Weblio Pre-reordering SMT System

Zhongyuan Zhu
(People call me Raphael Shu)
@raphaelshu
Overview of pre-reordering systems

- Reorder input text before translation

John hits a ball

John va_nsubj a ball va_obj hits

ジョンはボールを打った
Approaches of pre-reordering

• Syntactic pre-reordering with parse trees
  - Rule-based
    - Head-finalization (Isozaki et al., 2010)
  - Supervised learning with word alignments
    - Automatically learning Rewrite Patterns (Xia and McCord, 2004)

• Syntactic pre-reordering without parse tree
  - LADER (Neubig et al., 2012)
Pre-reordering model in our system
Overview of our pre-reordering system

Input

- $f$

CFG

CFG Parse Tree

Tree Restructuring

- DEP

Dependency Parse Tree

Reordering

- HRCFG

Head-restructured CFG Parse Tree

Pre-reordering Result

- $f'$
Head-restructured CFG Parse Tree

• Problem of CFG parse tree
  - Hard to capture long-distance reordering patterns

• Problem of Dependency parse tree
  - Fully lexicalized parse tree leads to a sparse reordering model
Head-restructured CFG Parse Tree

- Our approach
  - Restructure a CFG parse tree to inject head information into it

Head word is always lexicalized

CFG Parse Tree

Dependency Parse Tree

Head-restructured CFG Parse Tree (HRCFG)
Learning reordering model based on LM

- Extract tag sequences in golden order

Head-restructured CFG parse tree

- Train a language model on reordered tag sequences
Finding golden order with word alignments

- Given a bilingual sentence pair, source-side parse tree and word alignments, the golden order of a node layer is defined as

For nodes \((n_1, n_2, \ldots, n_k)\)

**Initial order:**

\[ o_0 = (1, 2, \ldots, k) \]

**Golden order:**

\[ \hat{o} = (a_1, a_2, \ldots, a_k) \]

- Average position (Ranked) \(a_1 = 1\), \(a_3 = 2\), \(a_2 = 3\)
Reordering a input parse tree

1. List all possible orders for a treelet

2. Score them with language model

3. Select the best order to adjust the treelet

- nsubj dobj hits
- dobj nsubj hits
- hits nsubj dobj
- hits dobj nsubj
- dobj hits nsubj
- nsubj hits dobj

-5.2
-6.1
-12.3
-19.5
-20.5
-22.7
**N-best reordering**

All 12 possible combinations here

Selected N-best results by accumulated scores (Cube Pruning is applied in the practice)

Reordered treelets with LM scores
Experiments
In-house experiments

<table>
<thead>
<tr>
<th></th>
<th>BLEU</th>
<th>RIBES</th>
</tr>
</thead>
<tbody>
<tr>
<td>1-best parse + 1 best reorder</td>
<td>34.46</td>
<td>0.7817</td>
</tr>
<tr>
<td>$N$-best parse + 1 best reorder</td>
<td>34.80</td>
<td>0.7851</td>
</tr>
<tr>
<td>1-best parse + $N$-best reorder</td>
<td>34.90</td>
<td>0.7857</td>
</tr>
<tr>
<td>$N$-best parse + $N$-best reorder</td>
<td>35.10</td>
<td>0.7887</td>
</tr>
</tbody>
</table>

- For “$N$-best reorder”, 10 candidate reordering results are considered.
- For “$N$-best parse”, 30 candidate parse trees are considered.
- We select the final translation by the sum of translation score (given by decoder) and the score of pre-reordering.
**N-best reordering & N-best parse tree inputs**

- Incorporating multiple reordering results and parse trees benefits automatic scores.
## Official evaluation results

<table>
<thead>
<tr>
<th>Method</th>
<th>BLEU</th>
<th>RIBES</th>
<th>HUMAN</th>
</tr>
</thead>
<tbody>
<tr>
<td>N-best reorder</td>
<td>34.87</td>
<td>0.7869</td>
<td>+43.25</td>
</tr>
<tr>
<td>N-best reorder + N-best parse</td>
<td>35.04</td>
<td>0.7900</td>
<td>+36.00</td>
</tr>
<tr>
<td>BASELINE PBMT</td>
<td>29.80</td>
<td>0.6919</td>
<td>0.00</td>
</tr>
</tbody>
</table>
Official evaluation results

**EJ BLEU score**

**EJ RIBES score**

**EJ HUMAN score**
Effect of pre-ordering

• Identical ordered sentences increases to 15%
Example of pre-reordering

Original input
the improvement of the life is a large problem of the practical application.

Restructured parse tree

Reordered input
the life of the improvement va_nsubjpass the practical application of a large problem is .

Reference
寿命 の 向上 が 実用 化 の 大きな 課題 で あ る。
Review

• Language model is just a quick solution to the reordering problem, sometimes it fails in simple cases.
  - Sparseness problem
• To gain more from forest input, it’s necessary to integrate it inside the pre-reordering model.
Online demonstrations

Head-structured CFG parse tree
http://raphael.uaca.com/demos/hdtree

Pre-reordering
http://raphael.uaca.com/demos/raphreorder
Thanks.