Using N-Gram LMs with SCFG TMs

March 20, 2013
Hypergraph review

Source label

Target label

Goal node

la lectura : reading

ayer : yesterday

1 de 2 : 2’s 1

1 de 2 : 1 from 2
Hypergraph review

Substitution sites / variables / non-terminals
Hypergraph review

For LM integration, we ignore the source!
Hypergraph review

For LM integration, we ignore the source!
Hypergraph review

How can we add the LM score to each string derived by the hypergraph?
LM Integration

- If LM features were purely local ...
  - “Unigram” model
  - Discriminative LM
- ... integration would be a breeze
  - Add an “LM feature” to every edge
- But, LM features are non-local!
Why is it hard?

Two problems:

1. What is the content of the variables?
Why is it hard?

Two problems:

1. What is the content of the variables?
Why is it hard?

Two problems:

1. What is the content of the variables?
Why is it hard?

Two problems:

1. What is the content of the variables?
Why is it hard?

Two problems:

1. What is the content of the variables?
2. What will be the left context when this string is substituted somewhere?
Why is it hard?

Two problems:

1. What is the content of the variables?
2. What will be the left context when this string is substituted somewhere?
Why is it hard?

Two problems:

1. What is the content of the variables?

2. What will be the left context when this string is substituted somewhere?
Why is it hard?

Two problems:

1. What is the content of the variables?

2. What will be the left context when this string is substituted somewhere?
Naive solution

• Extract the all (k-best?) translations from the translation model
• Score them with an LM
• What’s the problem with this?
Outline of DP solution

• Use $n$-order Markov assumption to help us
  
  • In an $n$-gram LM, words more than $n$ words away will not affect the local (conditional) probability of a word in context
  
  • This is not generally true, just the Markov assumption!

• General approach
  
  • Restructure the hypergraph so that LM probabilities decompose along edges.
  
  • Solves both “problems”
    
    • we will not know the full value of variables, but we will know “enough”.
    
    • defer scoring of left context until the context is established.
Hypergraph restructuring

• Note the following three facts:
  • If you know $n$ or more consecutive words, the conditional probabilities of the $n$th, $(n+1)$th, ... words can be computed.
    • Therefore: add a feature weight to the edge for words.
  • $(n-1)$ words of context to the left is enough to determine the probability of any word
    • Therefore: split nodes based on the $(n-1)$ words on the right side of the span dominated by every node
  • $(n-1)$ words on the left side of a span cannot be scored with certainty because the context is not known
    • Therefore: split nodes based on the $(n-1)$ words on the left side of the span dominated by every node
Hypergraph restructuring

• Note the following three facts:

  • If you know \( n \) or more consecutive words, the conditional probabilities of the \( n \)th, \((n+1)\)th, ... words can be computed.

  • \((n-1)\) words on the left side of a span cannot be scored with certainty because the context is not known.

  • Therefore: split nodes based on the \((n-1)\) words on the left side of the span dominated by every node.

Split nodes by the \((n-1)\) words on both sides of the convergent edges.
Hypergraph restructuring

- Algorithm ("cube intersection"):
  - For each node $v$ (proceeding in topological order through the nodes)
    - For each edge $e$ with head-node $v$, compute the $(n-1)$ words on the left and right; call this $q_e$
      - Do this by substituting the $(n-1)\times2$ word string from the tail node corresponding to the substitution variable
      - If node $vq_e$ does not exist, create it, duplicating all outgoing edges from $v$ so that they also proceed from $vq_e$
      - Disconnect $e$ from $v$ and attach it to $vq_e$
  - Delete $v$
Hypergraph restructuring
Hypergraph restructuring

-LM Viterbi:
  the stain’s the man
Hypergraph restructuring

Let's add a bi-gram language model!
Hypergraph restructuring

Let’s add a bi-gram language model!
Hypergraph restructuring

\[ p(\text{mancha}|\text{la}) \]

- 0.1 la mancha
- 0.7 the stain
- 0.2 the gray stain

- 0.6 the man
- 0.4 the husband

- 0.6
- 0.4

- From 1 to 2
Hypergraph restructuring

\[ p(\text{mancha}|\text{la}) \]

0.1 \text{ la mancha} \rightarrow X
0.7 \text{ the stain} \rightarrow X
0.2 \text{ the gray stain} \rightarrow X

0.6 \text{ the man} \rightarrow X
0.4 \text{ the husband} \rightarrow X

0.6 \text{ 's 1}
0.4 \text{ 1 from 2}
Hypergraph restructuring

\[ p(\text{stain}|\text{the}) \]

- 0.6: the man
- 0.4: the husband
- 0.1: la mancha
- 0.7: the stain
- 0.2: the gray stain

\[ 2 \text{'s} 1 \\
1 \text{ from} 2 \]

0.6

0.4
Hypergraph restructuring

\[ p(\text{stain}|\text{the}) \]

- the man: 0.6
- the husband: 0.4
- la mancha: 0.1
- the stain: 0.7
- the gray stain: 0.2

- 2's: 0.6
- 1 from 2: 0.4
Hypergraph restructuring
Hypergraph restructuring
Hypergraph restructuring
Hypergraph restructuring
Hypergraph restructuring

- the man
- the husband
- la mancha
- the stain
- the gray stain

Weights:
- 0.6
- 0.4
- 0.1
- 0.7
- 0.2

2's 1
1 from 2
0.6
0.4
Hypergraph restructuring

Every node “remembers” enough for edges to compute LM costs.
Complexity

• What is the run-time of this algorithm?
Complexity

• What is the run-time of this algorithm?

\[ O(|V||E||\Sigma|^{2(n-1)}) \]

Going to longer n-grams is exponentially expensive!
Cube pruning

• Expanding every node like this exhaustively is impractical

• Polynomial time, but really, really big!

• Cube pruning: minor tweak on the above algorithm

• Compute the k-best expansions at each node

• Use an estimate (usually a unigram probability) of the unscored left-edge to rank the nodes
Cube pruning

• Why “cube” pruning?
  • Cube-pruning only involves a “cube” when arity-2 rules are used!
  • More appropriately called “square” pruning with arity-1
  • Or “hypercube” pruning with arity > 2!
Cube Pruning

monotonic grid?

(VP_{3,6}^{\text{hold} \star \text{meeting}})

(VP_{3,6}^{\text{hold} \star \text{talk}})

(VP_{3,6}^{\text{hold} \star \text{conference}})

<table>
<thead>
<tr>
<th></th>
<th>1.0</th>
<th>3.0</th>
<th>8.0</th>
</tr>
</thead>
<tbody>
<tr>
<td>1.0</td>
<td>2.0</td>
<td>4.0</td>
<td>9.0</td>
</tr>
<tr>
<td>1.1</td>
<td>2.1</td>
<td>4.1</td>
<td>9.1</td>
</tr>
<tr>
<td>3.5</td>
<td>4.5</td>
<td>6.5</td>
<td>11.5</td>
</tr>
</tbody>
</table>
non-monotonic grid due to LM combo costs

<table>
<thead>
<tr>
<th></th>
<th>1.0</th>
<th>3.0</th>
<th>8.0</th>
</tr>
</thead>
<tbody>
<tr>
<td>(VP&lt;sub&gt;3,6&lt;/sub&gt;&lt;sup&gt;held&lt;/sup&gt; * meeting)</td>
<td>1.0</td>
<td>2.0 + 0.5</td>
<td>4.0 + 5.0</td>
</tr>
<tr>
<td>(VP&lt;sub&gt;3,6&lt;/sub&gt;&lt;sup&gt;held&lt;/sup&gt; * talk)</td>
<td>1.1</td>
<td>2.1 + 0.3</td>
<td>4.1 + 5.4</td>
</tr>
<tr>
<td>(VP&lt;sub&gt;3,6&lt;/sub&gt;&lt;sup&gt;hold&lt;/sup&gt; * conference)</td>
<td>3.5</td>
<td>4.5 + 0.6</td>
<td>6.5 + 10.5</td>
</tr>
</tbody>
</table>
Cube Pruning

Huang and Chiang

non-monotonic grid due to LM combo costs

- \( (VP_{3,6}^{\text{with meeting}}) \)
- \( (VP_{3,6}^{\text{held talk}}) \)
- \( (VP_{3,6}^{\text{hold conference}}) \)

<table>
<thead>
<tr>
<th></th>
<th>1.0</th>
<th>3.0</th>
<th>8.0</th>
</tr>
</thead>
<tbody>
<tr>
<td>1.0</td>
<td>2.0</td>
<td>+</td>
<td>2.5</td>
</tr>
<tr>
<td>1.0</td>
<td>2.0</td>
<td>+</td>
<td>2.5</td>
</tr>
<tr>
<td>1.1</td>
<td>2.1</td>
<td>+</td>
<td>2.5</td>
</tr>
<tr>
<td>3.5</td>
<td>4.5</td>
<td>+</td>
<td>4.5</td>
</tr>
</tbody>
</table>

V (held ⋆ meeting)

PP (with ⋆ Sharon)

VP (held ⋆ talk)

PP (1,3

VP (1,6

PP (1,3

PP (1,3

PP (1,6

Forest Rescoring 13
Cube Pruning

<table>
<thead>
<tr>
<th></th>
<th>1.0</th>
<th>3.0</th>
<th>8.0</th>
</tr>
</thead>
<tbody>
<tr>
<td>(VP\textsuperscript{held} \ast \text{meeting}) \text{3,6}</td>
<td>1.0</td>
<td>2.5</td>
<td>9.0</td>
</tr>
<tr>
<td>(VP\textsuperscript{held} \ast \text{talk}) \text{3,6}</td>
<td>1.1</td>
<td>2.4</td>
<td>9.5</td>
</tr>
<tr>
<td>(VP\textsuperscript{hold} \ast \text{conference}) \text{3,6}</td>
<td>3.5</td>
<td>5.1</td>
<td>17.0</td>
</tr>
</tbody>
</table>

non-monotonic grid due to LM combo costs

Huang and Chiang
**Cube Pruning**

**k-best parsing**  
(Huang and Chiang, 2005)

- a priority queue of candidates
- extract the best candidate

<table>
<thead>
<tr>
<th>(VP_3,6_held * meeting)</th>
<th>1.0</th>
<th>2.5</th>
<th>9.0</th>
<th>9.5</th>
</tr>
</thead>
<tbody>
<tr>
<td>(VP_3,6_held * talk)</td>
<td>1.1</td>
<td>2.4</td>
<td>9.5</td>
<td>9.4</td>
</tr>
<tr>
<td>(VP_3,6_hold * conference)</td>
<td>3.5</td>
<td>5.1</td>
<td>17.0</td>
<td>12.1</td>
</tr>
</tbody>
</table>

Huang and Chiang
### Cube Pruning

**$k$-best parsing**  
(Huang and Chiang, 2005)

- a priority queue of candidates  
- extract the best candidate  
- push the two successors

<table>
<thead>
<tr>
<th></th>
<th>PP with * Sharon</th>
<th>PP along * Sharon</th>
<th>PP with * Shalong</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>(VP</strong>&lt;sub&gt;3,6&lt;/sub&gt; <strong>held ⋆ meeting)</strong></td>
<td>1.0</td>
<td>3.0</td>
<td>8.0</td>
</tr>
<tr>
<td><strong>(VP</strong>&lt;sub&gt;3,6&lt;/sub&gt; <strong>held ⋆ talk)</strong></td>
<td>1.1</td>
<td>2.4</td>
<td>9.5</td>
</tr>
<tr>
<td><strong>(VP</strong>&lt;sub&gt;3,6&lt;/sub&gt; <strong>hold ⋆ conference)</strong></td>
<td>3.5</td>
<td>5.1</td>
<td>17.0</td>
</tr>
</tbody>
</table>

Huang and Chiang  

Forest Rescoring  16
Cube Pruning

\(k\)-best parsing
(Huang and Chiang, 2005)

- a priority queue of candidates
- extract the best candidate
- push the two successors

<table>
<thead>
<tr>
<th></th>
<th>PP with (\star) Sharon</th>
<th>PP along (\star) Sharon</th>
<th>PP with (\star) Shalong</th>
</tr>
</thead>
<tbody>
<tr>
<td>(\text{VP}_{3,6}^{\text{hold} \star \text{meeting}})</td>
<td>1.0</td>
<td>3.0</td>
<td>8.0</td>
</tr>
<tr>
<td></td>
<td>1.0</td>
<td>2.5</td>
<td>9.0</td>
</tr>
<tr>
<td></td>
<td>2.4</td>
<td>9.5</td>
<td>9.4</td>
</tr>
<tr>
<td>(\text{VP}_{3,6}^{\text{hold} \star \text{talk}})</td>
<td>1.1</td>
<td>2.4</td>
<td>9.5</td>
</tr>
<tr>
<td>(\text{VP}_{3,6}^{\text{hold} \star \text{conference}})</td>
<td>3.5</td>
<td>5.1</td>
<td>17.0</td>
</tr>
<tr>
<td></td>
<td>5.1</td>
<td>17.0</td>
<td>12.1</td>
</tr>
</tbody>
</table>
Cube pruning

- Widely used for phrase-based and syntax-based MT
- May be applied in conjunction with a bottom-up decoder, or as a second “rescoring” pass
- Nodes may also be grouped together (for example, all nodes corresponding to a certain source span)
- Requirement for topological ordering means translation hypergraph may not have cycles
## LM Integration

<table>
<thead>
<tr>
<th>Method</th>
<th>Settings</th>
<th>Time</th>
<th>BLEU</th>
</tr>
</thead>
<tbody>
<tr>
<td>rescore</td>
<td>$k = 10^4$</td>
<td>16</td>
<td>33.31</td>
</tr>
<tr>
<td>rescore</td>
<td>$k = 10^5$</td>
<td>139</td>
<td>33.33</td>
</tr>
<tr>
<td>intersect*</td>
<td></td>
<td>1455</td>
<td>37.09</td>
</tr>
<tr>
<td>cube prune</td>
<td>$\varepsilon = 0$</td>
<td>23</td>
<td>36.14</td>
</tr>
<tr>
<td>cube prune</td>
<td>$\varepsilon = 0.1$</td>
<td>35</td>
<td>36.77</td>
</tr>
<tr>
<td>cube prune</td>
<td>$\varepsilon = 0.2$</td>
<td>111</td>
<td>36.91</td>
</tr>
</tbody>
</table>