Semi-Supervised Learning for Audio, Speech and Language Processing

Mark Hasegawa-Johnson, Jui-Ting Huang, Xiaodan Zhuang, Xi Zhou, and Chi Hu

University of Illinois

February 25, 2011
Outline

1. Clusters vs. Classes
2. ML and MMI
   - Notation
   - A PAC Learning Approach
   - MMI+NCE
3. Modeling Prior Knowledge Using Finite State Transducers
4. Modeling Prior Knowledge Using MAP Adaptation
5. Discovering Unknown Classes: Anomaly Detection
6. Conclusions
Semi-Supervised Learning: Best-Case Scenario

Labels are Expensive
Labeling is expensive. Most classifiers learn from labeled data, therefore most classifiers learn from very small databases.

Data are Cheap
Unlabeled data are cheap. In an ideal world, clusters = classes. We learn the clusters from unlabeled data, then label each cluster.
Semi-Supervised Learning: Worst-Case Scenario

There is no guarantee that the label distribution will be predicted by any structure in the unlabeled data. If there is no correspondence between structure and labeling, then semi-supervised learning is useless. In that case, a training algorithm with $N_{\ell}$ labeled tokens and $N_u$ unlabeled tokens is no better (and, on average, no worse) than a training algorithm with only the $N_{\ell}$ labeled tokens.
Overlapping clusters can be learned.

How: regularized learning.
Regularized Learning: Maximum Likelihood

- Given a set of labeled data $\mathcal{D}_L$ and a set of unlabeled data $\mathcal{D}_U$:
  - Find an accurate description of the data...
  - Find the parameter set $\theta$ that maximizes

$$ J(\theta) = F_{ML}(\mathcal{D}_L)(\theta) + \alpha F_{ML}(\mathcal{D}_U)(\theta) $$

$$ = \frac{1}{l} \sum_{i=1}^{l} \ln p_{\theta}(x_i|y_i) + \alpha \frac{1}{u} \sum_{i=l+1}^{l+u} \ln p_{\theta}(x_i) $$
Regularized Learning: Discriminative

- Given a set of labeled data $\mathcal{D}_L$ and a set of unlabeled data $\mathcal{D}_U$:
- Make the separation between the classes as large as possible.
- Find the parameter set $\theta$ that maximizes

$$J(\theta) = F_{\text{MMI}}(\theta) + F_{\text{NCE}}(\theta)$$

$$= \frac{1}{l} \sum_{i=1}^{l} \ln p_\theta(y_i|x_i) + \alpha \frac{1}{u} \sum_{i=l+1}^{l+u} \sum_y p_\theta(y|x_i) \ln p_\theta(y|x_i)$$
Discriminative Training Criteria

Supervised: Maximum Mutual Information  Minimum probability of error = maximum probability of the correct class = maximum mutual information (MMI) between observations and labels

\[
F^{(DL)}_{MMI}(\theta) = \frac{1}{l} \sum_{i=1}^{l} \ln p_\theta(y_i|x_i)
\]

Unsupervised: Negative Conditional Entropy  Encourage the model to have the greatest possible certainty about its labeling decisions

\[
F^{(DU)}_{NCE}(\theta) = -\mathcal{H}^{(DU)}_{emp}(Y; X, \theta) = \frac{1}{l+u} \sum_{i=l+1}^{l+u} \sum_y p_\theta(y|x_i) \ln p_\theta(y|x_i)
\]
Hoeffding’s Inequality

\[ z_1, \ldots, z_n \text{ i.i.d., } P(z_i \in [0, R]) = 1, \text{ then} \]

\[ P \left( |E[z] - \langle z \rangle| \geq \epsilon \right) \leq 2e^{-\frac{2\epsilon^2 n}{R^2}} \]

for \( \langle z \rangle \equiv \frac{1}{n} \sum z_i \) and \( E[z] \equiv \int z p(z) dz \)

Probably Approximately Correct (PAC) Learning

- **Hypothesis Space:** \( h : \mathcal{X} \rightarrow \mathcal{Y} \) has cardinality \( N(\mathcal{H}) \)
- **Loss Function:** \( f(h(x_i), y_i) \in [0, R] \) w/probability one
- **Confidence:**
  \[ \delta \equiv P \left( \max_{h \in \mathcal{H}} |E[f(h(x), y)] - \langle f(h(x), y) \rangle| \geq \epsilon \right) \]
- **The Basic PAC Bound:**
  \[ \epsilon \leq R \sqrt{\frac{\ln 2N(\mathcal{H}) - \ln \delta}{2n}} \]
Continuous Hypothesis Spaces: Covering Number

\( N(\mathcal{H}) = \text{size of the } \epsilon\text{-covering set for empirical and stochastic averages of } f(\mathcal{H}), \text{ i.e., the smallest possible discrete set } \{h_1, \ldots, h_{N(\mathcal{H})}\} \text{ such that} \)

\[
\max_h \left( \min_{1 \leq j \leq N(\mathcal{H})} |E[f(h_j(x), y)] - E[f(h(x), y)]| \right) \leq \epsilon
\]

\[
\max_h \left( \min_{1 \leq j \leq N(\mathcal{H})} |\langle f(h_j(x), y) \rangle - \langle f(h(x), y) \rangle| \right) \leq \epsilon
\]

Continuous Hypothesis Spaces: Revised PAC Bound

\[
\delta \equiv P \left( \max_{h \in \mathcal{H}} |E[f(h(x), y)] - \langle f(h(x), y) \rangle| \geq 3\epsilon \right)
\]

\[
\epsilon \leq R \sqrt{\frac{\ln 2N(\mathcal{H}) - \ln \delta}{2n}}
\]
Kernel Estimators of Conditional Risk

Define $f_{X}(h(\xi), y)$ to be the kernel projection of $h(\xi)$ onto $x$,

$$f_{X}(h(\xi), y) \equiv f(h(\xi), y)K(x, \xi)$$

for some symmetric positive-definite kernel, $K(x, \xi) \in [0, 1]$.

Conditional Covering Number

Define $N(H|x)$ to be size of a set $h_{j}$ which is big enough to explain all of the losses incurred only by the data points that are “near” $x$, where the word “near” is defined by the kernel. Specifically,

$$\max_{h} \left( \min_{1 \leq j \leq N(H|x)} \left| E_{\xi,y}[f_{X}(h_{j}(\xi), y)] - E_{\xi,y}[f_{X}(h(\xi), y)] \right| \right) \leq \epsilon$$

$$\max_{h} \left( \min_{1 \leq j \leq N(H|x)} \left| \langle f_{X}(h_{j}(\xi), y) \rangle - \langle f_{X}(h(\xi), y) \rangle \right| \right) \leq \epsilon$$

Usually, $N(H|x) \ll N(H)$. 
### Covering Number: Example

- $\mathcal{X} = [0, 1]^2$, $\mathcal{Y} = [0, 1]$
- $f(h(x), y) = h(x) - y$

\[
N(\mathcal{H}) \sim \left(\frac{1}{\epsilon}\right)^3
\]

### Conditional Covering Number

Let's use a $2\epsilon$-width rectangular kernel:

\[
f_x(h(\xi), y) = \begin{cases} 
  h(\xi) - y & |x - \xi| < \epsilon \\
  0 & \text{else}
\end{cases}
\]

So

\[
N(\mathcal{H}|x) \sim \left(\frac{1}{\epsilon}\right)
\]
**Confidence of the Conditional Risk Estimate**

\[
\delta(x) \equiv P \left( \max_{h \in \mathcal{H}(x)} |E[f_x(h(\xi), y)] - \langle f_x(h(\xi), y) \rangle| \geq 3\epsilon \right)
\]

**A Semi-Supervised PAC Bound**

Suppose (1) \(p(x)\) is known, e.g., because we have lots and lots of unlabeled data, (2) we don’t really care about \(\delta(x)\), but only about

\[
\ln \delta \equiv E_x[\ln \delta(x)]
\]

If we’re willing to redefine “confidence” in this way, then it is possible to bound \(\epsilon\) much more tightly in the semi-supervised case than in the supervised case, for two reasons.

- **Range:** \(\langle f_x(h, y) \rangle \equiv \langle f(h, y)K(x, \xi) \rangle\) tends to be much smaller than \(\langle f(h, y) \rangle\). We compensate by rescaling \(R\).
- **VC Dimension:** \(\ln N(\mathcal{H}|x)\) is less than \(N(\mathcal{H})\). The reduced VC dimension creates a better bound.
PAC Bound for Semi-Supervised Learning

\[ \epsilon \leq \bar{R} \sqrt{\frac{E_x[\ln 2N(\mathcal{H}|x)] - \ln \delta}{2n}} \]

- **Range:** \( f_x(h, y) \) has a much smaller range than \( f(h, y) \). The root-harmonic-mean-squared radius, \( \bar{R} \ll R \), compensates for the difference in range.

\[ \bar{R} = R \left( E_x \left[ \left( \frac{1}{n} \sum_i K^2(x, x_i) \right)^{-1} \right] \right)^{-1/2} \]

- **VC Dimension:** In addition to the much smaller range, \( f_x(h, y) \) also typically has a much smaller covering number than \( f(h, y) \). The VC dimension, \( E_x[\ln N(\mathcal{H}|x)] \), may therefore be much smaller than the VC dimension, \( \ln N(\mathcal{H}) \), that can be achieved without the unlabeled data.
Maximum Mutual Information (MMI)

MMI is defined by the hypothesis and loss function

\[
\vec{h}(x) = \begin{bmatrix}
\ln \hat{p}(Y = 1|x) \\
\vdots \\
\ln \hat{p}(Y = c|x)
\end{bmatrix}, \quad f(\vec{h}, y) = \vec{h}^T \delta_y = -\ln \hat{p}(Y = y|x)
\]

MMI training chooses \( \vec{h} \in \mathcal{H} \) to minimize

\[
\langle f(\vec{h}, y) \rangle \equiv -\frac{1}{n} \sum_{i=1}^{n} \ln \hat{p}(Y = y_i|x_i)
\]

PAC bound on the resulting risk is

\[
E[f(\vec{h}, y)] \leq \langle f(\vec{h}, y) \rangle + R \sqrt{\frac{\ln 2N(\mathcal{H}) - \ln \delta}{2n}}
\]
Covering Number for the MMI Loss

\[ f(\vec{h}, y) = -\ln \hat{p}(Y = y|x) \] has infinite covering number. Finite covering number is possible for an exponentiated average:

\[
\max_h \left( \min_{1 \leq j \leq N(\mathcal{H}|x)} \left| e^{\langle f_x(h_j(\xi), y) \rangle} - e^{\langle f_x(h(\xi), y) \rangle} \right| \right) \leq \epsilon
\]

For example, suppose we choose some arbitrary entropy threshold \( E_{\text{max}} \), and limit the hypothesis space to:

\[
\mathcal{H}(x) = \left\{ h : - \sum_{y \in \mathcal{Y}} \hat{p}(y|x) \ln \hat{p}(y|x) \leq E_{\text{max}} \right\}
\]

then the covering number is

\[
N(\mathcal{H}|x) \sim e^{E_{\text{max}}}
\]
Semi-Supervised MMI

Estimate the VC dimension using unlabeled data, $\mathcal{D}_U = \{x_{n+1}, \ldots, x_{n+u}\}$:

$$E_x[\ln N(\mathcal{H}|x)] \approx -\frac{1}{u} \sum_{i=n+1}^{n+u} \sum_{y \in \mathcal{Y}} \hat{p}(x_i, y_i) \ln \hat{p}(y_i|x_i)$$

Choose $h(x)$ as

$$h = \arg \min -\frac{1}{n} \sum_{i=1}^{n} \ln \hat{p}(y_i|x_i), \quad \text{s.t.} \quad E_x[\ln N(\mathcal{H}|x)] \leq E_{max}$$

whose corresponding Lagrangian is

$$\mathcal{F}(\hat{h}) = -\frac{1}{n} \sum_{i=1}^{n} \ln \hat{p}(y_i|x_i) - \frac{\alpha}{u} \sum_{i=n+1}^{n+u} \sum_{y \in \mathcal{Y}} \hat{p}(x_i, y) \ln \hat{p}(y|x_i)$$
Supervised: Maximum Mutual Information  Minimum probability of error = maximum probability of the correct class = maximum mutual information (MMI) between observations and labels

\[ \mathcal{F}_{MMI}^{(D_L)}(\hat{h}) = \frac{1}{n} \sum_{i=1}^{n} \ln \hat{p}(y_i|x_i) \]

Unsupervised: Negative Conditional Entropy  Encourage the model to have the greatest possible certainty about its labeling decisions

\[ \mathcal{F}_{NCE}^{(D_U)}(\hat{h}) = \frac{1}{u} \sum_{i=n+1}^{l+u} \sum_{y} \hat{p}(x_i, y) \ln \hat{p}(y|x_i) \]
Experiments: Phone Classification

- On TIMIT corpus
  - Training: 462 speakers, 3696 utterances, 140225 segments
  - Development: 50 speakers, 400 utterances, 15057 segments
  - Test: 118 speakers, 944 utterances, segments, 35697 segments
- 48 phone classes
- To create a semi-supervised setting: Labels of s% of the training set are kept ((100-s)% are unlabeled)
- Segmental features [Halberstadt ’98]: a fixed length vector is calculated from the frame-based spectral features (12PLP coefficients plus energy)
  - Divide the frames for each phone segment into three regions with 3-4-3 proportion
  - Plus the 30 ms regions beyond the start and end time of the segment
  - Log duration
- Each phone is modeled by a GMM with two full-covariance Gaussian components
Optimization: Conjugate Gradient Descent

Classifier = Gaussian Mixture Model (GMM)

\[ p(x|y = c) = \sum_{m=1}^{M} w_{ck} \mathcal{N}(x; \mu_{ck}, \Sigma_{ck}) \]

Gradient of the MMI/NCE Criterion

\[ \nabla_{\mu_{cm}} \mathcal{F}_{\text{MMI}} = \frac{1}{l} \sum_{i=1}^{l} \xi_i(c, k) \left( \frac{\delta[y_i, c]}{p_\theta(c|x_i)} - 1 \right) \]

\[ \nabla_{\mu_{ck}} \mathcal{F}_{\text{NCE}} = \frac{1}{u} \sum_{i=l+1}^{l+u} \xi_i(c, k) \left( \sum_y \log p_\theta(y|x_i)(\delta[y, c] - p_\theta(y|x_i)) \right) \xi_i(c, k) = \sum_{ck}^{-1} (x_i - \mu_{ck}) p_\theta(k, c|x_i) \]
MMI/NCE Learns a Better Classifier...
...but MMI/NCE Doesn’t Learn the Data Distribution
Decision regions
(NOT data distribution)

MMIE with 10% of labels

MMIE with 100% of labels

MMIE with 10% of labels + 90% as unlabeled data
Conclusions: Phone Classification

- The influence of unlabeled data on the estimated model is not equivalent to that of labeled data.
- Unlabeled data are useful for finding more accurate distributions or a better classifier, depending on the training criterion.
  - Semi-supervised generative training (ML) finds likelihood functions that approximate true class distributions.
  - Semi-supervised discriminative training (MMI/NCE) focuses on class separability, and may lead to a better classifier.
Proximity of Latent Representations

What does it mean for similar tokens to have similar labels?

- GMM phone classifier encourages similar labels for tokens $x_i$, $x_j$ such that $\|x_i - x_j\|$ is small
- Sometimes $\|x_i - x_j\|$ is not a good measure of the \textit{a priori} probability that $y_i \neq y_j$
- Example: $y_i=$word label, $x_i=$pronunciation
  
  - warmth: $[\text{wɔrmθ}] \rightarrow [\text{wɔrmpθ}]$
  - I don’t know: $[\text{aɪdɒntno}] \rightarrow [\text{ɑrʊð}]$
  - several: $[\text{sɛvɜr}] \rightarrow [\text{sɛrv}]$
  - instruments: $[\text{ɪnstrʌmɛnts}] \rightarrow [\text{ɪstʃɪnts}]$
  - everybody: $[\text{ɛvriˈbʌdi}] \rightarrow [\text{ɛruai}]$
Pronunciation Variability = Temporal Overlap of Articulatory Gestures

“seven,” dictionary form: /s ɛ vən/

Lips: Fricative /v/
Tongue: Fricative /s/ Wide /ɛ/ Neutral /ə/ Closed /n/

“seven,” casual speech: /s ɛ vn/

Lips: Fricative /v/
Tongue: Fricative /s/ Wide /ɛ/ Neutral /ə/ Closed /n/
A Mapping Between Gestures and Phones

- Each phone corresponds to a canonical “gestural pattern vector” (GPV)
- There are more GPVs than phones; most GPVs correspond to non-English phones, allophones, or pseudo-phones
Proximity of Gestural Scores: “The”

A_1: uvulo-pharyngeal fast tongue body location
A_2: wide fast tongue body degree
A_3: slow release tongue tip location
A_4: wide slow tongue tip degree
Isolated word recognition: \( \hat{w} = \arg \max \ p(O|Q)p(Q|w) \)

- \( O = [\vec{o}_1, \ldots, \vec{o}_T] = \text{Articulograph observations} \)
- \( Q = [q_1, \ldots, q_T] = \text{GPV sequence} \)

Observation PDF \( p(O|Q) = \text{ANN-GMM-HMM}, \text{trained on 277 words, tested on 139 words} \)

Pronunciation model \( p(Q|w) \)
- Initialized using dictionary
- Expanded to include up to \( N_Q \) alternate pronunciations with similar gestural scores, \( N_Q \) fixed in advance
- No learning yet!! Similar gestural scores are assumed, \textit{a priori}, to be members of the same class (same word)
- (Future work: learning goes here?)
Word Recognition Accuracy

<table>
<thead>
<tr>
<th>Recognizer</th>
<th>Word Recognition Accuracy</th>
</tr>
</thead>
<tbody>
<tr>
<td>GPV Bigram</td>
<td>85%</td>
</tr>
<tr>
<td>(models local GPV sequence) (statistics, not global)</td>
<td></td>
</tr>
<tr>
<td>GPV-FST, $N_Q = 1$ pronunciation/word</td>
<td>88%</td>
</tr>
<tr>
<td>GPV-FST, $N_Q = 50$ pronunciations/word</td>
<td>90%</td>
</tr>
<tr>
<td>GPV-FST, $N_Q = 200$ pronunciations/word</td>
<td>90.7%</td>
</tr>
</tbody>
</table>
Conclusions: Pronunciation Modeling

Gestures encode pronunciation variability: GPV sequences with similar corresponding gestural scores are a priori likely to be instantiations of the same word.

A method that works: enumerate similar gestural scores, combine them into a finite state transducer, and use it to recognize speech.

A method that doesn’t work: enumerate similar phone symbol sequences, combine them into a finite state transducer, and use it to recognize speech.

Conclusion stated as a tautology: If...

- If you can find a representation \( Q \) in which intra-class variability is smaller than inter-class variability, then
- tokens with similar values of \( Q \) tend to have the same class label.
Surprising idea: Instead of saying that the class PDF generates instances,

- Say that the class PDF generates instance PDFs, and each instance PDF generates exactly one instance.

Why it’s useful: Instance PDF is drawn from an arbitrarily high-dimensional space (the space of all possible PDFs).

- It is always possible to find a transformation of that space in which intra-class variability is smaller than inter-class variability.

Obvious limitations:

- How do you estimate a PDF from one instance?
- In which transformation of the “space of all possible PDFs” is intra-class variability smaller than inter-class variability?
Estimating the Instance PDF: MAP Adaptation

**Mixture Gaussian Model**

\( \vec{x} \) is the signal log spectrum; \( c \) is the acoustic event label. The PDF \( p(\vec{x}|c) \) is modeled as a stochastic mixture of Gaussian kernels with means \( \vec{\mu}_k \) and covariances \( \Sigma_k \).

\[
p(\vec{x}|c) = \sum_{m} w_{ck} \mathcal{N}(\vec{x}; \vec{\mu}_k, \Sigma_k)
\]

**MAP Adaptation to the p’th instance**

\( \gamma_k(t) \) is the posterior probability that observation \( \vec{x}_t \), one of the observations from the \( p^{th} \) instance, belongs to Gaussian kernel \( k \).

\[
\gamma_k(t) = \frac{w_{ck} \mathcal{N}(\vec{x}_t; \vec{\mu}_k, \Sigma_k)}{\sum_j w_{cj} \mathcal{N}(\vec{x}_t; \vec{\mu}_j, \Sigma_j)}
\]

The adapted mean vectors, \( \vec{\mu}^{(p)}_k \), describe the \( p^{th} \) instance PDF. Their resemblance to the type PDF is controlled by the inertia parameter \( \nu \).

\[
\vec{\mu}^{(p)}_k = \frac{\sum_{t \in p} \gamma_k(t) \vec{x}_t + \nu \vec{\mu}_k}{\sum_{t \in p} \gamma_k(t) + \nu}
\]
Parameterizing and Normalizing the Instance PDF

Parameterize the $p^{th}$ instance

1. Instance PDF is parameterized by a supervector, $\vec{s}_p$.

$$\vec{s}_p = \begin{bmatrix} \Sigma_1^{-1/2}(\vec{\mu}_1^{(p)} - \vec{\mu}_1) \\ \vdots \\ \Sigma_K^{-1/2}(\vec{\mu}_K^{(p)} - \vec{\mu}_K) \end{bmatrix}$$

2. Inter-instance variability is parameterized by a within-class covariance matrix, $R = \text{COV}(\vec{s}_p)$.

Normalize the $p^{th}$ instance

3. Supervectors are then normalized using within-class covariance normalization (WCCN): $\tilde{s}_p = R^{-1}\vec{s}_p$.

4. In the WCCN supervector space $\tilde{s}_p$, intra-class variability is less than inter-class variability, therefore any classifier can work well (e.g., nearest-centroid).
Experimental Test: Non-Speech Acoustic Event Detection

**Difficulties**

- Negative SNR (speech is “background noise”)
- Unknown spectral structure
- Different spectral structure for each event type

**Key Jingle**

**Footsteps**

**Speech**
WCCN Supervectors Rescore Tandem MAP Decoding

MAP Decoding Using Tandem NN-GMM-HMM

Rescored Using WCCN Supervectors

MAP Decoding
For segmentation & classification

Hypothesized Boundaries and event labels
(one best or lattice)

Confidence rescoring / event classification
using new method

Refining output result
According to AED metric

Improved Detection Result
# Acoustic Event Detection Results

## Without Supervectors

<table>
<thead>
<tr>
<th>Inst.</th>
<th>AED Accuracy</th>
</tr>
</thead>
<tbody>
<tr>
<td>AIT</td>
<td>4.4</td>
</tr>
<tr>
<td>ITC</td>
<td>23.4</td>
</tr>
<tr>
<td>TUT</td>
<td>14.7</td>
</tr>
<tr>
<td>UIUC</td>
<td>36</td>
</tr>
<tr>
<td>STI2R</td>
<td>22.9</td>
</tr>
<tr>
<td>UPC</td>
<td>23</td>
</tr>
</tbody>
</table>

## With WCCN Supervectors

<table>
<thead>
<tr>
<th>Method</th>
<th>ap</th>
<th>cl</th>
<th>cm</th>
<th>co</th>
<th>ds</th>
<th>kj</th>
<th>kn</th>
<th>kt</th>
<th>la</th>
<th>pr</th>
<th>pw</th>
<th>st</th>
<th>Average</th>
</tr>
</thead>
<tbody>
<tr>
<td>MFCC</td>
<td>78.3</td>
<td>26.9</td>
<td>29.5</td>
<td>24.2</td>
<td>56.3</td>
<td>39.9</td>
<td>7.7</td>
<td>0.0</td>
<td>39.0</td>
<td>35.2</td>
<td>14.1</td>
<td>28.7</td>
<td>28.2</td>
</tr>
<tr>
<td>FB</td>
<td>34.5</td>
<td>21.8</td>
<td>25.4</td>
<td>24.9</td>
<td>38.9</td>
<td>27.2</td>
<td>11.7</td>
<td>0.0</td>
<td>49.1</td>
<td>13.8</td>
<td>11.7</td>
<td>28.1</td>
<td>27.8</td>
</tr>
<tr>
<td>Adaboost</td>
<td>44.4</td>
<td>25.5</td>
<td>31.3</td>
<td>31.2</td>
<td>57.3</td>
<td>33.2</td>
<td>13.5</td>
<td>1.9</td>
<td>51.3</td>
<td>36.7</td>
<td>17.6</td>
<td>36.8</td>
<td>34.0</td>
</tr>
<tr>
<td>Adaboost+T</td>
<td>52.6</td>
<td>21.9</td>
<td>37.2</td>
<td>51.3</td>
<td>63.0</td>
<td>29.6</td>
<td>11.5</td>
<td>0.0</td>
<td>54.2</td>
<td>42.7</td>
<td>25.8</td>
<td>34.6</td>
<td>35.3</td>
</tr>
<tr>
<td>Adaboost+S</td>
<td>44.4</td>
<td>25.0</td>
<td>33.7</td>
<td>31.2</td>
<td>56.6</td>
<td>33.2</td>
<td>20.9</td>
<td>35.5</td>
<td>51.3</td>
<td>36.7</td>
<td>19.2</td>
<td>41.3</td>
<td>37.5</td>
</tr>
<tr>
<td>Adaboost+T+S</td>
<td>52.6</td>
<td>21.9</td>
<td>39.6</td>
<td>49.8</td>
<td>63.0</td>
<td>29.6</td>
<td>15.4</td>
<td>36.8</td>
<td>54.7</td>
<td>41.7</td>
<td>26.0</td>
<td>38.3</td>
<td>38.6</td>
</tr>
</tbody>
</table>

MFCC = mel-frequency cepstral coefficients, FB = filterbank, T = Tandem, S = Supervector
Conclusions: Acoustic Event Detection

- The class PDF generates instance PDFs; the instance PDFs generate instances.
  - Instance PDF can be estimated using MAP (regularized) learning.
- The space of all possible PDFs is a very large space indeed; lots of interesting normalization methods are possible.
  - (Simple) Within-class covariance normalization is very effective.
  - After WCCN, (simple) minimum-centroid classification seems to work better (often) than any other classifier.
Class Discovery

Problem Statement
- The labeled dataset $\mathcal{D}_L$ contains examples drawn from only one of the two classes, say, $y_i = 1$ for $1 \leq i \leq l$
- The unlabeled dataset $\mathcal{D}_U$ contains examples drawn from two classes, say, $y_i \in \{0, 1\}$ for $l + 1 \leq i \leq l + u$
- Both of the two classes have piece-wise-compact distributions, e.g., GMMs

Likelihood Model

$$P(x, y, L) = \sum_{k=1}^{K} \mathcal{N}(x|\mu_k, \Sigma_k) P(y|k) P(L|y)$$

$L \in \{\text{labeled, unlabeled}\}$
Semi-Supervised Learning for Class Discovery

Parameter Set

$$\theta = \{\mu_k, \Sigma_k, P(y|k), P(L|y)\}$$

For class $y = 0$, we set $P(L = \text{labeled}|y = 0) = 0$.

EM Algorithm

**E-Step:**

$$Q(\theta, \theta^{(i-1)}) = E \left[ \log p_\theta(Y_L, X_L, X_U) \middle| X_L, X_U, \theta^{(i-1)} \right]$$

**M-Step:**

$$\hat{\theta} = \arg\max_\theta Q(\theta, \theta^{(i-1)})$$
Experimental Test: Prosodic Break Detection in Mandarin

- Every syllable is either $y_i = 1$ (prosodic phrase final) or $y_i = 0$ (nonfinal).
- Syllables that are known to be word-nonfinal are therefore also phrase-nonfinal. Call these $\mathcal{D}_L$.
- Syllables that are word-final may be either phrase-final or phrase-nonfinal. Call these $\mathcal{D}_U$.
- $x_i = 25$ acoustic features based on pitch, duration, and energy of the syllable.
- 402 training sentences, 25 testing sentences.
“Class Discovery” case: labeled examples of the phrase-nonfinal class, but no labeled examples of the phrase-final class.

“Semi-Supervised” case: All syllables followed by silence are automatically considered to be phrase-final, thus providing $\mathcal{D}_L$ with examples from both classes.

<table>
<thead>
<tr>
<th></th>
<th>nonbreak</th>
<th>break w/o silence</th>
<th>minor break, silence</th>
<th>major break, silence</th>
<th>total</th>
</tr>
</thead>
<tbody>
<tr>
<td>Chance</td>
<td>100%</td>
<td>0%</td>
<td>0%</td>
<td>0%</td>
<td>79%</td>
</tr>
<tr>
<td>Class Discovery</td>
<td>70</td>
<td>83</td>
<td>87</td>
<td>87</td>
<td>73</td>
</tr>
<tr>
<td>Semi-Supervised</td>
<td>84</td>
<td>66</td>
<td>77</td>
<td>91</td>
<td>82</td>
</tr>
<tr>
<td>Supervised</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>89</td>
</tr>
</tbody>
</table>
By assuming that class-conditional PDFs are piece-wise compact (e.g., GMM), it is possible to

- Use unlabeled data to improve the distribution estimates via ML estimation
- Use unlabeled data to improve classifier performance via MMI/NCE estimation
- Discover, in the unlabeled data, a class that did not exist in the labeled data

When semi-supervised learning depends on compact likelihood functions, choosing the right data representation is important

- Knowledge-based representation (e.g., gestures): estimate the pronunciation likelihood with no learning at all
- Instance PDF representation (e.g., supervectors): unknown sources of variability can be compensated using simple techniques