Improving Trees and Alignments for Syntax-Based Machine Translation

Kevin Knight

USC/Information Sciences Institute

joint work with Steven DeNeefe, Daniel Marcu, Wei Wang, and Jonathan May

SRI, July 12, 2007
Syntactic Approaches to MT

• Use of syntactic information (noun, verb, etc) in the translation process:
  – Manually constructed rule-based systems
  – Statistical systems
    • Wu & Wong, 1998
    • Yamada & Knight, 2001-2002
    • Galley et al, 2004
  – Contrast with phrase-based statistical approaches
Phrase-Based Output

Gunman of police killed .

Decoder Hypothesis #1
Gunman of police attack.
Gunman by police killed.
Killed gunman by police.
Phrase-Based Output

Gunman killed the police.

Decoder Hypothesis #9,329

枪手 被 警方 击毙．
Phrase-Based Output

Gunman killed by police .

Problematic –

- Output lacks English auxiliary and determiner
- Re-ordering relies on luck, instead of on Chinese passive marker
The gunman killed by police.
Gunman by police shot .
The gunman was killed by police.
Why Might Syntax Help?

• Phrase-based MT output is “n-grammatical”, not grammatical
  – Every sentence needs a subject and a verb

• Re-ordering is poorly explained as “distortion” -- better explained as syntactic transformation
  – Arabic to English, VSO → SVO

• Function words have syntactic effects even if they are not themselves translated
Why Might Syntax Hurt?

- Less freedom to glue pieces of output together -- search space has fewer output strings
- Search space is more difficult to navigate
- Rule extraction from bilingual text has limitations

available phrase-based translations

this talk
Why Might Syntax Hurt?

- Less freedom to glue pieces of output together -- search space has fewer output strings
- Search space is more difficult to navigate
- Rule extraction from bilingual text has limitations

this talk
Why Might Syntax Hurt?

- Less freedom to glue pieces of output together -- search space has fewer output strings
- Search space is more difficult to navigate
- Rule extraction from bilingual text has limitations
Comparing Phrase-Based Extraction with Syntax-Based Extraction

• Quantitatively compare
  – A typical phrase-based bilingual extraction algorithm (ATS, Och & Ney 2004)
  – A typical syntax-based bilingual extraction algorithm (GHKM, Galley et al 2004)
  – These algorithms picked from two good-scoring NIST-06 systems

• Identify areas of improvement for syntax-based rule coverage
Phrase-Based and Syntax-Based Pattern Extraction

ATS [Och & Ney, 2004]

phrase pairs consistent with word alignment

GHKM [Galley et al 2004]

syntax transformation rules consistent with word alignment
ATS (Och & Ney, 2004)

**PHRASE PAIRS ACQUIRED:**

- felt
- felt obliged
- felt obliged to do
- obliged
- obliged to do
- do
- part
- part

- 我 有 责任 尽 一份 力

- I felt obliged to do my part
ATS (Och & Ney, 2004)

PHRASE PAIRS ACQUIRED:

felt → 有
felt obliged → 有 责任
felt obliged to do → 有 责任 尽
obliged → 责任
obliged to do → 责任 尽
do → 尽
part → 一份
part → 一份 力
ATS (Och & Ney, 2004)

PHRASE PAIRS ACQUIRED:

- felt
- felt obliged
- felt obliged to do
- obliged
- obliged to do
- do
- part

→ 有
→ 有 责任
→ 有 责任 尽
→ 责任
→ 责任 尽
→ 尽
→ 一份
→ 一份 力
I felt obliged to do my part.

我有责任尽一份力

PHRASE PAIRS ACQUIRED:

- felt $\rightarrow$ 有
- felt obliged $\rightarrow$ 有责任
- felt obliged to do $\rightarrow$ 有责任尽
- obliged $\rightarrow$ 责任
- obliged to do $\rightarrow$ 责任尽
- do $\rightarrow$ 尽
- part $\rightarrow$ 一份
- part $\rightarrow$ 一份力
i felt obliged to do my part

我有责任尽一份力

RULES ACQUIRED:

VBD(felt) → 有
VBN(obliged) → 责任
VP(x0:VBD
   VP-C(x1:VBN
      x2:SG-C) → x0 x1 x2
VP(VBD(felt)
   VP-C(VBN(obliged))
      x0:SG-C) → 有 责任 x0
S(x0:NP-C x1:VP) → x0 x1
GHKM (Galley et al, 2004)

RULES ACQUIRED:

- VBD(felt) → 有
- VBN(obliged) → 责任
- VP(x0:VBD)
  - VP-C(x1:VBN)
    - x2:SG-C) → x0 x1 x2
- VP(VBD(felt)
  - VP-C(VBN(obliged))
    - x0:SG-C) → 有 责任 x0
- S(x0:NP-C x1:VP) → x0 x1
GHKM (Galley et al, 2004)

RULES ACQUIRED:

- VBD(felt) → 有
- VBN(obliged) → 责任

\[
\begin{align*}
\text{VP}(x0: VBD) &\rightarrow x0 \\
\text{VP-C}(x1: VBN) &\rightarrow x0 \ x1 \ x2 \\
\text{VP}(\text{VBD}(\text{felt})) &\rightarrow \text{有 责任} \ x0 \\
\text{VP-C}(\text{VBN(obliger)}) &\rightarrow \text{有 责任} \ x0 \\
\text{S}(x0: \text{NP-C} x1: \text{VP}) &\rightarrow x0 \ x1
\end{align*}
\]
GHKM (Galley et al., 2004)

Rules Acquired:

- VBD(felt) $\rightarrow$ 有
- VBN(obliged) $\rightarrow$ 责任
- VP(x0:VBD, x1:VBN, x2:SG-C) $\rightarrow$ x0 x1 x2
- VP(VBD(felt), VBN(obliged)) $\rightarrow$ 有 责任 x0
- S(x0:NP-C, x1:VP) $\rightarrow$ x0 x1
GHKM (Galley et al, 2004)

RULES ACQUIRED:

VBD(felt)        →  有
VBN(obliged)     →  责任
VP(x0:VBD
  VP-C(x1:VBN
    x2:SG-C) → x0 x1 x2

VP(VBD(felt)
  VP-C(VBN(obliged)))
    x0:SG-C) → 有 责任 x0

S(x0:NP-C x1:VP) → x0 x1
i felt obliged to do my part

我 有 责任 尽 一份 力

RULES ACQUIRED:

VBD(felt) → 有
VBN(obliged) → 责任
VP(x0:VBD VP-C(x1:VBN x2:SG-C) → x0 x1 x2
VP(VBD(felt) VP-C(VBN(obliged))) x0:SG-C) → 有 责任 x0
S(x0:NP-C x1:VP) → x0 x1
GHKM (Galley et al, 2004)

RULES ACQUIRED:

VBD(felt) → 有
VBN(obliged) → 责任
VP(x0:VBD
  VP-C(x1:VBN
    x2:SG-C)) → x0 x1 x2

VP(VBD(felt)
  VP-C(VBN(obliged))
    x0:SG-C) → 有 责任 x0

S(x0:NP-C x1:VP) → x0 x1

minimal rules tile the tree/string/alignment triple.

composed rules are made by combining those tiles.
GHKM Syntax Rules

Phrasal Translation

\[ \text{VP} \rightarrow \text{está, cantando} \]
\[ \text{VBZ} \quad \text{VBG} \]
\[ \text{is} \quad \text{singing} \]

Non-constituent Phrases

\[ \text{S} \rightarrow \text{hay, NP} \]
\[ \text{PRO} \quad \text{VP} \]
\[ \text{there} \quad \text{VB} \quad \text{NP} \]
\[ \text{are} \]

Non-contiguous Phrases

\[ \text{VP} \rightarrow \text{poner, NP} \]
\[ \text{VB} \quad \text{NP} \quad \text{PRT} \]
\[ \text{put} \quad \text{on} \]

Context-Sensitive
Word Insertion

\[ \text{NPB} \rightarrow \text{NNS} \]
\[ \text{DT} \quad \text{NNS} \]
\[ \text{the} \]

Multilevel Re-Ordering

\[ \text{S} \rightarrow \text{VB, NP1, NP2} \]
\[ \text{NP1} \quad \text{VP} \]
\[ \text{VB} \quad \text{NP2} \]

Lexicalized Re-Ordering

\[ \text{NP} \rightarrow \text{NP1, the, NP2} \]
\[ \text{NP2} \quad \text{PP} \]
\[ \text{of} \]
GHKM Syntax Rules

Phrasal Translation

\[
\text{VP} \rightarrow \text{está, cantando}
\]

\[
\text{VBZ} \quad \text{VBG} \\
\text{is} \quad \text{singing}
\]

Non-constituent Phrases

\[
\text{S} \rightarrow \text{hay, NP}
\]

\[
\text{PRO} \rightarrow \text{VP}
\]

\[
\text{there} \quad \text{VB} \quad \text{NP}
\]

\[
\text{are}
\]

Non-contiguous Phrases

\[
\text{VP} \rightarrow \text{poner, NP}
\]

\[
\text{VB} \quad \text{NP} \quad \text{PRT}
\]

\[
\text{put} \quad \text{on}
\]

Context-Sensitive Word Insertion

\[
\text{NPB} \rightarrow \text{NNS}
\]

\[
\text{DT} \quad \text{NNS} \\
\text{the}
\]

Multilevel Re-Ordering

\[
\text{S} \rightarrow \text{VB, NP1, NP2}
\]

\[
\text{NP1} \quad \text{VP} \\
\text{VB} \quad \text{NP2}
\]

Lexicalized Re-Ordering

\[
\text{NP} \rightarrow \text{NP1, of, NP2}
\]

\[
\text{NP2} \quad \text{PP} \\
\text{P} \quad \text{NP1}
\]

\[
of
\]
ATS and GHKM Methods Do Not Coincide

- GHKM Phrase Pairs Relevant to NIST-02
  - 43k
  - GHKM has no built-in phrase size limit -- ATS does.
  - GHKM pulls unaligned English words into phrases.
  - GHKM only gets \textit{minimal} rules to explain each segment pair.
  - GHKM forced to incorporate unaligned English words into phrases.
  - GHKM misses phrases due to parse failures.
- ATS Phrase Pairs Relevant to NIST-02
  - 161k
  - GHKM forced to incorporate some unaligned foreign words into phrases.
  - GHKM phrases come with applicability conditions.
ATS and GHKM Methods Overlap

GHKM Phrase Pairs Relevant to NIST-02

ATS Phrase Pairs actually used in 1-best decodings of NIST-02 (1,994 = 2 per sentence).

1,994

GHKM phrases come with applicability conditions.

GHKM only gets minimal rules to explain each segment pair.

GHKM forced to incorporate unaligned English words into phrases.

GHKM forced to incorporate some unaligned foreign words into phrases.

GHKM misses phrases due to parse failures.
ATS and GHKM Methods Overlap

**GOAL: REDUCE THIS NUMBER**

- GHKM Phrase Pairs Relevant to NIST-02
- ATS Phrase Pairs actually used in 1-best decodings of NIST-02 (1,994 = 2 per sentence).

GHKM phrases come with applicability conditions.
GHKM only gets *minimal* rules to explain each segment pair.
GHKM forced to incorporate unaligned English words into phrases.
GHKM forced to incorporate some unaligned foreign words into phrases.
GHKM misses phrases due to parse failures.

1,994
Four Ideas for Improving Syntax-Based Rule Extraction

- Acquire larger rules
  Composed rules (Galley et al, 06)
  Phrasal rules (Marcu et al, 06)
- Acquire more general rules
  Re-structure English trees (Wang et al, 07)
  Re-align tree/string pairs (May & Knight, 07)
Larger, Composed Rules

Minimal GHKM Rules:

- $B(\text{e}1 \text{ e}2) \rightarrow c1 \ c2$
- $C(\text{e}3) \rightarrow c3$
- $A(\text{x}0:B \ \text{x}1:C) \rightarrow \text{x}0 \ \text{x}1$

Additional Composed Rules:

- $A(B(\text{e}1 \text{ e}2) \ \text{x}0:C) \rightarrow c1 \ c2 \ \text{x}0$
- $A(\text{x}0:B \ C(\text{e}3)) \rightarrow \text{x}0 \ \text{c}3$
- $A(B(\text{e}1 \text{ e}2) \ C(\text{e}3)) \rightarrow c1 \ c2 \ c3$

“big phrasal rule”
Larger, Composed Rules

Minimal GHKM Rules:

\[ B(e1 \ e2) \rightarrow c1 \ c2 \]
\[ C(e3) \rightarrow c3 \]
\[ A(x0:B \ x1:C) \rightarrow x0 \ x1 \]

Additional Composed Rules:

\[ A(B(e1 \ e2) \ x0:C) \rightarrow c1 \ c2 \ x0 \]
\[ A(x0:B \ C(e3)) \rightarrow x0 \ c3 \]
\[ A(B(e1 \ e2) \ C(e3)) \rightarrow c1 \ c2 \ c3 \]

“big phrasal rule”
Larger, Composed Rules

Minimal GHKM Rules:

\[ B(e_1 e_2) \rightarrow c_1 c_2 \]
\[ C(e_3) \rightarrow c_3 \]
\[ A(x_0:B x_1:C) \rightarrow x_0 x_1 \]

Additional Composed Rules:

\[ A(B(e_1 e_2) x_0:C) \rightarrow c_1 c_2 x_0 \]
\[ A(x_0:B C(e_3)) \rightarrow x_0 c_3 \]
\[ A(B(e_1 e_2) C(e_3)) \rightarrow c_1 c_2 c_3 \]

“big phrasal rule”
Larger, Composed Rules

Minimal GHKM Rules:

\[ B(e_1 e_2) \rightarrow c_1 c_2 \]
\[ C(e_3) \rightarrow c_3 \]
\[ A(x_0:B \ x_1:C) \rightarrow x_0 x_1 \]

Additional Composed Rules:

\[ A(B(e_1 e_2) \ x_0:C) \rightarrow c_1 c_2 x_0 \]
\[ A(x_0:B \ C(e_3)) \rightarrow x_0 c_3 \]
\[ A(B(e_1 e_2) \ C(e_3)) \rightarrow c_1 c_2 c_3 \]

“big phrasal rule”
## Larger, Composed Rules

<table>
<thead>
<tr>
<th>Composed limit (internal nodes in composed rule)</th>
<th># of rules acquired</th>
<th>Unacquired phrase pairs used in ATS 1-best decodings</th>
</tr>
</thead>
<tbody>
<tr>
<td>0 = minimal</td>
<td>2.5m</td>
<td>1994</td>
</tr>
<tr>
<td>2</td>
<td>12.4m</td>
<td>1478</td>
</tr>
<tr>
<td>3</td>
<td>26.9m</td>
<td>1096</td>
</tr>
<tr>
<td>4</td>
<td>55.8m</td>
<td>900</td>
</tr>
</tbody>
</table>
“Phrasal” Syntax Rules

- SPMT Model 1 (Marcu et al 2006)
  - consider each foreign phrase up to length L
  - extract smallest possible syntax rule that does not violate alignments

<table>
<thead>
<tr>
<th>Method</th>
<th>Unacquired ATS Phrase Pairs</th>
</tr>
</thead>
<tbody>
<tr>
<td>Minimal</td>
<td>1994</td>
</tr>
<tr>
<td>Composed 4</td>
<td>900</td>
</tr>
<tr>
<td><strong>SPMT M1</strong></td>
<td><strong>676</strong></td>
</tr>
<tr>
<td><strong>Both</strong></td>
<td><strong>663</strong></td>
</tr>
</tbody>
</table>
Restructuring English Training Trees
# Restructuring English Training Trees

<table>
<thead>
<tr>
<th>Method</th>
<th>Unacquired ATS Phrase Pairs</th>
</tr>
</thead>
<tbody>
<tr>
<td>Minimal</td>
<td>1994</td>
</tr>
<tr>
<td>+ Composed 4</td>
<td>900</td>
</tr>
<tr>
<td>+ SPMT M1</td>
<td>663</td>
</tr>
<tr>
<td>+ Restructuring</td>
<td>458</td>
</tr>
</tbody>
</table>
## Effects of Coverage Improvements on Syntax-Based MT Accuracy

<table>
<thead>
<tr>
<th></th>
<th>Chinese/English</th>
<th>Arabic/English</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Dev-02</td>
<td>Test-03</td>
</tr>
<tr>
<td>ATS</td>
<td>36.00</td>
<td>34.31</td>
</tr>
<tr>
<td>GHKM minimal</td>
<td>39.11</td>
<td>38.85</td>
</tr>
<tr>
<td>GHKM composed 2</td>
<td>41.59</td>
<td>40.90</td>
</tr>
<tr>
<td>GHKM composed 3</td>
<td>42.28</td>
<td>41.62</td>
</tr>
<tr>
<td>GHKM composed 4</td>
<td>42.63</td>
<td>41.82</td>
</tr>
<tr>
<td>GHKM minimal + SPMT</td>
<td>41.01</td>
<td>40.34</td>
</tr>
<tr>
<td>GHKM composed 4 + SPMT</td>
<td>43.30</td>
<td>42.17</td>
</tr>
<tr>
<td>+ Left binarization of etrees</td>
<td>43.45</td>
<td>42.41</td>
</tr>
</tbody>
</table>

NIST Bleu r4n4
Can We Do Better?

- Improved binarization methods
- Improved word alignment of tree/string pairs
Why are Penn Treebank Trees Problematic?

维克多·切尔诺梅尔金及同事?

维克多·切尔诺梅尔金及同事
Why are Penn Treebank Trees Problematic?
Binarizing English Trees
Simple Binarizations

(1) unbinarized tree

(2) left-binarization

(3) right-/head-binarization

(4) left-binarization

(5) right-binarization

(6) left-binarization

(7) right-/head-binarization
Parallel Binarization
Parallel Binarization

维克多·切尔诺梅尔金
Forest-Based Rule Extraction

• Gets all minimal rules consistent with word alignment and some binarization
• Run EM algorithm to determine best binarization of each node in each tree
Binarization Using EM

e-tree

f-string,
alignment
Binarization Using EM

e-tree → parallel binarization → e-forest

f-string,
alignment
Binarization Using EM

e-tree → parallel binarization → e-forest → forest-based extraction of minimal rules

f-string, alignment → rules
Binarization Using EM

- e-tree
- parallel binarization
- e-forest
- forest-based extraction of minimal rules
- rules
- derivation forests
- EM

f-string, alignment
Binarization Using EM

1. e-tree → parallel binarization → e-forest
2. e-forest → forest-based extraction of minimal rules → rules → derivation forests
3. rules → derivation forests → EM
4. f-string, alignment
5. binarized e-tree → project e-tree → viterbi derivation for each example
Binarization Using EM

- e-tree → parallel binarization → e-forest
- forest-based extraction of minimal rules
  - rules
  - derivation forests

- f-string, alignment

- binarized e-tree
  - project e-tree
  - viterbi derivation for each example
  - EM

- composed rule extraction (Galley et al., 2006)
  - rules for decoding
Binarization Using EM

- e-tree
- parallel binarization
- e-forest
- forest-based extraction of minimal rules
- f-string, alignment
- rules
- derivation forests
- project e-tree
- viterbi derivation for each example
- EM
- composed rule extraction (Galley et al., 2006)
- rules for decoding
# Experimental Results

<table>
<thead>
<tr>
<th>Type of Binarization</th>
<th># of Rules Learned</th>
<th>Test Bleu (NIST-03)</th>
</tr>
</thead>
<tbody>
<tr>
<td>None</td>
<td>63.4m</td>
<td>36.94</td>
</tr>
<tr>
<td>Left</td>
<td>114.0m</td>
<td>37.47 (p=0.047)</td>
</tr>
<tr>
<td>Right</td>
<td>113.0m</td>
<td>37.49 (p=0.044)</td>
</tr>
<tr>
<td>Head</td>
<td>113.8m</td>
<td>37.54 (p=0.086)</td>
</tr>
<tr>
<td>EM</td>
<td>115.6m</td>
<td><strong>37.94 (p=0.0047)</strong></td>
</tr>
</tbody>
</table>
Tree binarized by EM training
Last Topic: Alignment

- GIZA++ string-based alignments
  - are errorful
  - don’t match our syntax-based MT system
- Would like to use the tree-based translation model to align data
Last Topic: Alignment

English trees
Foreign strings

GIZA++ → initial word alignments → GHKM syntax rule extraction → minimal rules

EM alignment (“Training Tree Transducers”, Graehl & Knight’04)

Viterbi derivations
→ Improved word alignments

Details in May & Knight, 07

Result: +0.5-1.0 Bleu

throw away GIZA alignments
Conclusions

• Phrase-based and syntax-based extraction algorithms have different coverage.
• Syntax-based coverage can be improved:
  – composed rules
  – phrasal rules
  – binarizing English trees with EM
  – re-aligning tree/string pairs with EM
• Improvements lead to better translation accuracy.
the end