Learning to translate with source and target syntax

David Chiang, USC Information Sciences Institute

14 July 2010
Overview

- Using source and target syntax
- Why is it hard?
- How can we make it better?

- Let the model learn how much syntax to use
- The model does choose syntax, for improvements of +0.6–0.8 BLEU
reference: An official from Japan ’s science and technology ministry said , " We are highly encouraged by Abraham ’s comment .

Hiero: Officials of the Japanese ministry of education and science , " said Abraham speeches , we are deeply encouraged by .

string-to-tree: Japan ’s ministry of education , culture , sports , science and technology , " Abraham ’s statement , which is most encouraging , " the official said .
## Previous work

<table>
<thead>
<tr>
<th>Type</th>
<th>Method/Authors</th>
<th>Year(s)</th>
</tr>
</thead>
<tbody>
<tr>
<td>string-to-string</td>
<td>ITG (Wu 1997)</td>
<td></td>
</tr>
<tr>
<td>string-to-tree</td>
<td>Yamada &amp; Knight 2001</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Hiero (Chiang 2005)</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Huang et al 2006</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Y Liu et al 2006</td>
<td></td>
</tr>
</tbody>
</table>
1. Phrases
   ✤ respect word alignments
   ✤ are syntactic constituents on both sides

2. Phrase pairs form rules

3. Subtract phrases to form rules
1. Phrases
   - respect word alignments
   - are syntactic constituents on both sides

2. Phrase pairs form rules

3. Subtract phrases to form rules
STSG extraction

1. Phrases
   - respect word alignments
   - are syntactic constituents on both sides

2. Phrase pairs form rules

3. Subtract phrases to form rules
<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>phrase</td>
<td>23.66</td>
<td>23.86</td>
<td>30.18</td>
</tr>
<tr>
<td>STSG</td>
<td>20.21</td>
<td>24.71</td>
<td>22.23</td>
</tr>
</tbody>
</table>
Why is tree-to-tree hard?

- too few rules
  - a la bruja verde
  - the green witch
  - DT JJ NN NP

- too few derivations
  - NJR IN CD NP NN NNS
  - more than 20
  - check points
Why is tree-to-tree hard?

too few rules
台湾在两岸贸易中顺差 Taiwan's surplus in trade between the two shores
台湾
在两岸贸易中
顺差

Taiwan’s surplus
in trade between the two shores
Extracting more rules

binarize head-out

台湾  在两岸贸易中  顺差

台湾  在两岸贸易中  顺差
Extracting more rules

Tree-Sequence Substitution Grammar
(M. Zhang et al., 2008)

Syntax-Augmented Machine Translation (Venugopal & Zollmann)
Why is tree-to-tree hard?

too few rules

too few derivations

\[
\text{the} \quad \text{green} \quad \text{witch} \quad a \ la \ bruja \ verde
\]

\[
\text{NP} \quad \text{JJ} \quad \text{NN} \quad \text{QP} \quad \text{IN} \quad \text{CD} \quad \text{NP} \quad \text{NN} \quad \text{NNS} \quad \text{NP} \quad \text{NN} \quad \text{NNS} \quad \text{check} \quad \text{points}
\]
Why is tree-to-tree hard?

too few derivations

NP

QP

JJR IN CD

more than 20

NP

NN NNS

check points
Allow more derivations

- STSG: allow only matching substitutions
- Hiero-like: allow any substitutions
- Let the model learn to choose:
  - matching substitutions
  - mismatching substitutions
  - monotone phrase-based
Allow more derivations

fire subst:NP→NP
fire subst:match

fire subst:NNS→NP
fire subst:unmatch
Allow more derivations

Cross-lingual features

NP

NN

| 堵卡站 |

| check |

| points |

fire root: NP, NN+NNS

suggested by Adam Pauls
Allow more derivations

Hiero-like decoding

STSG decoding

fuzzy STSG decoding

\[
\begin{align*}
[X, i, j] & \quad [X, j+1, k] \\
& \quad [X, i, k] \\
& \quad [VP, i, j] & [NP, j+1, k] \\
& \quad [NP, i, k] \\
& \quad [A, i, j] & [B, j+1, k] \\
& \quad [NP, i, k] \\
\end{align*}
\]

\[X \rightarrow X \text{ 的 } X\]
## Experiments

<table>
<thead>
<tr>
<th></th>
<th>Chinese-English</th>
<th>Arabic-English</th>
</tr>
</thead>
<tbody>
<tr>
<td>parallel text</td>
<td>240M+260M</td>
<td>190+220M</td>
</tr>
<tr>
<td>language model</td>
<td></td>
<td>2G</td>
</tr>
<tr>
<td>parser (source)</td>
<td>800k</td>
<td>600k</td>
</tr>
<tr>
<td>parser (target)</td>
<td></td>
<td>2.1M</td>
</tr>
</tbody>
</table>
## Results

<table>
<thead>
<tr>
<th>extraction</th>
<th>Chinese-English</th>
<th>Arabic-English</th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>rules</td>
<td>feats</td>
<td>BLEU</td>
<td>rules</td>
</tr>
<tr>
<td>Hiero</td>
<td>440M</td>
<td>1k</td>
<td>23.7</td>
<td>790M</td>
</tr>
<tr>
<td>fuzzy STSG</td>
<td>50M</td>
<td>5k</td>
<td>23.9</td>
<td>38M</td>
</tr>
<tr>
<td>fuzzy STSG +binarize</td>
<td>64M</td>
<td>5k</td>
<td>24.3</td>
<td>40M</td>
</tr>
<tr>
<td>fuzzy STSG +SAMT</td>
<td>440M</td>
<td>160k</td>
<td>24.3</td>
<td>790M</td>
</tr>
</tbody>
</table>
Example tree-to-tree translation

An official from Japan's science and technology ministry said, "We are highly encouraged by Abraham's comment."

Hiero: Officials of the Japanese ministry of education and science, "said Abraham's speeches, we are deeply encouraged by.

string-to-tree: Japan's ministry of education, culture, sports, science and technology, "Abraham's statement, which is most encouraging," the official said.

Fuzzy STSG, binarize: Officials of the Japanese ministry of education, culture, sports, science and technology, said, "we are very encouraged by the speeches of Abraham."
Rule usage (Chinese-English)

<table>
<thead>
<tr>
<th></th>
<th>Hiero</th>
<th>Fuzzy STSG+SAMT</th>
</tr>
</thead>
<tbody>
<tr>
<td>Chinese side</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Match</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Mismatch</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Glue</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th></th>
<th>Hiero</th>
<th>Fuzzy STSG+SAMT</th>
</tr>
</thead>
<tbody>
<tr>
<td>English side</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Match</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Mismatch</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Glue</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
Rule usage (Arabic-English)

- Hiero
- Fuzzy STSG+SAMT

Arabic side

English side

- Match
- Mismatch
- Glue
Conclusions

Why is tree-to-tree translation hard?

- Too few rules
- Too few derivations

How can we make it better?

- Extract more rules: even simple binarization works
- Allow more derivations: let model learn how much syntax to use