Co-Training &
Its applications in Vision

Santosh K. Divvala
Overview

• What is Co-Training?
  – Blum & Mitchell in COLT’98
  – Focus on general idea rather than the theory

• How has it been used in object detection?
  – Improving object detectors (ICCV’03, CVPR’05), Co-Tracking (ICCV’07)

• How could it be used for geometric context? (if time permits)
The Example: Faculty Web-page Classification

Faculty Pages

Vs

Other Web-Pages in cs domain
The “High Initial Expense” Problem

• Classifiers need labeled data

• But labeled data is expensive
  – Human annotation is boring
  – Might require expert knowledge
  – Graduate students have better things to do!

• Trick: Use unlabeled data for training

• How?: Co-training!
What is Co-Training (in theory)?

- Originally introduced by Blum & Mitchell [COLT’98]
- Problem
  - Given feature X, label Y, labeled data L, unlabeled data U
  - Wish to learn f: X -> Y, given L and U drawn from P(X,Y)
- Assumptions
  - A weekly useful classifier
  - Two (conditionally) independent views of the data i.e.,
    features describing X can be partitioned (X = X1 x X2),
    such that f can be computed from either X1 or X2
- Provide a PAC-style framework and analysis
- Theorem
  - If X1 and X2 are conditionally independent given Y
    and f is PAC learnable from noisy labeled data,
  - Then f is PAC learnable from weak initial classifier
    plus unlabeled data
What is Co-Training (in Practice)?

- Train a pair of classifiers using small set of labeled examples.

- Unlabelled examples which are confidently labeled by one of the classifiers are added, with labels, to the training set of the other classifier.
  - Margin magnitude give confidence.

- But not all confidently labeled examples are useful!
What is Co-Training (in practice)?

- Retrain the classifier with only *informative* examples
  - Sample only those examples that are close to the hyper-plane of the combined classifier
- Empirically shown to work even if assumptions do not hold strictly!
The Example: Faculty Web-page Classification

View 1: Page Text

View 2: Hyperlink Text
Experiment setting

- 1051 web pages from 4 CS depts.
  - 263 pages (25%) as test data
  - The remaining 75% of pages
    - Labeled data: 3 positive and 9 negative examples
    - Unlabeled data: the rest (776 pages)
- Manually labeled into a number of categories: e.g., “course home page”.
- Two views:
  - View #1 (page-based): words in the page
  - View #2 (hyperlink-based): words in the hyperlink
- Learner: Naïve Bayes

<table>
<thead>
<tr>
<th></th>
<th>Page-based classifier</th>
<th>Hyperlink-based classifier</th>
<th>Combined classifier</th>
</tr>
</thead>
<tbody>
<tr>
<td>Supervised training</td>
<td>12.9</td>
<td>12.4</td>
<td>11.1</td>
</tr>
<tr>
<td>Co-training</td>
<td>6.2</td>
<td>11.6</td>
<td>5.0</td>
</tr>
</tbody>
</table>

Table 2: Error rate in percent for classifying web pages as course home pages. The top row shows errors when training on only the labeled examples. Bottom row shows errors when co-training, using both labeled and unlabeled examples.
Understanding Co-Training

• Co-Training is a *training process* not a classification process

• Lets classifiers teach each other by working on unlabelled data

• Main Intuition
  – If the margins assigned by the classifiers are independent (not related),
  – then (it has been theoretically proven) that there will exist set of examples with high margin on one of the classifier and small or negative margin based on other
Recent Popularity of Co-Training in Data-driven based image parsing approaches

- “Co-Tracking using semi-supervised SVMs” – ICCV 2007
- “Online Detection and Classification of Moving Objects Using Progressively Improving Detectors” – CVPR '05
- “Unsupervised Improvement of Visual Detectors using Co-Training” – ICCV '03
Co-training Visual Detectors – ICCV’03

• First time Co-Training used in computer vision!

• Task: Identify Cars

• Improving the [Viola, Jones IJCV’02] detector

• Solution:
  – Pick any detector
  – Gather training data
  – Train detector
  – Test a new image using the detector

• Requires lot of labeled data $O(10^9)$ to achieve a low false positive rate!
Details

- Two views:
  - Grey-level (original image)
  - background-subtracted images (image – average background)
- Data: 50 labeled examples, 22000 unlabeled images
- Classifier: Adaboost (Use a cascade for speed-up, asymmetry)
- Features: Selected via LogAdaBoost (Simple linear function of rectangular sums and a threshold)
- Samples examples based on their weights
- Co-Training
  - Pseudo-online updates
  - 12 rounds of resampling
  - Each round add three new features
Revisiting Adaboost

Given: \((x_1, y_1), \ldots, (x_m, y_m)\) where \(x_i \in X, y_i \in Y = \{-1, +1\}\)
Initialize \(D_1(i) = 1/m\).
For \(t = 1, \ldots, T\):

- Train base learner using distribution \(D_t\).
- Get base classifier \(h_t : X \rightarrow \mathbb{R}\).
- Choose \(c_t \in \mathbb{R}\).
- Update:

\[
D_{t+1}(i) = \frac{D_t(i) \exp(-c_t y_i h_t(x_i))}{Z_t}
\]

where \(Z_t\) is a normalization factor (chosen so that \(D_{t+1}\) will be a distribution).

Output the final classifier:

\[
H(x) = \text{sign} \left( \sum_{t=1}^{T} c_t h_t(x) \right).
\]

Figure 1: The boosting algorithm AdaBoost.
Margin & Confidence

\[ \text{margin}_f(x,y) = \frac{y \sum_t \alpha_t h_t(x)}{\sum_t |\alpha_t|} \in [-1, +1] \]

- Positive iff \( H \) correctly classifies
- Margin magnitude is a measure of the confidence in classification
  - Close to -1 or +1 => high confidence
  - Close to 0 => low confidence
- Theoretical bound

\[
P \left[ f(\tilde{h}_i) \leq 0 \right] \leq \frac{1}{m} \sum_{i=1}^{m} 1 \left[ f(\tilde{h}_i) \leq \theta \right] + O \left( \frac{1}{\sqrt{m}} \left( \frac{d \log^2 (m/d)}{\theta^2} + \log(1/\delta) \right)^{1/2} \right)
\]
Interesting Visualization

$e_p$ is max score achieved by negative examples (likely to be positive)
$e_n$ is min score achieved by positive examples (likely to be negative)
Results

• On 90 labeled images
• Impressive performance

Figure 5: Detection results. (a)- The gray level classifier before co-training. (b)- The gray level classifier after co-training. (c)- The background subtracted classifier before co-training. (d)- The background subtracted classifier after co-training.

Figure 4: ROC curves. Green/Grey line: the original classifier. Black dashed: the co-trained classifier. TOP the GREY classifier. BOTTOM: the BackSub classifier.
Co-training Visual Detectors– CVPR’05

- An online version similar to ICCV’03 paper
- Improving pedestrian and car detectors online (3-class)
- Views: Top ‘m’ eigen vectors of the respective subspaces
- Classifiers: Adaboost using naïve bayes classifiers
- Samples examples for co-training based on the confidence of the final classifier
  - Thresholds: $T_{base}^{j,ci}, T_{ada}^{ci}$
Online Boosting

- Perform online updates to the classifiers

1. Online Updating the weak-classifiers
   - Depends on the weak learner

2. Online Updating the voting-weights [Oza, Russel - AISTATS’01]

3. Online Computation of weight distribution [Grabner, Bischof – CVPR’06]
Results

**Figure 3.** Some classification results from sequence 1.

**Figure 4.** Change in performance with increase in time for sequence 1, 2 and 3 respectively. The performance was measured over two minute intervals. Approximately 150 to 200 possible detections of vehicles or pedestrians were made in each time interval.
Co-Tracking – ICCV’07

- Tracking using co-training
- Two views: HOG and color histograms
- Classifiers: online SVMs
  - Uses most ideas from SVT of [Avidan – PAMI’04]
  - Combines in a Adaboost fashion
- Sampling
  - All positive (if confidence above a threshold)
  - K negative (peaks that do not overlap with object)

Figure 2. The results of our algorithm on a video of a pedestrian walking across a courtyard. As can be seen, the combined confidence map obtained through combining HoG and color information through our co-training framework is more reliable than the confidence map from either feature individually. a-d show the results from frame 183 of the sequence, and e-h show the results from frame 237 of the sequence.
Results

- Drastic improvement in results

Figure 3. A comparison of our confidence maps with those of the Ensemble Tracker. Original frames are given in the top row. The middle row contains the confidence maps generated by the Ensemble Tracking algorithm. The bottom row shows the confidence maps generated by our algorithm.
Excited about Co-Training?!

• Variations of Co-Training
  – “Co-EM” in ML’00: Behavior of Co-Training + EM
  – “Enhancing Supervised Learning with Unlabeled Data” in ICML’00: uses two learners of different types but both take the whole feature set
  – “Tri-Training” in TKDE’05: use three learners. If two agree, the data is used to teach the third learner.

• Readings
  – “Co-Training and Expansion” in NIPS’04
  – “Bayesian Co-Training” in NIPS’07
  – “Semi-Supervised Learning” – Tutorial at ICML’07