Semi-Supervised Structured Output Learning

Jun Suzuki

NTT Communication Science Laboratories, NTT Corporation
For many NLP applications,

- Framework for semi-supervised learning that handles both labeled and unlabeled data is more appropriate
  - A large amount of unlabeled data is available (eg. raw texts)
  - Making labeled data requires much cost

Traditional and important NLP tasks

- eg. POS-tagging: sequence modelling
- Parsing: tree modelling
  \[\Rightarrow\] Structured output prediction tasks
A (discriminative) learning framework for structured output domain that can handle both labeled and unlabeled data

This presentation focuses solely on sequence labeling tasks as structured output prediction tasks.
Related Work

- **Max-margin approach**
  - Graph Laplacian (pair-wise similarities)
    - “Maximum Margin Semi-Supervised Learning for Structured Variables”, [Altun+, NIPS’05]
  - Multi-view (Co-training)
    - “Semi-Supervised Learning for Structured Output Variables”, [Brefield+, ICML’06]

- **Probabilistic approach**
  - HMM + EM (Baum-Welch)
  - Kernel CRFs
  - Entropy regularization
    - “Semi-Supervised Conditional Random Fields for Improved Sequence Segmentation and Labeling”, [Jiao+, Coling/ACL’06]
  - (Hybrid model)
    - [Suzuki+, EMNLP’07]
Approaches from Supervised to Semi-supervised Learning

- **Generative models**
  - can naturally incorporate unlabeled data via marginal likelihood
    - Maximum marginal likelihood (MML) estimation
      eg. [Nigam+, ML-2000]

- **Discriminative models**
  - a new assumption is required
    - Entropy regularization criterion [Jiao+, Coling/ACL-2006]
    - Pair-wise similarity [Altun+, NIPS-2005]

→ Hybrid generative and discriminative model
   + a variant of MML estimation
Overview

- Main characteristics of our proposal
  - Based on a hybrid generative and discriminative approach
    - Can incorporate techniques of generative approach into discriminative learning (HMMs + CRFs)
  - Simple and easily understandable
    - (w/o constructing special efficient calculation algorithm, new assumptions)
  - Scalable against the size of unlabeled data
    - The same calculation cost as parameter estimation for HMM
    - Linear order against the size of unlabeled data \( (M) \Rightarrow O(M) \)

- Experiments
  - Syntactic Chunking on CoNLL’00 shared task data ⇒ Current best result
  - NER on CoNLL’03 shared task data ⇒ Current 2nd best result
    - Only with unlabeled data for additional information
      - (w/o feature engineering or sensitive hyper-parameter tuning)
Outline

- How to effectively incorporate unlabeled data
  - 1. Hybrid generative and discriminative model
  - 2. Maximum discriminant functions sum parameter estimation

- Experiments on NER and syntactic chunking
  - Performance compared with the current top systems
Discriminatively combines $J$-kinds of generative models as features of a discriminative model.
Extension for SSL

- Discriminatively combines $J$-kinds of generative models (trained by using **unlabeled** data) as features of a discriminative model

Sequence labeling tasks

Unlabeled data

First-order HMMs

GM-1: $\theta_1$

GM-J: $\theta_J$

Discriminative Model

Labeled data

Linear-chain CRF

GM-j: j-th generative model

Structured Output Predictor
Conditional Probabilistic Model for SSL

- The definition of conditional probability
- Conventional supervised CRF
  \[ R(y|x; \lambda) = \frac{1}{Z_{\lambda}(x)} \prod_c \Phi_c(y, x, \lambda) \]
  \[ \Phi_c(y, x, \lambda) = \exp(\lambda \cdot f_c(y, x)) \]
- Our Hybrid model
  \[ R(y|x; \Lambda, \Theta) = \frac{1}{Z_{\Lambda, \Theta}(x)} \prod_c \Phi_c(y, x, \Lambda, \Theta) \]
  \[ \Lambda = (\lambda_1, \ldots, \lambda_I, \lambda_{I+1}, \ldots, \lambda_{I+J}) \]
  \[ \Theta = (\theta_1, \ldots, \theta_J) \]
  \[ \Phi_c(y, x, \Lambda, \Theta) = \exp(\lambda \cdot f_c(y, x)) \cdot \prod_j p_{jc}(x_j, y; \theta_j)^{\lambda_{I+j}} \]
  \[ = \exp(\Lambda \cdot h_c(y, x, \Theta)) \quad h_c(y, x, \Theta) = (f_1, \ldots, f_I, \log p_1, \ldots, \log p_J) \]

A natural semi-supervised extension of conventional supervised CRFs
Parameter Estimation for $\Lambda$

- **MAP estimation**

  Given labeled data: $\mathcal{D}_l = \{(x^n, y^n)\}_{n=1}^N$

  - Conventional supervised CRF

    $$\mathcal{L}^{\text{CRF}}(\lambda) = \sum_n \log R(y^n|x^n; \lambda) + \log p(\lambda)$$

  - Our hybrid model

    Assume $\Theta$ is given and fixed

    $$\mathcal{L}^{1}(\Lambda|\Theta) = \sum_n \log R(y^n|x^n; \Lambda, \Theta) + \log p(\Lambda)$$

The same forward-backward algorithm as used in supervised CRFs is available for estimating parameter $\Lambda$. 
Outline

- How to effectively incorporate unlabeled data
  - 1. Hybrid generative and discriminative model
  - 2. Maximum discriminant functions sum parameter estimation

- Experiments on NER and syntactic chunking
  - Performance compared with the current top systems
How to Incorporate Unlabeled Data

We have to consider

- a parameter estimation criterion for “effective” incorporation of unlabeled data
- an efficient parameter estimation algorithm

\[ \text{GM-1: } \theta_1 \quad \ldots \quad \text{GM-J: } \theta_J \]

Unlabeled data

First-order HMMs

Discriminative Model

Labeled data

Linear-chain CRF

GM-j: j-th generative model

Structured Output Predictor
Individually and independently train each HMM

Given Unlabeled data: $\mathcal{D}_u = \{x^m\}_{m=1}^M$

MML estimation

$$\hat{\theta}_1 = \arg \max_{\theta_1} \sum_m \log \sum_y p_1(x^m_1, y; \theta_1)$$

$$\vdots$$

$$\hat{\theta}_J = \arg \max_{\theta_J} \sum_m \log \sum_y p_J(x^m_J, y; \theta_J)$$
Effective Parameter Estimation Criterion

- **Maximum Discriminant Functions sum (MDF) (parameter) estimation** [Fujino+, AAAI’05]
  - A variant of maximum marginal likelihood (MML) estimation

Given unlabeled data: \( D_u = \{ x^m \}_{m=1}^M \)

**MML estimation**

\[
\hat{\theta} = \arg \max_{\theta} \mathcal{L}^{\text{MML}}(\theta) \quad \mathcal{L}^{\text{MML}}(\theta) = \sum_m \log \sum_y p(x^m, y; \theta)
\]

**MDF estimation**

\[
\hat{\theta} = \arg \max_{\theta} \mathcal{L}^{\text{MDF}}(\theta) \quad \mathcal{L}^{\text{MDF}}(\theta) = \sum_m \log \sum_y g(x^m, y; \theta)
\]

e.g., CRF: \( R(y|x; \lambda) = \frac{1}{Z_A(x)} \prod_c \Phi_c(y, x, \lambda) \)

\( g(x, y) \): discriminant function
Intuitive Derivation of MDF Estimation

- Generative classifiers (eg., Naive Bayes classifiers, HMM)
  - $p(x, y)$ is used as the discriminant functions $g(x, y)$
  - ⇒ under the given $x$ \[ p(x, y; \theta) \propto p(y|x; \theta) \]

MML estimation

\[
\hat{\theta} = \arg \max_{\theta} \mathcal{L}^{\text{MML}}(\theta) \quad \mathcal{L}^{\text{MML}}(\theta) = \sum_{m} \log \sum_{y} p(x^m, y; \theta)
\]

MDF estimation

\[
\hat{\theta} = \arg \max_{\theta} \mathcal{L}^{\text{MDF}}(\theta) \quad \mathcal{L}^{\text{MDF}}(\theta) = \sum_{m} \log \sum_{y} g(x^m, y; \theta)
\]

(In the context of generative classifiers) MML = MDF
MDF Estimation in Our Hybrid Model

- Discriminant functions of our hybrid model
  - \( Z \) is independent from \( y \)

\[
R(y|x; \Lambda, \Theta) = \frac{1}{Z_{\Lambda, \Theta}(x)} \prod_c \Phi_c(y, x, \Lambda, \Theta)
\]

\[
g(x, y; \Lambda, \Theta) = \prod_c \Phi_c(y, x, \Lambda, \Theta) = \prod_c \exp(\lambda f_c(y, x)) \cdot \prod_j p_{jc}(x_j, y; \theta_j)^{\lambda_j + j}
\]

- Objective function for MDF estimation
Algorithm for MDF Estimation

- Assume $\Lambda$ is given and fixed

\[
\text{maximize: } \mathcal{L}^2(\Theta | \Lambda) = \sum_m \log \sum_y g(x^m, y; \Lambda, \Theta) + \log p(\Theta)
\]

The same forward-backward (aka Baum-Welch) algorithm as used in conventional HMMs can be used for the parameter estimation of $\Theta$

Q-function of MDF for our Hybrid model

\[
Q(\Theta^t, \Theta^{t-1}; \Lambda) = \sum_j \lambda_{I+j} \sum_m \sum_y R(y|x^m; \Lambda, \Theta^{t-1}) \log p_j(x^m_j, y; \theta^t_j) + \log p(\Theta^t)
\]

c.f. Q-function of MML for HMM

\[
Q(\theta^t, \theta^{t-1}) = \sum_m \sum_y p(y|x^m; \theta^{t-1}) \log p(x^m, y; \theta^t) + \log p(\theta^t)
\]
Advantages of using MDF Estimation

The posterior distribution, $R$, is used as $\Theta$ estimation

- **Calculation Cost**

$$Q(\Theta^t, \Theta^{t-1}; \Gamma) = \sum_j \lambda_{I+j} \sum_m \sum_y R(y|m; \Lambda, \Theta^{t-1}) \log p_J(x_j^m, y; \theta_j^t) + \log p(\Theta^t)$$

Independent from $J$

Number of executing FB-algorithm is independent from $J$

cf. Individually and independently train $J$-kinds of HMMs

$$Q(\theta_i^t, \theta_i^{t-1}) = \sum_m \sum_y p_1(y|x_i^m; \theta_i^{t-1}) \log p_1(x_i^m, y; \theta_i^t) + \log p(\theta_i^t)$$

$$\vdots$$

$$Q(\theta_J^t, \theta_J^{t-1}) = \sum_m \sum_y p_J(y|x_J^m; \theta_J^{t-1}) \log p_J(x_J^m, y; \theta_J^t) + \log p(\theta_J^t)$$

- **Effectiveness**

- $R$ includes the distribution of labeled data

Automatically incorporates information of labeled data distribution
Parameter Estimation Procedure

- The $\Theta$ and $\Lambda$ estimations are mutually dependent

$$
\mathcal{L}^1(\Lambda|\Theta) = \sum_n \log p(y^n|x^n; \Lambda, \Theta) + \log p(\Lambda)
$$

$$
\mathcal{L}^2(\Theta|\Lambda) = \sum_m \log \sum_y g(x^m, y; \theta) + \log p(\Theta)
$$

⇒ estimate $\Theta$ and $\Lambda$ iteratively and alternatively

- Initialize $t \leftarrow 1$, $\Theta^{(0)} = \text{uniform distribution}$

- 1. Estimate $\Lambda$
  - Maximize $\mathcal{L}^1(\Lambda|\Theta)$ with fixed $\Theta \leftarrow \Theta^{(t-1)}$, using labeled data

- 2. Estimate $\Theta^{(t)}$: (initial values = $\Theta^{(t-1)}$)
  - Maximize $\mathcal{L}^2(\Theta|\Lambda)$ with fixed $\Lambda$, using unlabeled data

- $t \leftarrow t + 1$, no

- 3. Converged ?

- yes
  - Structured Output Predictor

- output
Evaluation Phase (Decoding)

- The Viterbi algorithm as used in conventional CRFs is also available for finding the most likely output $y$ given $x$. 
Experiments

- Sequence labeling tasks
  - Syntactic chunking: CoNLL’00 shared task data
  - Named entity recognition (NER): CoNLL’03 shared task data

- Results
  - Comparison with current top systems
Data

<table>
<thead>
<tr>
<th></th>
<th>Chunking (CoNLL’00)</th>
<th>NER (CoNLL’03)</th>
</tr>
</thead>
<tbody>
<tr>
<td># of labels</td>
<td>23 (w/ IOB tagging)</td>
<td>9 (w/ IOB tagging)</td>
</tr>
<tr>
<td>Data</td>
<td># of sentences</td>
<td># of words</td>
</tr>
<tr>
<td>Training Set</td>
<td>8,936</td>
<td>211,727</td>
</tr>
<tr>
<td>Development Set</td>
<td>N/A</td>
<td>N/A</td>
</tr>
<tr>
<td>Test Set</td>
<td>2,012</td>
<td>47,377</td>
</tr>
</tbody>
</table>

These data sets are exactly the same as those provided for the shared task of CoNLL’00 and CoNLL’03.
Comparison with Current Top Systems (1/2)

- Syntactic chunking results
  (CoNLL’00 shared task data)

<table>
<thead>
<tr>
<th>Methods</th>
<th>F-score</th>
<th>Additional resources</th>
</tr>
</thead>
<tbody>
<tr>
<td>Hybrid+MDF</td>
<td>94.67</td>
<td>unlabeled data (15M words: WSJ)</td>
</tr>
<tr>
<td>ASO-semi (Ando+, ACL’05)</td>
<td>94.39</td>
<td>unlabeled data (15M words: WSJ)</td>
</tr>
<tr>
<td>Winnow (Zhang+, ML’02)</td>
<td>94.17</td>
<td>full parser output</td>
</tr>
<tr>
<td>SVM comb. (Kudo+, NAACL’01)</td>
<td>93.91</td>
<td>-</td>
</tr>
<tr>
<td>Supervised CRF (baseline)</td>
<td>93.88</td>
<td>-</td>
</tr>
</tbody>
</table>

15M word unlabeled data: obtained from WSJ-1991
Approximately 75 times larger than the size of labeled training data
Comparison with Current Top Systems (2/2)

- Named entity recognition results
  (CoNLL’03 shared task data)

<table>
<thead>
<tr>
<th>Methods</th>
<th>dev.</th>
<th>test</th>
<th>Additional resources</th>
</tr>
</thead>
<tbody>
<tr>
<td>ASO-semi (Ando+, ACL’05)</td>
<td>93.15</td>
<td>89.31</td>
<td>unlabeled data (27M words)</td>
</tr>
<tr>
<td>Hybrid+MDF</td>
<td>93.12</td>
<td>88.86</td>
<td>unlabeled data (27M words)</td>
</tr>
<tr>
<td>(Florian+, CoNLL’03)</td>
<td>93.87</td>
<td>88.76</td>
<td>their own large gazetteers, 2M-word labeled data</td>
</tr>
<tr>
<td>(Chieu+, CoNLL’03)</td>
<td>93.01</td>
<td>88.31</td>
<td>their own large gazetteers, very elaborated features</td>
</tr>
<tr>
<td>MEMM (Klain+, CoNLL’03)</td>
<td>92.27</td>
<td>86.31</td>
<td>Rule-based post processing</td>
</tr>
<tr>
<td>Supervised CRF (baseline)</td>
<td>91.54</td>
<td>86.44</td>
<td>-</td>
</tr>
</tbody>
</table>

27M word unlabeled data: obtained from Reuters Corpus
(Approximately 135 times larger than the size of labeled training data)
Conclusion

- We proposed a framework of discriminative learning for structured output domain that can handle both labeled and unlabeled data
  - Hybrid generative and discriminative model
  - MDF estimation (variant of MML estimation)

- Our method provides a framework that enables us to incorporate techniques developed for generative approaches into discriminative learning