AI Engineer Training: VII

In the Era of Deep Learning

IT21 Learning
Alvin Jin

Weekly AI News

- Demis Hassabis(DeepMind), was appointed as an advisor to the UK government national AI strategy.
- OpenAI Five plays against itself every day to beat human team at video game Dota 2.
- The 35th ICML is on going in Sweden with 600 accepted papers, many from Chinese researchers. Google:43, DeepMind:33, Google AI:24, Tencent: 11

Agenda

- Deep Learning in Computer Visions:
  - Transfer Learning
  - Network Design Patterns
- Case Studies:
  - Cats vs. Dogs: VGGNet

Transfer Learning

- Using a pre-trained network model to classify classes by learning patterns from new data it **wasn't** originally trained on.

- Feature Extraction
  - Treating networks as feature extractors, propagate the inputs until a given layer
  - Take these activations as feature vectors for new dataset.

- Fine-tuning
  - Removing the FC layers of an existing network, placing new FC layers, and fine-tuning their weights (and optionally previous layers) to specific classification task.

Feature Portability

- CONV layers come earlier in the model extract local, highly generic feature maps about the inputs, whereas layers that are higher up extract more abstract concepts about the targets.
- The representations learned by the classifier are specific to the classes on which the model is trained.
- The portability of learned features across different problems is a key advantage of deep learning.

Feature Extraction

- Feature extraction consists of taking the CONV base of a pre-trained network, running the new data **only once** through it.
- Activating feature maps to quantify the content of an image.
- Training a new model with either a standard ML classifier or a FC classifier on top of the CNN extracted features.

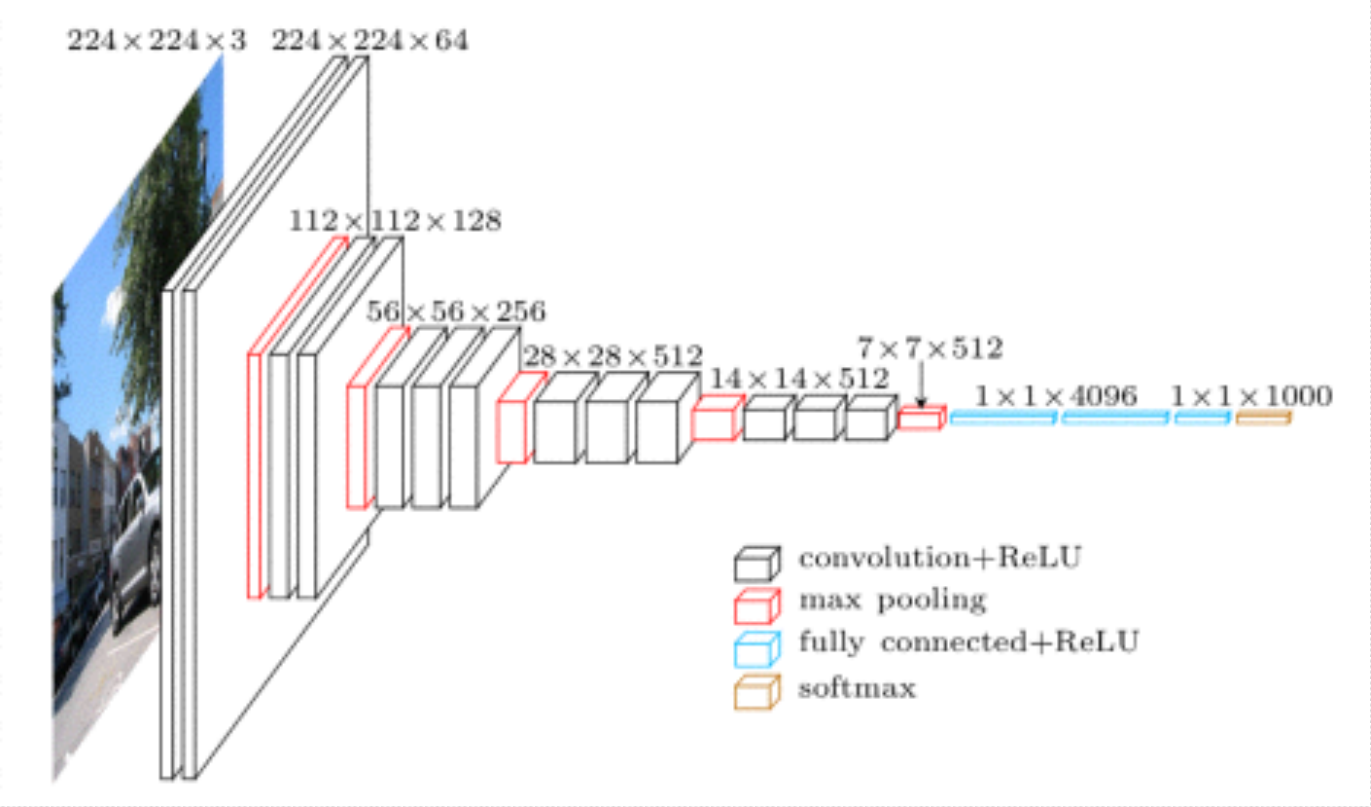
Feature Extraction Steps

- Cutting off the final set of FC layers from a pre-trained ConvNet
- Replacing the head with a new set of FC layers with random initializations. Normally, the new FC head will have fewer parameters than the original one.
- All layers below the head are frozen, so their weights cannot be updated during the training.
- Training the network using a small learning rate, so the new set of FC layers can learn discriminative filters from the previously learned CONV layers.

VGGNet

- ImageNet Dataset:
  - 1.2M training images
  - 50K validation and 100k testing images
  - 1000 object categories
- VGG won the 1st place of LSVRC2014
- VGG Network Size:
  - 16, 19 stands for the number of weight layers.
  - VGG16 model size 533MB

VGG16 Architecture



VGGNet Characteristics

- Using only 3x3 convolutional layers stacked on top of each other in increasing depth.
- Stacking multiple CONV => RELU layer sets before applying a single POOL layer.
- This allows the network to learn more rich features from the CONV layers prior to **downsampling** the spatial input size via POOL layer.
- Two fully-connected layers each with 4,096 nodes are then followed by a softmax classifier.
- The network weights are quite larger due to its depth and number of fully-connected nodes, and slow to train.

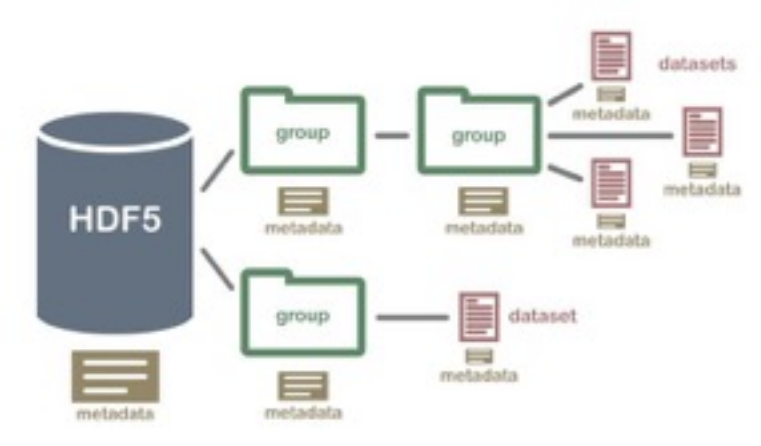
Downsampling

- Downsampling is to reduct the number of parameters ensures higher computational speeds.

- To downsample the feature maps by a factor of 2.
  - Max pooling is usually done with 2x2 windows and stride 2,
  - While convolution is typically done with 3x3 windows and stride 2.

Persisting Models in HDF5

- Feature Vectors/Model Weights are persisted into HDF5 format
- HDF5 is binary format to store large numerical datasets on disk.
- Data in HDF5 is stored hierarchically in groups, where a group is a container-like structure holding datasets and other groups.
- A dataset can be thought of as a multi-dimensional array of a homogeneous data type.



Fine-tuning

- Fine-tuning consists of:
  - Unfreeze a few top layers of a frozen base model
  - Jointly train these top layers and a new FC classifier.
- It slightly adjusts the abstract representations of the model being reused, to make them more relevant for the problem.
- The new classifier has to be trained in advance. Otherwise, the error signal propagating through the network is too large to train the unfrozen top layers.

Freezing Layers

- Training data is forward propagated through the network; The new added FC layers are randomly initialized.
- Freezing a set of layers means preventing their weights from being updated during training.
- Otherwise, large weight updates will be propagated through the network and destroy the representations previously learned.
- The back-propagation is stopped before the frozen layers, which allows unfrozen and new added layers to use patterns from the highly discriminative CONV layers

Fine-tuning Steps

- Replace the head with a new set of FC layers with fewer parameters and random initializations.
- Freeze the base network to avoid weights update.
- Train the FC layers(classifier) just added for warm-up.
- Unfreeze a few top CONV layers in the base network.
- Jointly train both the unfrozen CONV layers and the warm-up classifier layer.

Transfer Learning or Learn from Scratch?

- Considering dataset size and similarity with the pre-trained model was trained on.

	Similar Dataset	***Different Dataset***
Small Dataset	Feature extraction using FC layers + classifier	Feature extraction using lower level CONV layers+classifier
Large Dataset	Fine-tuning likely to work, but might have to train from scratch	Fine-tuning worth trying, but likely have to train from scratch

CNN Architecture Patterns

- INPUT
- => ((CONV => RELU) * N => POOL) * M
- => (FC => RELU) * K
- => FC
- 0 <= N <= 3; M >= 0; 0 <= K <= 2

- Stacking multiple CONV layers before applying a POOL layer allows the CONV layers to develop more complex features before the destructive pooling operation is performed.

- The depth of the feature maps progressively increases in the network, whereas the size of the feature maps decreases

- Dense layers learn global patterns in their input feature space, whereas convolution layers learn local patterns.

Rules of Thumb

- Using square inputs to take advantage of linear algebra optimization libraries.
- CONV layers should use smaller filter sizes 3x3 and 5x5.
- Using a stride 1 enables our CONV layers to learn filters, while the POOL layer is responsible for downsampling.
- Applying zero-padding when stacking multiple CONV layers increases classification accuracy.

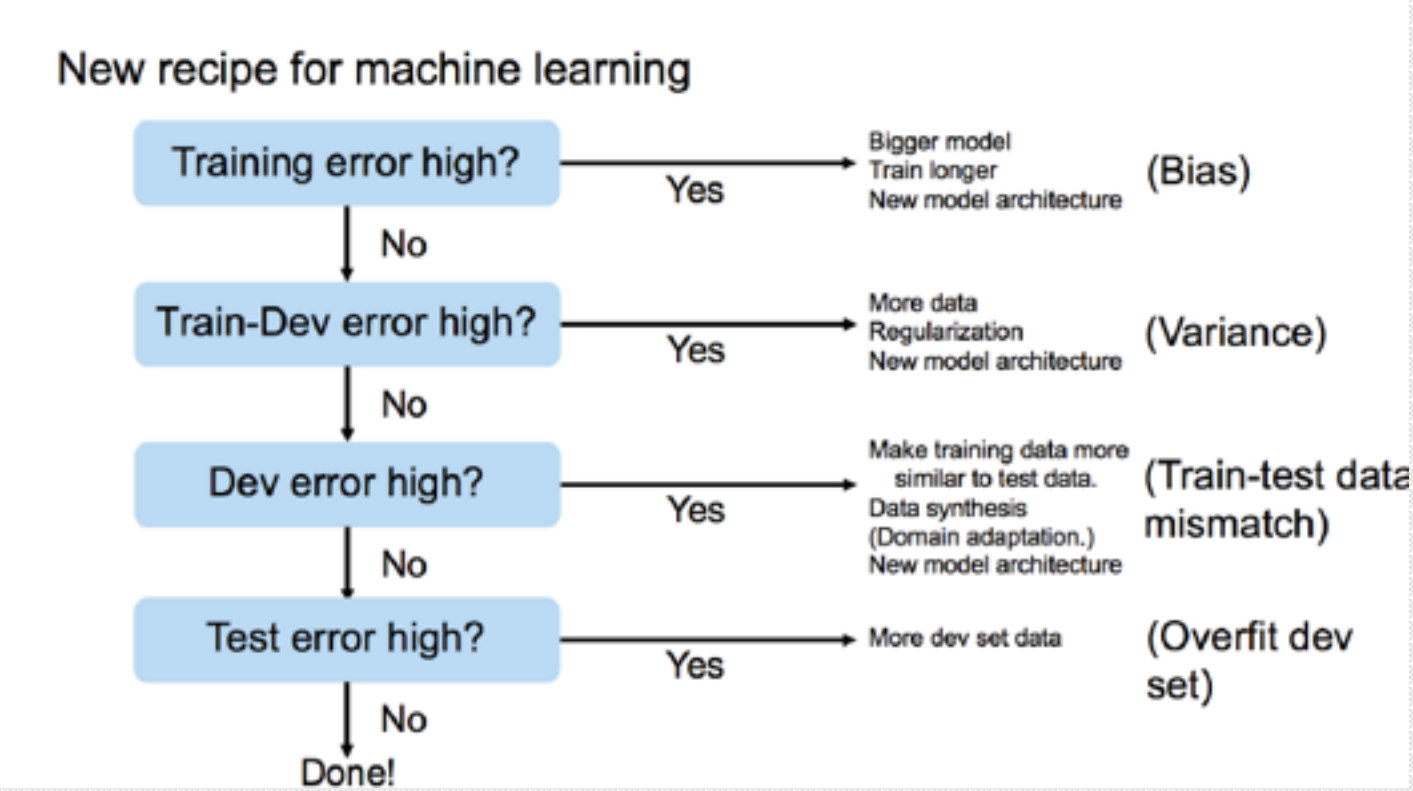
Rules of Thumb

- Most commonly, max pooling applied over a 2x2 receptive field size and a stride 2.

- Increasing the network capacity until overfitting becomes the primary obstacle. Then, fighting with overfitting to achieve high accuracy.

Recipes for Training

- Make sure your training data is representative of your validation and testing sets.



Q & A