Review

• Last lecture on model compression:
  • Motivation
  • Key idea: knowledge distillation (KD)
  • Pioneering KD model for image classification
  • Pioneering KD model for object detection
  • State-of-the-art for KD

• Assignments (Canvas):
  • Final project outline due earlier today
  • Project presentation (poster and video) due in 2.5 weeks

• Questions?
Efficient Learning: Today’s Topics

• Motivation

• Curriculum Learning

• Active Learning

• Data Distillation

• Few-shot Learning

• Interview by Lijun Chen
Efficient Learning: Today’s Topics

• Motivation

• Curriculum Learning

• Active Learning

• Data Distillation

• Few-shot Learning

• Interview by Lijun Chen
Trend: **Extensive Training**

Models perform better with more training data (paired with parameter-heavy architectures):

Hu et al. Scaling Up Vision-Language Pre-training for Image Captioning. CVPR 2022
Trend: **Extensive Training**

2022: How many training examples led to top performance in Vision Transformers?

- 3 million
- 30 million
- 300 million
- 3 billion
- 30 billion

It takes 2,500 TPUv3-core-days to train this model

Zhai et al. Scaling Vision Transformers. CVPR 2022
Boss: What did you do last month?
You: Trained the model for one epoch.

Boss: Umm, fine, what is your plan for next month?
You: Train… train the model for one more epoch?
Why Is Extensive Training Costly?

- Time-consuming
- Expensive
- Increased environmental impact from carbon emissions
When Is Extensive Training Unrealistic?

1. On-device adaptation (e.g., because of privacy concerns and poor/no internet connection):

2. Rare content for which there is a scarcity of data (e.g., private content including medical information, natural disasters, rare locations such as outer space)

Figure: https://aws.amazon.com/blogs/machine-learning/demystifying-machine-learning-at-the-edge-through-real-use-cases/
How to teach machines so they learn more efficiently: (1) faster and (2) with fewer resources?
Efficient Learning: Today’s Topics

• Motivation

• Curriculum Learning

• Active Learning

• Data Distillation

• Few-shot Learning

• Interview by Lijun Chen
Intuition: How to Teach a Child Math?

Random Order of Examples

Meaningful Order of Examples

Big Book of Math; Dinah Zike
Intuition: How to Teach a Child To Read

Random Order of Examples

Meaningful Order of Examples
Idea: Teach Machines As We Teach Humans

Curriculum

Train with simpler examples first and progressively harder examples over time

Key Evaluation Metrics

• Training convergence speed

• Generalization performance on test data
Pioneering Task: Shape Prediction

Classify each shape as rectangle, ellipse, or triangle

Solution: 3-layer neural network
1. Easy (Basic): less shape variability (squares, circles, and equilateral triangles); 10,000 examples
2. Hard (Geom): more shape variability (rectangles, ellipses, and triangles); 10,000 examples

Bengio et al., Curriculum Learning, 2009
Shape Prediction: Curriculum Learning

Results of training on “easy” examples for $n$ epochs and then training on “hard” examples until 256 epochs or validation error hits minimum (20 random initializations).

What are the benefits of curriculum learning?

How many epochs should the algorithm train with easy examples before switching to difficult examples?

No curriculum

Bengio et al., Curriculum Learning, 2009
EfficientTrain

Key idea: eliminate difficult patterns from all training examples at earlier learning stages by removing higher-frequency content

EfficientTrain: Exploring Generalized Curriculum Learning for Training Visual Backbones

Yulin Wang¹* Yang Yue¹* Rui Lu¹ Tianjiao Liu² Zhao Zhong² Shiji Song¹ Gao Huang¹,³✉

¹Department of Automation, BNRist, Tsinghua University ²Huawei Technologies Ltd. ³BAAI
{wang-y119, yueyang22}@mails.tsinghua.edu.cn, gaohuang@tsinghua.edu.cn
EfficientTrain: Key Idea

~20% training cost eliminated by initially training on lower resolution, low-frequency images to learn low-frequency information typically learned first during training.
Key Questions In Creating “Curriculum”

• How to define what is “easy” versus “hard”?

• How many levels to include in the curriculum from easy to hard?
Efficient Learning: Today’s Topics

• Motivation

• Curriculum Learning

• Active Learning

• Data Distillation

• Few-shot Learning

• Interview by Lijun Chen
How to teach machines with minimal human supervision?

e.g., limited access to (expert) annotators

e.g., limited funding
Idea: Choose Most Informative Data to Label

Stream-Based

- Consider one example at a time

Pool-Based

- Consider many examples at a time

Image Credit: https://www.cs.utah.edu/~piyush/teaching/10-11-slides.pdf
Active Learning for Neural Networks: Status Quo

Iteratively add more labelled training examples after $n$ epochs; different from curriculum learning because labels need to be collected for the added data.

Consider many examples at a time

Image Credit: https://www.cs.utah.edu/~piyush/teaching/10-11-slides.pdf
What approach might be effective in identifying the most informative data to label?
Common Approach: Uncertainty Sampling

Query instance(s) the classifier is most uncertain about.

True Representation (Assume Labels Are Not Known)

Passive Learner (Random Selection)

Active Learner (Uncertainty Sampling)

e.g., Uncertainty Estimation for Neural Networks Using Robustness Testing

Use model’s predictions on random augmentations of the input to measure consistency/uncertainty; e.g.,

Figure Source: https://learnopencv.com/understanding-alexnet/

Elezi et al. Not all labels are equal: rationalizing the labeling costs for training object detection. CVPR 2022
e.g., Uncertainty Estimation for Neural Networks Using Ensembles (Two Approaches)

1. Dropout with different masks at inference time

2. Multiple neural networks

Figure Source: Srivastava et al. Dropout: A Simple Way to Prevent Neural Networks from Overfitting. Journal of Machine Learning Research. 2014

Predicted softmax probabilities used to estimate uncertainty (e.g., entropy across softmax values), with average taken across all ensemble’s softmax distributions

Beluch et al. The power of ensembles for active learning in image classification. CVPR 2018
e.g., Uncertainty Estimation for Neural Networks Using Ensembles (Two Approaches)

Active learning methods lead to faster learning and reduced human annotation effort than passive (random) learning for two image classification datasets.

Beluch et al. The power of ensembles for active learning in image classification. CVPR 2018
Common AL Techniques Have Mixed Results

- **Successes**: image classification, object detection
- **Failure**: VQA (e.g., AL methods label 10% of overall pool per iteration; initial model trained on 10% of pool)

Karamcheti et al. Mind your outliers! Investigating the negative impact of outliers on active learning for visual question answering. Association for Computational Linguistics (ACL) 2021
Common AL Techniques Have Mixed Results

Why might AL methods perform comparable or worse to random selection?
- Challenging examples to learn are sampled; e.g.,

<table>
<thead>
<tr>
<th>VQA-2</th>
<th>GQA</th>
</tr>
</thead>
<tbody>
<tr>
<td><img src="vqa-2.jpg" alt="Image" /></td>
<td><img src="gqa.jpg" alt="Image" /></td>
</tr>
<tr>
<td><strong>External knowledge:</strong> What does the symbol on the blanket mean?</td>
<td><strong>OCR:</strong> What is the first word on the black car?</td>
</tr>
<tr>
<td><strong>Underspecification:</strong> What is on the shelf?</td>
<td><strong>Multi-hop reasoning:</strong> What is the vehicle that is driving down the road the box is on the side of?</td>
</tr>
</tbody>
</table>

Figure 7: Example groups of collective outliers in the VQA-2 and GQA datasets.

Karamcheti et al. Mind your outliers! Investigating the negative impact of outliers on active learning for visual question answering. Association for Computational Linguistics (ACL) 2021
Idea: Remove “Unlearnable” Data from Pool

Performance compared to random selection improves for AL approaches when removing “challenging” examples from data pool

Karamcheti et al. Mind your outliers! Investigating the negative impact of outliers on active learning for visual question answering. Association for Computational Linguistics (ACL) 2021
Efficient Learning: Today’s Topics

• Motivation

• Curriculum Learning

• Active Learning

• Data Distillation

• Few-shot Learning

• Interview by Lijun Chen
Key Idea: Distill Large Dataset Into a Small Number of Synthetic Images

Pioneering Paper

Synthetic training images enable more efficient training (i.e., less training images and so gradient descent steps) while reducing data storage costs and bypassing privacy concerns.

Wang et al. Dataset Distillation. arXiv 2018
Typical Approach: Many Optimization Objectives

**Algorithm 1: Dataset Distillation Framework**

**Input:** Original dataset $\mathcal{T}$

**Output:** Synthetic dataset $S$

Initialize $S$ $\triangleright$ Random, real, or core-set

**while not converge do**

- Get a network $\theta$ $\triangleright$ Random or from some cache
- Update $\theta$ and cache it if necessary $\triangleright$ Via $S$ or $\mathcal{T}$, for some steps
- Update $S$ via $\mathcal{L}(S, \mathcal{T})$ $\triangleright$ PerM, ParM, DisM, or their variants

**end**

**return $S$**
Many Optimization Objectives

Dataset Distillation

- Performance Matching
  - Meta Learning
  - KRR
- Distribution Matching
  - Single Layer
  - Multi Layer
- Parameter Matching
  - Single-Step
  - Multi-Step

“key idea... train the same network using synthetic datasets and original datasets for some steps, respectively, and encourage the consistency of their trained neural parameter”

“obtain synthetic data whose distribution can approximate that of real data” rather than match impacts on training

https://blog.roboflow.com/what-is-dataset-distillation/
Efficient Learning: Today’s Topics

• Motivation

• Curriculum Learning

• Active Learning

• Data Distillation

• Few-shot Learning

• Interview by Lijun Chen
Intuition: Generalize Current Knowledge

Given one example per category, identify the category of the query

Lake et al, 2013, 2015

https://www.youtube.com/watch?v=9j4iH9TPTd8
Problem Set-up: Learn from Few Examples

Often, 1-10 examples

https://daredevilmusicproduction.com/long-tail/
Problem Set-up: Learn from Few Examples

- Few shot learning: evaluate only for categories with few examples
- Generalized few shot learning: evaluate on all categories

[Diagram of model learning from sufficient and insufficient labeled data]
What are applications for which we might have limited examples?

- medical
- outer space
- natural disasters
Popular Approaches

• Design-time approach: fine-tuning

• Run-time approach: meta learning
Popular Approaches

• Design-time approach: fine-tuning

• Run-time approach: meta learning
Recall Fine-Tuning

e.g., Fine-Tuning for Object Detection

Faster R-CNN architecture: Why would we anticipate learned features would generalize well to locating novel categories?


e.g., Fine-Tuning for Object Detection

Stage I: Base training

Stage II: Few-shot fine-tuning

$K$ shots from both base and novel categories used for training

Why include shots from both base and novel categories?

### e.g., Fine-Tuning for Object Detection

Tested with cross validation on 3 splits from VOC

mAP scores for training with 1, 2, 3, 5, and 10 examples (shots) per category

<table>
<thead>
<tr>
<th>Method / Shot</th>
<th>Backbone</th>
<th>Novel Set 1</th>
<th>Novel Set 2</th>
<th>Novel Set 3</th>
</tr>
</thead>
<tbody>
<tr>
<td>YOLO-joint (Kang et al., 2019)</td>
<td>YOLOv2</td>
<td>0.0 0.0 1.8 1.8 1.8</td>
<td>0.0 0.1 0.0 1.8 0.0</td>
<td>1.8 1.8 1.8 3.6 3.9</td>
</tr>
<tr>
<td>YOLO-ft (Kang et al., 2019)</td>
<td>YOLOv2</td>
<td>3.2 6.5 6.4 7.5 12.3</td>
<td>8.2 3.8 3.5 3.5 7.8</td>
<td>8.1 7.4 7.6 9.5 10.5</td>
</tr>
<tr>
<td>YOLO-ft-full (Kang et al., 2019)</td>
<td>YOLOv2</td>
<td>6.6 10.7 12.5 24.8 38.6</td>
<td>12.5 4.2 11.6 16.1 33.9</td>
<td>13.0 15.9 15.0 32.2 38.4</td>
</tr>
<tr>
<td>FSRW (Kang et al., 2019)</td>
<td>FRCN w/VGG16</td>
<td>14.8 15.5 26.7 33.9 47.2</td>
<td>15.7 15.3 22.7 30.1 40.5</td>
<td>21.3 25.6 28.4 42.8 45.9</td>
</tr>
<tr>
<td>MetaDet (Wang et al., 2019b)</td>
<td>FRCN w/R-101</td>
<td>17.1 19.1 28.9 35.0 48.8</td>
<td>18.2 20.6 25.9 30.6 41.5</td>
<td>20.1 22.3 27.9 41.9 42.9</td>
</tr>
<tr>
<td>FRCN+joint (Wang et al., 2019b)</td>
<td>FRCN w/R-101</td>
<td>0.3 0.0 1.2 0.9 1.7</td>
<td>0.0 0.0 1.1 1.9 1.7</td>
<td>0.2 0.5 1.2 1.9 2.8</td>
</tr>
<tr>
<td>FRCN+joint-ft (Wang et al., 2019b)</td>
<td>FRCN w/VGG16</td>
<td>9.1 10.9 13.7 25.0 39.5</td>
<td>10.9 13.2 17.6 19.5 36.5</td>
<td>15.0 15.1 18.3 33.1 35.9</td>
</tr>
<tr>
<td>MetaDet (Wang et al., 2019b)</td>
<td>FRCN w/R-101</td>
<td>18.9 20.6 30.2 36.8 49.6</td>
<td>21.8 23.1 27.8 31.7 43.0</td>
<td>20.6 23.9 29.4 43.9 44.1</td>
</tr>
<tr>
<td>FRCN+joint (Yan et al., 2019)</td>
<td>FRCN w/R-101</td>
<td>2.7 3.1 4.3 11.8 29.0</td>
<td>1.9 2.6 8.1 9.9 12.6</td>
<td>5.2 7.5 6.4 6.4 6.4</td>
</tr>
<tr>
<td>FRCN+ft (Yan et al., 2019)</td>
<td>FRCN w/R-101</td>
<td>11.9 16.4 29.0 36.9 36.9</td>
<td>5.9 8.5 23.4 29.1 28.8</td>
<td>5.0 9.6 18.1 30.8 43.4</td>
</tr>
<tr>
<td>FRCN+ft-full (Yan et al., 2019)</td>
<td>FRCN w/R-101</td>
<td>13.8 19.6 32.8 41.5 45.6</td>
<td>7.9 15.3 26.2 31.6 39.1</td>
<td>9.8 11.3 19.1 35.0 45.1</td>
</tr>
<tr>
<td>Meta R-CNN (Yan et al., 2019)</td>
<td>FRCN w/R-101</td>
<td>19.9 25.5 35.0 45.7 51.5</td>
<td>10.4 19.4 29.6 34.8 45.4</td>
<td>14.3 18.2 27.5 41.2 48.1</td>
</tr>
<tr>
<td>FRCN+ft-full (Our Impl.)</td>
<td>FRCN w/R-101</td>
<td>15.2 20.3 29.0 40.1 45.5</td>
<td>13.4 20.6 28.6 32.4 38.8</td>
<td>19.6 20.8 28.7 42.2 42.1</td>
</tr>
<tr>
<td>TFA w/ fc (Ours)</td>
<td>FRCN w/R-101</td>
<td>36.8 29.1 43.6 55.7 57.0</td>
<td>18.2 29.0 33.4 35.5 39.0</td>
<td>27.7 33.6 42.5 48.7 50.2</td>
</tr>
<tr>
<td>TFA w/ cos (Ours)</td>
<td>FRCN w/R-101</td>
<td><strong>39.8</strong> 36.1 <strong>44.7</strong> 55.7 56.0</td>
<td><strong>23.5</strong> 26.9 <strong>34.1</strong> 35.1 39.1</td>
<td><strong>30.8</strong> 34.8 <strong>42.8</strong> 49.5 49.8</td>
</tr>
</tbody>
</table>

Consistently outperforms baselines by 2-20 points on novel categories

e.g., Fine-Tuning for Object Detection

Tested with cross validation on 3 splits from VOC

mAP scores for training with 1, 2, 3, 5, and 10 examples (shots) per category

<table>
<thead>
<tr>
<th>Method / Shot</th>
<th>Backbone</th>
<th>Novel Set 1</th>
<th>Novel Set 2</th>
<th>Novel Set 3</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td>1</td>
<td>2</td>
<td>3</td>
</tr>
<tr>
<td>YOLO-joint (Kang et al., 2019)</td>
<td>YOLOv2</td>
<td>0.0</td>
<td>0.0</td>
<td>1.8</td>
</tr>
<tr>
<td>YOLO-ft (Kang et al., 2019)</td>
<td></td>
<td>3.2</td>
<td>6.5</td>
<td>6.4</td>
</tr>
<tr>
<td>YOLO-ft-full (Kang et al., 2019)</td>
<td></td>
<td>6.6</td>
<td>10.7</td>
<td>12.5</td>
</tr>
<tr>
<td>FSRW (Kang et al., 2019)</td>
<td></td>
<td>14.8</td>
<td>15.5</td>
<td>26.7</td>
</tr>
<tr>
<td>MetaDet (Wang et al., 2019b)</td>
<td></td>
<td>17.1</td>
<td>19.1</td>
<td>28.9</td>
</tr>
<tr>
<td>FRCN+joint (Wang et al., 2019b)</td>
<td>FRCN w/VGG16</td>
<td>0.3</td>
<td>0.0</td>
<td>1.2</td>
</tr>
<tr>
<td>FRCN+joint-ft (Wang et al., 2019b)</td>
<td></td>
<td>9.1</td>
<td>10.9</td>
<td>13.7</td>
</tr>
<tr>
<td>MetaDet (Wang et al., 2019b)</td>
<td></td>
<td>18.9</td>
<td>20.6</td>
<td>30.2</td>
</tr>
<tr>
<td>FRCN+joint (Yan et al., 2019)</td>
<td>FRCN w/R-101</td>
<td>2.7</td>
<td>3.1</td>
<td>4.3</td>
</tr>
<tr>
<td>FRCN+ft (Yan et al., 2019)</td>
<td></td>
<td>11.9</td>
<td>16.4</td>
<td>29.0</td>
</tr>
<tr>
<td>FRCN+ft-full (Yan et al., 2019)</td>
<td></td>
<td>13.8</td>
<td>19.6</td>
<td>32.8</td>
</tr>
<tr>
<td>Meta R-CNN (Yan et al., 2019)</td>
<td></td>
<td>19.9</td>
<td>25.5</td>
<td>35.0</td>
</tr>
<tr>
<td>FRCN+ft-full (Our Impl.)</td>
<td></td>
<td>15.2</td>
<td>20.3</td>
<td>29.0</td>
</tr>
<tr>
<td>TFA w/ fc (Ours)</td>
<td>FRCN w/R-101</td>
<td>36.8</td>
<td>29.1</td>
<td>43.6</td>
</tr>
<tr>
<td>TFA w/ cos (Ours)</td>
<td></td>
<td>39.8</td>
<td>36.1</td>
<td>44.7</td>
</tr>
</tbody>
</table>

Similar performance boosts also observed on two more datasets (COCO and LVIS)

Fine-Tuning

What are limitations of this approach for real-world applications?

- Must retrain algorithm to add new categories
Popular Approaches

• Design-time approach: fine-tuning

• Run-time approach: meta learning
Meta Learner: Update Model with Support Set
Implementation: Trained Model Updates Itself to Generalize to Support Set Categories

Goal: learn features during training that are class-agnostic and so can generalize to novel test categories

Given support categories, detect which one the “query” matches

Implementation: Trained Model Updates Itself to Generalize to Support Set Categories

How many “shots” are observed at each training round?

- 4 (per category)
Implementation: Trained Model Updates Itself to Generalize to Support Set Categories

How many shots are observed at testing?

Implementation: Trained Model Updates Itself to Generalize to Support Set Categories

Detect which support category the “query” matches (recall support categories are never observed at training!)

Implementation: Trained Model Updates Itself to Generalize to Support Set Categories

How to train a model to do this?

Implementation: Trained Model Updates Itself to Generalize to Support Set Categories

Adapted from Finn ‘17

https://www.youtube.com/watch?v=9j4iH9TPTd8
Implementation: Trained Model Updates Itself to Generalize to Support Set Categories

- Santoro et al. ’16
- Duan et al. ’17
- Wang et al. ’17
- Munkhdalai & Yu ’17
- Mishra et al. ’17
- ...

Adapted from Finn ’17

e.g., learn set-invariant neural networks, such as those that rely on attention, to locate similarity

https://www.youtube.com/watch?v=9j4iH9TPTd8
Implementation: Trained Model Updates Itself to Generalize to Support Set Categories

Support
- Koch ’15
- Vinyals et al. ’16
- Snell et al. ’17
- Shyam et al. ’17
- Sung et al. ’17
- ...

Query

Compare query to each support category; e.g., establish a “prototype” for each support set


https://www.youtube.com/watch?v=9j4iH9TPTd8
Implementation: Trained Model Updates Itself to Generalize to Support Set Categories

Function to optimize is conditioned on the support set; e.g., tweak “forget” gate of LSTM

https://www.youtube.com/watch?v=9j4iH9TPTd8
What are limitations of this approach for real-world applications?

- Requires large amount of memory to process the support set on top of the query set

Popular Approaches

- Design-time approach: fine-tuning
- Run-time approach: meta learning
Efficient Learning: Today’s Topics

• Motivation

• Curriculum Learning

• Active Learning

• Data Distillation

• Few-shot Learning

• Interview by Lijun Chen
The End