Cross-Instance Tuning of Unsupervised Document Clustering Algorithms

Damianos Karakos, Jason Eisner, and Sanjeev Khudanpur

Center for Language and Speech Processing
Johns Hopkins University

Carey E. Priebe

Dept. of Applied Mathematics and Statistics
Johns Hopkins University

NAACL-HLT'07 - April 24, 2007

The talk in one slide

- **Scenario:** unsupervised learning under a wide variety of conditions (e.g., data statistics, number and interpretation of labels, etc.)
- Performance varies; can our knowledge of the task help?
- **Approach:** introduce ***tunable*** parameters into the unsupervised algorithm. Tune the parameters for each condition.
- Tuning is done in an unsupervised manner using ***supervised*** data from an ***unrelated*** instance (cross-instance tuning).
- **Application:** unsupervised document clustering.

The talk in one slide

- **Scenario:** unsupervised learning under a wide variety of conditions (e.g., data statistics, number and interpretation of labels, etc.)

- Performance varies; can our knowledge of the task help?

- **Approach:** introduce ***tunable*** parameters into the unsupervised algorithm. Tune the parameters for each condition.

- Tuning is done in an unsupervised manner using ***supervised*** data from an ***unrelated*** instance (cross-instance tuning).

- **Application:** unsupervised document clustering.

The talk in one slide

- **STEP 1:** *Parameterize* the unsupervised algorithm, i.e., convert into a supervised algorithm.
- **STEP 2:** *Tune* the parameter(s) using *unrelated* data; still unsupervised learning, since no labels of the task instance of interest are used.

The talk in one slide

- **STEP 1:** *Parameterize* the unsupervised algorithm, i.e., convert into a supervised algorithm.
- **STEP 2:** *Tune* the parameter(s) using *unrelated* data; still unsupervised learning, since no labels of the task instance of interest are used.

Applicable to any supervised scenario where training data ≠ test data

Combining Labeled and Unlabeled Data

- **Semi-supervised learning:** using a few labeled examples of the same kind as the unlabeled ones. E.g., bootstrapping (Yarowsky, 1995), co-training (Blum and Mitchell, 1998).

- **Multi-task learning:** labeled examples in many tasks, learning to do well in all of them.

- Special case: alternating structure optimization (Ando and Zhang, 2005).

- Mismatched learning: domain adaptation. E.g., (Daume and Marcu, 2006).

Reminder

- **STEP 1:** *Parameterize* the unsupervised algorithm, i.e., convert into a supervised algorithm.
- **STEP 2:** *Tune* the parameter(s) using *unrelated* data; still unsupervised learning, since no labels of the task instance of interest are used.

Reminder

- **STEP 1:** *Parameterize* the unsupervised algorithm, i.e., convert into a supervised algorithm.
- **STEP 2:** *Tune* the parameter(s) using *unrelated* data; still unsupervised learning, since no labels of the task instance of interest are used.

Document clustering.

Unsupervised Document Clustering

- Goal: Cluster documents into a pre-specified number of categories.

- Preprocessing: represent documents into fixed-length vectors (e.g., in tf/idf space) or probability distributions (e.g., over words).

- Define a "distance" measure and then try to minimize the intra-cluster distance (or maximize the inter-cluster distance).

- Some general-purpose clustering algorithms: K-means, Gaussian mixture modeling, etc.

Step I : Parameterization

Ways to parameterize the clustering algorithm:

- In the "distance" measure: e.g., L_p distance instead of Euclidean.

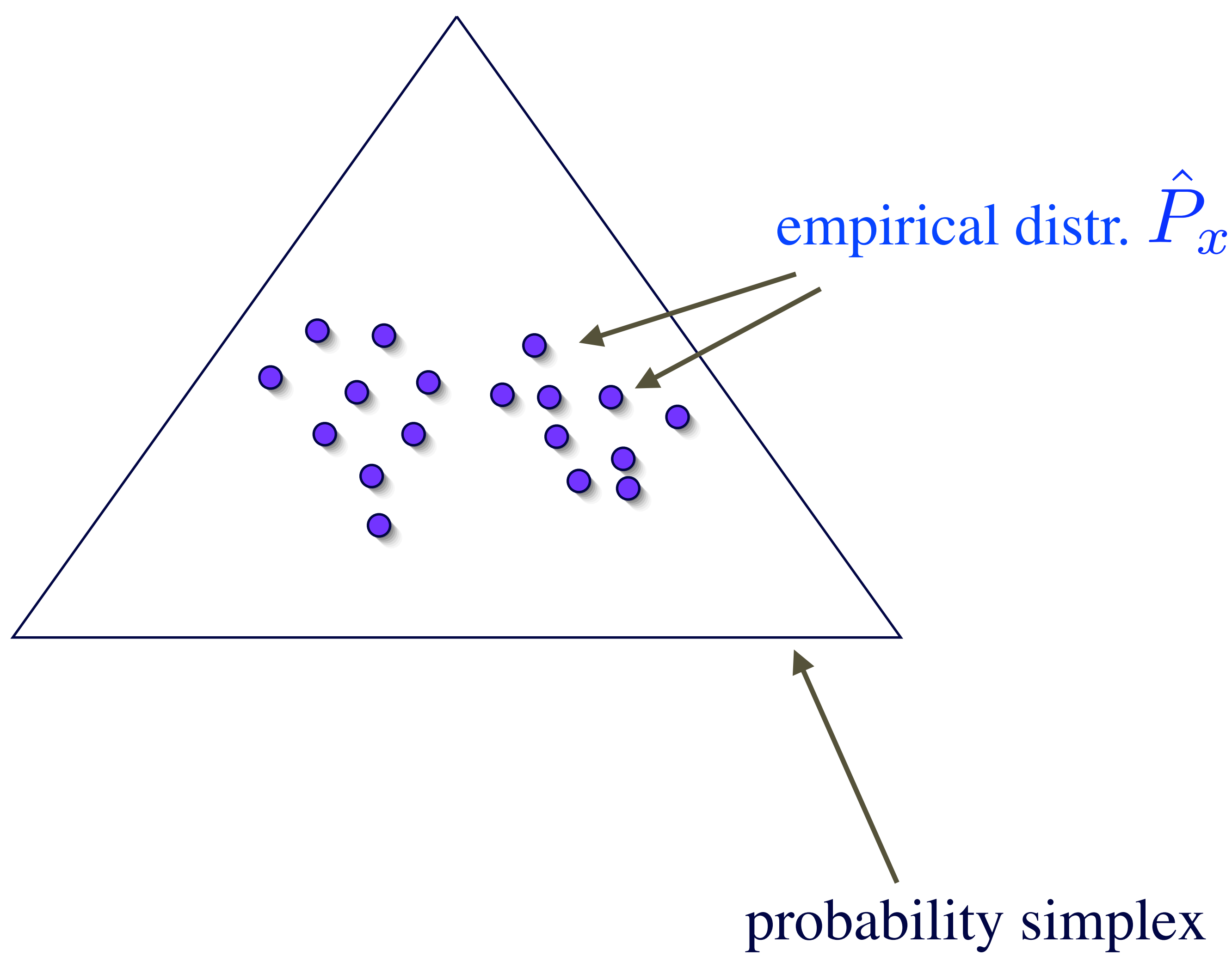
- In the dimensionality reduction: e.g., constrain the projection in the first p dimensions.

- In Gaussian mixture modeling: e.g., constrain the rank of the covariance matrices.

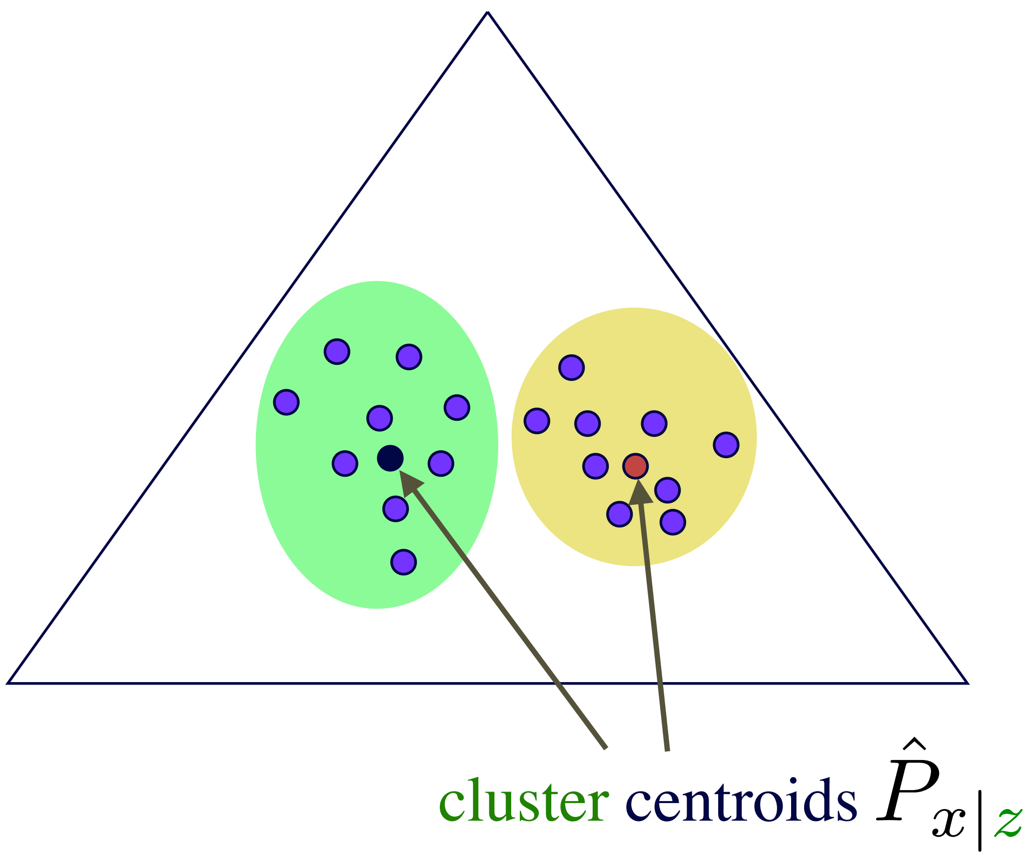
- In the smoothing of the empirical distributions: e.g., the discount parameter.

- Information-theoretic clustering: generalized information measures.

Information-theoretic Clustering



Information-theoretic Clustering



Information Bottleneck

- Considered state-of-the-art in unsupervised document classification.
- Goal: maximize the mutual information between words and assigned clusters.
- In mathematical terms:

$$\max_{\hat{P}_{x|z}} I(Z; X^n(Z))$$

$$= \max_{\hat{P}_{x|z}} \sum_{z} P(Z = z) D(\hat{P}_{x|z} \| \hat{P}_x)$$

Information Bottleneck

- Considered state-of-the-art in unsupervised document classification.
- Goal: maximize the mutual information between words and assigned clusters.
- In mathematical terms:

cluster index

$$\max_{\hat{P}_{x|z}} I(Z; X^n(Z))$$

$$= \max_{\hat{P}_{x|z}} \sum_z P(Z = z) D(\hat{P}_{x|z} \| \hat{P}_x)$$

empirical distr.

Integrated Sensing and Processing Decision Trees

- Goal: greedily maximize the mutual information between words and assigned clusters; top-down clustering.
- Unique feature: data are *projected* at each node before splitting (corpus-dependent-feature-extraction).
- Objective optimization via *joint* projection and clustering.
- In mathematical terms, at each node t :

$$\max_{\hat{\mathcal{P}}_{x\,|\,z}} I(Z_t; X^n(Z_t))$$

$$= \max_{\hat{\mathcal{P}}_{x\,|\,z}} \sum_{z} P(Z = z|t) D(\hat{\mathcal{P}}_{x|z} \| \hat{\mathcal{P}}_x |t)$$

Integrated Sensing and Processing Decision Trees

- Goal: greedily maximize the mutual information between words and assigned clusters; top-down clustering.
- Unique feature: data are *projected* at each node before splitting (corpus-dependent-feature-extraction).
- Objective optimization via *joint* projection and clustering.
- In mathematical terms, at each node t :

$$\max_{\hat{\mathcal{P}}_{x|z}} I(Z_t; X^n(Z_t))$$

$$= \max_{\hat{\mathcal{P}}_{x|z}} \sum_z P(Z = z|t) D(\hat{\mathcal{P}}_{x|z} || \hat{\mathcal{P}}_x | t)$$

projected empirical distr.

Integrated Sensing and Processing Decision Trees

- Goal: greedily maximize the mutual information between words and assigned clusters; top-down clustering.
- Unique feature: data are *projected* at each node before splitting (corpus-dependent-feature-extraction).
- Objective optimization via *joint* projection and clustering.
- In mathematical terms, at each node t :

See ICASSP-07 paper

$$\max_{\hat{\mathcal{P}}_{x|z}} I(Z_t; X^n(Z_t))$$

$$= \max_{\hat{\mathcal{P}}_{x|z}} \sum_z P(Z = z|t) D(\hat{\mathcal{P}}_{x|z} || \hat{\mathcal{P}}_x | t)$$

projected empirical distr.

Useful Parameterizations

- Of course, it makes sense to choose a parameterization that has the *potential* of improving the final result.
- Information-theoretic clustering: Jensen-Renyi divergence and Csiszar's mutual information can be less sensitive to sparseness than regular MI.
- I.e., instead of smoothing the sparse data, we create an optimization objective which works equally well with sparse data.

Useful Parameterizations

- Jensen-Renyi divergence:
-
$$I_\alpha(X;Z) = H_\alpha(X) - \sum_z P(Z=z) H_\alpha(X|Z=z)$$
-

- Csiszar's mutual information:

$$I_\alpha^C(X;Z) = \min_Q \sum P(Z=z) D_\alpha(P_{X|Z}(\cdot|Z=z) \| Q)$$

$$0 < \alpha \leq 1$$

Useful Parameterizations

- Jensen-Renyi divergence:

- $$I_\alpha(X;Z) = H_\alpha(X) - \sum_z P(Z=z) H_\alpha(X|Z=z)$$
-

- Csiszar's mutual information:

$$I_\alpha^C(X;Z) = \min_Q \sum P(Z=z) D_\alpha(P_{X|Z}(\cdot|Z=z)\|Q)$$

$$0 < \alpha \leq 1$$

Useful Parameterizations

- Jensen-Renyi divergence:
-

$$I_\alpha(X;Z) = H_\alpha(X) - \sum_z P(Z=z) H_\alpha(X|Z=z)$$

-

Renyi entropy

- Csiszar's mutual information:

Renyi divergence

$$I_\alpha^C(X;Z) = \min_Q \sum P(Z=z) D_\alpha(P_{X|Z}(\cdot|Z=z) \| Q)$$

$$0 < \alpha \leq 1$$

Step II : Parameter Tuning

Options for tuning the parameter(s) using labeled unrelated data (*cross-instance tuning*):

- Tune the parameter to do well on the unrelated data; use the *average value* of this optimum parameter on the test data.
- Use a *regularized* version of the above: instead of the "optimum" parameter, use an *average* over many "good" values.
- Use various "clues" to *learn a meta-classifier* that distinguishes good from bad parameters, i.e., "Strapping" (Eisner and Karakos, 2005).

Experiments

Unsupervised document clustering from the “20 Newsgroups” corpus:

- Test data sets have the same labels as the ones used by (Slonim *et al*., 2002).
    - “Binary”: *talk.politics.mideast, talk.politics.misc*
    - “Multi5”: *comp.graphics, rec.motorcycles, rec.sport.baseball, sci.space, talk.politics.mideast*
    - “Multi10”: *alt.atheism, comp.sys.mac.hardware, misc.forsale, rec.autos, rec.sport.hockey, sci.crypt, sci.electronics, sci.med, sci.space, talk.politics.guns*.

Experiments

Unsupervised document clustering from the “20 Newsgroups” corpus:

- Training data sets have *different* labels from the corresponding test set labels.
- Collected training documents from newsgroups which are close (in the tf/idf space) to the test newsgroups (in an unsupervised manner).
- For example, for the test set “Multi5” (with documents from the test newsgroups *comp.graphics*, *rec.motorcycles*, *rec.sport.baseball*, *sci.space*, *talk.politics.mideast)* we collected documents from the newsgroups *sci.electronics*, *rec.autos*, *sci.med*, *talk.politics.misc*, *talk.religion.misc*).

Experiments

Tuning of α (rounded-off to 0.1, 0.2, ... 1.0) using the labeled data

- **Option 1:** Used the average α that gave the lowest error on the training data.

- **Option 2:** Regularized least squares to approximate the probability that an α is the best:

$$\hat{\boldsymbol{p}} = \mathbf{K}(\lambda \mathbf{I} + \mathbf{K})^{-1} \boldsymbol{p}$$

where

$$\boldsymbol{p} = (0, \ldots, 1, \ldots, 0)$$

$$K(i, j) = \exp(-(\mathcal{E}(\alpha_i) - \mathcal{E}(\alpha_j))^2 / \sigma^2)$$

Value used:

$$\hat{\alpha} = \sum_{i=1}^{10} \hat{p}_i \, \alpha_i$$

Experiments

Tuning of α (rounded-off to 0.1, 0.2, ... 1.0) using the labeled data

- **Option 3:** "Strapping": from each training clustering, build a feature vector with clues that measure clustering goodness. Then, learn a model which predicts the best clustering from these clues.
- Clues:
  - 1 - avg. cosine of angle between documents and cluster centroid (in tf/idf space).
  - Avg. Renyi divergence between empirical distributions and assigned cluster centroid.
  - A value per α, which is decreasing with the avg. ranking of the clustering (as predicted by the above clues).

Experiments

Tuning of α (rounded-off to 0.1, 0.2, ... 1.0) using the labeled data

- **Option 3:** "Strapping": from each training clustering, build a feature vector with clues that measure clustering goodness. Then, learn a model which predicts the best clustering from these clues.
- Clues:

Do not require any knowledge of the true labels

- 1 - avg. cosine of angle between documents and cluster centroid (in tf/idf space).
- Avg. Renyi divergence between empirical distributions and assigned cluster centroid.
- A value per α, which is decreasing with the avg. ranking of the clustering (as predicted by the above clues).

Results

Algorithm	Method	Binary	Multi5	Multi10
ISPDT	MI (α=1)	11.3%	9.9%	42.2%
	avg. best α	**9.7%** (α=0.3)	10.4% (α=0.8)	42.5% (α=0.5)
	RLS	**10.1%**	10.4%	42.7%
	Strapping	**10.4%**	**9.2%**	**39.0%**
IB	MI (α=1)	12.0%	6.8%	38.5%
	avg. best α	**11.4%** (α=0.2)	7.2% (α=0.8)	**36.1%** (α=0.8)
	RLS	**11.1%**	7.4%	**37.4%**
	Strapping	**11.2%**	6.9%	**35.8%**

Results

Algorithm	Method	Binary	Multi5	Multi10
ISPDT	MI (α=1)	11.3%	9.9%	42.2%
	avg. best α	**9.7%*** (α=0.3)	10.4% (α=0.8)	42.5% (α=0.5)
	RLS	**10.1%***	10.4%	42.7%
	Strapping	**10.4%***	**9.2%**	**39.0%***
IB	MI (α=1)	12.0%	6.8%	38.5%
	avg. best α	**11.4%** (α=0.2)	7.2% (α=0.8)	**36.1%** (α=0.8)
	RLS	**11.1%**	7.4%	**37.4%**
	Strapping	**11.2%**	6.9%	**35.8%***

* : significance at $p < 0.05$

Conclusions

- Appropriate parameterization of unsupervised algorithms is helpful.
- Tuning the parameters requires (i) a different (unrelated) task instance and (ii) a method of selecting the parameter.
- "Strapping", which learns a meta-classifier for distinguishing good from bad classifications has the best performance (7-8% relative error reduction).