Structured Output Prediction: Discriminative Learning

CS6780 – Advanced Machine Learning
Spring 2019
Thorsten Joachims
Cornell University

Reading:
Murphy 19.7, 19.6

Structured Output Prediction

• Supervised Learning from Examples
  – Find function from input space X to output space Y
  \[ h: X \rightarrow Y \]
  such that the prediction error is low.
• Typical
  – Output space is just a single number
    • Classification: \{-1,+1\}
  – Regression: some real number
• General
  – Predict outputs that are complex objects

Training HMMs with Structural SVM

• HMM
  \[ P(x,y) = P(y_1)P(x_1|y_1) \prod_{i=2}^n P(x_i|y_i)P(y_i|y_{i-1}) \]
  \[ \log P(x,y) = \log P(y_1) + \sum_{i=2}^n \log P(x_i|y_i) + \log P(y_i|y_{i-1}) \]
• Define \( \phi(x,y) \) so that model is isomorphic to HMM
  – One feature for each possible start state
  – One feature for each possible transition
  – One feature for each possible output in each possible state
  – Feature values are counts

Joint Feature Map for Sequences

• Linear Chain HMM
  – Each transition and emission has a weight
  – Score of a sequence is the sum of its weights
  – Find highest scoring sequence \( h(x) = \arg \max_{y \in Y} \log \phi(x,y) \)

Joint Feature Map for Trees

• Weighted Context Free Grammar
  – Each rule \( r_t \) (e.g. \( S \rightarrow NP VP \)) has a weight
  – Score of a tree is the sum of its weights
  – Find highest scoring tree \( h(x) = \arg \max_{y \in Y} \log \phi(x,y) \)

Idea for Discriminative Training of HMM

Idea:
– \( h_{Bayes}(x) = \arg \max_{y \in Y} \log P(y|x) \)
– Model \( P(y|x) \) with \( \phi(x,y) \) so that \( \log P(y|x) = \log P(y|x) + \sum_{i=2}^n \log P(x_i|y_i) + \log P(y_i|y_{i-1}) \)

Hypothesis Space:
\( h(x) = \arg \max_{y \in Y} \log \phi(x,y) \) with \( \omega \in \mathbb{R}^N \)

Intuition:
– Tune \( \omega \) so that correct \( y \) has the highest value of \( \omega \cdot \phi(x,y) \)
– \( \phi(x,y) \) is a feature vector that describes the match between \( x \) and \( y \)
Structural Support Vector Machine

- Joint features $\phi(x, y)$ describe match between $x$ and $y$
- Learn weights $w$ so that $w \cdot \phi(x, y)$ is max for correct $y$

$\psi(x_1, y_1), \psi^T(x_2, y_2), \psi(x_3, y_3), \psi(x_n, y_n)$

Structural SVM Training Problem

- Training Set: $(x_1, y_1), \ldots, (x_n, y_n)$
- Prediction Rule: $h_{SVM}(x) = arg\max_{y \in Y} [w \cdot \phi(x, y)]$
- Optimization:
  - Correct label $y_i$ must have higher value of $w \cdot \phi(x, y)$ than any incorrect label $y$
  - Find weight vector with smallest norm

Soft-Margin Structural SVM

- Loss function $\Delta(y_i, y)$ measures match between target and prediction.

$\psi^T(x_1, y_1), \psi^T(x_2, y_2), \psi^T(x_3, y_3), \psi^T(x_n, y_n)$

Soft-Margin Structural SVM

- Loss function $\Delta(y_i, y)$ measures match between target and prediction.
- Representation $\Phi(x, y)$

Generic Structural SVM

- Application Specific Design of Model
  - Loss function $\Delta(y_i, y)$
  - Representation $\Phi(x, y)$

- Prediction:
  $\tilde{y} = arg\max_{y \in Y} [w^t \Phi(y)]$

- Training:

$\min_{w \geq 0} \frac{1}{2} w^T w + C \sum_{i=1}^{n} \xi_i$
  s.t. $\forall y \in Y \setminus y_i : w^T \Phi(x_i, y) \geq w^T \Phi(x_i, y_i) + \Delta(y_i, y) - \xi_i$
  $\forall y \in Y \setminus y_i : w^T \Phi(x_i, y) \geq w^T \Phi(x_i, y) + \Delta(y_i, y) - \xi_i$

- Applications: Parsing, Sequence Alignment, Clustering, etc.

Cutting-Plane Algorithm for Structural SVM

- Input: $(x_1, y_1), \ldots, (x_n, y_n), c, \epsilon$
- $S \leftarrow \emptyset$, $\tilde{w} \leftarrow 0$, $\xi \leftarrow 0$
- REPEAT
  - FOR $i = 1, \ldots, n$
    - compute $\tilde{y} = arg\max_{y \in Y} [\Delta(y_i, y) + w^T \Phi(x_i, y)]$
    - IF $\tilde{y} \neq y_i$
      - $S \leftarrow S \cup \{(x_i, y_i), (x_i, \tilde{y})\}$
      - $\xi_i \leftarrow \Delta(y_i, \tilde{y}) - \xi_i$
    - ENDIF
  - ENDFOR
- UNTIL $S$ has not changed during iteration

Find most violated constraint
Viilated by more than $\epsilon$
Add constraint to working set
Polynomial Sparsity Bound

- Theorem: The sparse-approximation algorithm finds a solution to the soft-margin optimization problem after adding at most
  \[ n \frac{4CA^2R^2}{\varepsilon^2 S} \]
  constraints to the working set, so that the Kuhn-Tucker conditions are fulfilled up to a precision \( \varepsilon \). The loss has to be bounded \( 0 \leq \Delta(y_i, y) \leq A \), and \( \|\Phi(x, y)\| \leq R \).

More Expressive Features

- Linear composition: \( \Phi(x, y) = \sum \phi(x, y_j) \)
- So far: \( \phi(x, y) = \phi_{kernel}(\phi(x, [\text{rule, start, end}])) \)
- General: \( \phi(x, y) = \phi_{kernel}(\phi(x, [\text{start, end}])) \)
- Example: \( \phi(x, y) = \begin{cases} 1 & \text{if } x_{\text{start}} = "\text{while}\" \text{ and } x_{\text{end}} = "," \vspace{-2mm} \\ (\text{start} - \text{end})^2 & \text{span contains } "\text{and}" \end{cases} \)

Applying StructSVM to New Problem

- Basic algorithm implemented in SVM-struct
  - http://svmlight.joachims.org
- Application specific
  - Loss function \( \Delta(y_i, y) \)
  - Representation \( \Phi(x, y) \)
  - Algorithms to compute
    - \( \hat{y} = \arg\max_{y \in Y} [w \cdot \Phi(x, y)] \)
    - \( \hat{y} = \arg\max_{y \in Y} [\Delta(y_i, y) + w \cdot \Phi(x, y)] \)

→ Generic structure covers OMM, MPD, Finite-State Transducers, MRF, etc.

Conditional Random Fields (CRF)

- Model:
  - \( P(y|x, w) = \frac{\exp(w \cdot \Phi(x, y))}{\sum_{y'} \exp(w \cdot \Phi(x, y'))} \)
  - \( P(w) = N(w|0, \lambda) \)
- Conditional MAP training:
  - \( \hat{\lambda} = \arg\max_{w} \left\{ -w \cdot w + \lambda \sum_{i} \log(P(y_i|x_i, w)) \right\} \)
- Prediction for zero/one loss:
  - \( \hat{y} = \arg\max_{y \in Y} [w \cdot \Phi(x, y)] \)

Encoder/Decoder Networks

- Encoder: Build fixed-size representation of input sequence \( x \).
- Decoder: Generate output sequence \( y \) from encoder output.

\[
\begin{align*}
  h_t &= h(W_h h_{t-1} + V_h x_t) \\
  g_t &= g(W_g g_{t-1} + V_g y_{t-1}) \\
  p &= f(V_f g_t)
\end{align*}
\]