Search-based Learning

Michael Collins and Brian Roark. Incremental Parsing with the Perceptron Algorithm.
Hal Daume III, Daniel Marcu. Learning as search optimization: Approximate Large Margin Methods for Structured Prediction
Hal Daume III, John Langford, Daniel Marcu. Search-Based Structured Prediction.

Presented by Veselin Stoyanov

Structured Learning

• In the heart of all algorithms so far:
  \[ y^* = \arg \max_{y \in Y} f(x, y, w) \]
  • I.e., an exhaustive search over all y
  • This computation can be very expensive or computationally impossible

When the \( \text{argmax} \) is intractable…

• E.g. joint label+sequence inference, parsing, multi-document compression
• An idea:
  – Take advantage of the sequential nature of the decision
  – Decompose the problem and limit the search space

Approach 1: Collins and Roark
Incremental Parsing with the Perceptron Algorithm

• Reminiscent of the Structured Perceptron algorithm

Given \((x_i, y_i)\)
For \(t = 1 \ldots T, j = 1 \ldots n\)
  \[ z_i = \arg \max_{z \in \text{Gen}(x_i)} f(x_i, z, w) \]
  if \((z_i \neq y_i)\) then \(w = w + \Phi(x_i, y_i) - \Phi(x_i, z_i)\)

Noun Verb Adj. Noun
Yejin likes merino sheep.

Approach 1: Collins and Roark
Incremental Parsing with the Perceptron Algorithm

Yejin likes merino sheep.
Approach 1: Collins and Roark
Incremental Parsing with the Perceptron Algorithm

Yejin likes merino sheep.
Approach 1: Collins and Roark
Incremental Parsing with the Perceptron Algorithm

Noun   Verb
Verb    Noun
Verb    Adj.
Yejin likes merino sheep.

Approach 1: Collins and Roark
Incremental Parsing with the Perceptron Algorithm

• New incremental Perceptron algorithm

Given \((x_i, y_i)\)
For \(t = 1, \ldots, T, i = 1, \ldots, n\)
\[ z_i = \text{argmax}_{z \in \mathcal{F}(x_i, z, w)} f(x_i, z, w) \]
if \((z_i \neq y_i)\) then \(w = w + \Phi(x_i, y_i) - \Phi(z_i, y_i)\)

Approach 1: Collins and Roark
Incremental Parsing with the Perceptron Algorithm

• Refinements to the learning algorithm
  – Repeated use of hypothesis
    • Construct the \(F\)-sets and use them for a few iterations of the perceptron update
  – Early update
    • If the correct labeling falls out of the beam, stop decoding and use the partially decoded sequence

Approach 1: Collins and Roark
Incremental Parsing with the Perceptron Algorithm

• Experimental evaluation: Parsing (Pen Treebank)
• Results:
  – With standard features identical to generative model
  – Using the output of the generative model as a feature shows improvement (f1 of 88.8 vs. 86.7)

Approach 2: Daume and Marcu:
Learning as Search Optimization

• Similar to Collins and Roark
• Theoretical contributions:
  – Generalizes learning as search
  – Gives theoretical justifications of the algorithm
• Differences in the parameter update:
  – When gold standard falls out of the beam:
    • Update parameters and continue from correct solutions
  – Approximate maximum margin update

Approach 3: Daume, Langford and Marcu:
Search-based Structured Prediction

• Advances the idea one step further
• Keep the beam size 1
  – i.e. at every step, predict the best output given the partial output so far
• Thus, at every step you have a multiclass classification problem
Approach 3: Daume, Langford and Marcu: Search-based Structured Prediction

- Removes the search from the process
- Can incorporate any multiclass classifier
- Can handle more general features and loss function
- Theoretically sound
  - Proofs that good performance on the multiclass classification leads to good performance on the structured prediction

Yejin likes merino sheep.

Yejin likes merino sheep.

Yejin likes merino sheep.

Yejin likes merino sheep.
Approach 3: Daume, Langford and Marcu: Search-based Structured Prediction

• Problem: Once the classifier makes an error (exits the Garden Path) performance deteriorates

• Solution: Start with the optimal policy and slowly move to a learned policy

Approach 3: Daume, Langford and Marcu: Search-based Structured Prediction

• A few definitions:
  – Policy
    • A policy $h$ is a distribution over actions conditioned on an input $x$ and state $s$.
    • (The multiclass prediction)
  – Optimal policy
    • A policy that, for a given state, input and output always predicts the best action to take.
    • $\pi^*$

Approach 3: Daume, Langford and Marcu: Search-based Structured Prediction

• Cost-sensitive examples:
  – How are they computed
    • Given a state $s$
    • For every possible action $a$
      – Compute the expected loss of the action by decoding the full sequence using the current policy $\pi$
      – Create a cost vector for state $s$
        $l_a^n = \min l_a^n; l_a^n = \min l_a^n; \ldots; l_a^n = \min l_a^n$)
    • These costs will be used to train the multiclass classifier

Approach 3: Daume, Langford and Marcu: Search-based Structured Prediction

• The algorithm

\[
\begin{align*}
\theta^{(n+1)} &\leftarrow \theta^n \\
&\quad \text{for } l = 1, \ldots, L_{\text{max}} \text{ do} \\
&\quad S_l \leftarrow 0 \\
&\quad \text{for } n = 1, \ldots, N \text{ do} \\
&\quad \langle x_1, \ldots, x_N \rangle \leftarrow \text{path}(x, h^{l-1}, 0) \\
&\quad \text{Add}(\Phi(x_h), \pi^{(l-1)}(x, s_l) \pi^{(l-1)}(x, s_l) \ldots \pi^{(l-1)}(x, s_l)) \in S_l \\
&\quad \text{end for} \\
&\quad h' \leftarrow \text{Learn}(S_l) \\
&\quad \theta^{(n+1)} \leftarrow \beta \theta^n + (1 - \beta) \theta^{(n+1)} \\
&\quad \text{end for} \\
&\quad \text{return } h^{(n+1)} = \pi^*
\end{align*}
\]
Approach 3: Daume, Langford and Marcu: Search-based Structured Prediction

• Computational requirements
  – Optimal policy assumption
    \[
    \arg \min_{y \in \mathcal{Y}} w^T \Phi(x, y^*) + l(y, y^*)
    \]
  – Additionally, optimal policy is not required

• Beyond greedy search
  – Could add the beam search back in the algorithm
  – Similar to reinforcement learning

Approach 3: Daume, Langford and Marcu: Search-based Structured Prediction

• Spanish named entity recognition

<table>
<thead>
<tr>
<th>CRF</th>
<th>SVM</th>
<th>SVM ISO</th>
<th>SEARN</th>
</tr>
</thead>
<tbody>
<tr>
<td>Acc</td>
<td>94.83</td>
<td>94.94</td>
<td>94.90</td>
</tr>
</tbody>
</table>

Approach 3: Daume, Langford and Marcu: Search-based Structured Prediction

• Empirical evaluation
  – Spanish named entity recognition
  – Handwriting recognition
  – Joint sequence labeling
  – Multi-document compression

Approach 3: Daume, Langford and Marcu: Search-based Structured Prediction

• Handwriting recognition

| Acc-sm | 81.00 | 87.00 | 87.50 | 70.17 | 73.81 | 62.12 | 87.55 | 88.20 |
|        | 76.88 | 79.28 | 90.58 | 90.91 | 90.22 |
Approach 3: Daume, Langford and Marcu: Search-based Structured Prediction

• Joint sequence labeling

<table>
<thead>
<tr>
<th>Method</th>
<th>Joint Acc</th>
<th>Chunk F-score</th>
<th>POS Acc</th>
</tr>
</thead>
<tbody>
<tr>
<td>SEARN</td>
<td>96.09</td>
<td>92.99</td>
<td>98.90</td>
</tr>
<tr>
<td>Joint Acc (G)</td>
<td>96.81</td>
<td>93.62</td>
<td>99.07</td>
</tr>
<tr>
<td>Chunk F (G)</td>
<td>-</td>
<td>93.80</td>
<td>-</td>
</tr>
<tr>
<td>Joint Acc (B)</td>
<td>94.62</td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td>Chunk F (B)</td>
<td>94.62</td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td>Incr. Perceptron (G)</td>
<td>91.40</td>
<td>90.82</td>
<td>98.94</td>
</tr>
<tr>
<td>Incr. Perceptron (B)</td>
<td>93.12</td>
<td>92.44</td>
<td>98.05</td>
</tr>
<tr>
<td>LaSO (G)</td>
<td>94.74</td>
<td>93.11</td>
<td>98.89</td>
</tr>
<tr>
<td>LaSO (B)</td>
<td>96.72</td>
<td>93.49</td>
<td>98.90</td>
</tr>
<tr>
<td>Factored CRF</td>
<td>96.48</td>
<td>93.87</td>
<td>98.92</td>
</tr>
<tr>
<td>CRF</td>
<td>-</td>
<td>94.77</td>
<td>-</td>
</tr>
</tbody>
</table>

Approach 3: Daume, Langford and Marcu: Search-based Structured Prediction

• Multi-document compression

<table>
<thead>
<tr>
<th>Method</th>
<th>100 words</th>
<th>200 words</th>
</tr>
</thead>
<tbody>
<tr>
<td>Oracle</td>
<td>7.32</td>
<td>22.72</td>
</tr>
<tr>
<td>Vine-Growth</td>
<td>4.98</td>
<td>15.80</td>
</tr>
<tr>
<td>Extraction</td>
<td>4.28</td>
<td>12.33</td>
</tr>
<tr>
<td>BQFS+DUC05</td>
<td>3.97</td>
<td>10.89</td>
</tr>
<tr>
<td>BQFS+DUC03</td>
<td>3.54</td>
<td>9.47</td>
</tr>
<tr>
<td>Baseline</td>
<td>2.41</td>
<td>5.82</td>
</tr>
<tr>
<td>Best DUC05</td>
<td>-</td>
<td>10.19</td>
</tr>
</tbody>
</table>

Summary

• Collins and Roark’s Incremental Perceptron
  – Incremental decoding and perceptron-style learning

• Daume and Marcu’s LaSO
  – Generalize the search procedure
  – Theoretical justification
  – Approximate max-margin update

• Daume, Langford and Marcu’s SEARN
  – Separate incremental learning and the perceptron
  – Can incorporate any multiclass classifier
  – Can handle more general features and loss function
  – Theoretically sound