A Case Study of Semi-supervised Classification Methods for Imbalanced Data Set Situation

11742 IR-Lab Project Fall 2004

Yanjun Qi
Road Map

- Introduction of Semi-supervised Learning
- Three semi-supervise classifiers we compared
- Experiments and Results
## Introduction

- **Learning:** *Supervised* (classification, regression, etc.) *vs.* *Unsupervised* (clustering etc).

<table>
<thead>
<tr>
<th>Usage</th>
<th>Supervised learning</th>
<th>Unsupervised learning</th>
</tr>
</thead>
<tbody>
<tr>
<td>{(x,y)} labeled data</td>
<td>Yes</td>
<td>No</td>
</tr>
<tr>
<td>{x} unlabeled data</td>
<td>No</td>
<td>Yes</td>
</tr>
</tbody>
</table>
But in some applications

- Labeled data are often **hard** to obtain
  - Text categorization: time-consuming for subjects manually
  - Protein Structure, Protein interaction: laborious and expensive experimental efforts
  - etc.

- Unlabeled data are often **easy** to obtain: A lot

<table>
<thead>
<tr>
<th>Usage</th>
<th>Supervised learning</th>
<th>Semi-supervised learning</th>
<th>Unsupervised learning</th>
</tr>
</thead>
<tbody>
<tr>
<td>{(x,y)} labeled data</td>
<td>Yes</td>
<td>Yes</td>
<td>No</td>
</tr>
<tr>
<td>{x} unlabeled data</td>
<td>No</td>
<td>Yes</td>
<td>Yes</td>
</tr>
</tbody>
</table>
A Brief Review of Semi-supervised Learning

- **Semi-supervised classification**
  - Training also exploits additional unlabeled data
  - Aiming to result more accurate classification function

- **Semi-supervised clustering**
  - In recent years, some researchers successfully use labeled style constraints to help the unsupervised clustering
  - Labeled style constraints: like “must-link” or “cannot-link”, etc
Representative methods of semi-supervised classification

- Generative Model
- Large Margin based methods
- Graph based methods
- Co-training
Generative Models

- Unlabeled data $P(X)$ \(\rightarrow\) Classification $P(Y|X)$

- Generative models for joint probability
  - Gaussian [David 96, Castelli&Cover95, etc]
  - Multinomial [Nigam 98, 00]

- Use EM to combine small labeled set and large unlabeled set
  - Consider a joint model $P(x,y|\theta)$, unlabeled examples can be used to estimate parameter “theta”
  - For instance, by maximizing the joint likelihood
Large Margin Separation

- To maximize the classification margin
  - on both labeled and unlabeled data
  - while classifying the labeled data as correctly as possible

- Some existing works
  - Joachims 99: Transductive SVM
  - Kristin 2002: Boosting Decision Tree
  - Jaakkola 1999: maximum entropy
  - Et al.
Graph Based Method

- Generally based up an assumption that similar unlabeled examples should be given the same classification.
  - Place the data points on to a graph based on the distance relationships between examples
  - Then use the known labels to perform some type of graph partitioning

- Markov random walk: [Szummer and Jaakkola 2000]
- Graph Mincut: [Blum 2001, 2004]
- Gaussian Random Field [Zhu 2003, 2004]
- Tree structure [Griffiths 2003]
Co-Training

- Available data features are so redundant that we can train two classifiers using different features.

- Unlabeled data reduce the hypothesis space by forcing $h_1$ and $h_2$ to agree:
  - The two classifiers should at least agree on the classification for each unlabeled example.

- Some existing works:
  - Avrim Blum, Tom Mitchell 1998
  - F. Denis, etc (2003)
Three Methods We Compared

- Generative Models
  - Mixture Gaussian
- Large Margin based methods
  - Transductive SVM
- Graph based methods
  - Semi-Supervised learning using Gaussian random Fields
- Co-training
  - Not sure how to split the features
(1) Mixture Gaussian - EM

- David Miller & Hasn Uyar NIPS 1996
- Maximization of the total data likelihood, i.e. over both the labeled and unlabelled data
- EM used to do the iterative maximization

- The generalized mixture (GM) model
  - Assumes the class posterior for each mixture component is independent of the feature value
  - Each component is modeled by a Gaussian.
(1) Mixture Gaussian - EM

(a) Unlabeled data $U$

(b) Density $p(x)$ from infinite $U$

(c) One Gaussian component

(d) The other Gaussian component
1) Mixture Gaussian - EM

- The learning process:
  - E step: calculate each data point’s component posterior probability
  - M step:
    - update each component’s mean and variance parameter;
    - update the weight parameter;
    - update the different class given different component’s probabilities
(2) Transductive SVM

- Intuition behind
  - Assume decision boundaries lie in low-density regions of feature space
  - unlabeled examples help to find these areas.
(3) Semi-supervised learning using Gaussian random Fields


- This method can be viewed as a form of nearest neighbor approach, where the nearest labeled examples are computed in terms of a random walk on graph
(3) Semi-supervised learning using Gaussian random Fields

- Labeled and unlabeled data
  - Represented as vertices in the weighted graph
  - Edge weights encoding the similarity between instances

- Propagate label from labeled nodes to unlabeled nodes on the graph
Experiments

- Empirical comparison of three methods for a specific situation:
  - only two classes
  - have unbalanced class distribution

- 7 data sets from UCI Machine learning Repository
  - All transformed to Binary Classification task
  - Having different level of class imbalance
## Data Sets

<table>
<thead>
<tr>
<th>No.</th>
<th>DATASET</th>
<th>% MINORITY EXAMPLES</th>
<th>DATASET SIZE</th>
<th>FEATURE / CLASS SITUATION</th>
<th>CLASS USED</th>
<th>UNLABEL DATA SIZE IN EACH EXPERIMENTAL RUN</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Letter-a</td>
<td>3.9</td>
<td>20000</td>
<td>16 numeric (integer) features 17 classes</td>
<td>Letter “A” against all other letter</td>
<td>2000</td>
</tr>
<tr>
<td>2</td>
<td>Pendigits</td>
<td>8.3</td>
<td>7494</td>
<td>16 attributes (All input attributes are integers 0..100) 10 classes</td>
<td>Digits “0” against all other digits</td>
<td>2000</td>
</tr>
<tr>
<td>3</td>
<td>Letter-a-subset</td>
<td>17.0</td>
<td>4639</td>
<td>16 numeric (integer) features 17 classes</td>
<td>Letter “A” against Letter “BCDEF”</td>
<td>2000</td>
</tr>
<tr>
<td>4</td>
<td>Yeast</td>
<td>28.9</td>
<td>1484</td>
<td>8 attributes (numerical ) 10 classes</td>
<td>“NUC” against all the other localizations (429 positive)</td>
<td>1350</td>
</tr>
<tr>
<td>5</td>
<td>Pima</td>
<td>34.7</td>
<td>768</td>
<td>8 attributes (numerical ) 2 classes</td>
<td>(268 positive)</td>
<td>650</td>
</tr>
<tr>
<td>6</td>
<td>Bupa</td>
<td>42.0</td>
<td>345</td>
<td>6 attributes (numerical ) 2 classes</td>
<td>(145 positive)</td>
<td>240</td>
</tr>
<tr>
<td>7</td>
<td>Pendigits - Subset</td>
<td>50.0</td>
<td>1438</td>
<td>16 numeric (integer) features 17 classes</td>
<td>Digit “3” against digits “9” (719 positive)</td>
<td>1300</td>
</tr>
</tbody>
</table>
Experimental Design

- For each data set, various labeled set sizes to be tested:
  - \{5, 10, 20, 30, 40, 60, 80, 100\}.
  - For each labeled set’ certain size tested, perform 10 trials.

- In each trial:
  - Randomly sample labeled data from the entire dataset.
  - Randomly sample a fixed number of items from the rest as unlabeled data.
Performance Measurement

- We use error rate, average error rate and AUC area
  - Balanced error rate
    - (BER = the average of the error rate on positive class examples and the error rate on negative class examples).
    - If there are fewer positive examples, the errors on positive examples will count more.
  - Error rate
  - The area under the ROC curve (AUC score)
Performance – Set 1

DataSet1: Letter-a 3.9% “A” against All Other, Test 2000,
average error rate - four methods comparison

![Graph showing average error rate comparison for different methods.

auC score - four methods comparison

![Graph showing auC score comparison for different methods.]}
Performance – Set 2
Performance – Set 3

Dataset3: Letter-a-subset; 17.0%; Test Items: 2000; "A" against "BCDEF"
average error rate - four methods comparison

Dataset3: Letter-a-subset; 17.0%; Test Items: 2000; "A" against "BCDEF"
auc score - four methods comparison
Performance – Set 5
Performance – Set 6

average error rate - four methods comparison

auc score - four methods comparison
Performance – Set 7

![Graphs showing average error rate and auc score for four methods comparison.](image)
Performance – Set 8

Average error rate - four methods comparison

AUC score - four methods comparison
Discussion

- Harmonic and TransductiveSVM perform much better than the EM-Mixture method.

- Overall, TransductiveSVM gives a little help compared to the SVM itself by using the unlabeled data.

- Harmonic function seems a bit more stable than Transductive SVM.
Discussion

- Bad performance of EM-Mixture
  - Both labeled and unlabeled data contribute to a reduction of variance, but unlabeled data may lead to an increase in bias when modeling assumptions are incorrect!
  - If the train set is too small, the learning updating is very similar with the GMM clustering, with training points to do the initialization.
Discussion

• Bad performance of EM-Mixture

  – Compared to the small labeled set, too many unlabeled data has too big effect on the total likelihood function

  – The covariance matrix is hard to get when too small label set. Must take some ways to reduce the effect of this problem. For instance, Naïve model
Discussion

- From these experiments
  - Unlabeled data does help in the small train set case somehow
  - But it also happens that sometimes using the unlabeled data degrades the performance of the classification
Discussion

- From the results on these data set with different class ratio
  - It seems that the imbalanced distribution is not the main problem for a concrete classification task.
  - If classification perform badly under some imbalance distribution
    - most likely caused by the too small training set’s size
The End !