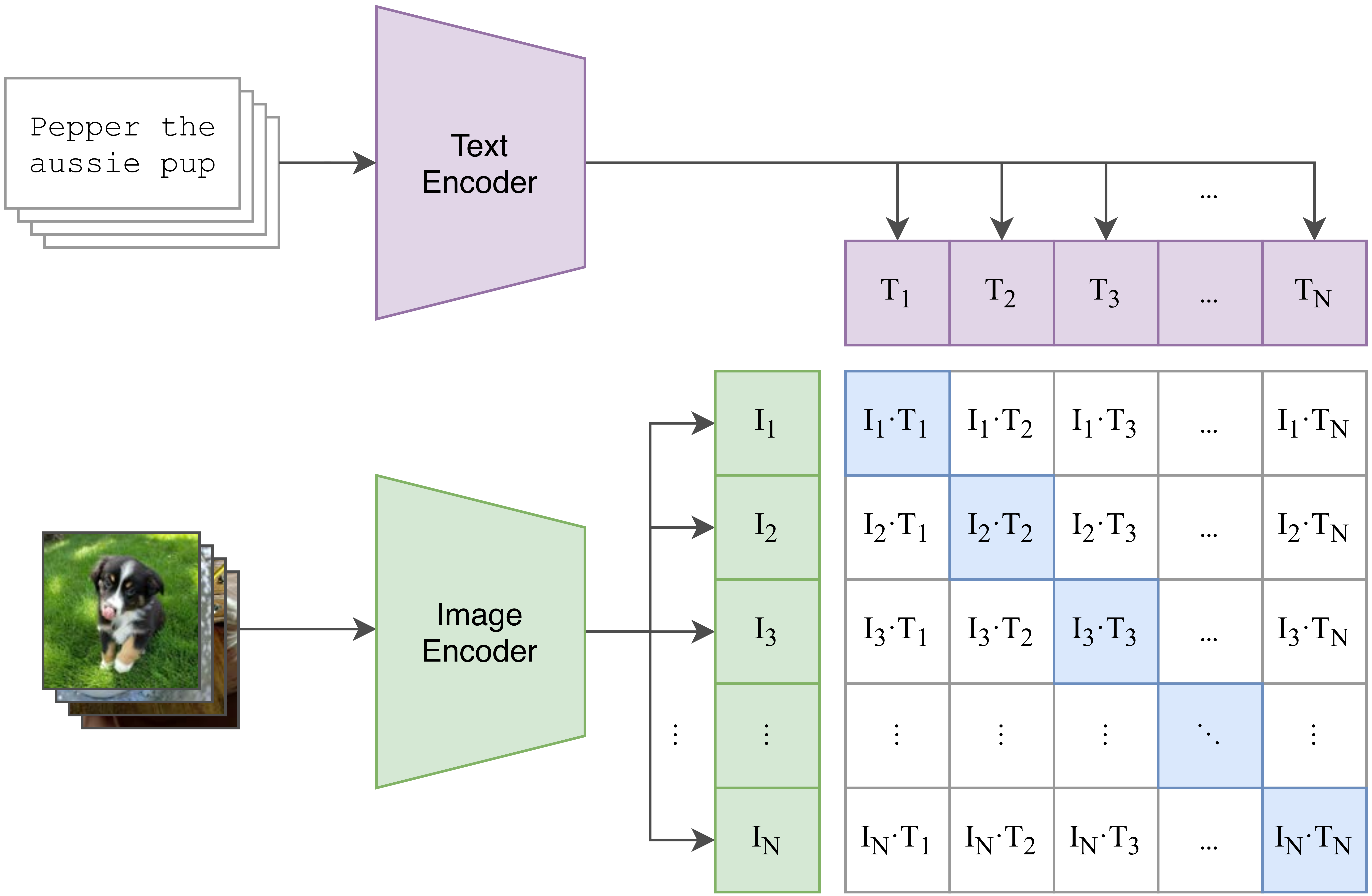
Chain-of-thought prompting

[Wei, Wang, Schuurmans et al. 2022]



CLIP [Radford et al., 2021] https://arxiv.org/pdf/2103.00020.pdf

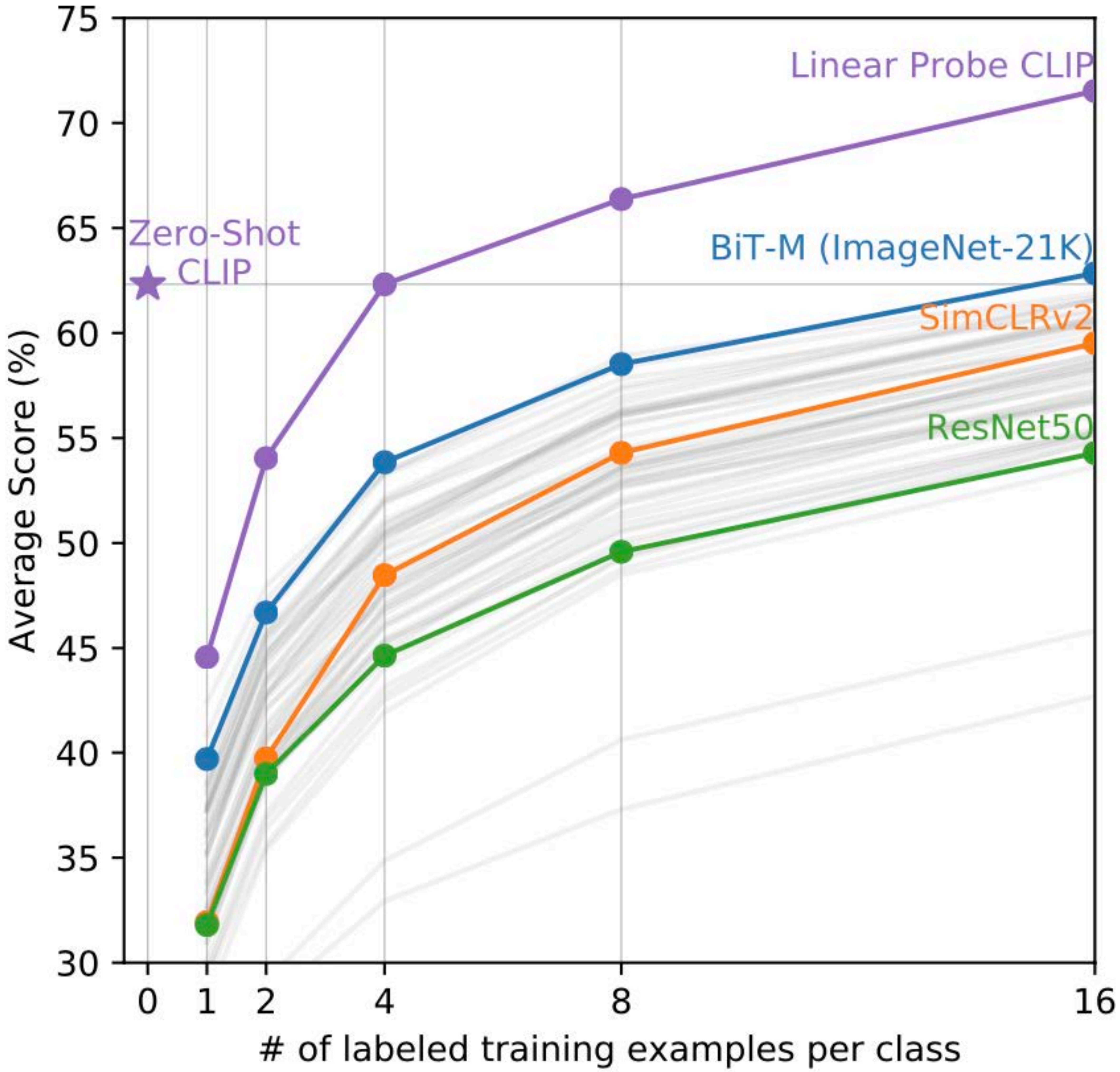
1. **Learner**: Contrastive Pretraining



[https://openai.com/blog/clip/]

CLIP [Radford et al., 2021] https://arxiv.org/pdf/2103.00020.pdf

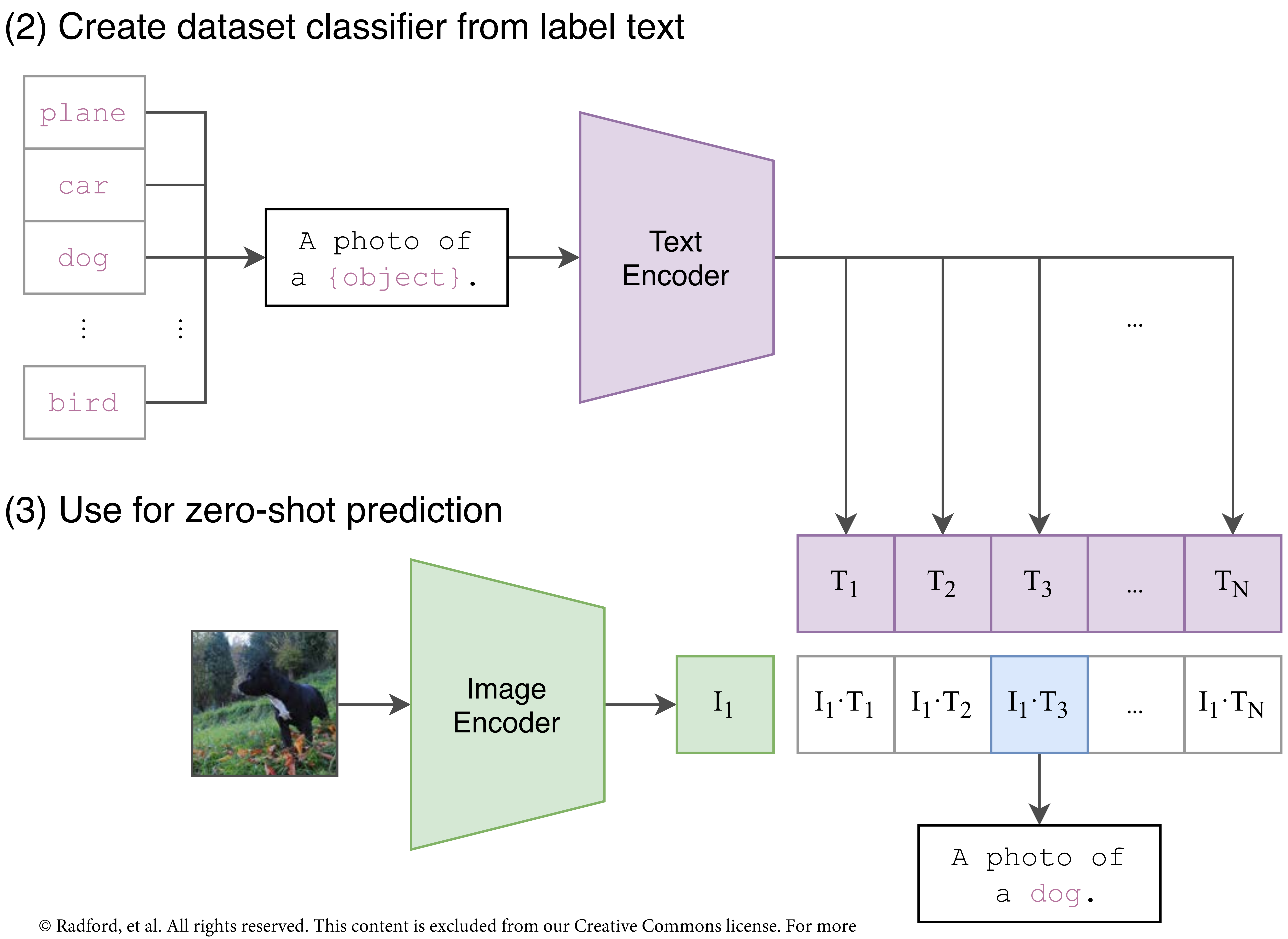
2. **Adaptor**:
Linear classifer on top of image encodings



[https://openai.com/blog/clip/]

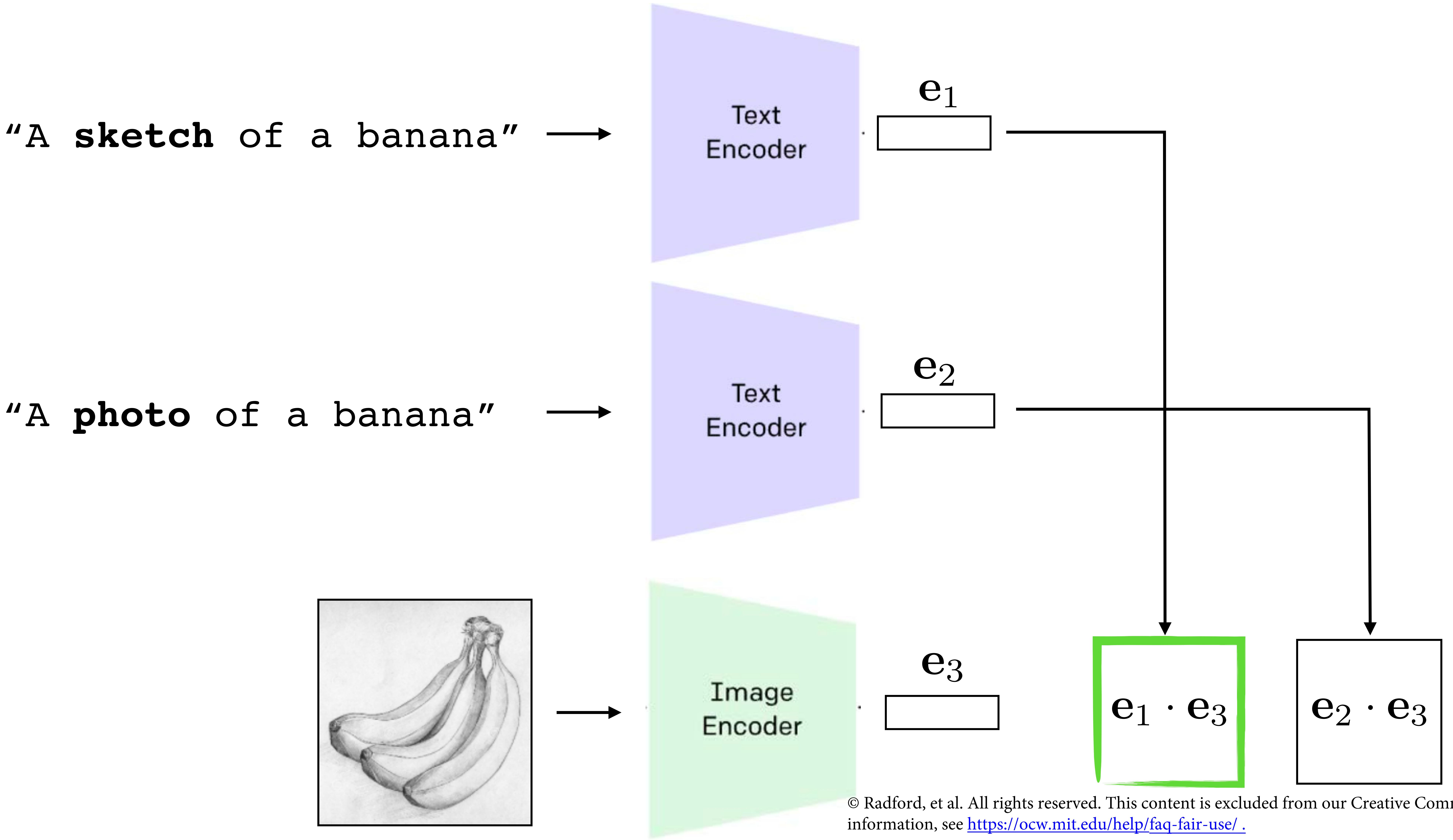
CLIP [Radford et al., 2021] https://arxiv.org/pdf/2103.00020.pdf

2. **Adaptor:**
Just ask



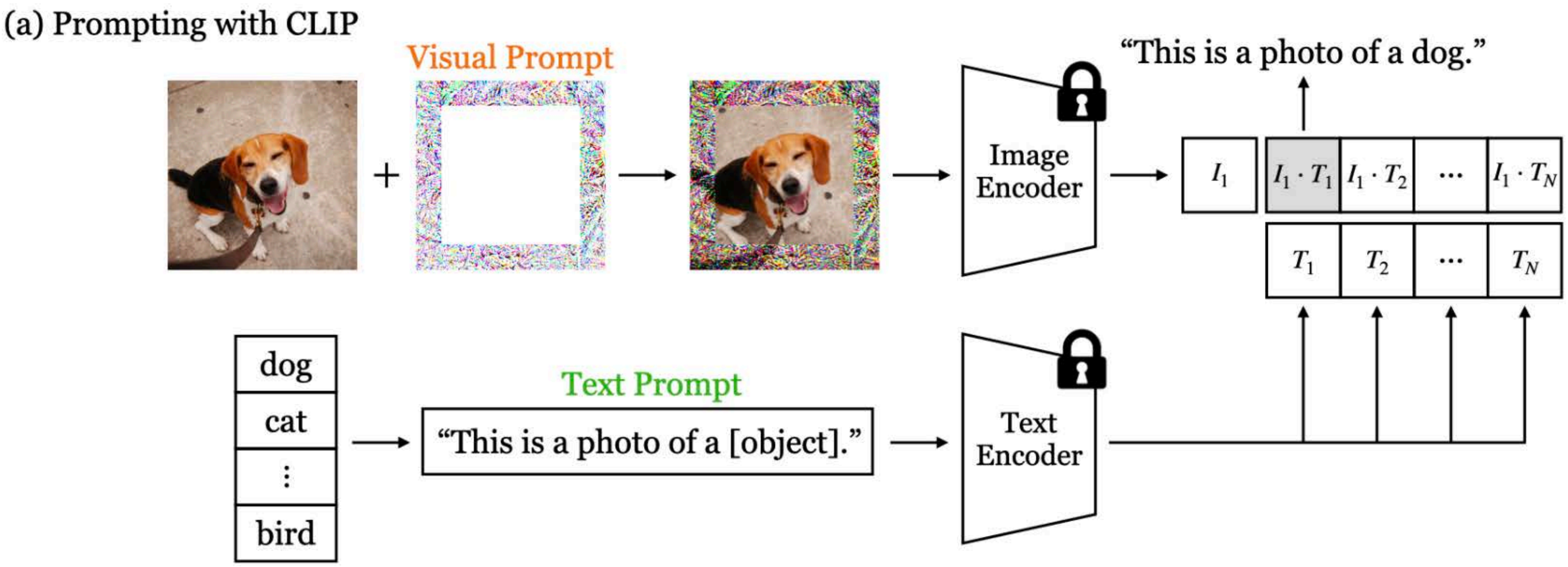
[https://openai.com/blog/clip/]

New capabilities by just asking



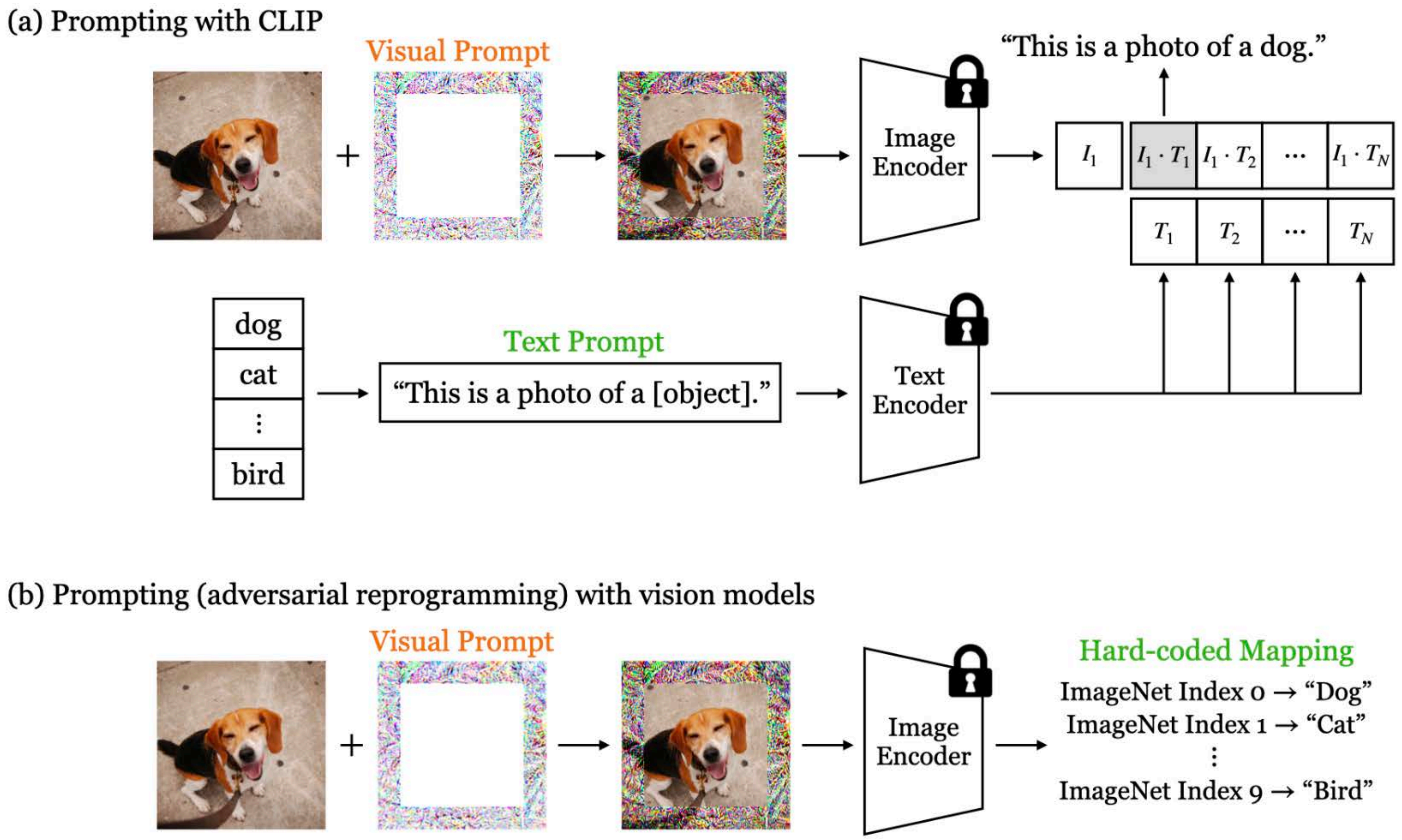
Prompting in other modalities

New capabilities by "visual prompting"



[Bahng, Jahanian, Sankaranarayanan, Isola 2022]

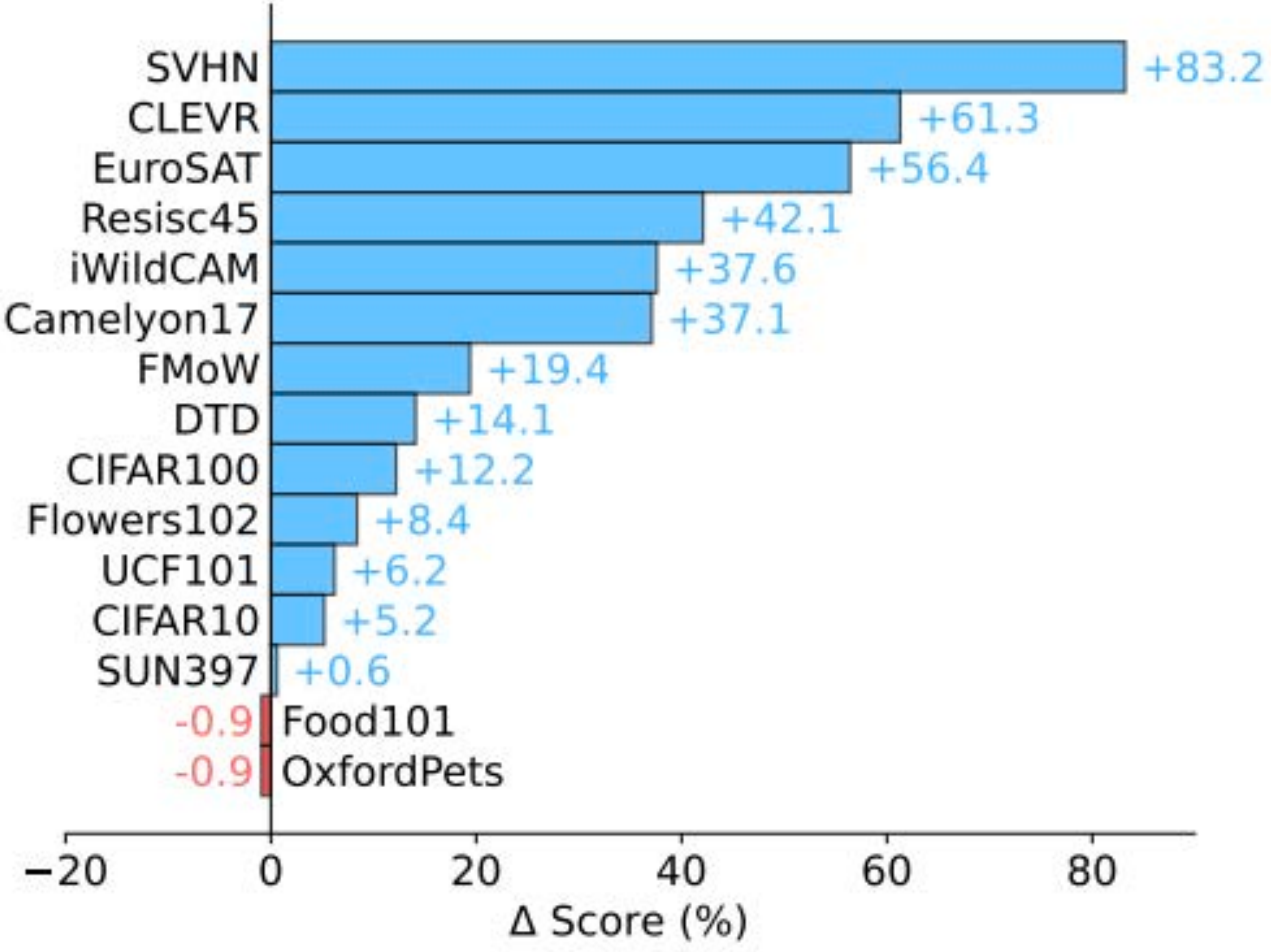
New capabilities by "visual prompting"



[Bahng, Jahanian, Sankaranarayanan, Isola 2022]

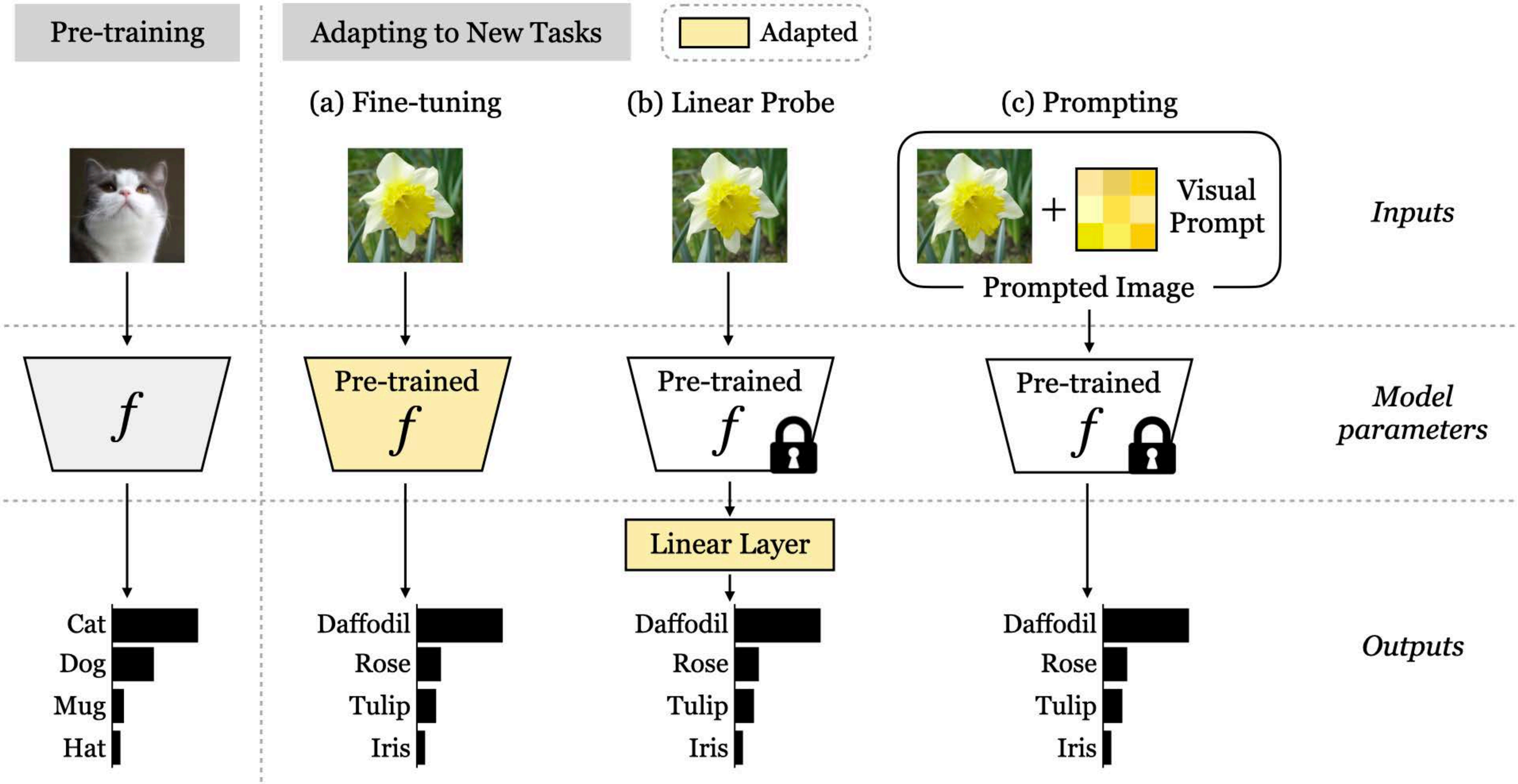
New capabilities by "visual prompting"

Adding a single pink pixel gives +3% acc boost



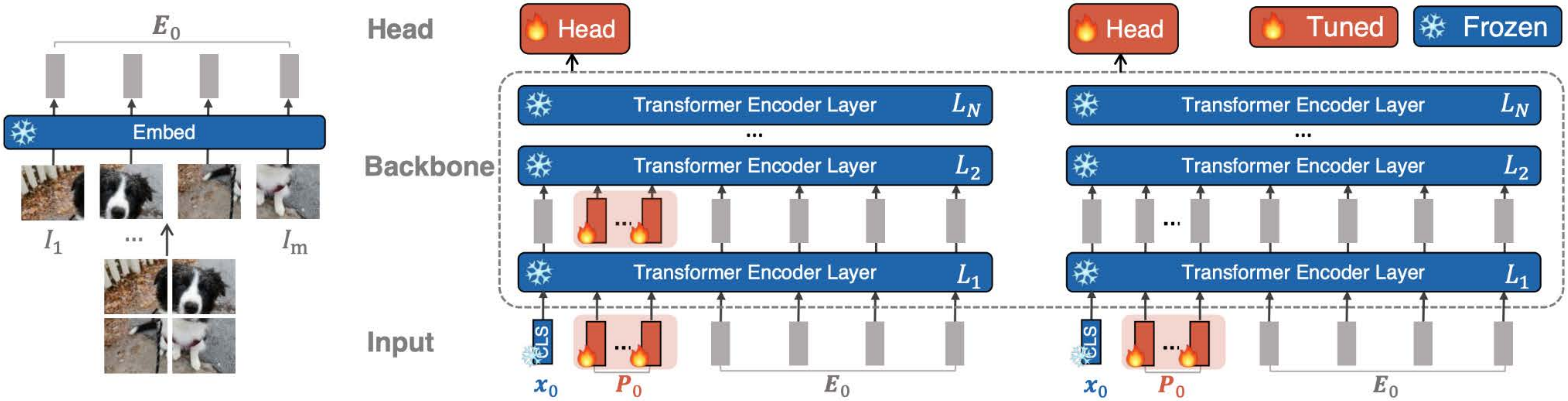
[Bahng, Jahanian, Sankaranarayanan, Isola 2022]

Adaptation can occur on inputs, mapping, or outputs



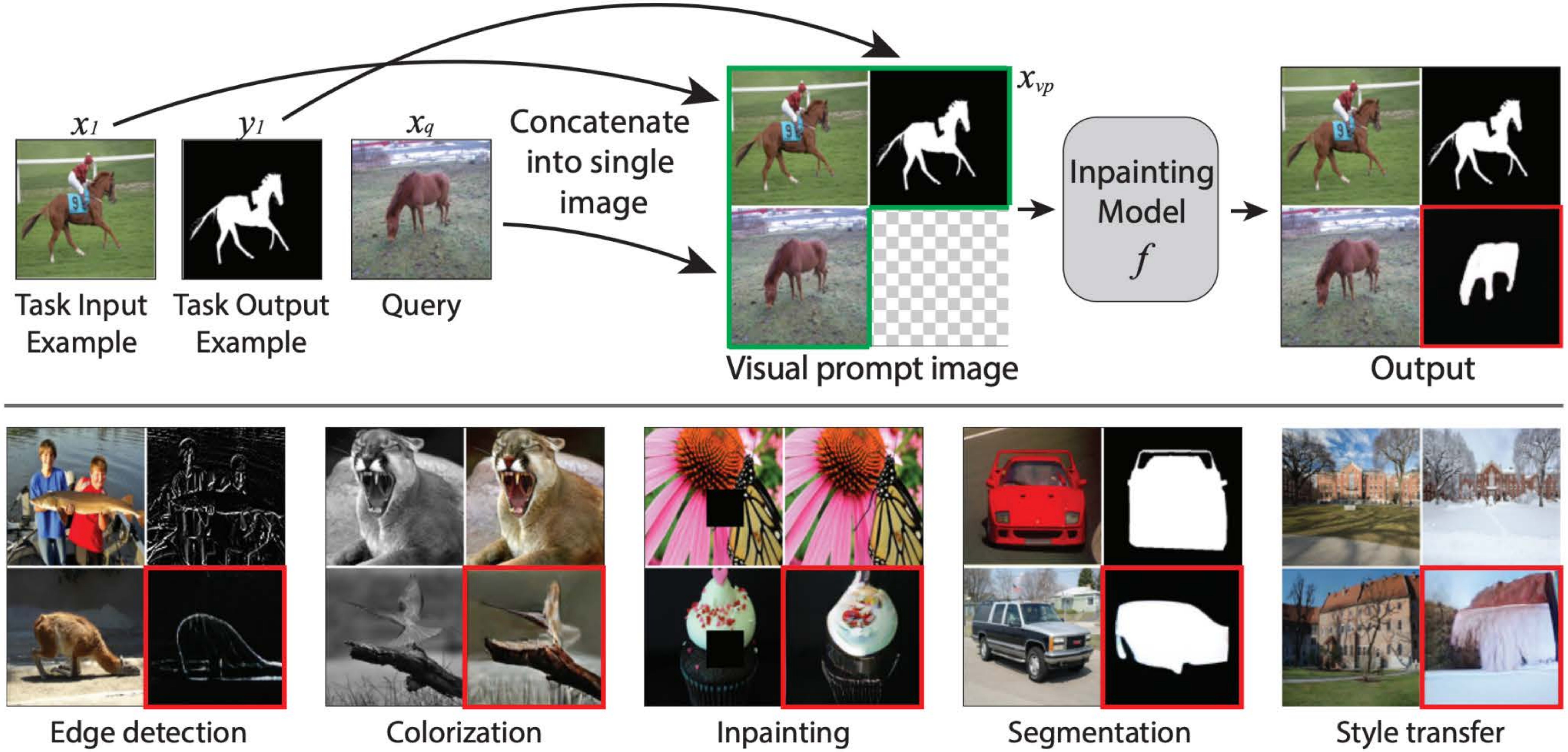
[Bahng, Jahanian, Sankaranarayanan, Isola 2022]

Visual Prompt Tuning (Input Encodings)



(a) Visual-Prompt Tuning: Deep (b) Visual-Prompt Tuning: Shallow

Visual Prompting via Image Inpainting



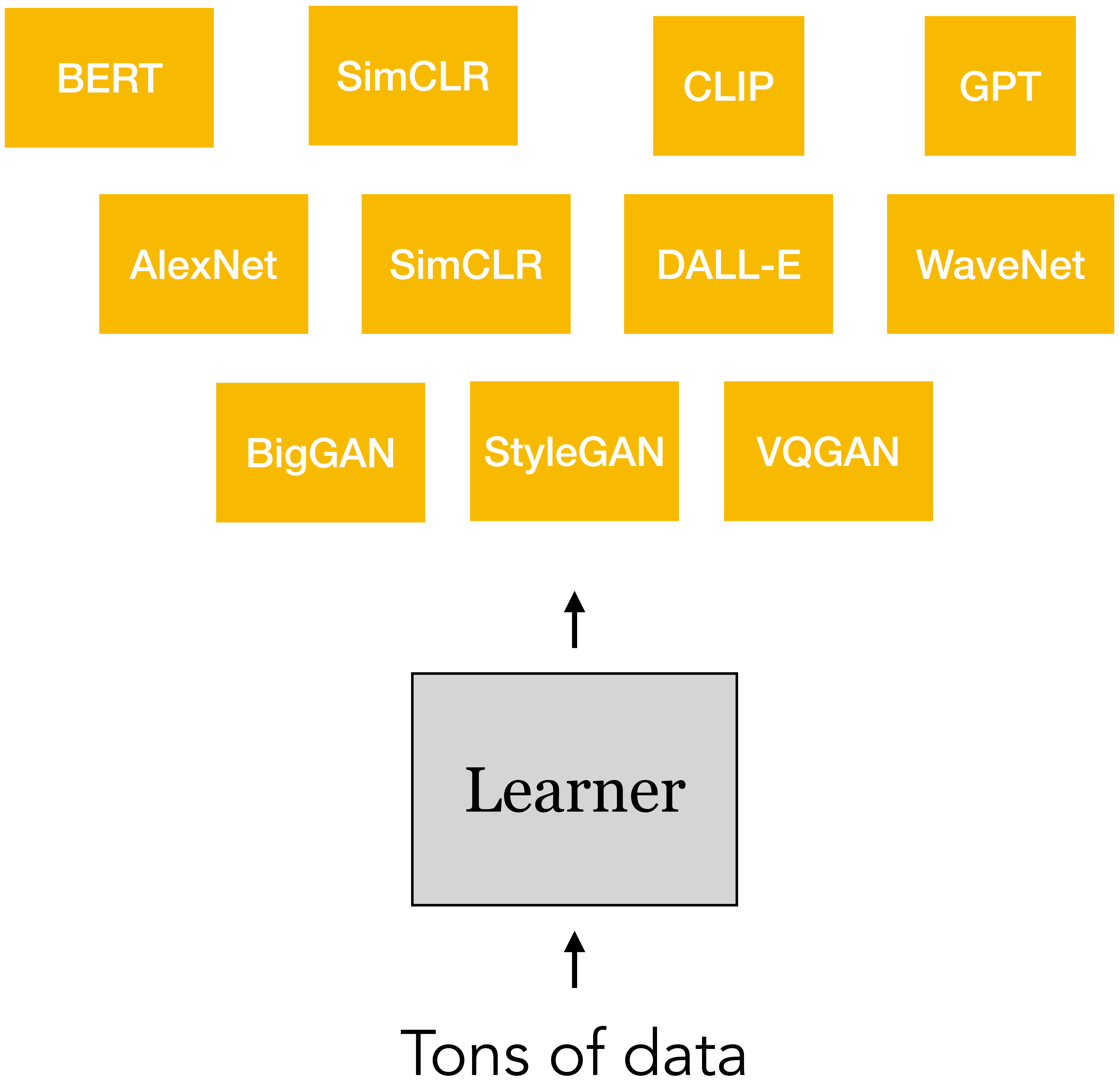
Visual Prompting via Image Inpainting.

Good Resources

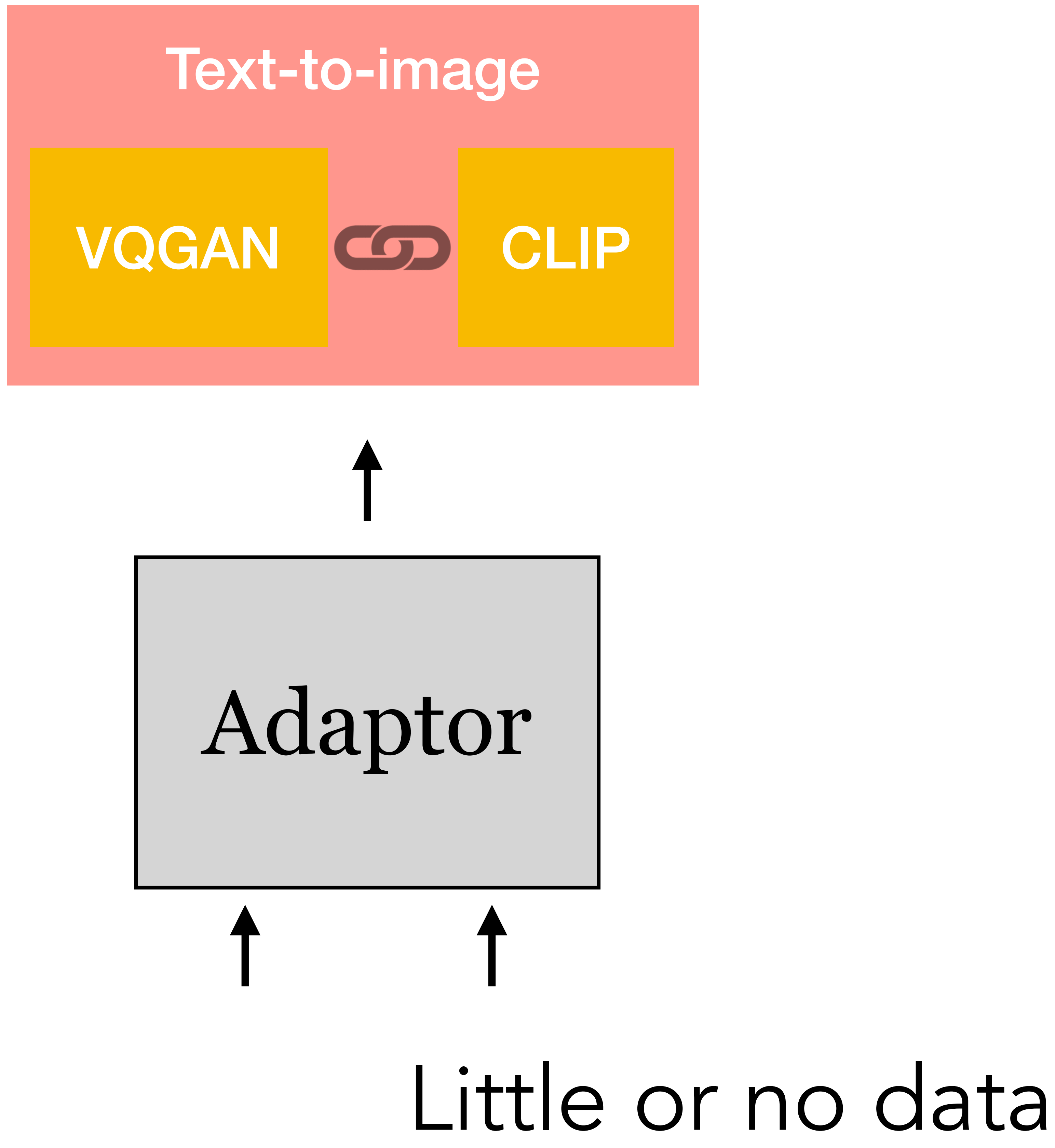
- https://github.com/thunlp/PromptPapers
- https://thegradient.pub/prompting/

Combining Foundation Models

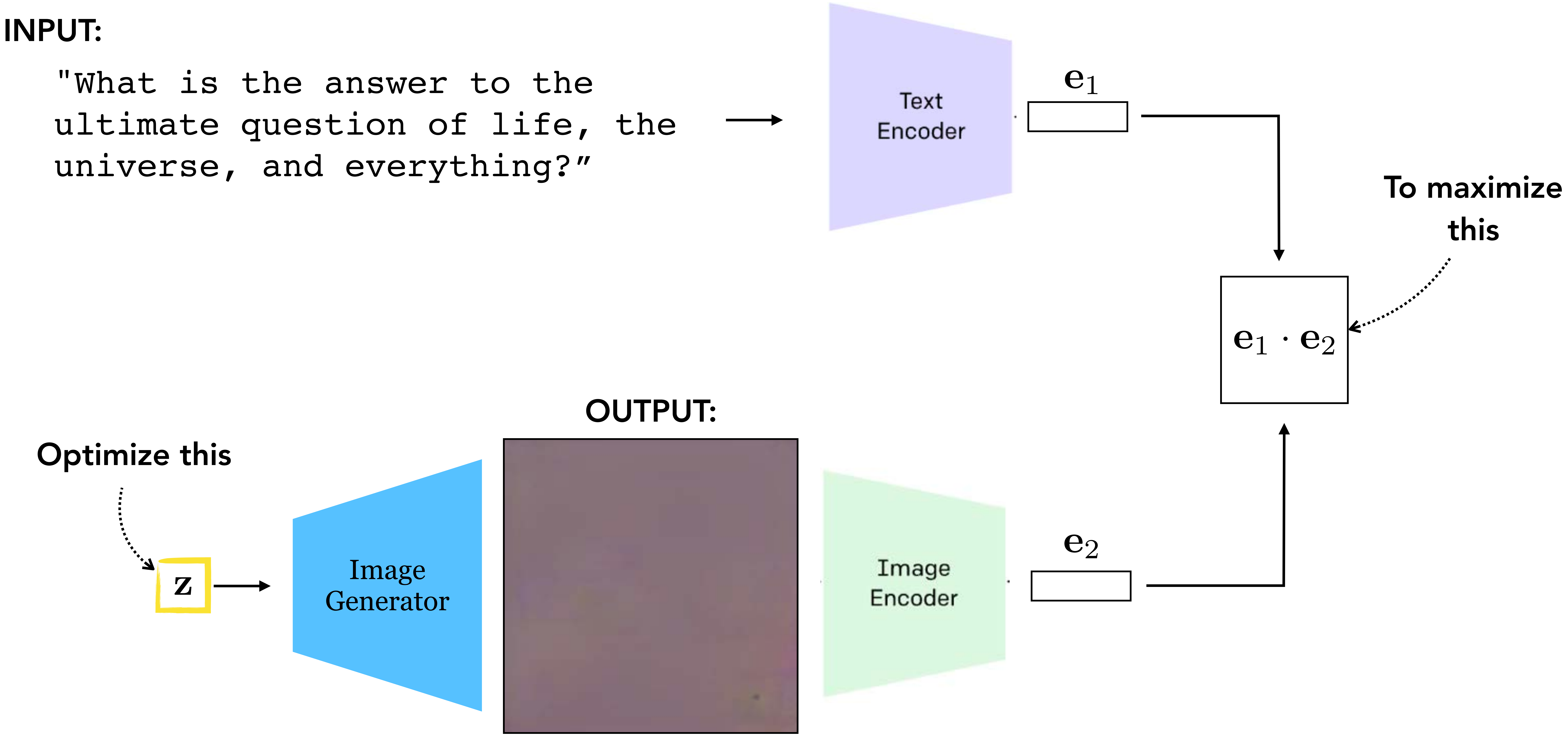
Learn foundation models



Use/adapt foundations to solve new problems

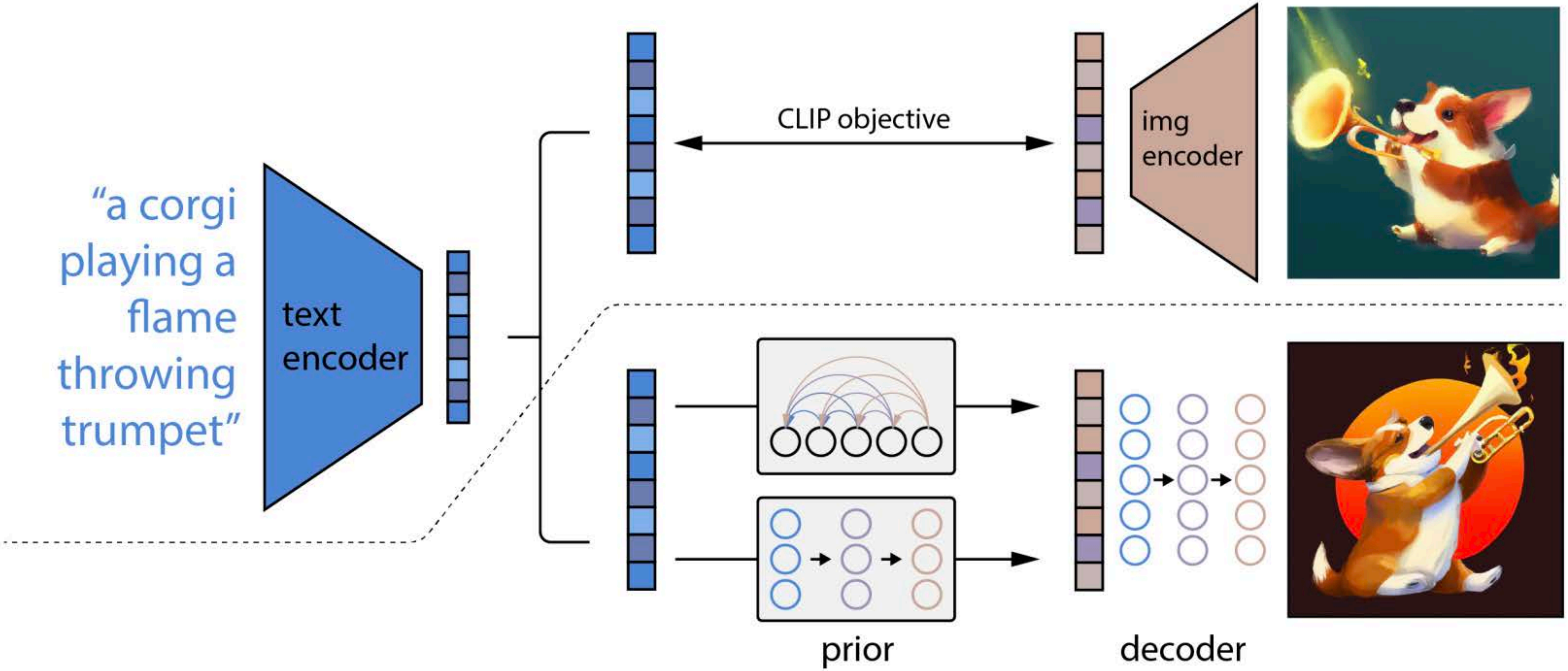


New capabilities by plugging pretrained models together: CLIP+GAN



Code: https://colab.research.google.com/drive/1_4PQqzM_0KKytCzWtn-ZPi4cCa5bwK2F?usp=sharing

DALL-E 2



[Ramesh et al. 2022]

Transfer Learning I

- Learning with little data
- Transferring knowledge about the mapping
    - Finetuning
    - Domain adaptation
- Transferring knowledge about the outputs
    - Knowledge distillation
- Foundation models