The Data Challenge
Neural networks need data

• ImageNet contains millions of labeled images
  • Extremely expensive to collect
  • Host of ethical issues

• Most domains do not have large labeled datasets

• What can we do?
The “fundamental law” of neural networks

• Neural networks must be trained on a large dataset
• If not enough labeled data for target task, then what?
  • Unlabeled data from target domain: *Self-supervised learning*
  • Labeled + Unlabeled data for target task: *Semi-supervised learning*
  • Labeled data from a related problem domain: *Few-shot / transfer learning*
Learning from unlabeled data: Self-supervised learning

- Two classes of approaches

- *Pretext*-based learning
  - Design a “pretext” task that leads to good features

- *Contrastive* learning
  - Spread images out in feature space
Pretext tasks

• Transform input, task network with predicting transformation
Pretext tasks

• Remove data, then task network with predicting it
Pretext tasks

• Use some source with additional data
• E.g. videos
1. Collect videos
2. Segment using motion
3. Train ConvNet

Ego-motion ↔ vision: view prediction

After moving:

Slide credit: Dinesh Jayaraman
Approach idea: Ego-motion equivariance

Training data
Unlabeled video + motor signals

Equivariant embedding
organized by ego-motions

Pairs of frames related by similar ego-motion should be related by same feature transformation

Learn

Slide credit: Dinesh Jayaraman
Self-supervision from multimodal data

Owens et al, CVPR 2016
Comparison

• Train on ImageNet w/o labels
• Use features to train linear classifier on scene classification (Places205)
Contrastive learning

• Training for classification is great!
• However, no class labels 😞
• Idea: let data define the classes
DeepCluster

Use pseudo-labels to produce representation

Use representation to cluster dataset
Instance Discrimination

• Simpler idea: \textit{let each image (+ data augmentations) be its own class}

• Challenge: number of classes too many!
SimCLR

- Sample a batch of images $x_1, \ldots, x_n$
- Augment each to produce $x_{n+1}, \ldots, x_{2n}$
- Loss = $-\log \sum_i \frac{e^{-d(x_i, x_{i+n})}}{\sum_{k \neq i} e^{-d(x_k, x_i)}}$
Why does this work?

• Data augmentation?
Curioser and curioser

Exponential moving average

input image $x$ to input view $v$: $f_{\theta}$

representation $y_{\theta}$: $g_{\theta}$

projection $z_{\theta}$: $q_{\theta}$

prediction $q_{\theta}(z_{\theta})$ online

Exponential moving average

loss

target

$sg(z'_{\xi})$
Why does this work

• Simple mechanism:

• *Spread images out in feature space while ensuring invariance to augmentation*

• Current techniques appear to be as good as supervised training

• But need much longer training, large datasets
Semi-supervised learning

• What if we have both labeled and unlabeled data?
• E.g., dataset only partially labeled
Semi-supervised learning I – Self-training / Pseudo-labeling
Semi-supervised learning II – Entropy minimization

• Loss function on labeled examples: standard negative log likelihood
• Loss function on unlabeled examples: entropy
  • \( H(p) = -\sum p_i \log p_i \)
  • Entropy is high when probabilities are uniform
  • Minimize entropy \(\rightarrow\) encourage classifier to be more confident
Semi-supervised learning III – Consistency regularization

• Loss on unlabeled images: *consistency* between predictions on augmented versions

\[
l_{lU}^{TS} = \sum_{j=1}^{n-1} \sum_{k=j+1}^{n} \| f^j(T^j(x_i)) - f^k(T^k(x_i)) \|_2^2
\]
Semi-supervised learning IV - FixMatch
Semi-supervised learning V – S4L

• Simple idea: use self-supervised loss on unlabeled data
• “Self-supervision for semi-supervised learning”
Few-shot learning

Base classes (many training examples) → Representation learning → Feature extractor → Low-shot learning → Novel classes (few training examples) → Classifier (base and novel categories)
The challenge: Intra-class variation
Philippine Tarsier

“Train set”

Philippine Tarsier

“Test set”

Mouse lemur

Beaver
Key cue: shared modes of variation
How do humans do this?

Bird, grey color, long beak, long legs, black markings on head,...

More invariant representations

Inductive biases during learning
Better representations: metric learning

True class boundary
Better representations: metric learning

“One-shot” class boundary
Metric learning

- Pull same-class pairs closer and different-class pairs apart
- Contrastive loss (DrLIM)
  - $= d(x, x')^2$ if $y = y'$
  - $= \max(0, m - d(x, x'))^2$ if $y \neq y'$
- Triplet loss
  - $= \max(d(x, x_+) - d(x, x_-) + \gamma, 0)$

Meta-learning

• Given:

Small training set
(few training examples)

• Produce:

• Idea: Make this a learnable function!
Meta-learning

\[ h = \mathcal{A}(S_{train}) \]
\[ \hat{p} = m(x) \]
Meta-learning: training

$D \xrightarrow{S_{\text{train}}} h \xrightarrow{w} \hat{p} \xrightarrow{L} S_{\text{test}}$

$\hat{p}$

Image

Label
An army of meta-learners


Meta-learning : MAML

• Given training set $S$, query example $q$, need function $h(S, q ; w)$
• Idea:
  • $w$ is initialization of neural network
  • $h$ does a few SGD steps using $S$ and then classifies $q$
  • Backpropagating through $h$ is difficult but can be done
Meta-learning: Prototypical Networks
Meta-learning: FEAT
Meta-learning: FRN