Advanced Meta-Learning Topics

Task Construction

CS 330

Course Reminders

Homework 2 due **today**.

Homework 3 out today, due **next Wednesday**.

Note: This homework is brand new.

Following up on some high-res feedback:

- We'll consider autograders for future quarters.
- Homework 3 includes non-classification problems
- Clarification on train/test terminology in today's lecture

Course Roadmap

(start of week 5!)

So far: Multi-task & transfer learning basics

Core meta-learning algorithms

Core unsupervised pre-training algorithms

Next two weeks:
(more advanced topics!)

Advanced meta-learning topics

- Task construction (today)
- Large-scale meta-optimization (Weds)

Bayesian meta-learning

Question of the Day

How should tasks be defined for good meta-learning performance?

Plan for Today

Brief Recap of Meta-Learning & Supervised Task Construction

Memorization in Meta-Learning (problem in meta-learning)
- When it arises
- Potential solutions

} Part of (optional) Homework 4

Meta-Learning without Tasks Provided
- Unsupervised Meta-Learning
- Semi-Supervised Meta-Learning

Goals for by the end of lecture:
- Understand when & how memorization in meta-learning may occur
- Understand techniques for constructing tasks automatically

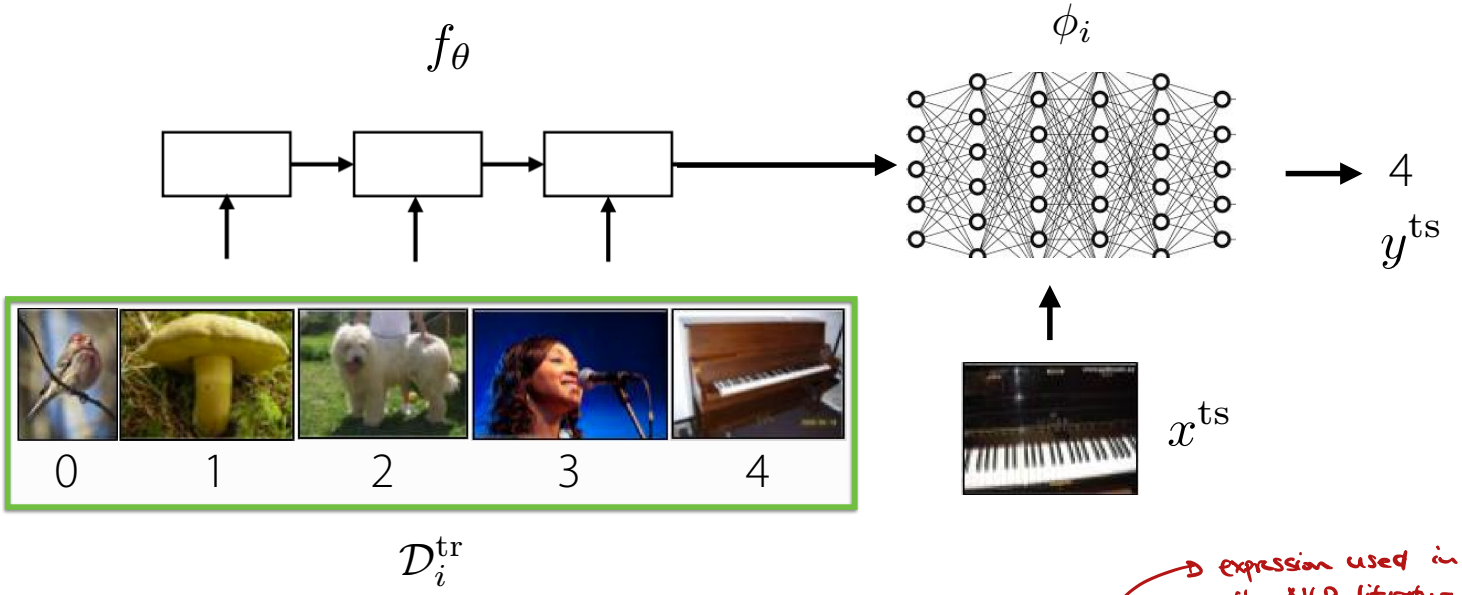
Revisiting meta-learning terminology

Homeworks 1 & 2 sometimes refer to these as **train** & **test**

(which is ambiguous & confusing!)

Recap: Black-Box Meta-Learning

Key idea: parametrize learner as a neural network



expression used in the NLP literature

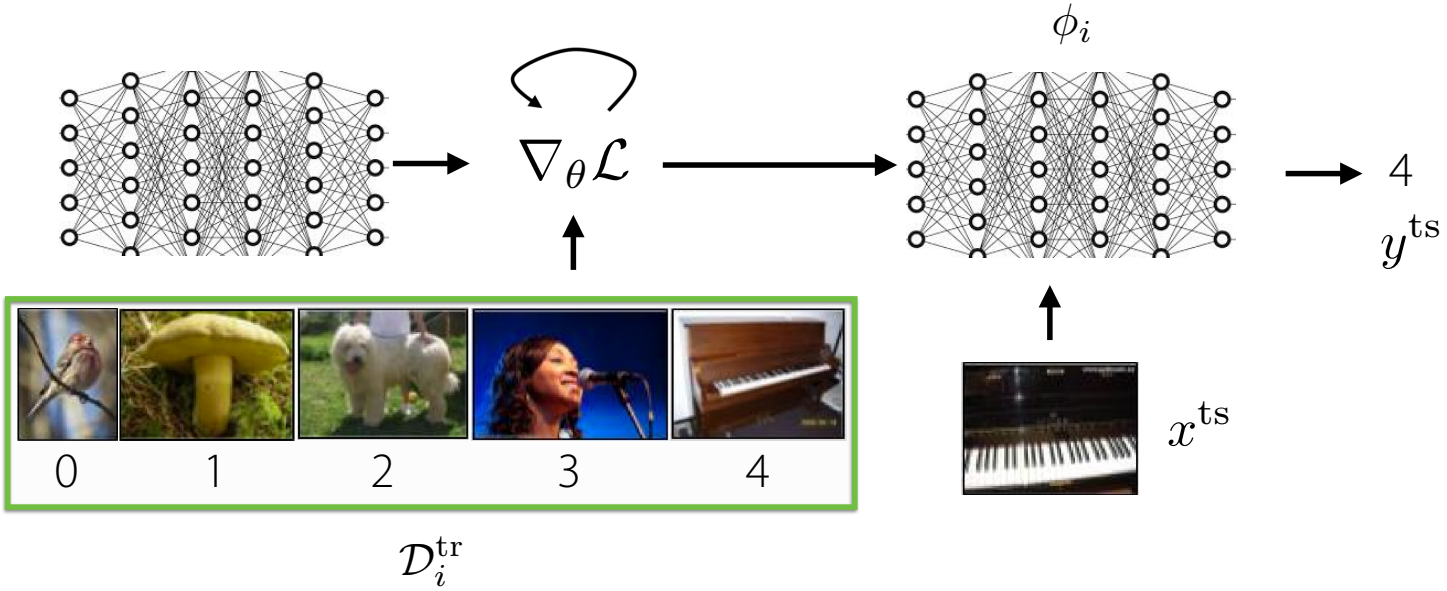
This network: inner loop, in-context learning

Training this network: outer loop

+ **expressive** - **challenging optimization** problem

Recap: Optimization-Based Meta-Learning

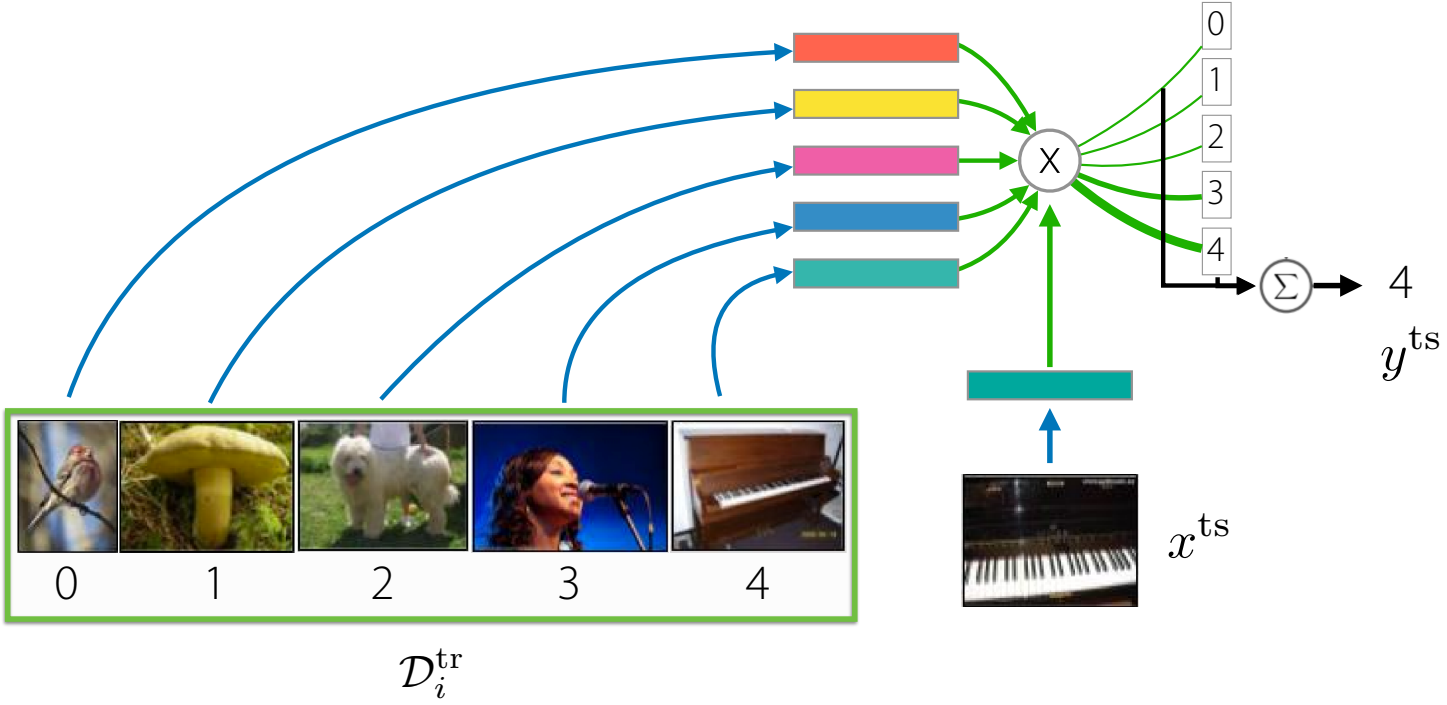
Key idea: embed optimization inside the inner learning process



+ **structure** of **optimization** embedded into meta-learner

- typically requires **second-order optimization**

Recap: Non-Parametric Meta-Learning



Key idea: *non-parametric learner* with *parametric* embedding / distance
(e.g. kNN to examples/prototypes)

+ easy to optimize, computationally fast

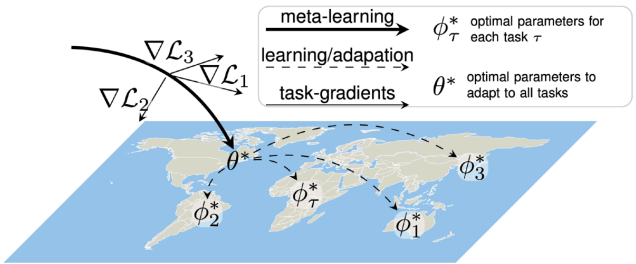
- largely restricted to classification

Supervised Task Construction

For N-way image classification

Use **labeled images** from prior classes

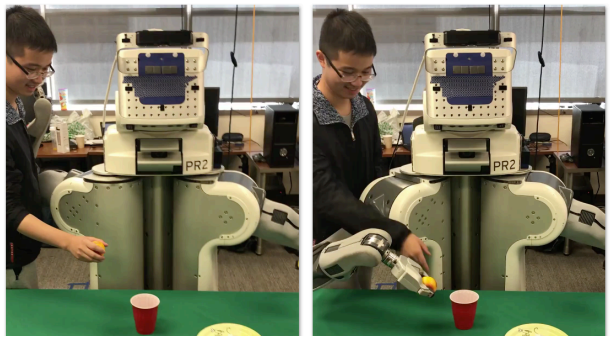
For adapting to regional differences



Rußwurm et al. Meta-Learning for Few-Shot Land Cover Classification. CVPR 2020 EarthVision Workshop

Use **labeled images** from prior regions

For few-shot imitation learning



Yu et al. One-Shot Imitation Learning from Observing Humans. RSS 2018

Use **demonstrations** for prior tasks

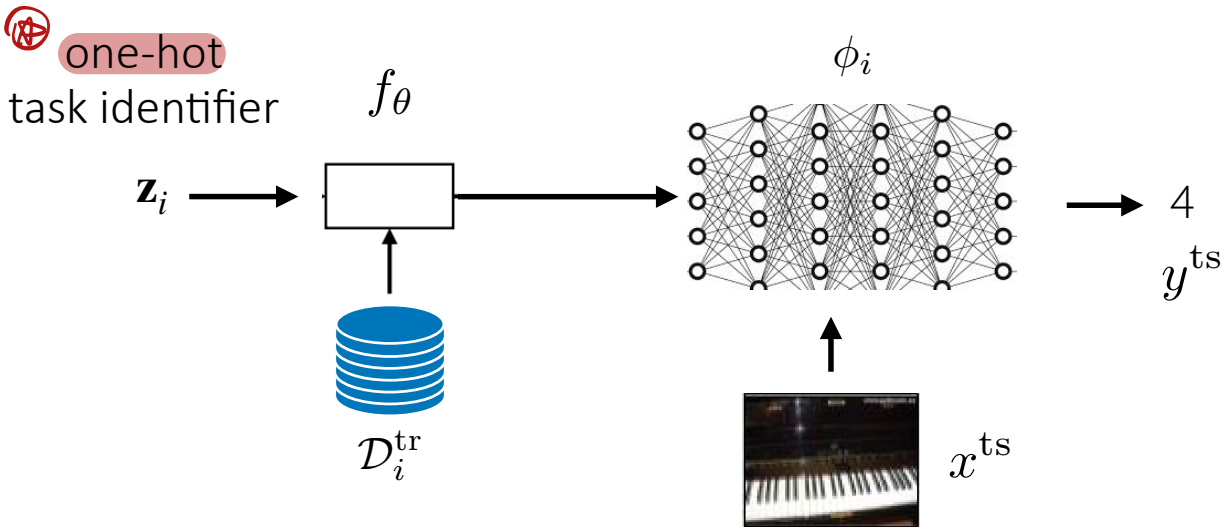
Plan for Today

Brief Recap of Meta-Learning & Task Construction

Memorization in Meta-Learning
- When it arises
- Potential solutions

Meta-Learning without Tasks Provided
- Unsupervised Meta-Learning
- Semi-Supervised Meta-Learning

Thought Exercise #1



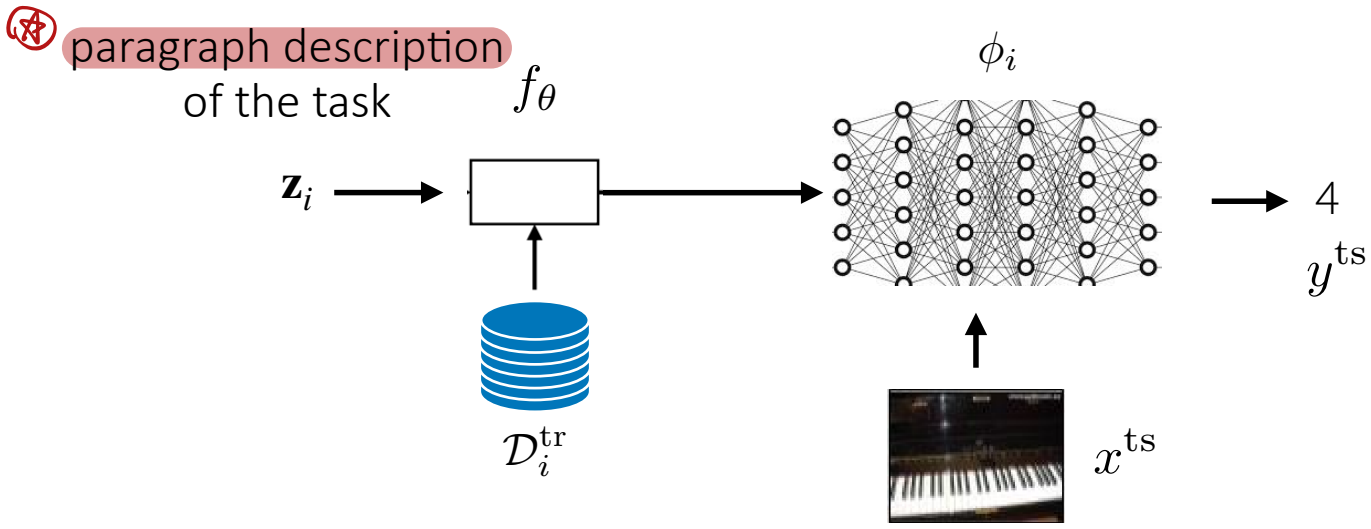
Question: What happens during meta-training if you pass in D_i^{tr} ***and*** the task identifier?

If it is difficult to learn from the data, the model will learn rely on $\mathbf{z}_i$.

Question: What happens at meta-test time if you pass in D_j^{tr} ***and*** the task identifier for a new task?

It won't generalize to the new task.

Thought Exercise #2



Question: What happens during meta-training if you pass in D_i^{tr} ***and*** the task identifier?

It depends on whether using the **description** or the **data** is simpler.

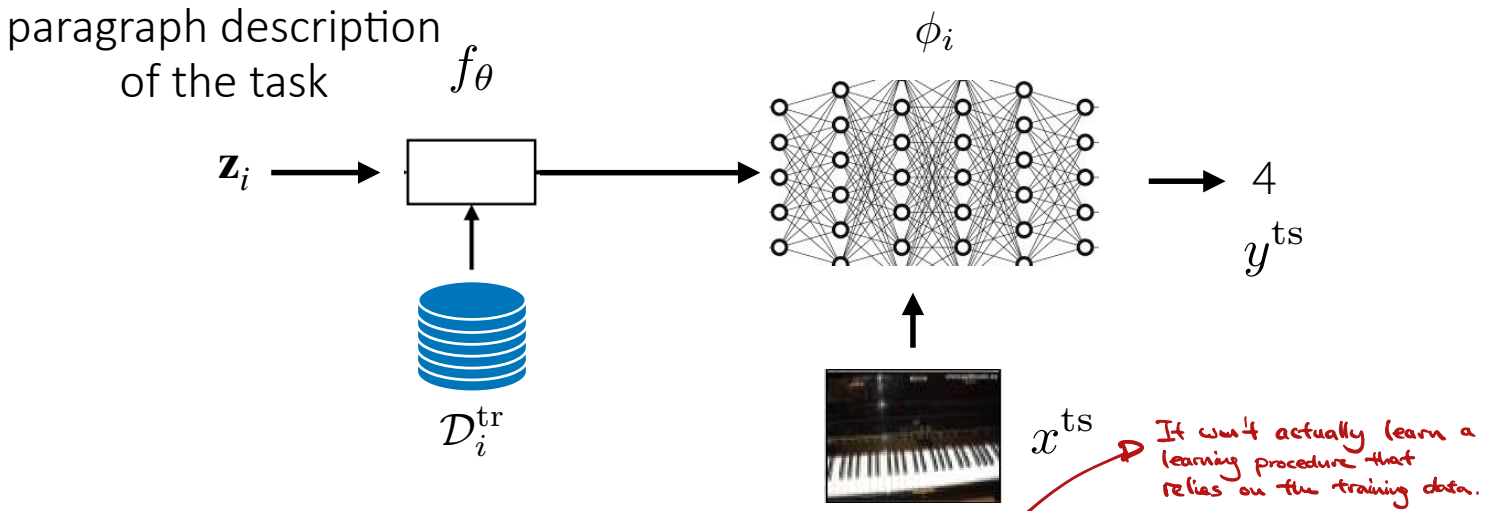
→ Meta model will learn to learn from the simpler of the two.

Question: What happens at meta-test time if you pass in D_j^{tr} ***and*** the task identifier for a new task?

It depends on what it learns to use during meta-training.

→ if it's learned sufficiently general representations of natural language text, it'll work ok. If not, then it'll see a new word in the parag for the new task, and it'll fail to interpret the new goal

Thought Exercise #2



Key problem: Model can minimize meta-training loss without looking at D_i^{tr}

Question: What happe…*and* the task identifier?

It depends …a is simpler.

Question: What happens at meta-test time if you pass in D_j^{tr} ***and*** the task identifier for a new task?

It depends on what it learns to use during meta-training.

How we construct tasks for meta-learning.

Randomly assign class labels to image classes for each task —> Tasks are *mutually exclusive*.

Algorithms **must** use training data to infer label ordering.

Thought Exercise #3: What if label assignment is consistent across tasks?

Q: Is it ok to have some images (⊛) sometimes appear in the support set & sometimes in the query set?

A: Yes, as long as they're not being used both in the support & query set in the same task. Mixing it up is actually preferred.

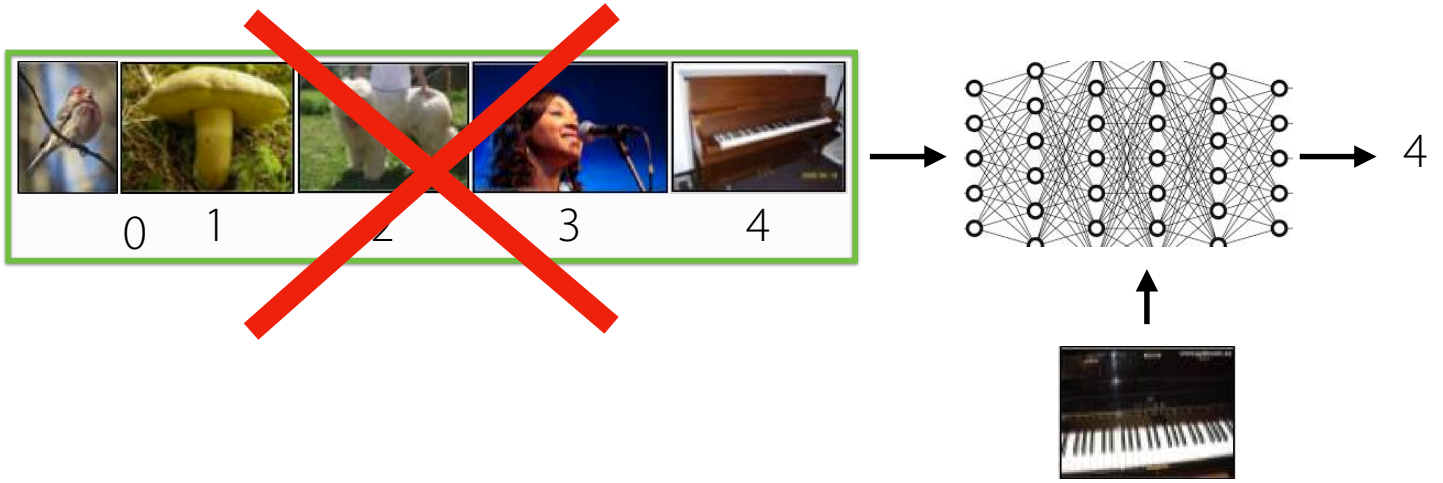
Tasks are **non-mutually exclusive**: a single function can solve all tasks.

The network can simply learn to classify inputs, irrespective of $\mathcal{D}_{\text{tr}}$ ∴

⊛ Not what we want!

→ See next slide.

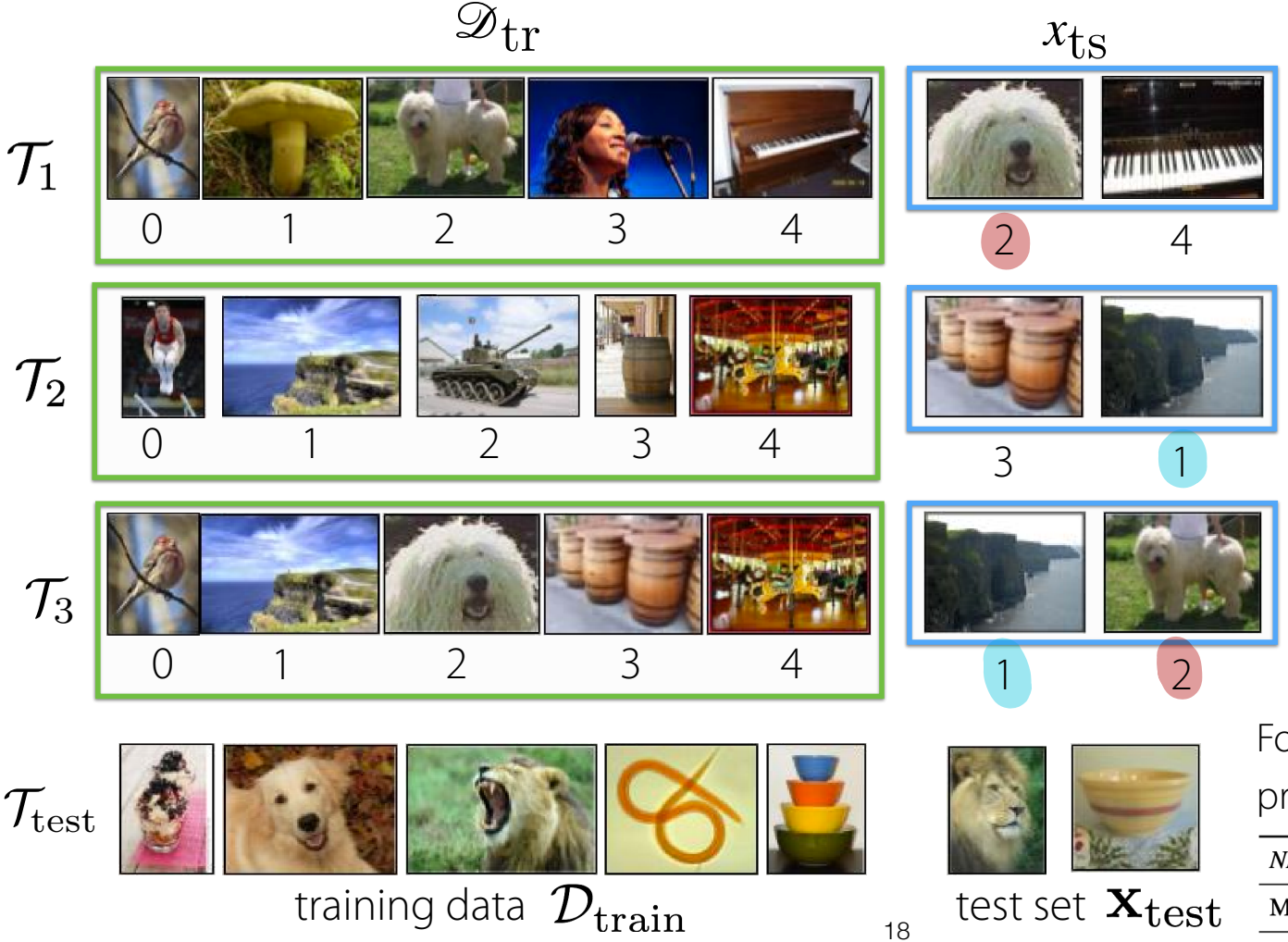
The network can simply learn to classify inputs, irrespective of $\mathscr{D}_{\text{tr}}$



What if label order is consistent?

→ Meta-testing will have poor accuracy (for new image classes)

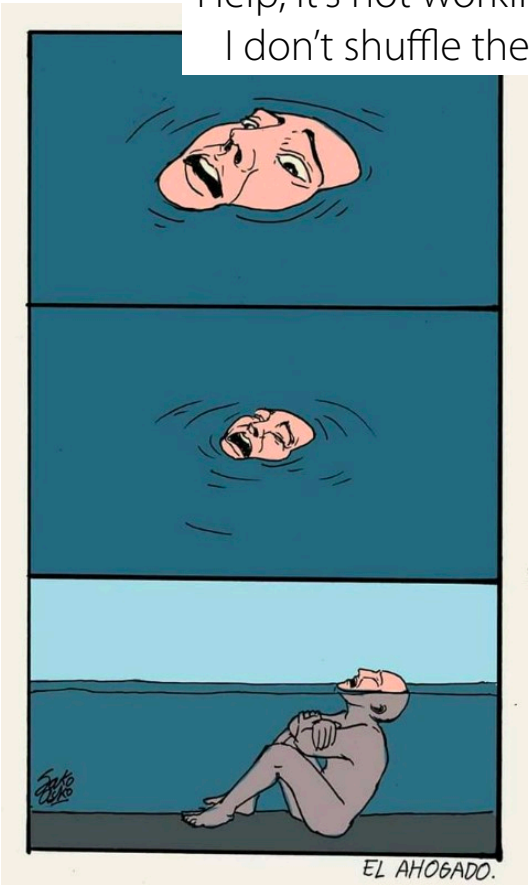
→ For image classes it saw at meta training, it'll have good acc. → but not what we want from meta learning.



For new image classes: can't make predictions w/o $\mathcal{D}_{\text{tr}}$

NME Omniglot	20-way 1-shot	20-way 5-shot
MAML	7.8 (0.2)%	50.7 (22.9)%

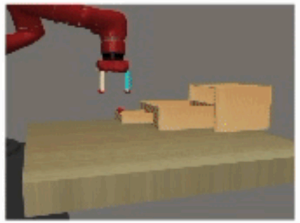
Is this a problem?

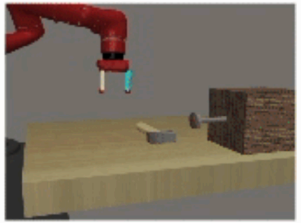


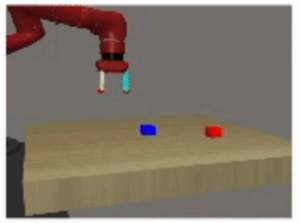
- **No**: for image classification, just shuffle labels*
- **No**, if we see the same image classes as training (no need to adapt at meta-test time)
- But, ***yes***, if we want to be able to adapt with data for new tasks.

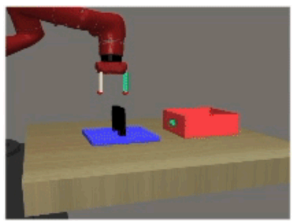
Another example

meta-training

"close drawer" $\mathcal{T}_1$ 

"hammer" $\mathcal{T}_2$ 

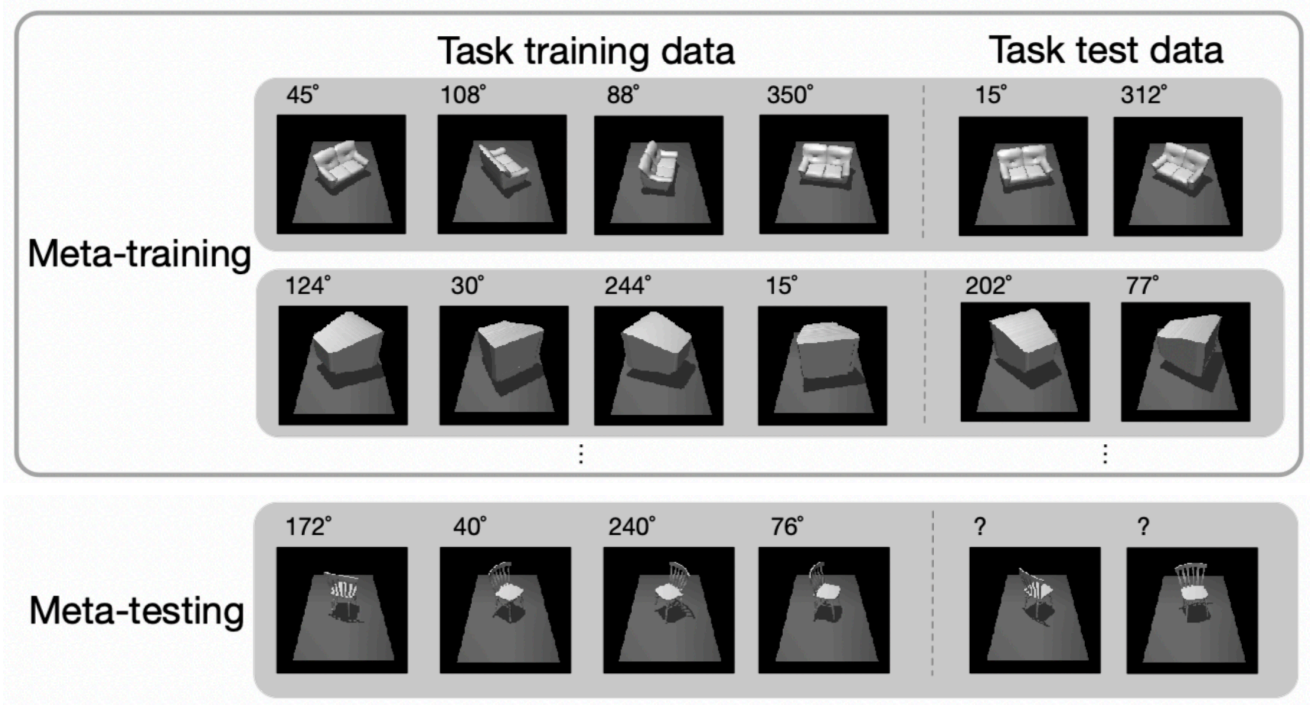
... "stack" $\mathcal{T}_{50}$ 

"close box" $\mathcal{T}_{\text{test}}$ 

If you tell the robot the task goal, the robot can **ignore** the trials.

T Yu, D Quillen, Z He, R Julian, K Hausman, C Finn, S Levine. ***Meta-World***. CoRL '19

Another example



Model can memorize the canonical orientations of the training objects.

Yin, Tucker, Yuan, Levine, Finn. **Meta-Learning without Memorization**. ICLR '19

Can we do something about it?

If tasks *mutually exclusive*: single function cannot solve all tasks

(i.e. due to label shuffling, hiding information)

If tasks are *non-mutually exclusive*: single function can solve all tasks

If can just ignore the meta-training data

Reason why memorization could occur

multiple solutions to the meta-learning problem

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

meta params

support set.

One solution: memorize canonical pose info in θ & ignore $\mathcal{D}_i^{\text{tr}}$

Another solution: carry no info about canonical pose in θ, acquire from $\mathcal{D}_i^{\text{tr}}$

An entire spectrum of solutions based on how **information** flows.

Suggests a potential approach: control information flow.

Yin, Tucker, Yuan, Levine, Finn. **Meta-Learning without Memorization**. ICLR '19

If tasks are *non-mutually exclusive*: single function can solve all tasks

multiple solutions to the meta-learning problem

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

Info can be stored in one of these three places
⇒ Primarily focused on θ and D_i^{tr}.
$\theta \rightarrow \hat{y}^{ts}$
$D_i^{tr} \rightarrow \hat{y}^{ts}$

One solution: memorize canonical pose info in θ & ignore $\mathcal{D}_i^{\text{tr}}$

Another solution: carry no info about canonical pose in θ, acquire from $\mathcal{D}_i^{\text{tr}}$

An entire spectrum of solutions based on how **information** flows.

Meta-regularization one option: $\max I(\hat{\mathbf{y}}_{\text{ts}}, \mathcal{D}_{\text{tr}} \mid \mathbf{x}_{\text{ts}})$

↳ difficult to implement

⇒ We can encourage the meta-learning process to use info from D_i^{tr} more than from θ

⇒ Can maximize mutual info between D_i^{tr} and $\hat{y}^{ts}$, but this is difficult

$I(a;b) = H(a) - H(b)$
$= KL(p(a,b) \| p(a) \cdot p(b))$

minimize meta-training loss + information in θ

$$\mathcal{L}(\theta, \mathcal{D}_{meta-train}) + \beta D_{KL}(q(\theta; \theta_\mu, \theta_\sigma) \| p(\theta))$$

prior dist (Standard Gaussian)

↳ distribution of θ

Places precedence on using information from $\mathcal{D}_{\text{tr}}$ over storing info in θ.

Can combine with your favorite meta-learning algorithm.

Add noise to θ
→ increasing info entropy in θ
→ telling it to meta-train w/ less information in θ

$q(\theta; \theta_\mu, \theta_\sigma)$ → Sampling from the Gaussian distribution corresponds to adding noise to the mean of the Gaussian distribution
↔ adding noise to the variable θ.
↳ Using info from θ now becomes more difficult
⇒ Re-sample from Gaussian every time you use θ.

Ref: "Bayes by Backprop" paper
→ impose Gaussian dist. on NN weights

Mutual info: How interrelated are these two variables?
eg) Completely independent → 0 mutual info
⇒ Minimize information coming from θ

Yin, Tucker, Yuan, Levine, Finn. **Meta-Learning without Memorization**. ICLR'19

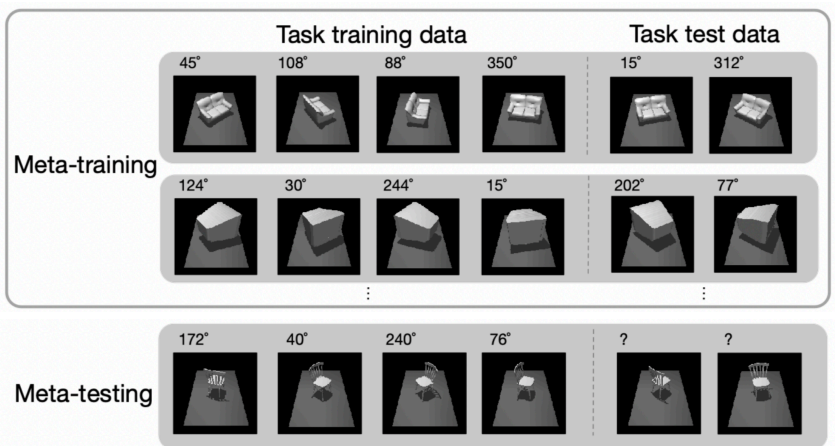
Omniglot without label shuffling: "non-mutually-exclusive" Omniglot

NME Omniglot	20-way 1-shot	20-way 5-shot
MAML	7.8 (0.2)%	50.7 (22.9)%
TAML	9.6 (2.3)%	67.9 (2.3)%
MR-MAML (W) (ours)	**83.3 (0.8)%**	**94.1 (0.1)%**

← Added noise to the weights, basically.

KL-Divergence
- Measure of how different two probability distributions are.
- Not a metric!
  i.e.) KL(P||Q) ≠ KL(Q||P)
- Non-negative measure
- Zero if two distributions are equal
- Measures the amount of extra info that is needed to describe a random variable when using a particular probability distribution instead of another.

On **pose prediction** task:



"Conditional neural processes" + black-box meta learner

Method	MAML	MR-MAML(W) (ours)	CNP	MR-CNP(W) (ours)
MSE	5.39 (1.31)	**2.26 (0.09)**	8.48 (0.12)	2.89 (0.18)

→ much lower MSE

(and it's not just as simple as standard regularization)

CNP	CNP + Weight Decay	CNP + BbB	MR-CNP (W) (ours)
8.48 (0.12)	6.86 (0.27)	7.73 (0.82)	**2.89 (0.18)**

TAML: Jamal & Qi. **Task-Agnostic Meta-Learning for Few-Shot Learning**. CVPR'19

Yin, Tucker, Yuan, Levine, Finn. **Meta-Learning without Memorization**. ICLR'19

Does meta-regularization lead to better generalization?

Let $P(\theta)$ be an arbitrary distribution over θ that doesn't depend on the meta-training data.

(e.g. $P(\theta) = \mathcal{N}(\theta; \mathbf{0}, \mathbf{I})$)

For MAML, with probability at least $1 - \delta$,

$$\underbrace{er(\theta_\mu, \theta_\sigma)}_{\text{generalization error}} \leq \underbrace{\frac{1}{n}\sum_{i=1}^{n} \hat{er}(\theta_\mu, \theta_\sigma, \mathcal{D}_i, \mathcal{D}_i^*)}_{\text{error on the meta-training set}} + \left(\sqrt{\frac{1}{2(K-1)}} + \sqrt{\frac{1}{2(n-1)}}\right) \underbrace{\sqrt{D_{KL}(\mathcal{N}(\theta; \theta_\mu, \theta_\sigma) \| P) + \log \frac{n(K+1)}{\delta}}}_{\text{meta-regularization}}, \quad \forall \theta_\mu, \theta_\sigma$$

With a Taylor expansion of the RHS + a particular value of β —> *recover the MR MAML objective*.

Proof: draws heavily on Amit & Meier '18

Yin, Tucker, Yuan, Levine, Finn. **Meta-Learning without Memorization**. ICLR '19

Summary of Memorization Problem

meta-learning ⟷ **standard supervised learning**

meta overfitting — analogous — standard overfitting

memorize training functions f_i — memorize training datapoints (x_i, y_i)

corresponding to tasks in your meta-training dataset — in your training dataset

meta regularization — **standard regularization**

control information flow — regularize hypothesis class

regularizes description length of meta-parameters — analogous — (though not always for DNNs)

- MDL (Minimum Description Length) Principle
- Kolmogorov complexity

] See next page (measure of information)

The minimum description length (MDL) principle is a method of inductive inference that states that the best explanation of a set of data is the one that requires the least amount of information to describe. This principle is often used in machine learning and statistics to select the best model for a given set of data.

The MDL score of a model is minimized when the model is both simple and accurate. A simple model is one that requires a short description, while an accurate model is one that can accurately predict the data. The MDL principle therefore favors models that are both simple and accurate.

Plan for Today

The Kolmogorov complexity of an object is the length of the shortest computer program that produces the object as output. It is a measure of the computational resources needed to specify the object, and is also known as algorithmic complexity, Solomonoff–Kolmogorov–Chaitin complexity, program-size complexity, descriptive complexity, or algorithmic entropy.

Brief Recap of Meta-Learning & Task Construction

Memorization in Meta-Learning

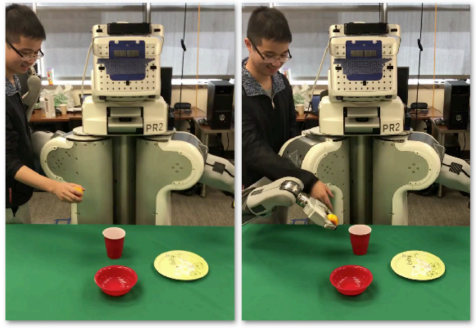
- When it arises
- Potential solutions

Meta-Learning without Tasks Provided

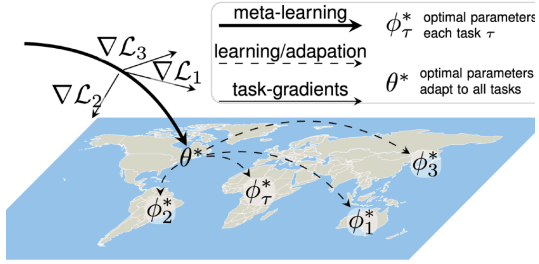
- Unsupervised Meta-Learning
- Semi-Supervised Meta-Learning

Where do tasks come from?

Requires tasks constructed from labeled data



Requires demos for many previous tasks



Rußwurm et al. Meta-Learning for Few-Shot Land Cover Classification. 2020

Requires labeled data from other regions

↑ All had labelled data

What if we only have unlabeled data? e.g., unlabeled images, unlabeled text

Last week: Pre-train representations & fine-tune → [Contrastive learning, Masked autoencoders] + fine-tune

Today: Explicit meta-learning with unlabeled data. → instead of the above.

A general recipe for unsupervised meta-learning

Given unlabeled dataset(s) → Propose tasks → Run meta-learning

Goal of unsupervised meta-learning methods:

Automatically construct tasks from unlabeled data

Make it so that they're close/similar to what you'd do at meta test time.
eg) Don't want to meta-train on 10000-way classification task when at meta test time, you'll most likely get a 10-way clf task.

Question: What do you want the task set to look like?

1. diverse (more likely to cover test tasks)
2. structured (so that few-shot meta-learning is possible)

If you propose completely random tasks, your meta learner won't actually be able to learn those tasks with just a few examples.

Next:

Task construction from unlabeled image data

Task construction from unlabeled text data

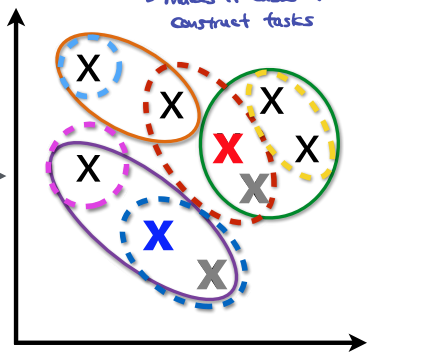
Can we meta-learn with only **unlabeled** images?

— — Task construction — —

Unsupervised learning
(to get an embedding space)

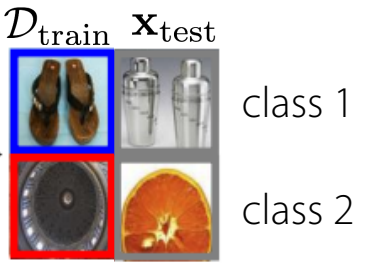
Lower dimensional → makes it easier to construct tasks

$\{\mathbf{x}_i\}$



Examples of tasks:
① classify btw lined clusters
② classify btw dotted line clusters

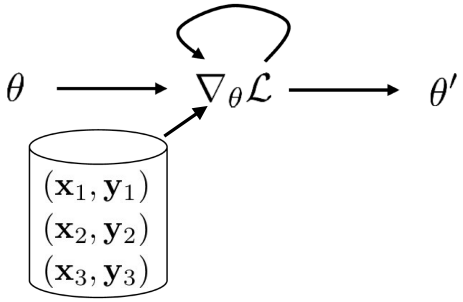
Propose cluster discrimination tasks



These tasks don't correspond to exactly categorizing actual image categories, but they still have a lot of structure to them.

Example of a 2-way clf task.
⇒ Can do N-way clf task.

Run meta-learning



Result: representation suitable for learning downstream tasks

Hsu, Levine, Finn. **Unsupervised Learning via Meta-Learning**. ICLR'19

Can we meta-learn with only **unlabeled** images?

Unsupervised learning
(to get an embedding space)

→

Propose cluster discrimination tasks

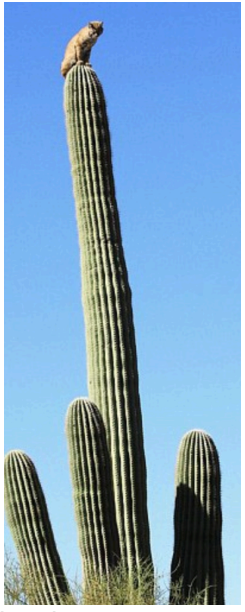
→

Run meta-learning

For example:
A few options:
- BiGAN — Donahue et al. '17
- DeepCluster — Caron et al. '18

Clustering to Automatically Construct Tasks for Unsupervised Meta-Learning (CACTUs) → from paper ref. below.

MAML — Finn et al. '17
ProtoNets — Snell et al. '17



CACTUs MAML

miniImageNet 5-way 5-shot

No labels at meta-train time (creating our own labels)
Labels at meta-test time

method	accuracy
MAML with labels ← Target	62.13%
BiGAN kNN	31.10%
BiGAN logistic	33.91%
BiGAN MLP + dropout	29.06%
BiGAN cluster matching	29.49%
BiGAN CACTUs MAML	51.28%
DeepCluster CACTUs MAML	**53.97%**

Same story for:

- **4 different embedding methods**
- **4 datasets** (Omniglot, CelebA, miniImageNet, MNIST)
- **2 meta-learning methods** (*)
- Test tasks with **larger datasets**

*ProtoNets underperforms in some cases.

Hsu, Levine, Finn. **Unsupervised Learning via Meta-Learning**. ICLR'19

Can we use **domain knowledge** when constructing tasks?

e.g. **image's label** often **won't change** when you: → apply various augmentations ← domain knowledge determines which augmentations can be used or not.

- drop out some pixels
- translate the image
- reflect the image

Task construction:

For each task $\mathcal{T}_i$:

i. Randomly sample N images & assign labels $1,\ldots,N$

—> Store in $\mathcal{D}_i^{\text{tr}}$

ii. For each datapoint in $\mathcal{D}_i^{\text{tr}}$, augment image using domain knowledge

—> Store in $\mathcal{D}_i^{\text{ts}}$

Khodadadeh, Bölöni, Shah. **Unsupervised Meta-Learning for Few-Shot Image Classification**. NeurIPS '19

Can we use **domain knowledge** when constructing tasks?

For each task $\mathcal{T}_i$:

i. Randomly sample N images & assign labels $1,\ldots,N$ —> Store in $\mathcal{D}_i^{\text{tr}}$

ii. For each datapoint in $\mathcal{D}_i^{\text{tr}}$, augment image using domain knowledge —> Store in $\mathcal{D}_i^{\text{ts}}$

How to augment in practice?

Omniglot: translation & random pixel dropout

↳ reflection won't work. Same w/ rotation (>90°)

MiniImagenet: AutoAugment* (translation, rotation, shear)

		Omniglot				Mini-Imagenet			
Algorithm (N, K)	**Clustering**	**(5,1)**	**(5,5)**	**(20,1)**	**(20,5)**	**(5,1)**	**(5,5)**	**(5,20)**	**(5,50)**
Training from scratch	N/A	52.50	74.78	24.91	47.62	27.59	38.48	51.53	59.63
linear classifier	ACAI / DC	61.08	81.82	43.20	66.33	29.44	39.79	56.19	65.28
MLP with dropout	ACAI / DC	51.95	77.20	30.65	58.62	29.03	39.67	52.71	60.95
cluster matching	ACAI / DC	54.94	71.09	32.19	45.93	22.20	23.50	24.97	26.87
CACTUs-MAML	ACAI / DC	68.84	87.78	48.09	73.36	39.90	**53.97**	**63.84**	**69.64**
CACTUs-ProtoNets	ACAI / DC	68.12	83.58	47.75	66.27	39.18	53.36	61.54	63.55
UMTRA (ours)	N/A	**83.80**	**95.43**	**74.25**	**92.12**	**39.93**	50.73	61.11	67.15
MAML (Supervised)	N/A	94.46	98.83	84.60	96.29	46.81	62.13	71.03	75.54
ProtoNets (Supervised)	N/A	98.35	99.58	95.31	98.81	46.56	62.29	70.05	72.04

- outstanding Omniglot performance
  (where we have good domain knowledge!)
- MiniImageNet: slightly underperforms CACTUs
  ↳ clustering approach

Khodadadeh, Bölöni, Shah. **Unsupervised Meta-Learning for Few-Shot Image Classification**. NeurIPS '19

* Cubuk et al. 2018

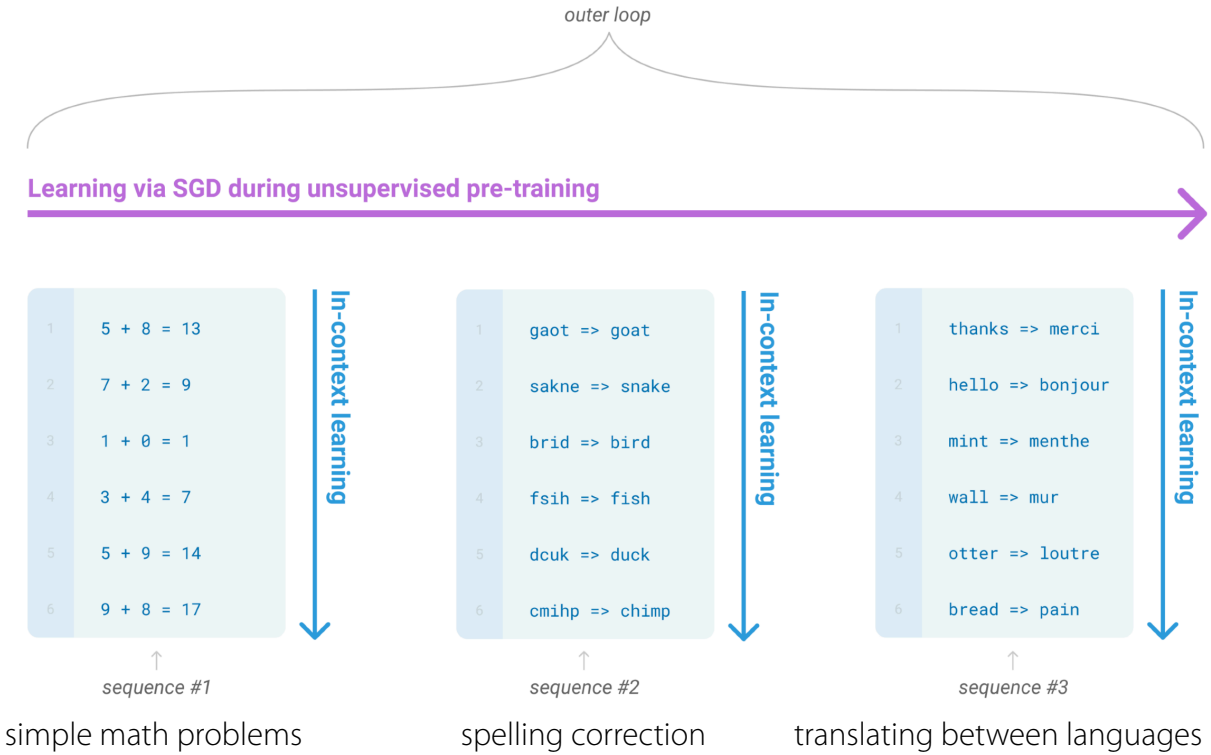
Can we meta-learn with only **unlabeled** text?

Option A: Formulate it as a language modeling problem.

Recall: GPT-3

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

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



When might we not use this option?

- harder to combine w/ **optimization-based meta-learning**
- harder to apply to **classification** tasks (e.g. sentiment, political bias, etc)

Brown, Mann, Ryder, Subbiah et al. **Language Models are Few-Shot Learners**. arXiv '20

Can we meta-learn with only **unlabeled** text?

Option B: Construct tasks by masking out words

Task: Classify the masked word.

For each task $\mathcal{T}_i$:

i. Sample subset of N unique words & assign unique ID.

{Democratic, Capital} 1 2

ii. Sample $K+Q$ sentences with that word, *masking the word out*

iii. Construct $\mathcal{D}_i^{\text{tr}}$ and $\mathcal{D}_i^{\text{ts}}$ with masked sentences & corresponding word IDs

$\mathcal{D}_i^{\text{tr}}$

Support set

Sentence	Class
A member of the [m] Party, he was the first African American to be elected to the presidency.	1
The [m] Party is one of the two major contemporary political parties in the United States, along with its rival, the Republican Party.	1
Honolulu is the [m] and largest city of the U.S. state of Hawaii.	2
Washington, D.C., formally the District of Columbia and commonly referred to as Washington or D.C., is the [m] of the United States.	2

$\mathcal{D}_i^{\text{ts}}$

Query: New Delhi is an urban district of Delhi which serves as the [m] of India
Correct Prediction: 2

Bansal, Jha, Munkhdalai, McCallum. **Self-Supervised Meta-Learning for Few-Shot Natural Language Classification Tasks**. EMNLP '20

entirely unsupervised pre-training | supervised or semi-supervised pre-training

Can take a semi-supervised approach.

Task	N	k	BERT	SMLMT	MT-BERT$_{\text{softmax}}$	MT-BERT	LEOPARD	Hybrid-SMLMT
CoNLL	4	4	50.44 ± 08.57	46.81 ± 4.77	52.28 ± 4.06	55.63 ± 4.99	54.16 ± 6.32	**57.60** ± 7.11
		8	50.06 ± 11.30	61.72 ± 3.11	65.34 ± 7.12	58.32 ± 3.77	67.38 ± 4.33	**70.20** ± 3.00
		16	74.47 ± 03.10	75.82 ± 4.04	71.67 ± 3.03	71.29 ± 3.30	76.37 ± 3.08	**80.61** ± 2.77
		32	83.27 ± 02.14	84.01 ± 1.73	73.09 ± 2.42	79.94 ± 2.45	83.61 ± 2.40	**85.51** ± 1.73
MITR	8	4	49.37 ± 4.28	46.23 ± 3,90	45.52 ± 5.90	50.49 ± 4.40	49.84 ± 3.31	**52.29** ± 4.32
		8	49.38 ± 7.76	61.15 ± 1.91	58.19 ± 2.65	58.01 ± 3.54	62.99 ± 3.28	**65.21** ± 2.32
		16	69.24 ± 3.68	69.22 ± 2.78	66.09 ± 2.24	66.16 ± 3.46	70.44 ± 2.89	**73.37** ± 1.88
		32	78.81 ± 1.95	78.82 ± 1.30	69.35 ± 0.98	76.39 ± 1.17	78.37 ± 1.97	**79.96** ± 1.48
Airline	3	4	42.76 ± 13.50	42.83 ± 6.12	43.73 ± 7.86	46.29 ± 12.26	54.95 ± 11.81	**56.46** ± 10.67
		8	38.00 ± 17.06	51.48 ± 7.35	52.39 ± 3.97	49.81 ± 10.86	61.44 ± 03.90	**63.05** ± 8.25
		16	58.01 ± 08.23	58.42 ± 3.44	58.79 ± 2.97	57.25 ± 09.90	62.15 ± 05.56	**69.33** ± 2.24
		32	63.70 ± 4.40	65.33 ± 3.83	61.06 ± 3.89	62.49 ± 4.48	67.44 ± 01.22	**71.21** ± 3.28
Disaster	2	4	55.73 ± 10.29	**62.26** ± 9.16	52.87 ± 6.16	50.61 ± 8.33	51.45 ± 4.25	55.26 ± 8.32
		8	56.31 ± 09.57	**67.89** ± 6.83	56.08 ± 7.48	54.93 ± 7.88	55.96 ± 3.58	63.62 ± 6.84
		16	64.52 ± 08.93	**72.86** ± 1.70	65.83 ± 4.19	60.70 ± 6.05	61.32 ± 2.83	70.56 ± 2.23
		32	73.60 ± 01.78	**73.69** ± 2.32	67.13 ± 3.11	72.52 ± 2.28	63.77 ± 2.34	71.80 ± 1.85
Emotion	13	4	09.20 ± 3.22	09.84 ± 1.09	09.41 ± 2.10	09.84 ± 2.14	11.71 ± 2.16	**11.90** ± 1.74
		8	08.21 ± 2.12	11.02 ± 1.02	11.61 ± 2.34	11.21 ± 2.11	12.90 ± 1.63	**13.26** ± 1.01
		16	13.43 ± 2.51	12.05 ± 1.18	13.82 ± 2.02	12.75 ± 2.04	13.38 ± 2.20	**15.17** ± 0.89
		32	16.66 ± 1.24	14.28 ± 1.11	13.81 ± 1.62	**16.88** ± 1.80	14.81 ± 2.01	16.08 ± 1.16
Political Bias	2	4	54.57 ± 5.02	57.72 ± 5.72	54.32 ± 3.90	54.66 ± 3.74	60.49 ± 6.66	**61.17** ± 4.91
		8	56.15 ± 3.75	63.02 ± 4.62	57.36 ± 4.32	54.79 ± 4.19	61.74 ± 6.73	**64.10** ± 4.03
		16	60.96 ± 4.25	**66.35** ± 2.84	59.24 ± 4.25	60.30 ± 3.26	65.08 ± 2.14	66.11 ± 2.04
		32	65.04 ± 2.32	**67.73** ± 2.27	62.68 ± 3.21	64.99 ± 3.05	64.67 ± 3.41	67.30 ± 1.53

More results & analysis in the paper!

BERT - standard self-supervised learning + fine-tuning

SMLMT - proposed unsupervised meta-learning

MT-BERT - multi-task learning + fine-tuning (on supervised tasks)

LEOPARD - optimization-based meta-learner (only on supervised tasks)

Hybrid-SMLMT - meta-learning on proposed tasks + supervised tasks

Bansal, Jha, Munkhdalai, McCallum. **Self-Supervised Meta-Learning for Few-Shot Natural Language Classification Tasks**. EMNLP '20

Summary of Unsupervised Meta-Training

Unlike what we did last week, which was:
(Contrastive learning / masked autoencoders) → fine-tune.

→ Go directly to meta-learning without explicit labels.

Given unlabeled dataset(s) → Propose tasks → Run meta-learning

Existing task proposal techniques:

- Classify between clusters of images
- Classify augmented image vs. different image instance
- Generate text from a particular context
- Classify a masked word

Q: If you have enough meta-test data, would the unsupervised pre-training methods outperform what we just talked about?

A: Yes. These methods will shine if you're in a few-shot regime at meta-test time because they're explicitly optimizing for few-shot learning. The settings in which unsupervised pre-training are going to do well are settings where you have a bit more data of your target task.

Plan for Today

Brief Recap of Meta-Learning & Task Construction

Memorization in Meta-Learning
- When it arises
- Potential solutions

Meta-Learning without Tasks Provided
- Unsupervised Meta-Learning
- Semi-Supervised Meta-Learning

} Part of (optional) Homework 4

Goals for by the end of lecture:
- Understand when & how memorization in meta-learning may occur
- Understand techniques for constructing tasks automatically

Course Reminders

Homework 2 due **today**.

Homework 3 out today, due **next Wednesday**.

Wednesday's lecture: large-scale meta-optimization

by Yoonho Lee
(ML PhD student)