Lecture 20: Model Adaptation

Machine Learning
April 15, 2010
Today

• Adaptation of Gaussian Mixture Models
  – Maximum A Posteriori (MAP)
  – Maximum Likelihood Linear Regression (MLLR)
• Application: Speaker Recognition
  – UBM-MAP + SVM
The Problem

• I have a little bit of labeled data, and a lot of unlabeled data.
• I can model the training data fairly well.
• But we always fit training data better than testing data.
• Can we use the wealth of unlabeled data to do better?
Let’s use a GMM

- GMMs to model labeled data.
- In simplest form, one mixture component per class.
Labeled training of GMM

• MLE estimators of parameters

\[
\pi_t = \frac{\sum_t p(i|x_t)}{N} = \frac{n_i}{N} \quad \mu_i = \frac{\sum_t p(i|x_t)x_t}{\sum_t p(i|x_t)} = \frac{\sum_{x \in t} x}{n_k}
\]

\[
\Sigma_i = \frac{\sum_{x \in t} (x_t - \mu)(x_t - \mu)^T}{n_k}
\]

• Or these can be used to seed EM.
Adapting the mixtures to new data

- Essentially, let EM start with MLE parameters as seeds.
- Expand the available data for EM, proceed until convergence
Adapting the mixtures to new data

- Essentially, let EM start with MLE parameters as seeds.
- Expand the available data for EM, proceed until convergence
Problem with EM adaptation

• The initial labeled seeds could contribute very little to the final model
One Problem with EM adaptation

• The initial **labeled** seeds could contribute very little to the final model
MAP Adaptation

• Constrain the contribution of unlabeled data.

\[ \hat{\pi}_i = \alpha_i^{\pi} \frac{\sum_u p(i|x_u)}{U} + (1 - \alpha_i^{\pi}) \pi_i \]

\[ \hat{\mu}_i = \alpha_i^{\mu} \frac{\sum_u p(i|x_u)x_u}{\sum_u p(i|x_u)} + (1 - \alpha_i^{\mu}) \mu_i \]

\[ \hat{\Sigma}_i = \alpha_i^{\Sigma} \frac{\sum_u p(i|x_u)(x_u - \mu_i)(x_u - \mu_i)^T}{U} + (1 - \alpha_i^{\Sigma}) \Sigma_i \]

• Let the alpha terms dictate how much weight to give to the new, unlabeled data compared to the exiting estimates.
MAP adaptation

• The movement of the parameters is constrained.
MLLR adaptation

• Another idea...
• “Maximum Likelihood Linear Regression”.
• Apply an affine transformation to the means.
• Don’t change the covariance matrices

\[ \hat{\mu} = W \mu \]
MLLR adaptation

• Another view on adaptation.
• Apply an affine transformation to the means.
• Don’t change the covariance matrices

\[ \hat{\mu} = W \mu \]
MLLR adaptation

- The new means are the MLE of the means with the new data.

\[ \hat{\mu}_i = W_i \mu_i = \frac{\sum_x p(i|x, \pi_i, \mu_i, \Sigma_i) x_i}{\sum_x p(i|x, \pi_i, \mu_i, \Sigma_i)} \]
MLLR adaptation

• The new means are the MLE of the means with the new data.

\[ \hat{\mu}_i = W_i \mu_i = \frac{\sum_x p(i|x, \pi_i, \mu_i, \Sigma_i) x_i}{\sum_x p(i|x, \pi_i, \mu_i, \Sigma_i)} \]
MLLR adaptation

- The new means are the MLE of the means with the new data.

\[
\hat{\mu}_i = W_i \mu_i = \frac{\sum_x p(i| x, \pi_i, \mu_i, \Sigma_i) x_i}{\sum_x p(i| x, \pi_i, \mu_i, \Sigma_i)}
\]

\[
W_i = \frac{\sum_x p(i| x, \pi_i, \mu_i, \Sigma_i) x_i}{\sum_x p(i| x, \pi_i, \mu_i, \Sigma_i)} (\mu_i^{-1})^T
\]
Why MLLR?

• We can tie the transformation matrices of mixture components.
• For example:
  – You know that the red and green classes are similar
  – Assumption: Their transformations should be similar
Why MLLR?

• We can tie the transformation matrices of mixture components.
• For example:
  – You know that the red and green classes are similar
  – Assumption: Their transformations should be similar
Application of Model Adaptation

• Speaker Recognition.
• Task: Given speech from a known set of speakers, identify the speaker.
• Assume there is training data from each speaker.
• Approach:
  – Model a generic speaker.
  – Identify a speaker by its difference from the generic speaker
  – Measure this difference by adaptation parameters
Speech Representation

• Extract a feature representation of speech.
• Samples every 10ms.

MFCC – 16 dims
Similarity of sounds

MFCC₂

/s/

/o/  /u/  /b/

MFCC₁
Universal Background Model

• If we had labeled phone information that would be great.
• But it’s expensive, and time consuming.
• So just fit a GMM to the MFCC representation of all of the speech you have.
  – Generally all but one example, but we’ll come back to this.
MFCC Scatter

MFCC\textsubscript{1} vs MFCC\textsubscript{2}

/s/

/o/  /u/

/b/
UBM fitting

MFCC$_2$  

MFCC$_1$

/s/

/o/  /u/  /b/
MAP adaptation

• When we have a segment of speech to evaluate,
  – Generate MFCC features.
  – Use MAP adaptation on the UBM Gaussian Mixture Model.
MAP Adaptation

MFCC$_2$

/s/

/o/

/u/

/b/

MFCC$_1$
MAP Adaptation

MFCC$_2$

/s/

/o/

/u/

MFCC$_1$

/b/
UBM-MAP

• Claim:
  – The differences between speakers can be represented by the movement of the mixture components of the UBM.

• How do we train this model?
UBM-MAP training

- **Supervector**
  - A vector of adapted means of the gaussian mixture components
  
  \[ \mathbf{x}_i = \left[ \hat{\mu}_0, \hat{\mu}_1, \ldots, \hat{\mu}_k \right]^T \]

  \[ t_i = \text{Speaker ID} \]

Train a supervised model with these labeled vectors.
UBM-MAP training

Training Data → UBM Training → MAP → Supervector → Multiclass SVM Training

Repeat for all training data

\[ x_i = [\hat{\mu}_0 \quad \hat{\mu}_1 \quad \ldots \quad \hat{\mu}_k]^T \]

\[ t_i = \text{Speaker ID} \]
UBM-MAP Evaluation

UBM

MAP

Supervector

Multiclass SVM

Prediction

Test Data
• Do we need all this?
• What if we just train an SVM on labeled MFCC data?
Results

• UBM-MAP (with some variants) is the state-of-the-art in Speaker Recognition.
  – Current state of the art performance is about 97% accuracy (~2.5% EER) with a few minutes of speech.

• Direct MFCC modeling performs about half as well ~5% EER.
Model Adaptation

• Adaptation allows GMMs to be seeded with labeled data.
• Incorporation of unlabeled data gives a more robust model.
• Adaptation process can be used to differentiate members of the population — UBM-MAP
Next Time

- Spectral Clustering