Semi-supervised Learning for Structured Output Variables
Ulf Brefeld  Tobias Scheffer
ICML 2006

presented by Jean-Baptiste Jeannin
CS6784  February 25th, 2010

Framework and notations
• Structured input $\mathbf{x}$ and output $\mathbf{y}$ with dependencies
• Joint feature representation $\Phi(\mathbf{x}, \mathbf{y})$
• Learn $f: \mathcal{X} \times \mathcal{Y} \rightarrow \mathbb{R}$ such that
  \[ \hat{\mathbf{y}} = \arg\max_{\mathbf{y} \in \mathcal{Y}} f(\mathbf{x}, \mathbf{y}) \] is as desired
• Linear model
  \[ f(\mathbf{x}, \mathbf{y}) = \langle \mathbf{w}, \Phi(\mathbf{x}, \mathbf{y}) \rangle \]
• Labeled examples $(x_1, y_1), \ldots, (x_n, y_n)$
• Unlabeled examples $x_{n+1}, \ldots, x_{n+m}$

Outline
• Semi-supervised learning by co-training
• Structured output variables
• Using co-training for structured output variables

Co-training
• Semi-supervised learning: using both labeled and unlabeled data for learning
• Idea of training: exploit two sufficiently redundant representations
  \[ \Phi(\mathbf{x}, \mathbf{y}) = (\Phi^0(\mathbf{x}, \mathbf{y}), \Phi^1(\mathbf{x}, \mathbf{y})) \]

Co-training
• Hypotheses are compatible if and only if for all examples
  \[ f^0(\mathbf{x}, \mathbf{y}) = f^1(\mathbf{x}, \mathbf{y}) \]
• Perfect classifiers do not disagree

Co-training
• Idea of training: exploit two sufficiently redundant representations
  • Training example: $(\langle \Phi^0(\mathbf{x}, \mathbf{y}), \Phi^1(\mathbf{x}, \mathbf{y}) \rangle, y)$
  • Test example: $(\Phi^0(\mathbf{x}, \mathbf{y}), \Phi^1(\mathbf{x}, \mathbf{y}))$
  • Hypotheses
    \[ f^0(\mathbf{x}, \mathbf{y}) = \langle \mathbf{w}^0, \Phi^0(\mathbf{x}, \mathbf{y}) \rangle \]
    \[ f^1(\mathbf{x}, \mathbf{y}) = \langle \mathbf{w}^1, \Phi^1(\mathbf{x}, \mathbf{y}) \rangle \]
  • are compatible if and only if for all test examples
    \[ f^0(\mathbf{x}, \mathbf{y}) = f^1(\mathbf{x}, \mathbf{y}) \]
Co-training

- Joint decision function
\[ f(x, y) = f^0(x, y) + f^1(x, y) \]
\[ = \langle w^0, \Phi^0(x, y) \rangle + \langle w^1, \Phi^1(x, y) \rangle \]

Structured output variables for supervised learning
[Tsochantaridis et al.]

- Support vector learning with slack variables \( \xi_i \geq 0 \)
- Introducing a loss function \( \Delta : Y \times Y \to \mathbb{R}^+ \)
- We would like \( y_i = \text{argmax}_y \langle w, \Phi(x_i, y) \rangle \)
- Minimize over all \( w \) and \( \xi_i \)
\[ \frac{1}{2}||w||^2 + C \sum_{i=1}^{n} \xi_i \]

Semi-supervised and co-learning

- Consensus maximizing principle:
  - Minimize the number of errors in labeled examples
  - Minimize the disagreement for unlabeled examples
- Let \( \hat{y}_i^1 \) be the prediction of \( x_i \) using \( f^1 \)
- \( \hat{y}_i^1 \) is treated as correct output
- For unlabeled examples \( x_{n+1}, \ldots, x_{n+m} \)
  \[ \hat{y}_i^1 = \text{argmax}_y \langle w^0, \Phi^0(x_i, y) \rangle \]
  \[ f^0(x_i, \hat{y}_i^1) - \max_{y \neq \hat{y}_i^1} f^0(x_i, y) = \gamma_i^0 \geq 1 \]
and vice-versa

Semi-supervised and co-learning

- Minimize over all \( w \) and \( \xi_i \)
\[ \frac{1}{2}||w||^2 + C \sum_{i=1}^{n} \xi_i + C C_u \sum_{i=n+1}^{n+m} \min \{ \gamma_i^1, 1 \} \xi_i \]
such that \( \forall i=1, \forall y \neq y_i \)
\[ \langle w, \Phi(x_i, y) - \Phi(x_i, \hat{y}_i) \rangle \geq 1 - \frac{\xi_i}{\Delta(y_i, y)} \]
and \( \forall i=n+1, \forall y \neq y_i \)
\[ \langle w^0, \Phi^0(x_i, \hat{y}_i^1) - \Phi^0(x_i, y) \rangle \geq 1 - \frac{\xi_i}{\Delta(y_i, y)} \]
and vice-versa
- \( \gamma_i^1 \) is the margin for the prediction of \( \hat{y}_i^1 \)

Dual problem – Empirical results

- Algebra transforms this optimization problem introducing Lagrange multipliers, like in normal Support Vector Machines, for resolution
- 3 cases are studied:
  - Multi-class classification
  - Label sequence learning
  - Natural language parsing
- Co-trained SVM outperforms SVM in most tasks

Example: label sequence learning

- Mapping sequential input to sequential output
- Datasets: sentences where we discriminate gene/other or person/organization/location
- The two views are a random split of the attributes
- Results: SVM and coSVM beat HMM. SVM is outperformed by coSVM in all but one setting
Conclusion

• A semi-supervised approach for structured output variables

• Combines the ideas of:
  • Co-learning (Blum & Mitchell, 1998)
  • Structured output variables (Tsochantaridis, Joachims, Hofmann & Altun, 2005)