Advanced Meta-Learning Topics
Task Construction

CS 330
Course Reminders

Homework 2 due today.

Homework 3 out today, due Mon Nov 6.
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: Advanced meta-learning topics
- Task construction (today)
- Large-scale meta-optimization (Weds)
Bayesian meta-learning

So far: Multi-task & transfer learning basics
Core meta-learning algorithms
Core unsupervised pre-training algorithms

Next two weeks: 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
- When it arises
- Potential solutions

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

{Part of (optional) Homework 4}
Revisiting meta-learning terminology

- Task training set: $D_{i}^{tr}$ ("support set")
- Task test dataset: $D_{i}^{test}$ ("query set")

**Meta-training**

**Meta-testing**
Recap: **Black-Box** Meta-Learning

**Key idea:** parametrize learner as a neural network

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 to rely on $z_i$.

**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 $D_{i \text{tr}}$ and the task identifier? It depends on whether using the description or the data is simpler.

**Question:** What happens at meta-test time if you pass in $D_{j \text{tr}}$ and the task identifier for a new task? It depends on what it learns to use during meta-training.
Thought Exercise #2

**Question:** What happens during meta-training if you pass in the task identifier? It depends on what it learns to use during meta-training.

**Key problem:** Model can minimize meta-training loss without looking at $D_{tr}^i$ and the task identifier.

**Question:** What happens at meta-test time if you pass in $D_{tr}^j$ 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?

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

The network can simply learn to classify inputs, irrespective of $D_{tr}$.
The network can simply learn to classify inputs, irrespective of $\mathcal{D}_{tr}$.
What if label order is consistent?

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

<table>
<thead>
<tr>
<th>NME Omniglot</th>
<th>20-way 1-shot</th>
<th>20-way 5-shot</th>
</tr>
</thead>
<tbody>
<tr>
<td>MAML</td>
<td>7.8 (0.2)%</td>
<td>50.7 (22.9)%</td>
</tr>
</tbody>
</table>
Is this a problem?

Help, it’s not working when I don’t shuffle the labels.

- 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

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

*multiple solutions* to the meta-learning problem

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

**One solution:** memorize canonical pose info in \( \theta \) & ignore \( \mathcal{D}_i^{\text{tr}} \)

**Another solution:** carry no info about canonical pose in \( \theta \), 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

\[ y^{ts} = f_\theta(D_i^{tr}, x^{ts}) \]

One solution: memorize canonical pose info in \( \theta \) & ignore \( D_i^{tr} \)
Another solution: carry no info about canonical pose in \( \theta \), acquire from \( D_i^{tr} \)

An entire spectrum of solutions based on how information flows.

Meta-regularization one option: \( \max I(\hat{y}_{ts}, D_{tr} | x_{ts}) \)

minimize meta-training loss + information in \( \theta \)

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

Places precedence on using information from \( D_{tr} \) over storing info in \( \theta \).
Can combine with your favorite meta-learning algorithm.

Yin, Tucker, Yuan, Levine, Finn. Meta-Learning without Memorization. ICLR'19
Omniglot without label shuffling: “non-mutually-exclusive” Omniglot

<table>
<thead>
<tr>
<th>NME Omniglot</th>
<th>20-way 1-shot</th>
<th>20-way 5-shot</th>
</tr>
</thead>
<tbody>
<tr>
<td>MAML</td>
<td>7.8 (0.2)%</td>
<td>50.7 (22.9)%</td>
</tr>
<tr>
<td>TAML</td>
<td>9.6 (2.3)%</td>
<td>67.9 (2.3)%</td>
</tr>
<tr>
<td>MR-MAML (W) (ours)</td>
<td>83.3 (0.8)%</td>
<td>94.1 (0.1)%</td>
</tr>
</tbody>
</table>

On pose prediction task:

<table>
<thead>
<tr>
<th>Method</th>
<th>MAML</th>
<th>MR-MAML(W) (ours)</th>
<th>CNP</th>
<th>MR-CNP(W) (ours)</th>
</tr>
</thead>
<tbody>
<tr>
<td>MSE</td>
<td>5.39 (1.31)</td>
<td><strong>2.26 (0.09)</strong></td>
<td>8.48 (0.12)</td>
<td><strong>2.89 (0.18)</strong></td>
</tr>
</tbody>
</table>

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

<table>
<thead>
<tr>
<th>CNP</th>
<th>CNP + Weight Decay</th>
<th>CNP + BbB</th>
<th>MR-CNP(W) (ours)</th>
</tr>
</thead>
<tbody>
<tr>
<td>8.48 (0.12)</td>
<td>6.86 (0.27)</td>
<td>7.73 (0.82)</td>
<td><strong>2.89 (0.18)</strong></td>
</tr>
</tbody>
</table>

TAML: Jamal & Qi. Task-Agnostic Meta-Learning for Few-Shot Learning. CVPR ’19
Yin, Tucker, Yuan, Levine, Finn. Meta-Learning without Memorization. ICLR ’19
Summary of Memorization Problem

**meta-learning**

- **meta overfitting**
  - memorize training functions $f_i$
corresponding to tasks in your meta-training dataset

- **meta regularization**
  - control information flow
  - regularizes description length
  - of meta-parameters

**standard supervised learning**

- **standard overfitting**
  - memorize training datapoints $(x_i, y_i)$
in your training dataset

- **standard regularization**
  - regularize hypothesis class
  (though not always for DNNs)
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
Where do tasks come from?

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

Last two lectures: Pre-train representations & fine-tune

Today: Explicit meta-learning with unlabeled data.

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

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)

Next:
- Task construction from unlabeled image data
- Task construction from unlabeled text data
Can we use **domain knowledge** when constructing tasks from unlabeled data?

e.g. **image’s label** often **won’t change** when you:
- 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 $\mathcal{D}_i^{tr}$

ii. For each datapoint in $\mathcal{D}_i^{tr}$, augment image using domain knowledge → Store in $\mathcal{D}_i^{ts}$
Can we use **domain knowledge** when constructing tasks from unlabeled data?

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

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

2. For each datapoint in $\mathcal{D}_i^{tr}$, augment image using domain knowledge  
   $\rightarrow$ Store in $\mathcal{D}_i^{ts}$

How to augment in practice?

**Omniglot**: translation & random pixel dropout  
**MiniImageNet**: AutoAugment* (translation, rotation, shear)

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

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

* Cubuk et al. 2018
Unsupervised meta-learning without domain knowledge?

--- Task construction ---

Unsupervised learning (to get an embedding space) → Propose cluster discrimination tasks → Run meta-learning

Result: representation suitable for learning downstream tasks

Hsu, Levine, Finn. Unsupervised Learning via Meta-Learning. ICLR’19
Unsupervised meta-learning without domain knowledge?

Unsupervised learning (to get an embedding space)

A few options:
- BiGAN — Donahue et al. ’17
- DeepCluster — Caron et al. ’18

Propose cluster discrimination tasks

Clustering to Automatically Construct Tasks for Unsupervised Meta-Learning (CACTUs)

Run meta-learning

- MAML — Finn et al. ’17
- ProtoNets — Snell et al. ’17

minilimageNet 5-way 5-shot

<table>
<thead>
<tr>
<th>method</th>
<th>accuracy</th>
</tr>
</thead>
<tbody>
<tr>
<td>MAML with labels</td>
<td>62.13%</td>
</tr>
<tr>
<td>BiGAN kNN</td>
<td>31.10%</td>
</tr>
<tr>
<td>BiGAN logistic</td>
<td>33.91%</td>
</tr>
<tr>
<td>BiGAN MLP + dropout</td>
<td>29.06%</td>
</tr>
<tr>
<td>BiGAN cluster matching</td>
<td>29.49%</td>
</tr>
<tr>
<td>BiGAN CACTUs MAML</td>
<td>51.28%</td>
</tr>
<tr>
<td>DeepCluster CACTUs MAML</td>
<td>53.97%</td>
</tr>
</tbody>
</table>

Same story for:
- 4 different embedding methods
- 4 datasets (Omniglot, CelebA, minilimageNet, 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 meta-learn with only **unlabeled** text?

**Option A:** Formulate it as a language modeling problem.

Recall: GPT-3

\( \mathcal{D}_{tr} \): sequence of characters

\( \mathcal{D}_{ts} \): 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 $T_i$:

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

---

Support set

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

---

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

<table>
<thead>
<tr>
<th>Task</th>
<th>$N$</th>
<th>$k$</th>
<th>BERT</th>
<th>SMLMT</th>
<th>MT-BERT$_{softmax}$</th>
<th>MT-BERT</th>
<th>LEOPARD</th>
<th>Hybrid-SMLMT</th>
</tr>
</thead>
<tbody>
<tr>
<td>CoNLL</td>
<td>4</td>
<td>16</td>
<td>50.44 ± 0.57</td>
<td>46.81 ± 4.77</td>
<td>52.28 ± 4.06</td>
<td>55.63 ± 4.99</td>
<td>54.16 ± 6.32</td>
<td>57.60 ± 7.11</td>
</tr>
<tr>
<td></td>
<td>8</td>
<td>32</td>
<td>50.06 ± 11.30</td>
<td>61.72 ± 3.11</td>
<td>65.34 ± 7.12</td>
<td>58.32 ± 3.77</td>
<td>67.38 ± 4.33</td>
<td>70.20 ± 3.00</td>
</tr>
<tr>
<td></td>
<td>16</td>
<td></td>
<td>74.47 ± 0.10</td>
<td>75.82 ± 4.04</td>
<td>71.67 ± 3.03</td>
<td>71.29 ± 3.20</td>
<td>76.37 ± 3.08</td>
<td>80.61 ± 2.77</td>
</tr>
<tr>
<td></td>
<td>32</td>
<td></td>
<td>83.27 ± 0.24</td>
<td>84.01 ± 1.73</td>
<td>73.09 ± 2.42</td>
<td>79.94 ± 2.45</td>
<td>83.61 ± 2.40</td>
<td>85.51 ± 1.73</td>
</tr>
<tr>
<td>MTTR</td>
<td>8</td>
<td>4</td>
<td>49.37 ± 4.28</td>
<td>46.23 ± 3.90</td>
<td>45.52 ± 5.90</td>
<td>50.49 ± 4.40</td>
<td>49.84 ± 3.31</td>
<td>52.29 ± 4.32</td>
</tr>
<tr>
<td></td>
<td>8</td>
<td>16</td>
<td>49.38 ± 7.76</td>
<td>61.15 ± 1.91</td>
<td>58.19 ± 2.65</td>
<td>58.01 ± 3.54</td>
<td>62.99 ± 3.28</td>
<td>65.21 ± 2.32</td>
</tr>
<tr>
<td></td>
<td>16</td>
<td>32</td>
<td>69.24 ± 3.68</td>
<td>69.22 ± 2.78</td>
<td>66.09 ± 2.24</td>
<td>66.16 ± 3.46</td>
<td>70.44 ± 2.89</td>
<td>73.37 ± 1.88</td>
</tr>
<tr>
<td></td>
<td>32</td>
<td></td>
<td>78.81 ± 1.95</td>
<td>78.82 ± 1.30</td>
<td>69.35 ± 0.98</td>
<td>76.39 ± 1.17</td>
<td>78.37 ± 1.97</td>
<td>79.96 ± 1.48</td>
</tr>
<tr>
<td>Airline</td>
<td>3</td>
<td>4</td>
<td>42.76 ± 13.50</td>
<td>42.83 ± 6.12</td>
<td>43.73 ± 7.86</td>
<td>46.29 ± 12.36</td>
<td>54.95 ± 11.81</td>
<td>56.46 ± 10.67</td>
</tr>
<tr>
<td></td>
<td>8</td>
<td>16</td>
<td>38.00 ± 17.06</td>
<td>51.48 ± 7.35</td>
<td>52.39 ± 3.97</td>
<td>49.81 ± 10.86</td>
<td>61.44 ± 03.90</td>
<td>63.05 ± 8.25</td>
</tr>
<tr>
<td></td>
<td>16</td>
<td>32</td>
<td>58.01 ± 28.33</td>
<td>58.42 ± 3.44</td>
<td>58.79 ± 2.97</td>
<td>57.25 ± 09.90</td>
<td>62.15 ± 05.96</td>
<td>69.33 ± 2.24</td>
</tr>
<tr>
<td></td>
<td>32</td>
<td></td>
<td>63.70 ± 4.40</td>
<td>65.33 ± 3.83</td>
<td>61.06 ± 3.89</td>
<td>62.49 ± 4.48</td>
<td>67.44 ± 01.22</td>
<td>71.21 ± 3.28</td>
</tr>
<tr>
<td>Disaster</td>
<td>2</td>
<td>4</td>
<td>55.73 ± 10.29</td>
<td>51.26 ± 9.16</td>
<td>52.87 ± 6.16</td>
<td>50.61 ± 8.33</td>
<td>51.45 ± 4.25</td>
<td>55.26 ± 8.32</td>
</tr>
<tr>
<td></td>
<td>8</td>
<td>16</td>
<td>56.31 ± 97.57</td>
<td>67.89 ± 6.83</td>
<td>56.08 ± 7.48</td>
<td>54.93 ± 7.88</td>
<td>55.96 ± 3.58</td>
<td>63.62 ± 6.84</td>
</tr>
<tr>
<td></td>
<td>16</td>
<td>32</td>
<td>64.52 ± 68.93</td>
<td>72.86 ± 1.70</td>
<td>65.83 ± 4.19</td>
<td>60.70 ± 6.06</td>
<td>61.32 ± 2.83</td>
<td>70.56 ± 2.23</td>
</tr>
<tr>
<td></td>
<td>32</td>
<td></td>
<td>73.60 ± 0.18</td>
<td>73.69 ± 2.32</td>
<td>67.13 ± 3.11</td>
<td>72.52 ± 2.28</td>
<td>63.77 ± 2.34</td>
<td>71.80 ± 1.85</td>
</tr>
<tr>
<td>Emotion</td>
<td>13</td>
<td>4</td>
<td>09.20 ± 3.22</td>
<td>09.84 ± 1.09</td>
<td>09.41 ± 2.10</td>
<td>09.84 ± 2.14</td>
<td>11.71 ± 2.16</td>
<td>11.90 ± 1.74</td>
</tr>
<tr>
<td></td>
<td>8</td>
<td>16</td>
<td>08.21 ± 2.12</td>
<td>11.02 ± 1.02</td>
<td>11.61 ± 2.34</td>
<td>11.21 ± 2.11</td>
<td>12.90 ± 1.63</td>
<td>13.26 ± 1.01</td>
</tr>
<tr>
<td></td>
<td>16</td>
<td>32</td>
<td>13.43 ± 2.51</td>
<td>12.05 ± 1.18</td>
<td>13.82 ± 2.02</td>
<td>12.75 ± 2.04</td>
<td>13.38 ± 2.20</td>
<td>15.17 ± 0.89</td>
</tr>
<tr>
<td></td>
<td>32</td>
<td></td>
<td>16.66 ± 1.24</td>
<td>14.28 ± 1.11</td>
<td>13.81 ± 1.62</td>
<td>16.88 ± 1.80</td>
<td>14.81 ± 2.01</td>
<td>16.08 ± 1.16</td>
</tr>
<tr>
<td>Political Bias</td>
<td>2</td>
<td>4</td>
<td>54.57 ± 5.02</td>
<td>57.72 ± 5.72</td>
<td>54.32 ± 3.90</td>
<td>54.66 ± 3.74</td>
<td>60.49 ± 6.66</td>
<td>61.17 ± 4.91</td>
</tr>
<tr>
<td></td>
<td>8</td>
<td>16</td>
<td>56.15 ± 3.75</td>
<td>63.02 ± 4.62</td>
<td>57.36 ± 4.32</td>
<td>54.79 ± 4.19</td>
<td>61.74 ± 6.73</td>
<td>64.10 ± 4.03</td>
</tr>
<tr>
<td></td>
<td>16</td>
<td>32</td>
<td>60.96 ± 4.25</td>
<td>66.35 ± 2.84</td>
<td>59.24 ± 4.23</td>
<td>60.30 ± 3.26</td>
<td>65.08 ± 2.14</td>
<td>66.11 ± 2.04</td>
</tr>
<tr>
<td></td>
<td>32</td>
<td></td>
<td>65.04 ± 2.32</td>
<td>67.73 ± 2.27</td>
<td>62.68 ± 3.21</td>
<td>64.99 ± 3.05</td>
<td>64.67 ± 3.41</td>
<td>67.30 ± 1.53</td>
</tr>
</tbody>
</table>

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

More results & analysis in the paper!
Summary of Unsupervised Meta-Training

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
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

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 Mon Nov 6.

Next week: Bayesian meta-learning