Transfer Learning with CNN

Mai H. Nguyen
What is Transfer Learning?

- To overcome challenges of training model from scratch:
  - Insufficient data
  - Very long training time
- **Use pre-trained model**
  - Trained on another dataset
  - This serves as starting point for model
  - Then train model on current dataset for current task
Transfer Learning Approaches

• **Feature extraction**
  • Remove last fully connected layer from pre-trained model
  • Treat rest of network as feature extractor
  • Use features to train new classifier (“top model”)

• **Fine tuning**
  • Tune weights in some layers of original model (along with weights of top model)
  • Train model for current task using new dataset
CNNs for Transfer Learning

- Popular architectures
  - AlexNet
  - GoogLeNet
  - VGGNet
  - ResNet
- All winners of ILSVRC
  - ImageNet Large Scale Visual Recognition Challenge
  - Annual competition on vision tasks on ImageNet data
ImageNet

**Database**
- Developed for computer vision research
- > 14,000,000 images hand-annotated
- > 22,000 categories

**ILSVRC History**
- Started in 2010
- Various vision tasks on 1,000 object categories
- Object classification error rate
  - 2011: ~25%
  - 2012: 15.3% (AlexNet)
  - 2015: 3.57% (ResNet; better than human performance)
  - 2016: 2.99% (ensemble CNNs)
Why Does Transfer Learning Work?

Lee et al. ‘Convolutional Deep Belief Networks for Scalable Unsupervised Learning of Hierarchical Representations’ ICML 2009
VGG as Pre-Trained Network

Source: https://www.cs.toronto.edu/~frossard/post/vgg16/
Transfer Learning – Feature Extraction

Extract features here and feed to separate classifier

Source: https://www.cs.toronto.edu/~frossard/post/vgg16/
Transfer Learning – Fine Tuning

Adjust weights in top layers using new dataset

Source: https://www.cs.toronto.edu/~frossard/post/vgg16/
When & How to Fine Tune

• New dataset is small & similar to original dataset
  • Extract features from higher layer and feed to separate classifier

• New dataset is large & similar to original dataset
  • Fine tune top or all layers

• New dataset is small & different from original dataset
  • Extract features from lower layer and feed to separate classifier

• New dataset is large & different from original dataset
  • Fine tune top or all layers
Other Practical Tips

• **Learning rate**
  • Use very small learning rate for fine tuning. Don’t want to destroy what was already learned.

• **Start with properly trained weights**
  • Train top-level classifier first, then fine tune lower layers.
  • Top model with random weights may have negative effects on when fine tuning weights in pre-trained model

• **Data augmentation**
  • Simple ways to slightly alter images
    • Horizontal/vertical flips, random crops, translations, rotations, etc.
  • Use to artificially expand your dataset
References


Additional CNN Resources

- Keras Documentation. https://keras.io/
Questions?