Addressing the data challenge: Transfer learning and semi-supervised learning
The data challenge

• Fundamentally, neural networks need a lot of data

• Why?
  • Lots of parameters
  • Deeper, bigger models are better in CV, and they all have many more parameters

• Large datasets are problematic
  • Expensive to collect
  • Expensive to curate
  • Expensive to label
  • Associated issues of bias
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
Classical unsupervised learning

• PCA (Principal Components Analysis)
  • Reduces dimensionality
  • But is a linear approach
Pretext tasks

- Transform input, task network with predicting transformation

[Diagram with images and a table showing numbers 1, 3, 4, and 2]
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

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: 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
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
Classical unsupervised learning

• Unsupervised learning is *old*
• Even with handcrafted features, some feature transformations are necessary
• E.g.: *spurious correlations between features* cause problems doing learning
  • If a car is always seen on a road, then learning algorithm may latch on to the road
Classical unsupervised learning

• Typically want features to be *independent* and *uncorrelated*
• What do uncorrelated features look like?
• If each feature dimension is normally distributed, and features are all independent
  • Multivariate Gaussian with identity covariance!
Classical unsupervised learning

• Whitening
  • Linear transformation to make the data have identity covariance
  • Closely related to LDA (Linear discriminant analysis), one of the earliest classification algorithm
Classical unsupervised learning

• But classical whitening is limited by linear transforms
  • Will remove only first order correlations
Deep unsupervised learning

• Key question: can we get a deep network to remove all correlations?
• Has been the subject of study for many years
• Contrastive learning turns out to be very good at this!
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
  • $H(p) = -\sum_i p_i \log p_i$
  • 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

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

• Simple idea: use *self-supervised loss* on unlabeled data
• “Self-supervision for semi—upervised learning”
Limitations of semi-supervised learning

• Still needs at least 10s of examples per class
• Need unlabeled data
Few-shot learning

Base classes (many training examples) | Feature extractor | Novel classes (few training examples) | Classifier (base and novel categories)

Representation learning | Low-shot learning
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

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

$S_{\text{train}}$

$S_{\text{test}}$

$h$

$w$

$\hat{p}$
Meta-learning: training
An army of meta-learners

Meta-learning : MAML

• Given training set $S$, query example $q$, need function $h(S, q ; \mathbf{w})$

• Idea:
  • $\mathbf{w}$ 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

(a) Instance Embedding
Meta-learning: FRN

Support Images $X_s$

Query Image
Transfer vs self-supervision vs few-shot on new domains

**ISIC**
- Best FSL
- Naïve transfer
- SimCLR

**EuroSAT**
- Best FSL
- Naïve transfer
- SimCLR
A magic ingredient

Step 1: Train classifier in source domains

Step 2: Pseudo-label target domain unlabeled data

Step 3: Use pseudo-labels + self-supervision to train target domain features
The magic of self-training in 3 steps

1. Pre-train convnet on source domain (ImageNet)

2. Use pre-trained convnet on unlabeled data from target domain to get pseudolabels

3. Use pseudo-labels to train target domain representation (+SimCLR as potential aux. loss)

Self Training for Adapting Representations To Unseen Problems (under review)
STARTUP – what does it do?
Why does STARTUP work?

• Induced grouping can be still meaningful in the target domain
  • STARTUP performance correlated with this

• Training with induced grouping forces network to learn domain-specific features