Large-scale Image Classification

Brendan Jou, Joe Ellis, & Jie Feng
{bwj2105, jge2105, jf2776}@columbia.edu
“Large-scale” a decade ago

UIUC Cars (2004)
S. Agarwal, A. Awan, D. Roth

CMU/VASC Faces (1998)
H. Rowley, S. Baluja, T. Kanade

FERET Faces (1998)
P. Phillips, H. Wechsler, J. Huang, P. Raus

COIL Objects (1996)
S. Nene, S. Nayar, H. Murase

MNIST Digits (1998-10)
Y LeCun & C. Cortes

KTH Human Action (2004)
I. Leptev & B. Caputo

Sign Language (2008)
P. Buehler, M. Everingham, A. Zisserman

Segmentation (2001)

3D Textures (2005)
S. Lazebnik, C. Schmid, J. Ponce

CuRET Textures (1999)
K. Dana B. Van Ginneken S. Nayar J. Koenderink

CAVIAR Tracking (2005)
R. Fisher, J. Santos-Victor J. Crowley

Middlebury Stereo (2002)
D. Scharstein R. Szeliski

Slide courtesy: Fei-Fei Li
“Large-scale” today
“Large-scale” today
“Large-scale” today

[Graph showing classes vs. images for CALTECH101 and SCENE15 datasets]

Scene 15
15 classes, 5K images

- bedroom (FP)
- coast (OT)
- forest (OT)
- highway (OT)
- industrial (L)
- inside city (OT)
- kitchen (FP)
- living room (FP)
- mountain (OT)
- office (FP)
- open country (OT)
- store (L)
- street (OT)
- suburb (FP)
- tall building (OT)
“Large-scale” today

PASCAL VOC’07
20 classes, 10K images

Image courtesy: xerox & Inria
“Large-scale” today

Tiny (32x32) images
75K classes, 80M images

Image courtesy: xerox & Inria
“Large-scale” today

ImageNet: 1st release
5K classes, 3M images

Image courtesy: Xerox & Inria
“Large-scale” today

ImageNet: cur. release
22K classes, 14M images

Image courtesy: Xerox & INRIA
“Large-scale” today

where is the delineation of “large-scale”?
ImageNet is an image database organized according to the WordNet hierarchy (currently only the nouns), in which each node of the hierarchy is depicted by hundreds and thousands of images. Currently we have an average of over five hundred images per node. We hope ImageNet will become a useful resource for researchers, educators, students and all of you who share our passion for pictures.

Click here to learn more about ImageNet, Click here to join the ImageNet mailing list.

What do these images have in common? Find out!

Hierarchies and WordNet

more "discriminable" synsets | less "discriminable" synsets
---|---
"Basic-Level" | "Subordinate-" or "Superordinate-" Level
To deal with more samples from more categories, feature dimension gets higher to achieve more discriminative power.

Standard feature in image classification:

- BoW
- Large Codebook
- Spatial Pyramid
- Fisher/Super Vector
- Normal Codebook

Perronnin et al. Towards Good Practice in Large-Scale Learning for Image Classification. CVPR, 2012
When SVMs meet Large Scale

- SVM still works!

- Popular paradigm: One-vs-all SVMs + Batch optimization for each SVM

- But we are talking about large scale ~ terabytes feature set using standard coding and pooling
When SVMs meet Large Scale

• Issues with large training set
  A. Not possible to pre-load into memory
  B. Slow convergence
  C. Multiple classifier training

• How to solve these problems?
  – Stochastic Gradient Descent -> A + B
  – Parallel Training -> C
Large Scale Learning for Linear SVMs

• SVM objective:

\[ C(w) = \frac{1}{N} \sum_{i=1}^{N} Q(z_i; w) \]

• Binary linear SVM primal formulation:

\[ \frac{1}{N} \sum_{i=1}^{N} \ell(x_i, y_i; w) + \frac{\lambda}{2} \|w\|^2 \]

\[ \ell(x_i, y_i; w) = \max\{0, 1 - y_i w' x_i\} \]

\[ Q(z_i; w) = \ell(x_i, y_i; w) + \frac{\lambda}{2} \|w\|^2 \]

• Optimization: BGD

\[ w_{t+1} = w_t - \eta_t \nabla_{w=w_t} C(w) \]

\[ = w_t - \eta_t \frac{1}{N} \sum_{i=1}^{N} \nabla_{w=w_t} Q(z_i; w) \]

• Memory requirement O(N)

• Slow convergence: gradient for each iteration involves all samples

Courtesy: Florent Perronnin, Herve Jegou
SGD for Large Scale Learning in SVMs

• SGD: use one random sample to compute gradient for each iteration

$$w_{t+1} = w_t - \eta_t \nabla_{w=w_t} Q(z_t; w)$$

$$\nabla_{w=w_t} Q(z_t; w) = \begin{cases} 
\lambda w - y_i x_i, & \text{if } y_i w' x_i < 1 \\
\lambda w, & \text{if } y_i w' \geq 1 
\end{cases}$$

• Very simple to implement

$$w_{t+1} = (1 - \eta_t \lambda)w_t + \delta_t \eta_t y_t x_t$$

$$\delta_t = 1 \text{ if } \ell(x_t, y_t; w) > 0, 0 \text{ otherwise}$$

• Tricky to choose a reasonable learning rate

Courtesy: Florent Perronnin, Herve Jegou
SGD for Large Scale Learning in SVMs

- Merit of SGD
  - Memory $O(1)$, online learning
  - Very fast iteration
  - Improve performance even with redundant samples
SGD for Large Scale Learning in SVMs

- Merit of SGD
  - Memory $O(1)$, online learning
  - Very fast iteration
  - Improve performance even with redundant samples

- Comparison with traditional SVM learning (BGD)

<table>
<thead>
<tr>
<th>Model</th>
<th>Algorithm</th>
<th>Training Time</th>
<th>Objective</th>
<th>Test Error</th>
</tr>
</thead>
<tbody>
<tr>
<td>Hinge loss $\lambda = 10^{-4}$</td>
<td>SVMLight</td>
<td>23,642 secs</td>
<td>0.2275</td>
<td>6.02%</td>
</tr>
<tr>
<td></td>
<td>SVMPerf</td>
<td>66 secs</td>
<td>0.2278</td>
<td>6.03%</td>
</tr>
<tr>
<td></td>
<td>SGD</td>
<td>1.4 secs</td>
<td>0.2275</td>
<td>6.02%</td>
</tr>
</tbody>
</table>

SGD for Large Scale Learning in SVMs

• Merit of SGD
  – Memory $O(1)$, online learning
  – Very fast iteration
  – Improve performance even with redundant samples

• Comparison with traditional SVM learning (BGD)

<table>
<thead>
<tr>
<th>Model</th>
<th>Algorithm</th>
<th>Training Time</th>
<th>Objective</th>
<th>Test Error</th>
</tr>
</thead>
<tbody>
<tr>
<td>Hinge loss</td>
<td>SVMLight</td>
<td>23,642 secs</td>
<td>0.2275</td>
<td>6.02%</td>
</tr>
<tr>
<td>$\lambda = 10^{-4}$</td>
<td>SVMPerf</td>
<td>66 secs</td>
<td>0.2278</td>
<td>6.03%</td>
</tr>
<tr>
<td></td>
<td>SGD</td>
<td>1.4 secs</td>
<td>0.2275</td>
<td>6.02%</td>
</tr>
</tbody>
</table>


• Variants
  – Min-Batch: K samples each iteration
  – 2\textsuperscript{nd} Order SGD
  – ASGD: $\bar{w}_t = \frac{1}{T} \sum_t w_t$
Large Scale Learning for Non-Linear SVMs

- Non-linear SVMs usually give better performance than linear SVMs

- Training
  - Explicitly mapping to high-dim feature space + $O(n)$ training (Primal form)
  - Implicitly mapping to feature space + $O(n^2)$ or $O(n^3)$ training (Dual form, kernel trick)

- For large scale training samples, infeasible for non-linear training

- How: Non-linear classification with explicit embedding $K \rightarrow \varphi$
Large Scale Learning for Non-Linear SVMs

- Histogram Intersection kernel

\[ \sum_{i=1}^{n} \min (h_a(i), h_b(i)) \]

- Indicator function:

\[ 1_{(a,b)}(t) = \begin{cases} 
1 & \text{if } a \leq t \leq b \\
0 & \text{otherwise} 
\end{cases} \]

- Compute intersection value:

\[ \min(x, z) = \int_{t=0}^{\infty} 1_{(0,x)}(t)1_{(0,z)}(t) \, dt \]

- Approximate infinite dimension feature map with a finite one:

- Replace integral by sum

Maji et al. Classification using intersection kernel support vector machines is efficient. CVPR 2008

Courtesy: Florent Perronnin, Herve Jegou
Large Scale Learning for Non-Linear SVMs

- Histogram Intersection kernel

\[ \sum_{i=1}^{n} \min (h_a(i), h_b(i)) \]

- Idea: split value range [0,1] into discrete bins N, count common bin number

- Bin vector indicator:

\[ U_1(t) = \left[ \begin{array}{c} x^t \cdot (N-t) \\ 1 \cdots 1 \end{array} \right] \]

\[ R(t) = [t + 0.5] \]

- Mapping:

\[ \varphi_1(t) = \frac{U_1(R(Nt))}{\sqrt{N}} \]

\[ \approx \frac{\varphi_1(0.33) \cdot \varphi_1(0.54)}{\sqrt{10} \cdot \sqrt{10}} \]

\[ = \frac{1}{10} [11100 \cdots 0]' [111110 \cdots 0] \]

\[ = 0.3 \]

Maji et al. Classification using intersection kernel support vector machines is efficient. CVPR 2008

Courtesy: Florent Perronnin, Herve Jegou
Challenges

• Theoretical Challenge
  – Algorithm: approximate optimization. How to find global optimum?

• Practical Challenge
  – What if no ‘large scale labeling’?
  – Limited public dataset: ImageNet (strong labeling) & Tiny Image (weak labeling)
  – Crowdsourcing: user inconsistency

• Approach: leverage unlabeled data
Why Leverage Unlabeled Data?

- Unlabeled data is cheap! (generally)
Why Leverage Unlabeled Data?

• Unlabeled data is cheap! (generally)
• Labeled data can be hard to get
Why Leverage Unlabeled Data?

• Unlabeled data is cheap! (generally)
• Labeled data can be hard to get
  • Human annotation is boring
Why Leverage Unlabeled Data?

- Unlabeled data is cheap! (generally)
- Labeled data can be hard to get
  - Human annotation is boring
  - Labels require experts

Male or female?
Why Leverage Unlabeled Data?

- Unlabeled data is cheap! (generally)
- Labeled data can be hard to get
  - Human annotation is boring
  - Labels require experts

Male or female?
Why Leverage Unlabeled Data?

- Unlabeled data is cheap! (generally)
- Labeled data can be hard to get
  - Human annotation is boring
  - Labels require experts
  - Labels may require special devices

Fluorescently tagged LB3 protein in HeLa cells
Why Leverage Unlabeled Data?

- Unlabeled data is cheap! (generally)
- Labeled data can be hard to get
  - Human annotation is boring
  - Labels require experts
  - Labels may require special devices
- Learner may overfit to labeled data (esp. when limited)
Transfer Learning

Lack of labeled training data always happens

When we have some related source domains

Slide courtesy: Qiang Yang
Transfer Learning

Multiple Domain Data

Feature Spaces

Heterogeneous

Instance Alignment?

Yes

Multi-view Learning

No

Heterogeneous Transfer Learning

Data Distribution?

Different

Transfer Learning across Different Distributions

Same

Traditional Machine Learning

Source Domain

Target Domain

Slide courtesy: Qiang Yang
Semi-supervised Learning (SSL)

Generative Models

Graph-based Methods

Matrix Completion

Semi-supervised SVMs (S³VM)


[R. Cabral, F. Torre, J. Costeria, A. Bernardino. Matrix Completion for Multi-label Image Classification. NIPS 2011.]

Image courtesy: Xiaojin Zhu

[Columbia University In the City of New York]
Semi-supervised Learning (SSL)

Generative Models

Graph-based Methods

Matrix Completion

Semi-supervised SVMs (S³VM)

Image courtesy: Xiaojin Zhu


Deep Learning

• Goal and Focus:
  – We can leverage deep learning techniques to “learn” important features for classification problems for a variety of domains.

• Advantages
  – Portable to many different datasets, with little domain knowledge of the data.
  – State of the art technique for many large scale problems.

• Disadvantages
  – Time-consuming to train model
  – Architecture of implementation can be very specialized
Neural Networks

Neuron

\[ x = w_1 f(z_1) + w_2 f(z_2) + w_3 f(z_3) \]

- \( x \) is called the total input to the neuron, and \( f(x) \) is its output.

Neural Network

A neural network computes a differentiable function of its input. For example, ours computes:
\[ p(\text{label} \mid \text{an input image}) \]
Convolutional Neural Networks


Local connections are enforced in CNNs
ImageNet Challenge

• ImageNet Challenge
  – ~1000 images in 1000 Categories.

• Training Set
  – 1.2M labeled images

• Validation and Test Set
  – 50,000 Validation Images
  – 150,000 Testing Images
System Overview

Raw Pixel Input from Image

Convolutional Layers

Fully-Connected Layers

Output Layer
Model Specifics

- Overlapped Max-Pooling Used
- Size
  - ~650,000 neurons
  - ~60,000,000 parameters
  - ~630,000,000 connections
- Train the weights with Stochastic Gradient Descent
Non-Saturating Neurons

\[ f(x) = \tanh(x) \]

Very bad (slow to train)

\[ f(x) = \max(0, x) \]

Very good (quick to train)
Counter Acting Over-Fitting

• Data Augmentation

• “Dropout”
  – Randomly set the output of each hidden neuron in the network to 0 with 50% probability.
    • They multiplied the output of each neuron by 0.5 to simulate this process.
ImageNet Challenge Results

• Results:
  – Image Net Challenge 2010 Results
    • Error Rates: Top 1 = 37.5% and Top 5 = 17.0%
    • Best Previous: Top 1 = 45.7% and Top 5 = 25.7%
  
  – Image Net Challenge 2012 Results
    • Error Rates: Top 1 = 38.1% and Top 5 = 16.4%
Classification Results

mite  container ship  motor scooter  leopard
mite  container ship  motor scooter  leopard
black widow  lifeboat  go-kart  jaguar
cockroach  amphibian  moped  cheetah
tick  fireboat  bumper car  snow leopard
starfish  drilling platform  golfcart  Egyptian cat

grille  mushroom  cherry  Madagascar cat
grille  mushroom  cherry  Madagascar cat
convertible  agaric  dalmatian  squirrel monkey
grille  mushroom  cherry  dalmatian
pickup  agaric  grape  spider monkey
beach wagon  jelly fungus  elderberry  titi
fire engine  gill fungus  elderberry  indri
dead-man's-fingers  fungus  currant  howler monkey
How Many Computers to Identify a Cat? 16,000
Unsupervised Feature Learning

Locally Receptive Field Networks

RICA Features

Raw Image Pixels
Asynchronous Stochastic Gradient Descent

Parameter Server

Calculation Servers

RICA

RICA

RICA

RICA
ImageNet Classification

- **ImageNet Dataset**
  - 20,000 Categories
  - 16M images

- **Model Training**
  - 10M 200x200 unlabeled images
  - 1000 machines (16000 cores)
    - 1 week
  - 1.15B parameters
Results

• The Face Neuron

• The Human Body Neuron
Classification Results

• State of the Art Classification on full ImageNet Dataset
  – Previous Results:
    • 9.5% Classification Accuracy on 22,000 categories
      – (Weston, Bengio ‘11)
  – Their Results:
    • 15.8% Classification Accuracy on 22,000 categories
      – Recently saw a presentation where their new claim is 22.3% accuracy (not sure what changed).
“Large-scale” tomorrow

[Graph showing the relationship between images and classes, with markers for TINY (2008), IMAGENET (2009), and current release.]
“Large-scale” tomorrow
“Large-scale” tomorrow

are there even this many semantic categories?
“Large-scale” tomorrow

will numerable categories saturate and research only extend along the x-axis?
"Large-scale" tomorrow

will numerable categories saturate and research only extend along the x-axis?