

Black-Box Meta-Learning & In-Context Learning

CS 330

Logistics

Project group form due **Wednesday Oct 11**

(Used for assigning TA mentors)

Homework 1 due **Monday October 16**

Following up on high-resolution feedback:

- Unfortunately not enough course staff for live remote questions this year (but feel free to ask on Ed!)
- I'll work on clarifying some terminology (e.g. "task")

Plan for Today

Meta-Learning

- Problem formulation
- General recipe of meta-learning algorithms
- Black-box adaptation approaches
- Case study of GPT-3

} **Topic of Homework 1!**

Goals for by the end of lecture:

- Training set-up for few-shot meta-learning algorithms
- How to implement black-box meta-learning techniques

Plan for Today

Meta-Learning

- **Problem formulation**
- General recipe of meta-learning algorithms
- Black-box adaptation approaches
- Case study of GPT-3

Meta-Learning Problem

Transfer Learning with Many Source Tasks

Given data from $\mathcal{T}_1, \dots, \mathcal{T}_n$, solve new task $\mathcal{T}_{\text{test}}$ more quickly / proficiently / stably

Key assumption: meta-training tasks and meta-test task drawn i.i.d. from same task distribution

$$\mathcal{T}_1, \dots, \mathcal{T}_n \sim p(\mathcal{T}), \mathcal{T}_{\text{test}} \sim p(\mathcal{T})$$

Like before, tasks must share structure.

What do the tasks correspond to?

- recognizing handwritten digits from different languages (see homework 1!)
- giving feedback to students on different exams
- classifying species in different regions of the world
- a robot performing different tasks



How many tasks do you need?

The more the better.

(analogous to more data in ML)

Two ways to view meta-learning algorithms

Mechanistic view

- Deep network that can read in an entire dataset and make predictions for new datapoints
- Training this network uses a meta-dataset, which itself consists of many datasets, each for a different task

Probabilistic view

- Extract prior knowledge from a set of tasks that allows efficient learning of new tasks
- Learning a new task uses this prior and (small) training set to infer most likely posterior parameters

How does meta-learning work? An example.

Given 1 example of 5 classes:



training data $\mathcal{D}_{\text{train}}$

Classify new examples



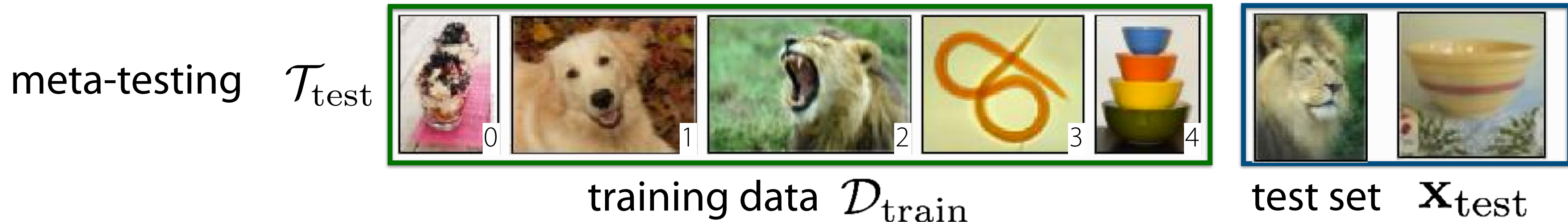
test set \mathbf{X}_{test}

How does meta-learning work? An example.



Given 1 example of 5 classes:

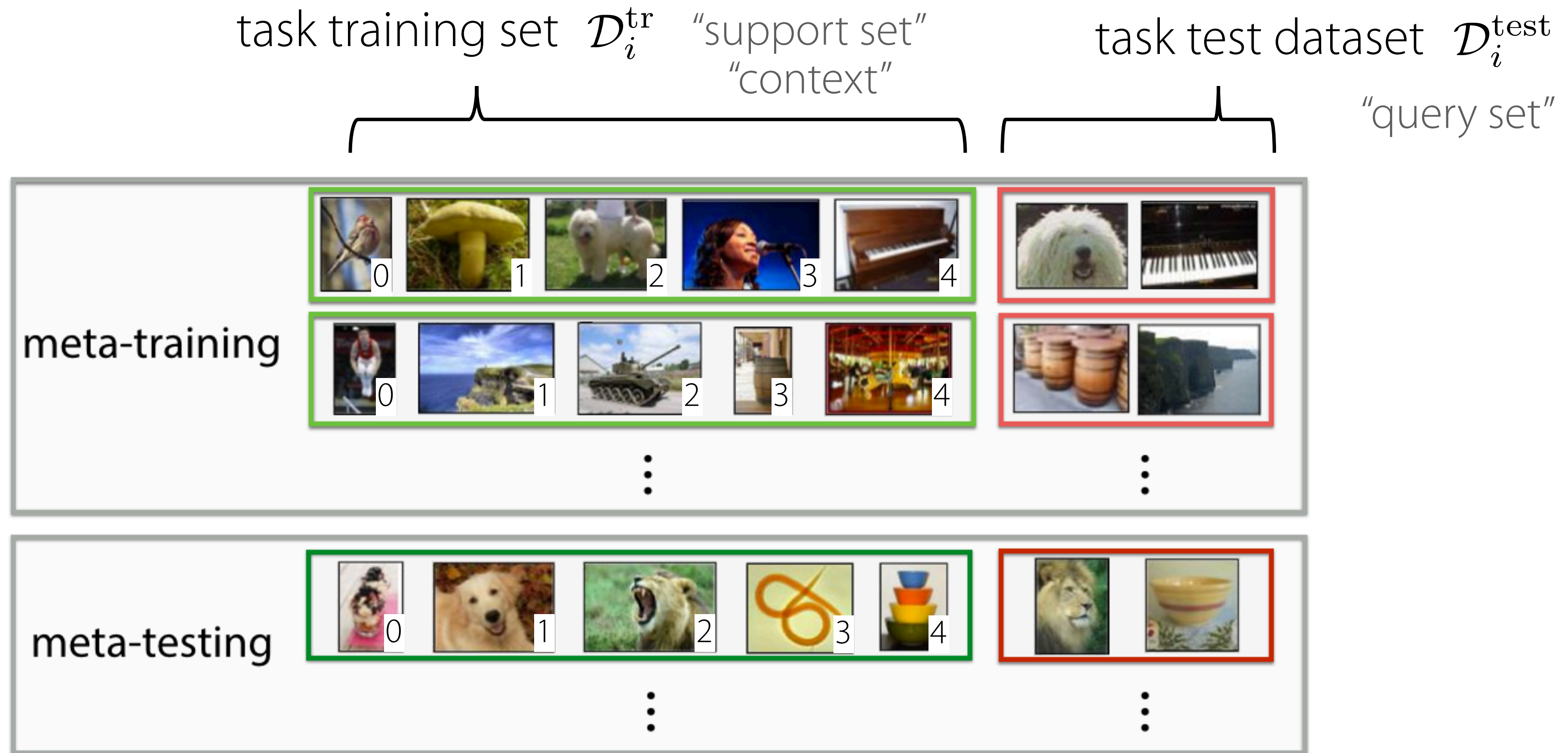
Classify new examples



Can replace image classification with: regression, language generation, skill learning,

any ML problem

Some terminology



k-shot learning: learning with **k** examples per class
(or **k** examples total for regression)

N-way classification: choosing between N classes

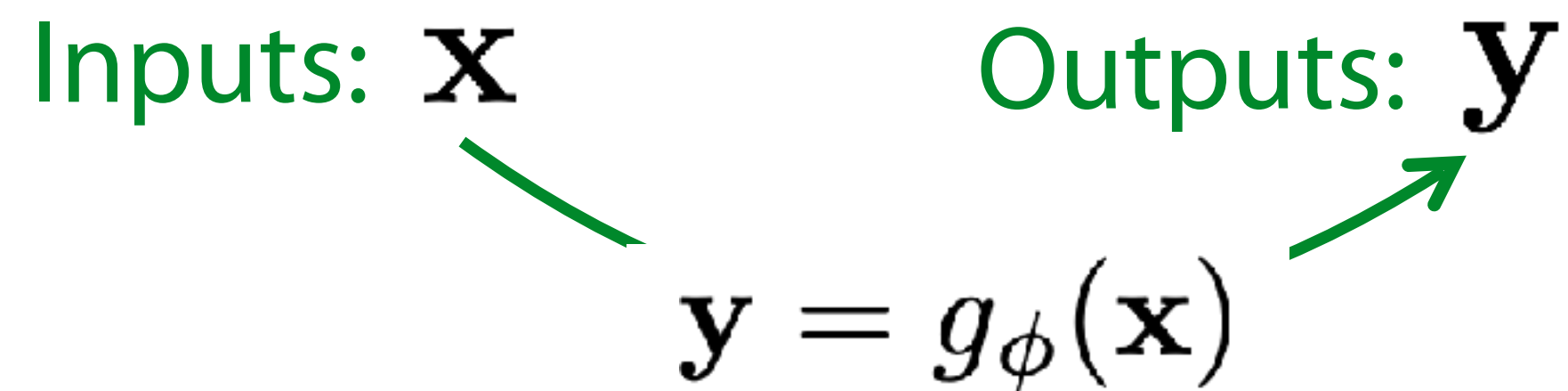
Plan for Today

Meta-Learning

- Problem formulation
- **General recipe of meta-learning algorithms**
- Black-box adaptation approaches
- Case study of GPT-3 (time-permitting)

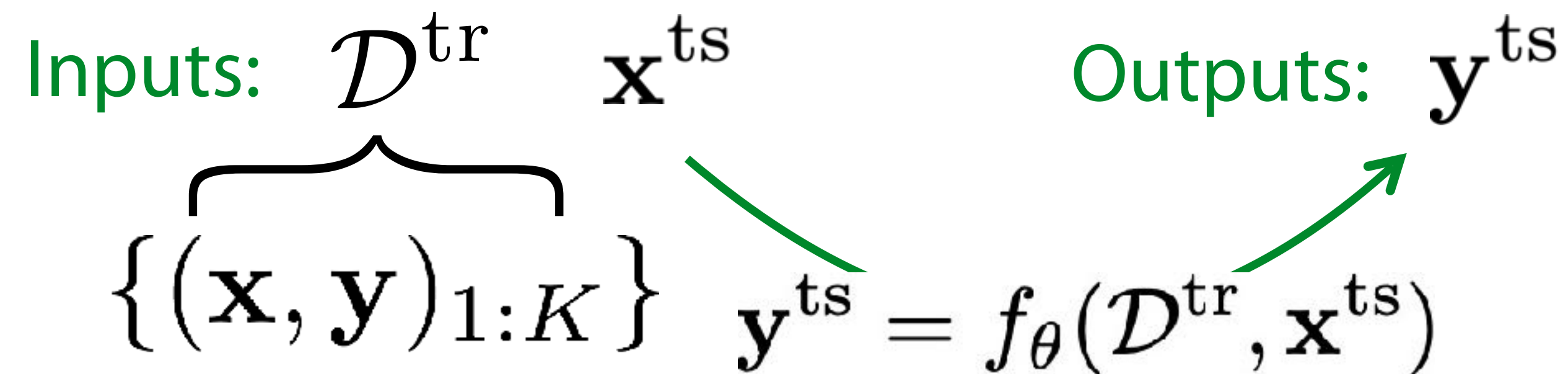
One View on the Meta-Learning Problem

Supervised Learning:



Data: $\{(\mathbf{x}, \mathbf{y})_i\}$

Meta Supervised Learning:



Data: $\{\mathcal{D}_i\}$

$\mathcal{D}_i : \{(\mathbf{x}, \mathbf{y})_j\}$

Why is this view useful?

Reduces the meta-learning problem to the design & optimization of f .

General recipe

How to design a meta-learning algorithm

1. Choose a form of $f_{\theta}(\mathcal{D}^{\text{tr}}, \mathbf{x}^{\text{ts}})$

2. Choose how to optimize θ w.r.t. max-likelihood objective using meta-training data

 meta-parameters

Plan for Today

Meta-Learning

- Problem formulation
- General recipe of meta-learning algorithms
- **Black-box adaptation approaches**
- Case study of GPT-3 (time-permitting)

Running example

Omniglot dataset Lake et al. Science 2015

1623 characters from 50 different alphabets

Hebrew	Bengali	Greek	Futurama
א ב ג ד ה	ঐ ঐ আ ন ত শ ঙ	φ λ β δ λ	ঐ ঐ ঐ ঐ ঐ ঐ
ו ז ח ט י	ক য় ঞ ও ট ব	μ α κ χ ν	ঐ ঐ ঐ ঐ ঐ ঐ
כ ল ম ন ס	দ থ শ ঞ এ ই ঙ	υ θ γ τ σ	ঐ ঐ ঐ ঐ ঐ ঐ
פ צ	শ হ ঙ ড ঞ গ য়	ω π η ο ε	ঐ ঐ ঐ ঐ ঐ ঐ
ק	ঙ ত ছ ঞ য় উ থ	ρ ξ ζ ψ	ঐ ঐ ঐ ঐ ঐ ঐ
ר	চ গ ত ঞ ঙ ট ঞ		
ש	ঐ ফ ঞ ব		

20 instances of each character

whiteboard

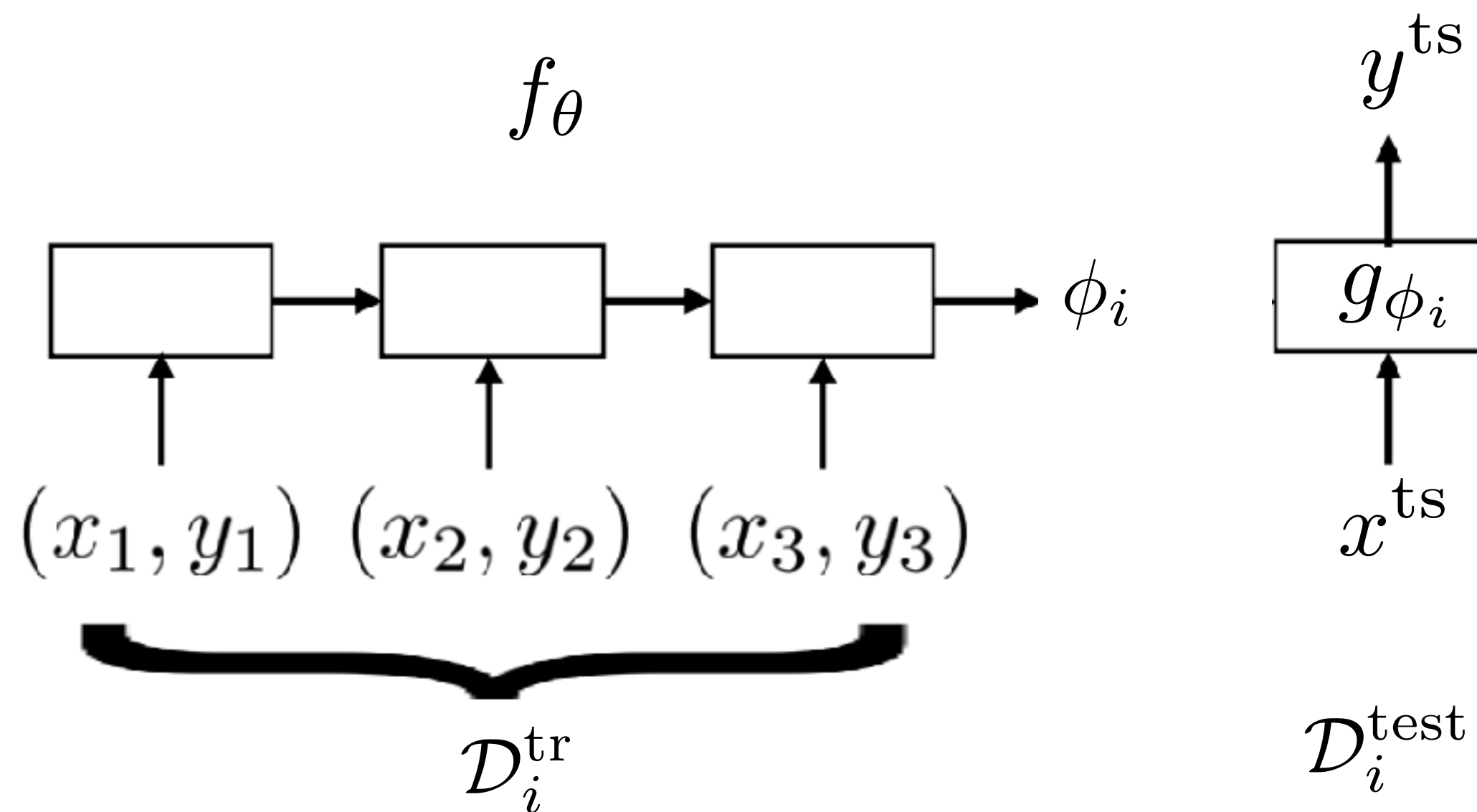
many classes, few examples
the “transpose” of MNIST
statistics more reflective
of the real world

More few-shot image recognition datasets: tieredImageNet, CIFAR, CUB, CelebA, ORBIT, others

More benchmarks: molecular property prediction (Ngyugen et al. '20), object pose prediction (Yin et al. ICLR '20), channel coding (Li et al. '21)

Black-Box Adaptation

Key idea: Train a neural network to represent $\phi_i = f_\theta(\mathcal{D}_i^{\text{tr}})$ “learner”
Predict test points with $\mathbf{y}^{\text{ts}} = g_{\phi_i}(\mathbf{x}^{\text{ts}})$



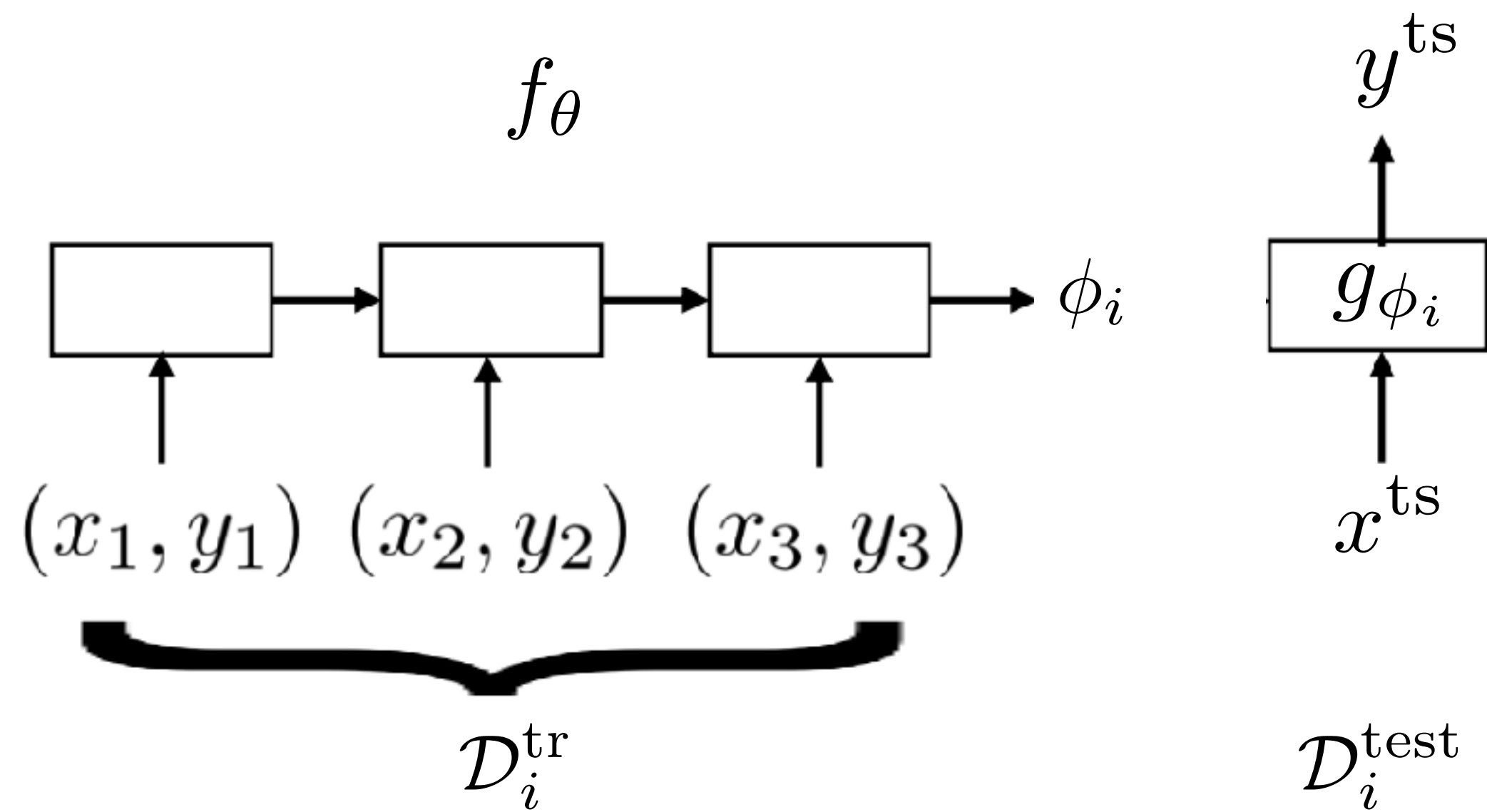
Train with standard supervised learning!

$$\min_{\theta} \sum_{\mathcal{T}_i} \sum_{(x,y) \sim \mathcal{D}_i^{\text{test}}} \underbrace{-\log g_{\phi_i}(y | x)}_{\mathcal{L}(\phi_i, \mathcal{D}_i^{\text{test}})}$$

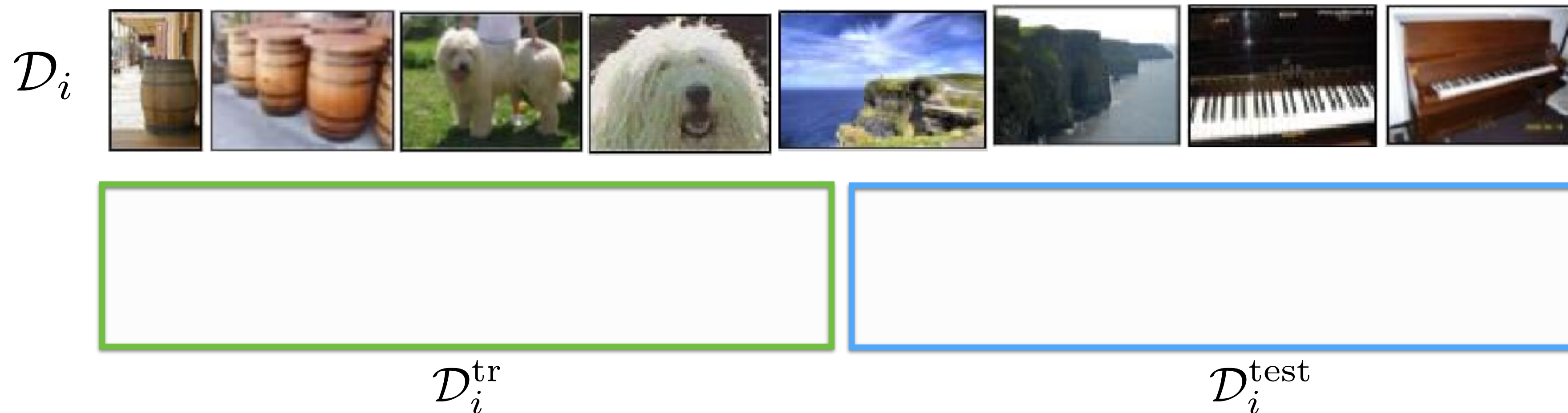
$$\min_{\theta} \sum_{\mathcal{T}_i} \mathcal{L}(f_\theta(\mathcal{D}_i^{\text{tr}}), \mathcal{D}_i^{\text{ts}})$$

Black-Box Adaptation

Key idea: Train a neural network to represent $\phi_i = f_\theta(\mathcal{D}_i^{\text{tr}})$.

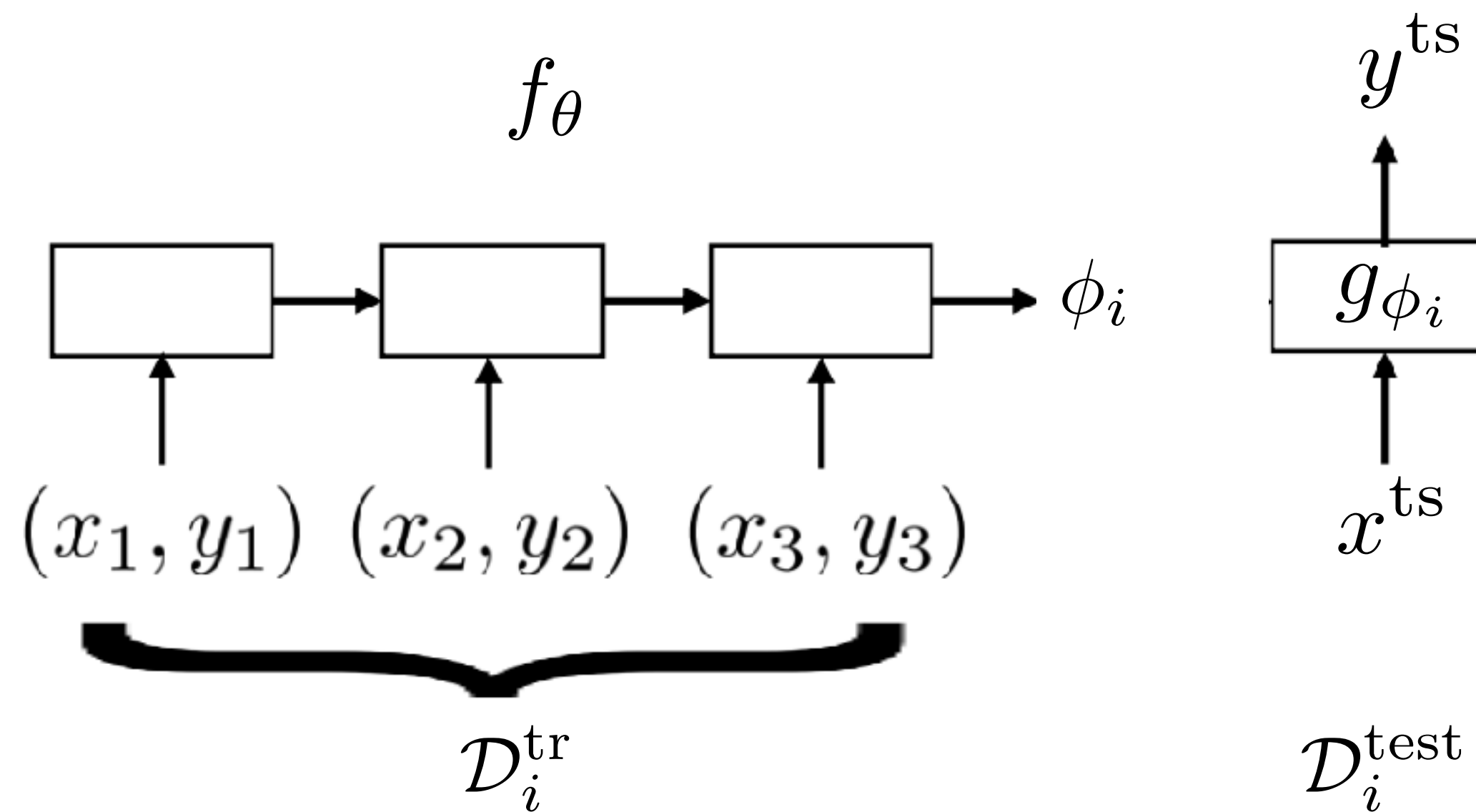


1. Sample task \mathcal{T}_i (or mini batch of tasks)
2. Sample disjoint datasets $\mathcal{D}_i^{\text{tr}}$, $\mathcal{D}_i^{\text{test}}$ from \mathcal{D}_i

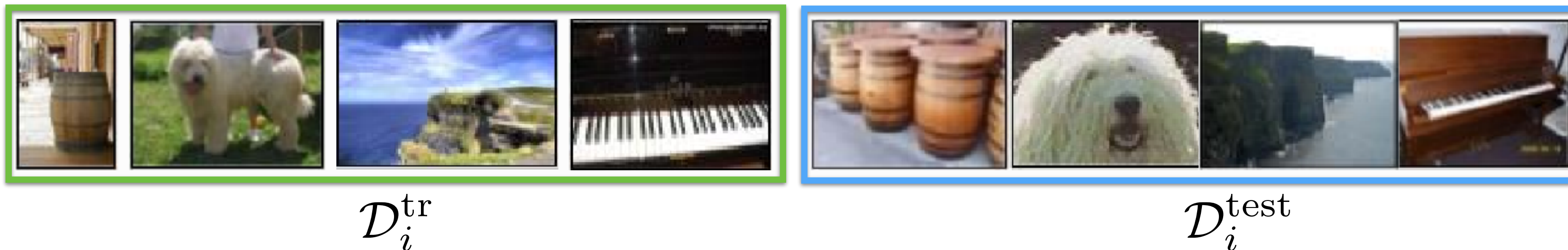


Black-Box Adaptation

Key idea: Train a neural network to represent $\phi_i = f_\theta(\mathcal{D}_i^{\text{tr}})$.



1. Sample task \mathcal{T}_i (or mini batch of tasks)
2. Sample disjoint datasets $\mathcal{D}_i^{\text{tr}}$, $\mathcal{D}_i^{\text{test}}$ from \mathcal{D}_i
3. Compute $\phi_i \leftarrow f_\theta(\mathcal{D}_i^{\text{tr}})$
4. Update θ using $\nabla_\theta \mathcal{L}(\phi_i, \mathcal{D}_i^{\text{test}})$



Black-Box Adaptation

Key idea: Train a neural network to represent $\phi_i = f_\theta(\mathcal{D}_i^{\text{tr}})$.

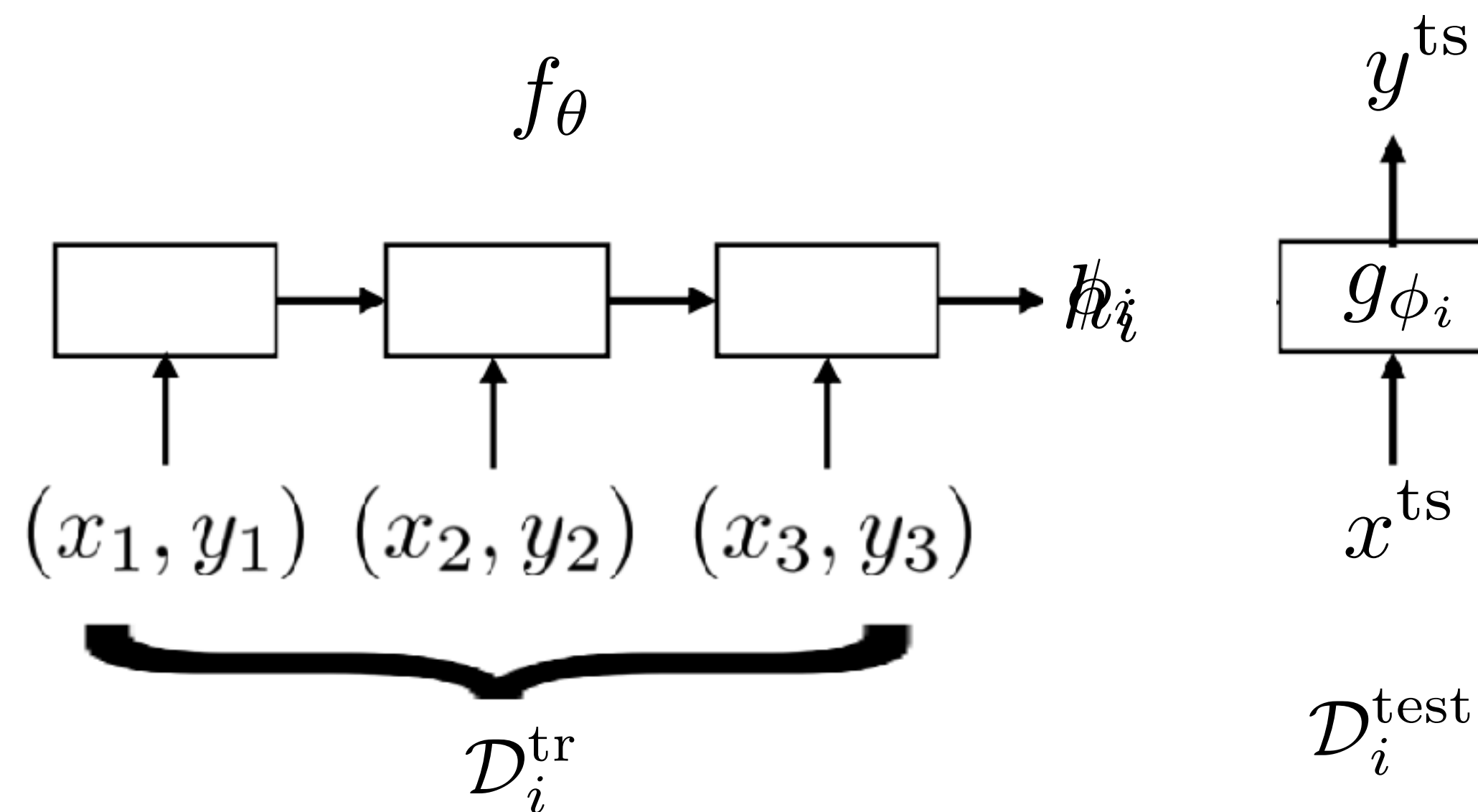
Challenge

Outputting all neural net parameters does not seem scalable?

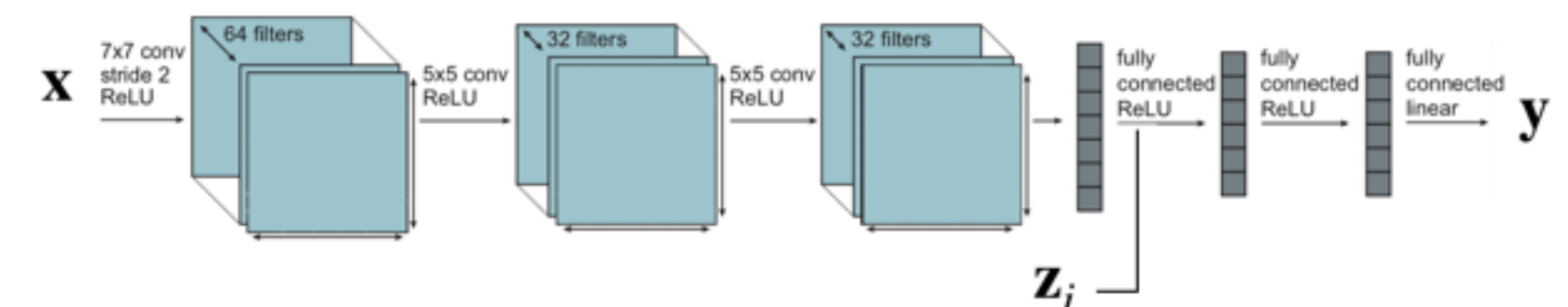
Idea: Do not need to output **all** parameters of neural net, only sufficient statistics (Santoro et al. MANN, Mishra et al. SNAIL)

low-dimensional vector h_i
represents contextual task information

$$\phi_i = \{h_i, \theta_g\}$$



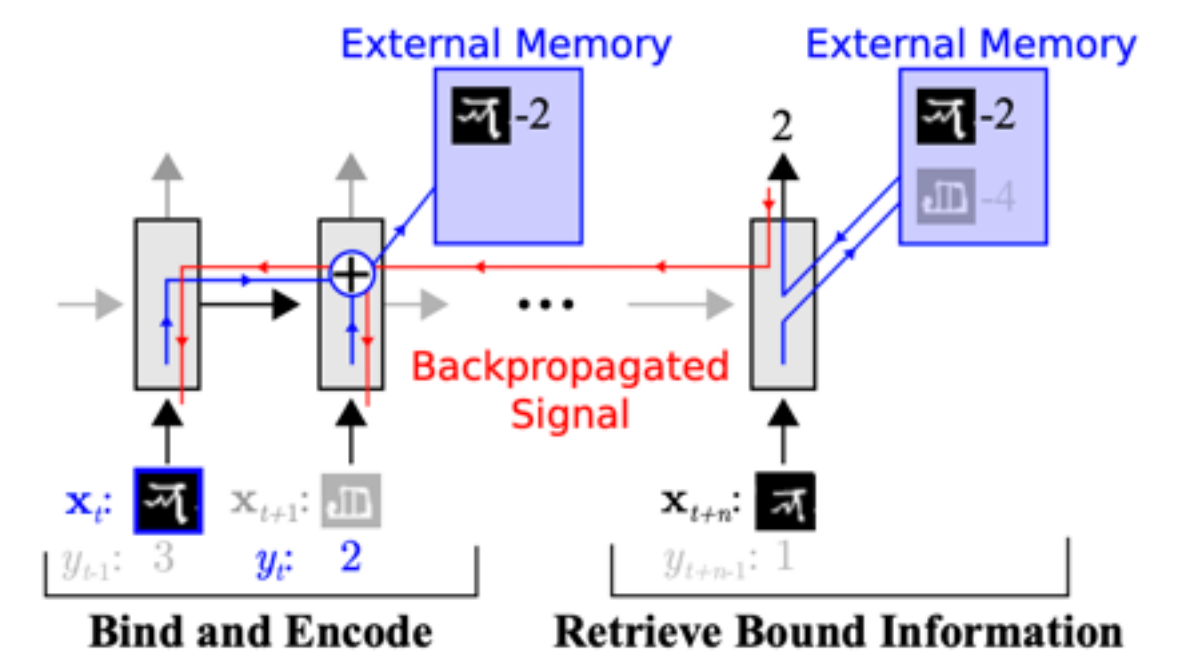
recall:



general form: $y^{\text{ts}} = f_\theta(\mathcal{D}_i^{\text{tr}}, x^{\text{ts}})$

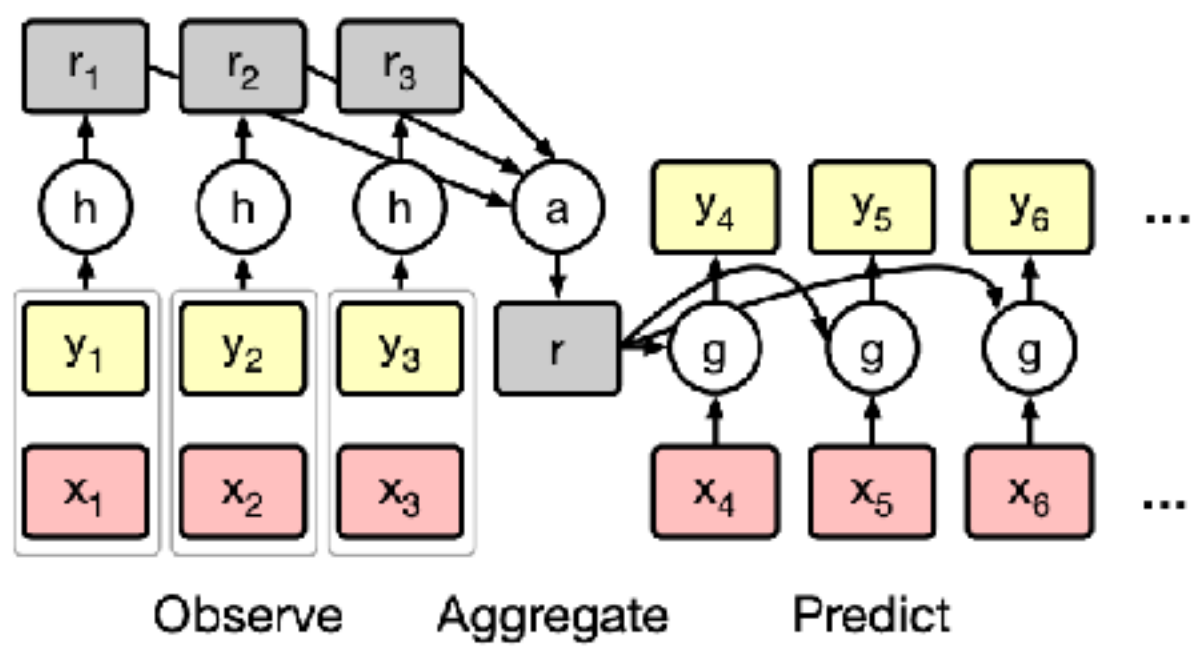
Black-Box Adaptation Architectures

LSTMs or Neural Turing Machine (NTM)



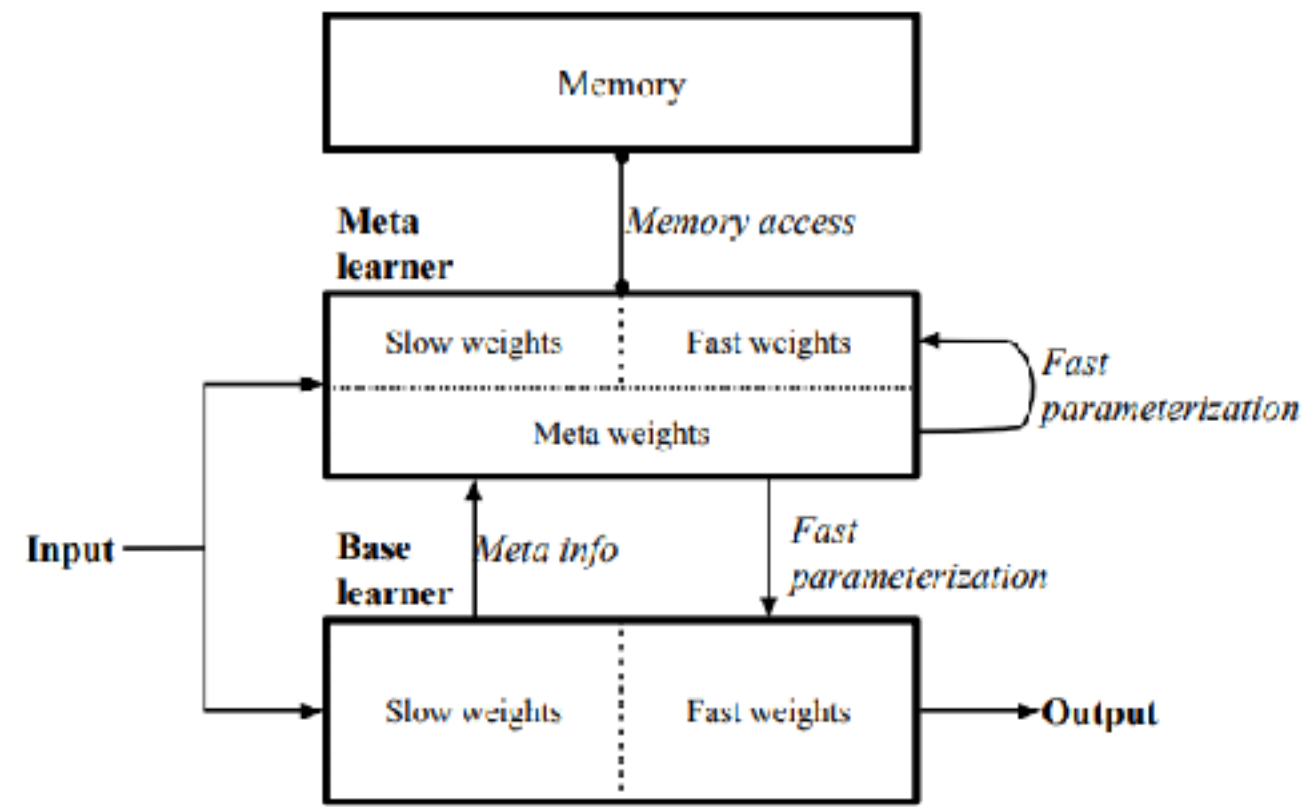
Meta-Learning with Memory-Augmented Neural Networks
Santoro, Bartunov, Botvinick, Wierstra, Lillicrap. ICML '16

Feedforward + average



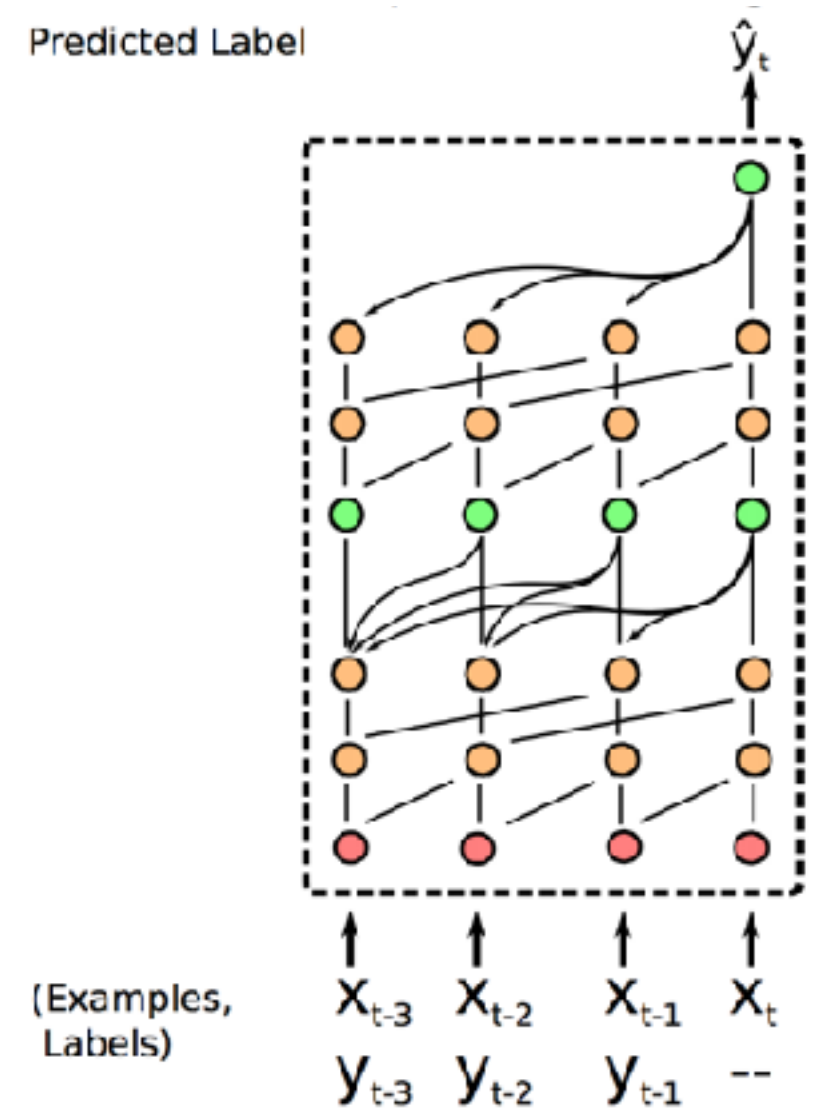
Conditional Neural Processes. Garnelo, Rosenbaum, Maddison, Ramalho, Saxton, Shanahan, Teh, Rezende, Eslami. ICML '18

Other external memory mechanisms



Meta Networks
Munkhdalai, Yu. ICML '17

Convolutions & attention



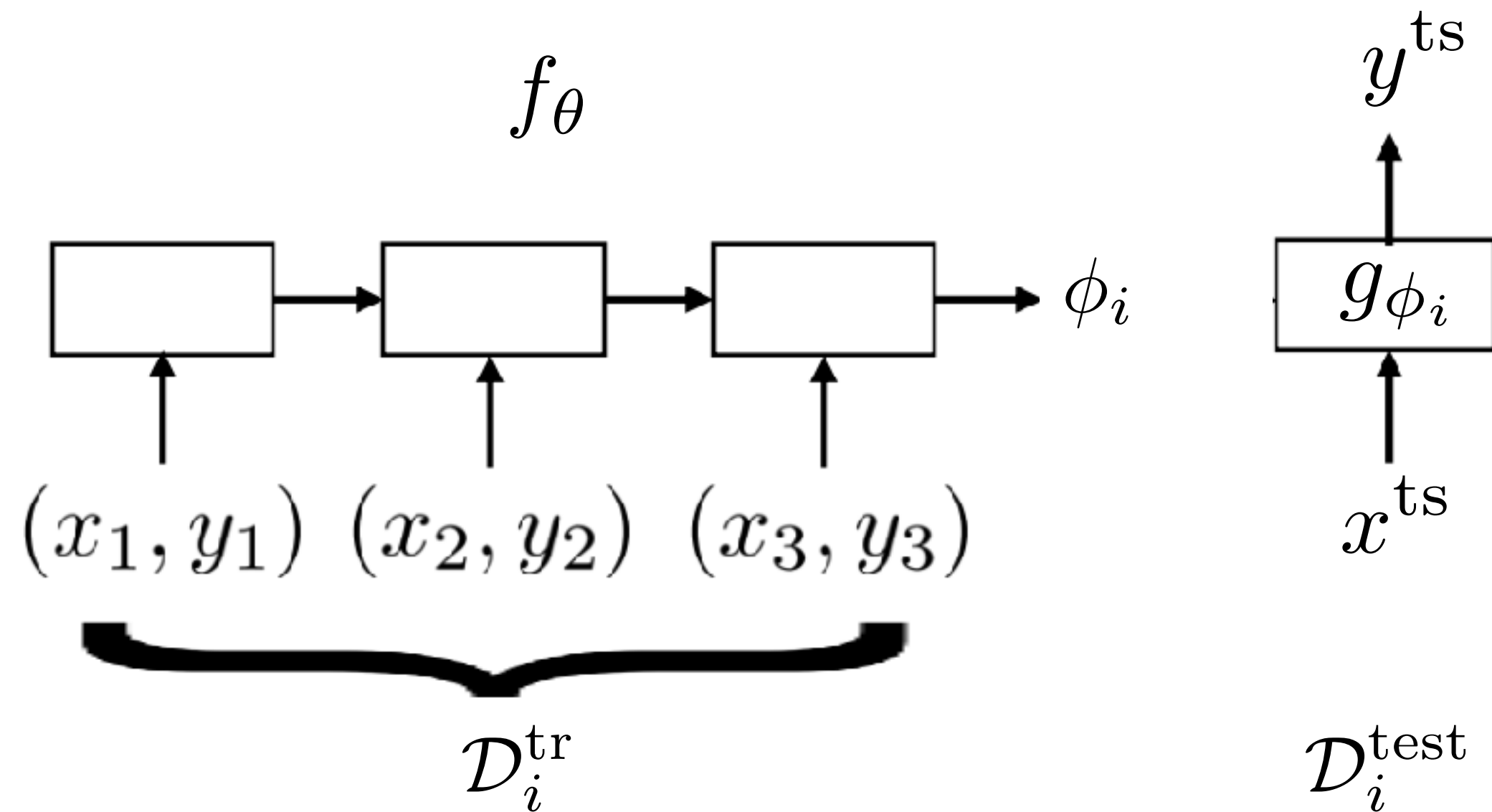
A Simple Neural Attentive Meta-Learner
Mishra, Rohaninejad, Chen, Abbeel. ICLR '18

Method	HW 1:
SNAIL, Ours	- implement data processing
	- implement simple black-box meta-learner
	- train few-shot Omniglot classifier

Question: Why might feedforward+average be better than a recurrent model?

Black-Box Adaptation

Key idea: Train a neural network to represent $\phi_i = f_\theta(\mathcal{D}_i^{\text{tr}})$.



+ **expressive**

+ easy to combine with **variety of learning problems** (e.g. SL, RL)

- **complex model w/ complex task:**
challenging optimization problem
- often **data-inefficient**

How else can we represent $\phi_i = f_\theta(\mathcal{D}_i^{\text{tr}})$?

Next time (Monday): What if we treat it as an **optimization** procedure?

Plan for Today

Meta-Learning

- Problem formulation
- General recipe of meta-learning algorithms
- Black-box adaptation approaches
- **Case study of GPT-3**

Case Study: GPT-3

Language Models are Few-Shot Learners

Tom B. Brown* **Benjamin Mann*** **Nick Ryder*** **Melanie Subbiah***
Jared Kaplan[†] **Prafulla Dhariwal** **Arvind Neelakantan** **Pranav Shyam** **Girish Sastry**
Amanda Askell **Sandhini Agarwal** **Ariel Herbert-Voss** **Gretchen Krueger** **Tom Henighan**
Rewon Child **Aditya Ramesh** **Daniel M. Ziegler** **Jeffrey Wu** **Clemens Winter**
Christopher Hesse **Mark Chen** **Eric Sigler** **Mateusz Litwin** **Scott Gray**
Benjamin Chess **Jack Clark** **Christopher Berner**
Sam McCandlish **Alec Radford** **Ilya Sutskever** **Dario Amodei**

OpenAI

May 2020

“emergent” few-shot learning

What is GPT-3?

a language model

black-box meta-learner trained on language generation tasks

$\mathcal{D}_i^{\text{tr}}$: sequence of characters $\mathcal{D}_i^{\text{ts}}$: the following sequence of characters

[meta-training] dataset: crawled data from the internet, English-language Wikipedia, two books corpora

architecture: giant “Transformer” network 175 billion parameters, 96 layers, 3.2M batch size

What do different tasks correspond to?

spelling correction

simple math problems

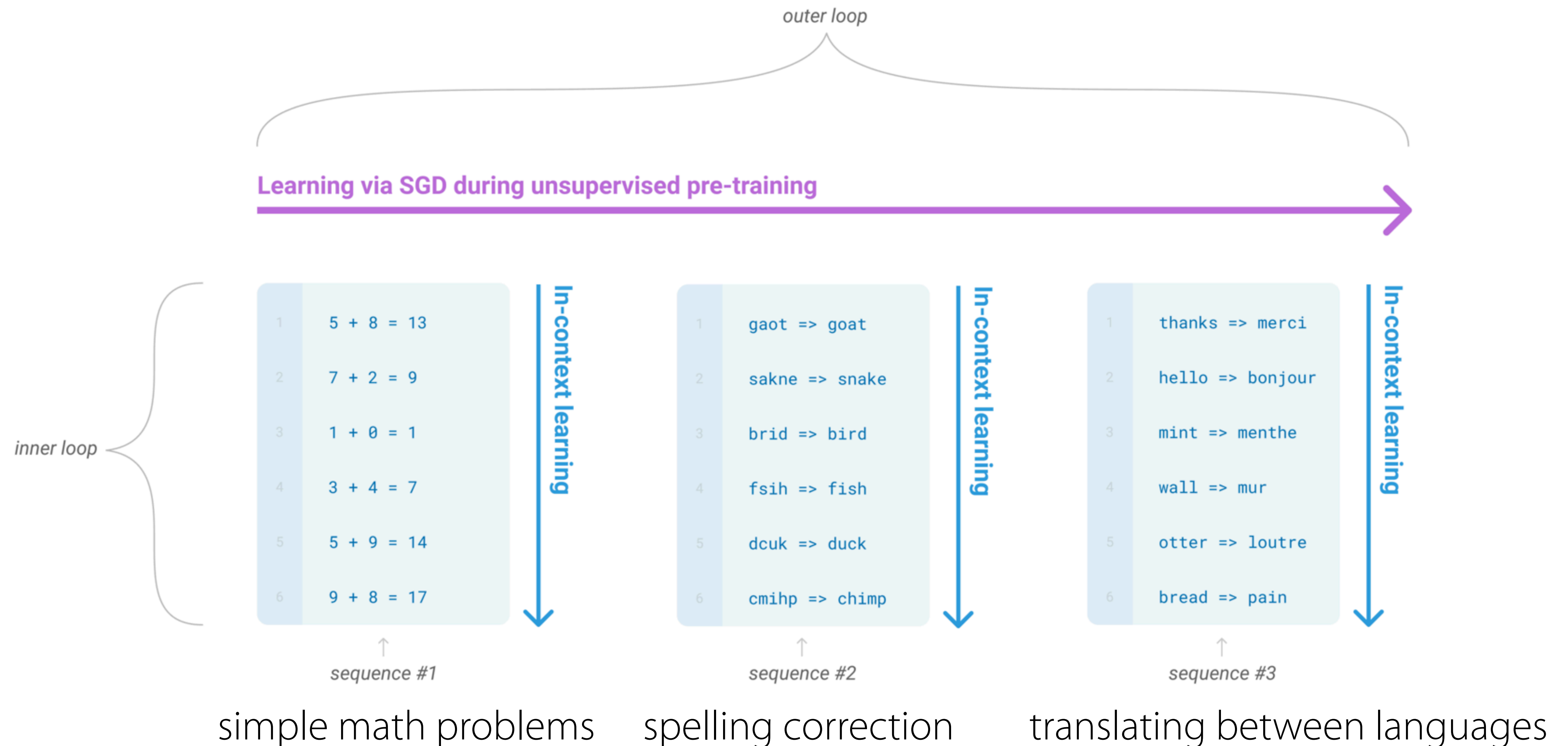
translating between languages

a variety of other tasks

How can those tasks all be solved by a single architecture?

How can those tasks all be solved by a single architecture? Put them all in the form of text!

Why is that a good idea? Very easy to get a lot of meta-training data.



Some Results

One-shot learning from dictionary definitions:

To "screeg" something is to swing a sword at it. An example of a sentence that uses the word screeg is:
We screeghed at each other for several minutes and then we went outside and ate ice cream.

Few-shot language editing:

Poor English input: I eated the purple berries.
Good English output: I ate the purple berries.
Poor English input: Thank you for picking me as your designer. I'd appreciate it.
Good English output: Thank you for choosing me as your designer. I appreciate it.
Poor English input: The mentioned changes have done. or I did the alteration that you requested. or I changed things you wanted and did the modifications.
Good English output: The requested changes have been made. or I made the alteration that you requested. or I changed things you wanted and made the modifications.
Poor English input: I'd be more than happy to work with you in another project.
Good English output: I'd be more than happy to work with you on another project.

Poor English input: Please provide me with a short brief of the design you're looking for and that'd be nice if you could share some examples or project you did before.
Good English output: Please provide me with a brief description of the design you're looking for and that would be nice if you could share some examples or projects you have done before.

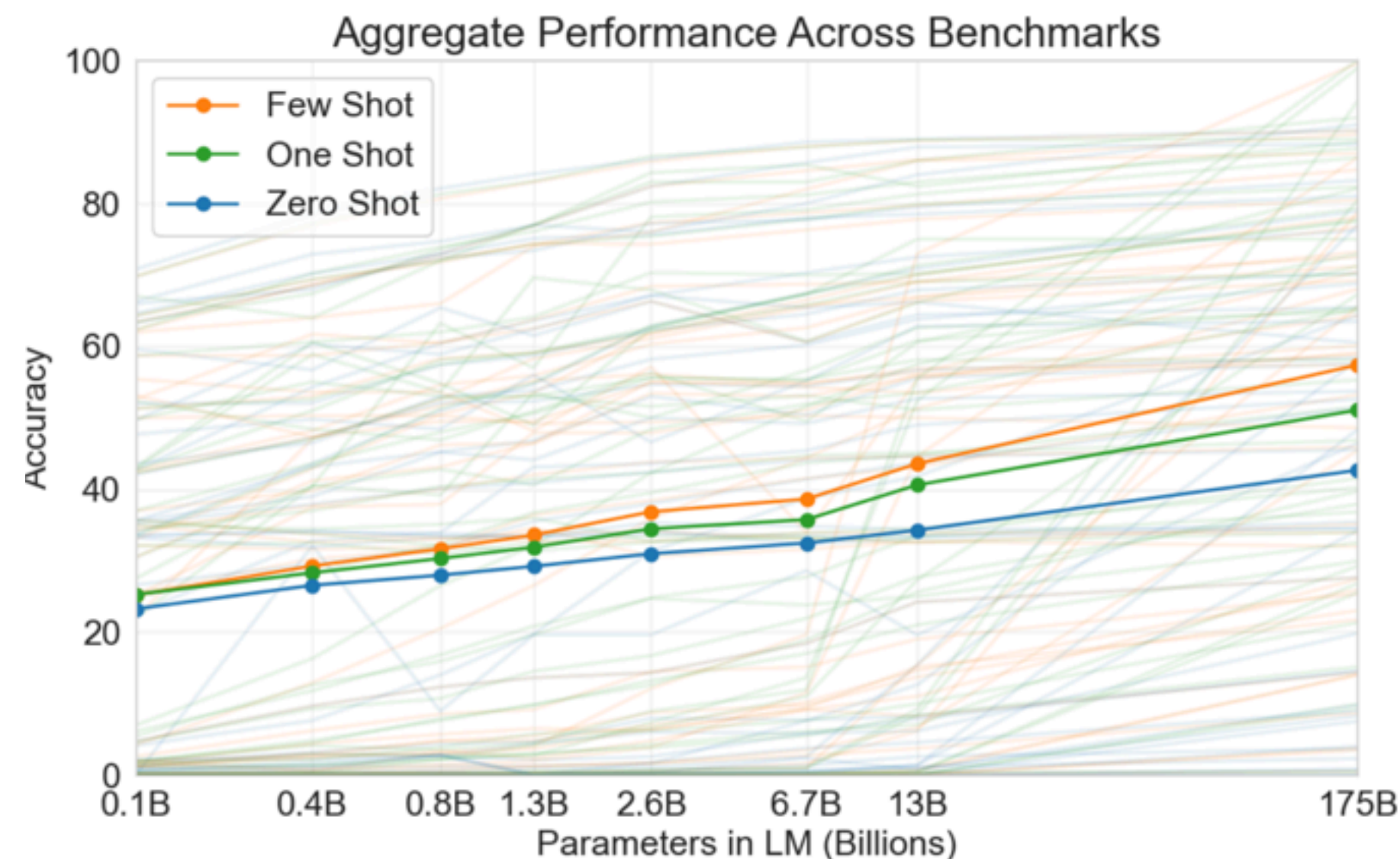
Non-few-shot learning tasks:

Title: United Methodists Agree to Historic Split
Subtitle: Those who oppose gay marriage will form their own denomination
Article: **After two days of intense debate, the United Methodist Church has agreed to a historic split - one that is expected to end in the creation of a new denomination, one that will be "theologically and socially conservative," according to The Washington Post. The majority of delegates attending the church's annual General Conference in May voted to strengthen a ban on the ordination of LGBTQ clergy and to write new rules that will "discipline" clergy who officiate at same-sex weddings. But those who opposed these measures have a new plan: They say they will form a separate denomination by 2020, calling their church the Christian Methodist**

General Notes & Takeaways

The results are extremely impressive.

The model is far from perfect.



The model fails in unintuitive ways.

Q: How many eyes does a giraffe have?

A: A giraffe has two eyes.

Q: How many eyes does my foot have?

A: Your foot has two eyes.

Q: How many eyes does a spider have?

A: A spider has eight eyes.

Q: How many eyes does the sun have?

A: The sun has one eye.

Source: <https://lacker.io/ai/2020/07/06/giving-gpt-3-a-turing-test.html>

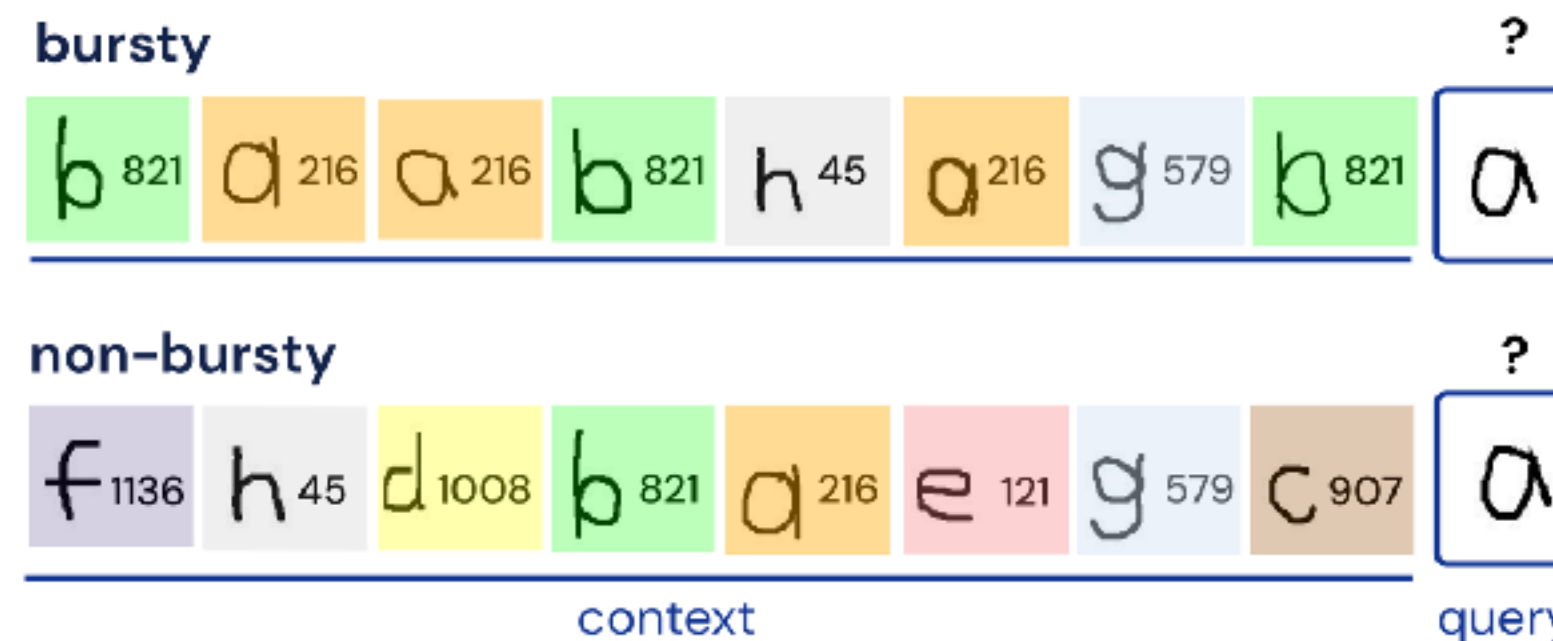
The choice of \mathcal{D}_i^{tr} at test time is important. ("prompting")

Source: <https://github.com/shreyashankar/gpt3-sandbox/blob/master/docs/priming.md>

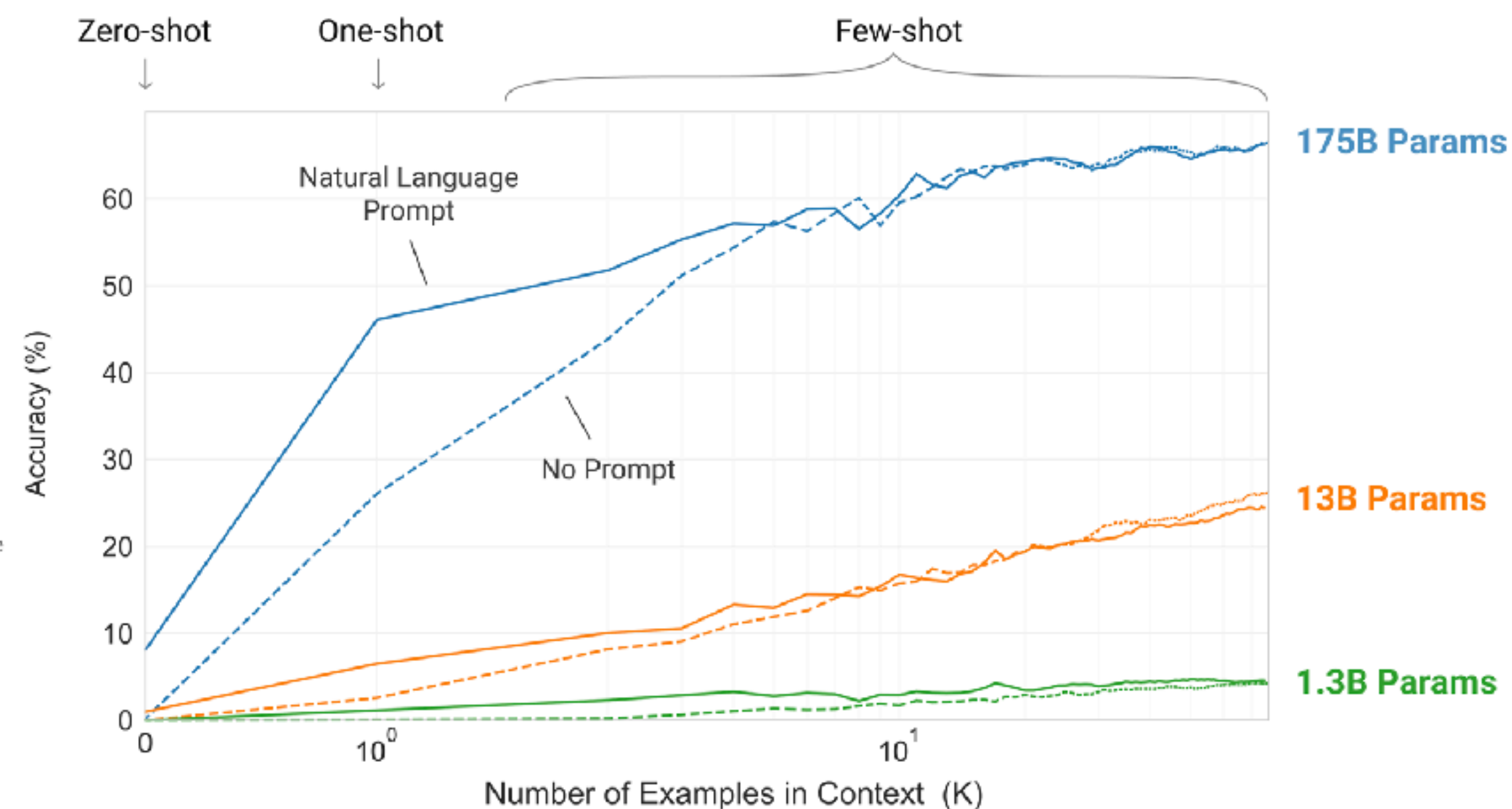
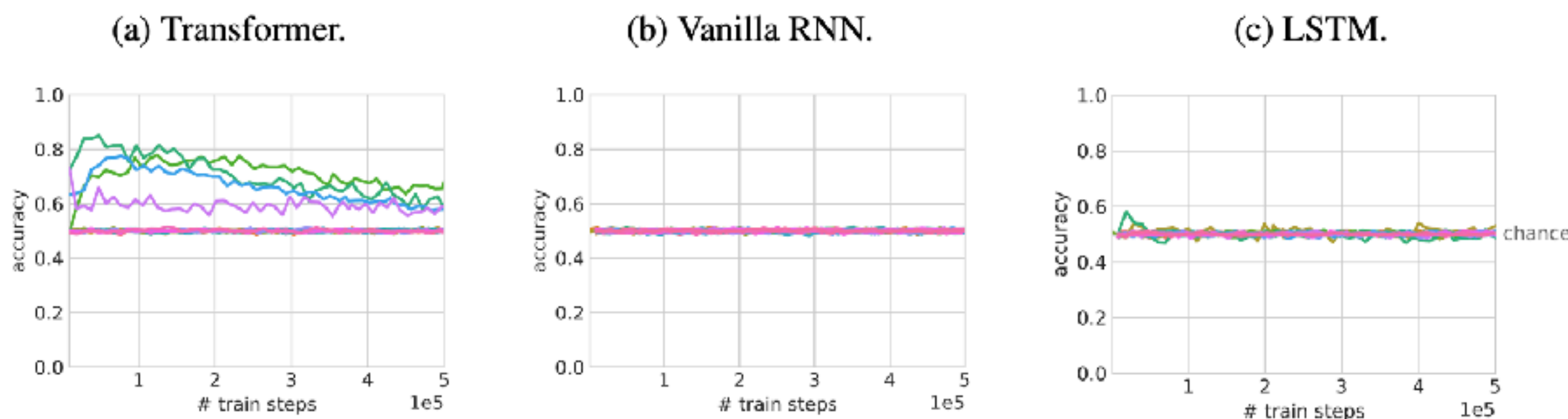
What is needed for few-shot learning to emerge?

An active research topic!

- Data:**
- temporal correlation
 - dynamic meaning of words



- Model:**
- large capacity models
 - transformers > RNNs
 - large models > small models



Plan for Today

Meta-Learning

- Problem formulation
- General recipe of meta-learning algorithms
- Black-box adaptation approaches
- Case study of GPT-3 (time-permitting)

} **Topic of Homework 1!**

Goals for by the end of lecture:

- Training set-up for few-shot meta-learning algorithms
- How to implement black-box meta-learning techniques

Reminders

Project group form due **Wednesday Oct 11**

(Used for assigning TA mentors)

Homework 1 due **Monday October 16**

Next time: Optimization-based meta-learning