Algorithms for NLP

Machine Translation III

Taylor Berg-Kirkpatrick – CMU

Slides: Dan Klein – UC Berkeley
Phrase-Based Translation Overview

**Input:** lo haré rápidamente.

**Translations:** l’ll do it quickly.

The decoder... tries different segmentations, translates phrase by phrase, and considers reorderings.

**Objective:**

$$\arg \max_e [P(f|e) \cdot P(e)]$$

$$\arg \max_e \left[ \prod_{\langle e,f \rangle} P(\bar{f}|\bar{e}) \cdot \prod_{i=1}^{\text{|e|}} P(e_i|e_{i-1}, e_{i-2}) \right]$$
Phrase-Based Decoding

Decoder design is important: [Koehn et al. 03]
<table>
<thead>
<tr>
<th>Maria</th>
<th>no</th>
<th>dio</th>
<th>una</th>
<th>bofetada</th>
<th>a</th>
<th>la</th>
<th>bruja</th>
<th>verde</th>
</tr>
</thead>
<tbody>
<tr>
<td>Mary</td>
<td>not</td>
<td>give</td>
<td>a</td>
<td>slap</td>
<td>to</td>
<td>the</td>
<td>witch</td>
<td>green</td>
</tr>
<tr>
<td>did not</td>
<td>a slap</td>
<td>by</td>
<td>green witch</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>no</td>
<td>slap</td>
<td>to the</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>did not give</td>
<td>to</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>the</td>
<td>the witch</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
Monotonic Word Translation

- Cost is $LM \times TM$
- It’s an HMM?
  - $P(e|e_{-1}, e_{-2})$
  - $P(f|e)$
- State includes
  - Exposed English
  - Position in foreign
- Dynamic program loop?

```plaintext
for (fPosition in 1…|f|)
  for (eContext in allEContexts)
    for (eOption in translations[fPosition])
      score = scores[fPosition-1][eContext] * LM(eContext+eOption) * TM(eOption, fWord[fPosition])
      scores[fPosition][eContext[eContext][2]+eOption] = \max score
```

<table>
<thead>
<tr>
<th>Maria</th>
<th>no</th>
<th>dio</th>
<th>una</th>
<th>bofetada</th>
<th>a</th>
<th>la</th>
<th>bruja</th>
<th>verde</th>
</tr>
</thead>
<tbody>
<tr>
<td>Mary</td>
<td>not</td>
<td>give</td>
<td>a</td>
<td>slap</td>
<td>to</td>
<td>the</td>
<td>witch</td>
<td>green</td>
</tr>
<tr>
<td>did not</td>
<td>no</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

\[
P(e|e_{-1},e_{-2})
\]

\[
P(f|e)
\]

\[
\text{Exposed English}
\]

\[
\text{Position in foreign}
\]
Beam Decoding

- For real MT models, this kind of dynamic program is a disaster (why?)
- Standard solution is beam search: for each position, keep track of only the best k hypotheses

```plaintext
for (fPosition in 1...|f|)
  for (eContext in bestEContexts[fPosition])
    for (eOption in translations[fPosition])
      score = scores[fPosition-1][eContext] * LM(eContext+eOption) * TM(eOption, fWord[fPosition])
      bestEContexts.maybeAdd(eContext[2]+eOption, score)
```

- Still pretty slow... why?
- Useful trick: cube pruning (Chiang 2005)
### Phrase Translation

<table>
<thead>
<tr>
<th>Maria</th>
<th>no</th>
<th>dio</th>
<th>una</th>
<th>bofetada</th>
<th>a</th>
<th>la</th>
<th>bruja</th>
<th>verde</th>
</tr>
</thead>
</table>

- **If monotonic, almost an HMM; technically a semi-HMM**

  for (fPosition in 1…|f|)
  for (lastPosition < fPosition)
  for (eContext in eContexts)
   for (eOption in translations[fPosition])
     … combine hypothesis for (lastPosition ending in eContext) with eOption

- **If distortion... now what?**
Non-Monotonic Phrasal MT

```
e: Mary
f: *-------
p: .534

e: Mary did not
f: **--------
p: .122

e: Mary slap
f: *-***-----
p: .043

e: witch
f: --------*
p: .182
```

e:  
f:  
p: 1
Pruning: Beams + Forward Costs

Problem: easy partial analyses are cheaper

- Solution 1: use beams per foreign subset
- Solution 2: estimate forward costs (A*-like)
The Pharaoh Decoder

<table>
<thead>
<tr>
<th>Maria</th>
<th>no</th>
<th>dio</th>
<th>una</th>
<th>bofetada</th>
<th>a</th>
<th>la</th>
<th>bruja</th>
<th>verde</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>not</td>
<td>give</td>
<td>a</td>
<td>slap</td>
<td>to</td>
<td>the</td>
<td>witch</td>
<td>green</td>
</tr>
<tr>
<td>did not</td>
<td></td>
<td></td>
<td>a slap</td>
<td>by</td>
<td></td>
<td></td>
<td>green witch</td>
<td></td>
</tr>
<tr>
<td>no</td>
<td></td>
<td></td>
<td>a slap</td>
<td>to the</td>
<td></td>
<td></td>
<td>to</td>
<td></td>
</tr>
<tr>
<td>did not give</td>
<td></td>
<td></td>
<td></td>
<td>to</td>
<td>the</td>
<td>the witch</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Maria</th>
<th>no</th>
<th>dio</th>
<th>una</th>
<th>bofetada</th>
<th>a</th>
<th>la</th>
<th>bruja</th>
<th>verde</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Mary</th>
<th>did not</th>
<th>slap</th>
<th>the</th>
<th>green</th>
<th>witch</th>
</tr>
</thead>
</table>
Hypothesis Lattices

<table>
<thead>
<tr>
<th>Maria</th>
<th>no</th>
<th>dio</th>
<th>una</th>
<th>bofetada</th>
<th>a</th>
<th>la</th>
<th>bruja</th>
<th>verde</th>
</tr>
</thead>
<tbody>
<tr>
<td>Mary</td>
<td>not</td>
<td>give</td>
<td>a</td>
<td>slap</td>
<td>to</td>
<td>the</td>
<td>witch</td>
<td>green</td>
</tr>
<tr>
<td>did not</td>
<td></td>
<td>a slap</td>
<td>by</td>
<td>green witch</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>no</td>
<td>slap</td>
<td>to the</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>did not give</td>
<td>to</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>slap</td>
<td>the witch</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

```
 Joe               
p=0.092

Mary
p=1

p=0.534
did not give

p=0.164
give
```
Parameter Tuning
Counting Phrase Pairs

**Input:**

Gracias, lo haré de muy buen grado.
Thank you, I shall do so gladly.

**Gloss**

Thanks,
that
do [first; future]
of
very
good
degree.
A real word alignment
(GIZA++ Model 4 with
grow-diag-final combination)

Gracias,
lo
haré
de
muy
buen
grado.

Thank you, I shall do so gladly.

Gloss
Thanks,
that
do [first; future]
of
very
good
degree.
What Happens in Practice

A real word alignment
(GIZA++ Model 4 with grow-diag-final combination)

Gracias
, 
lo
haré
de
muy
buen
grado
.

Thank you , I shall do so gladly .

Gloss

Thanks
, that
do [first; future]
of very
good degree
What Happens in Practice

A real word alignment
(GIZA++ Model 4 with grow-diag-final combination)

Gracias
,  
lo
haré
de
muy
buen
grado
.

Thank you , I shall do so gladly .

Gloss
Thanks
,  
that
do [first; future]
of
very
good
degree
**Phrase Scoring**

\[
\phi_{new}(\bar{e}_j | \bar{f}_i) = \frac{c(f_i, \bar{e}_j)}{c(f_i)}
\]

- Learning weights has been tried, several times:
  - [Marcu and Wong, 02]
  - [DeNero et al, 06]
  - ... and others

- Seems not to work well, for a variety of partially understood reasons

- Main issue: big chunks get all the weight, obvious priors don’t help
  - Though, [DeNero et al 08]
Phrases do help

- But they don’t need to be long
- Why should this be?
\[
\phi(\bar{f}_i|\bar{e}_i) = \frac{\text{count}(\bar{f}_i, \bar{e}_i)}{\text{count}(\bar{e}_i)} p_w(\bar{f}_i|\bar{e}_i)
\]

\[
\begin{align*}
P_w(\bar{f} | \bar{e}, a) &= P_w(f_1f_2f_3|e_1e_2e_3, a) \\
&= w(f_1|e_1) \\
&\quad \times \frac{1}{2}(w(f_2|e_2) + w(f_2|e_3)) \\
&\quad \times w(f_3|\text{NULL})
\end{align*}
\]
Tuning for MT

- Features encapsulate lots of information
  - Basic MT systems have around 6 features
  - \( P(e|f), P(f|e) \), lexical weighting, language model

- How to tune feature weights?

- Idea 1: Use your favorite classifier
Why Tuning is Hard

- **Problem 1: There are latent variables**
  - Alignments and segmentations
  - Possibility: forced decoding (but it can go badly)

\[\begin{align*}
x & : \text{le parlement} & \text{adopte} & \text{la} & \text{résolution} & \text{légaliste} \\
h & : \\
y & : \text{parliament} & \text{has} & \text{adopted} & \text{the} & \text{resolution}
\end{align*}\]
Why Tuning is Hard

- **Problem 3: Computational constraints**
  - Discriminative training involves repeated decoding
  - Very slow! So people tune on sets much smaller than those used to build phrase tables
Minimum Error Rate Training

- **Standard method:** minimize BLEU directly (Och 03)
  - MERT is a discontinuous objective
  - Only works for max ~10 features, but works very well then
  - Here: k-best lists, but forest methods exist (Machery et al 08)
  - Recently, lots of alternatives being explored for more features
MERT

Model Score

BLEU Score

\( \theta \)
Syntactic Models
Translating with Tree Transducers

Input

lo haré de muy buen grado .

Output

Grammar
Translating with Tree Transducers

Input

lo haré de muy buen grado .

Output

Grammar

ADV → 〈 de muy buen grado ; gladly 〉
Translating with Tree Transducers

**Input**

| ADV | lo haré de muy buen grado |

**Output**

| ADV | I gladly |

**Grammar**

$$ADV \rightarrow \langle \text{de muy buen grado ; gladly} \rangle$$
Translating with Tree Transducers

Input

\[ S \rightarrow \langle \text{lo haré} \quad \text{ADV} \quad . \quad ; \quad \text{I will do it} \quad \text{ADV} \quad . \quad \rangle \]

\[ \text{ADV} \rightarrow \langle \text{de muy buen grado} \quad ; \quad \text{gladly} \quad \rangle \]
Translating with Tree Transducers

Input

```
S
<table>
<thead>
<tr>
<th>lo haré</th>
<th>ADV</th>
</tr>
</thead>
<tbody>
<tr>
<td>de muy buen grado .</td>
<td></td>
</tr>
</tbody>
</table>
```

Output

```
S
<table>
<thead>
<tr>
<th>ADV</th>
</tr>
</thead>
<tbody>
<tr>
<td>I will do it</td>
</tr>
<tr>
<td>gladly .</td>
</tr>
</tbody>
</table>
```

Grammar

```
S → ⟨ lo haré ADV . ; I will do it ADV . ⟩

ADV → ⟨ de muy buen grado ; gladly ⟩
```
Translating with Tree Transducers

Input

<table>
<thead>
<tr>
<th>S</th>
</tr>
</thead>
<tbody>
<tr>
<td>lo haré</td>
</tr>
<tr>
<td>de muy buen grado</td>
</tr>
</tbody>
</table>

Output

<table>
<thead>
<tr>
<th>S</th>
</tr>
</thead>
<tbody>
<tr>
<td>ADV</td>
</tr>
<tr>
<td>I</td>
</tr>
<tr>
<td>do</td>
</tr>
<tr>
<td>it</td>
</tr>
<tr>
<td>gladly</td>
</tr>
</tbody>
</table>

Grammar

```
S -> ⟨ lo haré ADV . ; I will do it ADV . ⟩
ADV -> ⟨ de muy buen grado ; gladly ⟩
```
Translating with Tree Transducers

<table>
<thead>
<tr>
<th>Input</th>
<th>Output</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>ADV</strong></td>
<td><strong>ADV</strong></td>
</tr>
<tr>
<td>lo haré</td>
<td>I</td>
</tr>
<tr>
<td>de muy buen grado .</td>
<td>gladly</td>
</tr>
</tbody>
</table>

**Grammar**

\[
S \rightarrow \langle lo haré ADV . ; I will do it ADV . \rangle
\]

\[
ADV \rightarrow \langle de muy buen grado ; gladly \rangle
\]
Translating with Tree Transducers

Input

\[ \text{lo haré de muy buen grado .} \]

Output

\[ \text{I gladly} \]

Grammar

\[ \text{VP} \rightarrow \langle \text{lo haré ADV ; will do it ADV} \rangle \]
\[ \text{S} \rightarrow \langle \text{lo haré ADV . ; I will do it ADV .} \rangle \]
\[ \text{ADV} \rightarrow \langle \text{de muy buen grado ; gladly} \rangle \]
Translating with Tree Transducers

**Input**

<table>
<thead>
<tr>
<th>VP</th>
<th>ADV</th>
</tr>
</thead>
<tbody>
<tr>
<td>lo haré</td>
<td>de muy buen grado</td>
</tr>
</tbody>
</table>

**Output**

VP

<table>
<thead>
<tr>
<th>ADV</th>
<th>will do it</th>
</tr>
</thead>
<tbody>
<tr>
<td>I</td>
<td>gladly</td>
</tr>
</tbody>
</table>

**Grammar**

\[
VP \rightarrow \langle \text{lo haré ADV ; will do it ADV} \rangle \\
S \rightarrow \langle \text{lo haré ADV . ; I will do it ADV .} \rangle \\
ADV \rightarrow \langle \text{de muy buen grado ; gladly} \rangle
\]
Translating with Tree Transducers

Input

<table>
<thead>
<tr>
<th>VP</th>
<th>ADV</th>
</tr>
</thead>
<tbody>
<tr>
<td>lo haré</td>
<td>de muy buen grado</td>
</tr>
</tbody>
</table>

Output

<table>
<thead>
<tr>
<th>VP</th>
<th>ADV</th>
</tr>
</thead>
<tbody>
<tr>
<td>will</td>
<td>do it</td>
</tr>
<tr>
<td>I</td>
<td>gladly</td>
</tr>
</tbody>
</table>

Grammar

\[ S \rightarrow \langle \text{VP} . ; \ I \ \text{VP} . \rangle \]
\[ \text{VP} \rightarrow \langle \text{lo haré} \ \text{ADV} ; \ \text{will do it} \ \text{ADV} \rangle \]
\[ S \rightarrow \langle \text{lo haré} \ \text{ADV} . ; \ I \ \text{will do it} \ \text{ADV} . \rangle \]
\[ \text{ADV} \rightarrow \langle \text{de muy buen grado} ; \ \text{gladly} \rangle \]
Translating with Tree Transducers

**Input**

```
S
   VP
|   |   |
|   |   |
| lo haré | de muy buen grado |
```

**Output**

```
S
   VP
    |   |
    |   |
    ADV l
    |   |
    |   |
   I will do it gladly
```

**Grammar**

```
S → ⟨ VP ; I VP ⟩
VP → ⟨ lo haré ADV ; will do it ADV ⟩
S → ⟨ lo haré ADV ; I will do it ADV ⟩
ADV → ⟨ de muy buen grado ; gladly ⟩
```
Translating with Tree Transducers

Grammar

\[
S \rightarrow \langle \text{VP . ; I VP .} \rangle \quad \text{OR} \quad S \rightarrow \langle \text{VP . ; you VP .} \rangle \\
VP \rightarrow \langle \text{lo haré ADV ; will do it ADV} \rangle \\
S \rightarrow \langle \text{lo haré ADV . ; I will do it ADV .} \rangle \\
ADV \rightarrow \langle \text{de muy buen grado ; gladly} \rangle
\]
Learning Grammars for Translation

Thank you, I will do it gladly.

Gracias, lo haré de muy buen grado.
Learning Grammars for Translation

S

S

VP

NP

PRP

NP

MD

VP

VP

VB

PRP

PRP

ADV

Thank you, I will do it gladly.

Gracias,

lo haré de muy buen grado.
Learning Grammars for Translation

Thank you, I will do it gladly.

Gracias,

lo haré de muy buen grado.

Grammar Rules

⟨haré ; will do⟩
Thank you, I will do it gladly.

Gracias,
lo haré de muy buen grado.
Learning Grammars for Translation

Grammar Rules

Gracias
,  
lo  
haré  
de  
muy  
buen

grado  
.
Learning Grammars for Translation

Thank you, I will do it gladly.

Gracias, lo haré de muy buen grado.

Grammar Rules:

\[
\text{VP} \rightarrow \text{lo haré de ... grado ; will do it gladly}
\]
Learning Grammars for Translation

Thank you, I will do it gladly.

Gracias, lo haré de muy buen grado.

Grammar Rules

\textbf{haré ; will do}

\textbf{lo haré de ... grado ; will do it gladly}

\textbf{VP}$\rightarrow$
Thank you, I will do it gladly.

Gracias, lo haré de muy buen grado.
Learning Grammars for Translation

Grammar Rules

\(<\text{haré ; will do}\>

VP \rightarrow

\(<\text{lo haré de ... grado ; will do it gladly}\>
Learning Grammars for Translation

Thank you, I will do it gladly.

Gracias, lo haré de muy buen grado.

Grammar Rules

\[
\begin{align*}
\langle \text{haré} \rangle & \rightarrow \langle \text{will do} \rangle \\
\langle \text{lo haré de ... grado} \rangle & \rightarrow \langle \text{will do it gladly} \rangle \\
\langle \text{lo haré ADV} \rangle & \rightarrow \langle \text{will do it ADV} \rangle
\end{align*}
\]
The Size of Tree Transducer Grammars

- Extracted a transducer grammar from a 220 million word bitext
- Relativized the grammar to each test sentence
- Kept all rules with at most 6 non-terminals
The Size of Tree Transducer Grammars

- Extracted a transducer grammar from a 220 million word bitext
- Relativized the grammar to each test sentence
- Kept all rules with at most 6 non-terminals

Rules matching an example 40-word sentence

The graph shows the distribution of rule counts based on the size of the source-side yield. The x-axis represents the size of the yield (1 to 10+), and the y-axis shows the rule count. The graph indicates that the majority of rules match sentences of size 5, with a smaller number of rules matching sentences of size 4 and 6, and a significant drop for larger yields.
The Size of Tree Transducer Grammars

Extracted a transducer grammar from a 220 million word bitext
Relativized the grammar to each test sentence
Kept all rules with at most 6 non-terminals

Rules matching an example 40-word sentence

\[ S \rightarrow NP \ VP ; NP \ VP \]

Size of the source-side yield:

- Rule Count
- 90,000
- 67,500
- 45,000
- 0

- 1
- 2
- 3
- 4
- 5
- 6
- 7
- 8
- 9
- 10+
The Size of Tree Transducer Grammars

- Extracted a transducer grammar from a 220 million word bitext
- Relativized the grammar to each test sentence
- Kept all rules with at most 6 non-terminals

Rules matching an example 40-word sentence

- $S \rightarrow NP$ no es ni $ADJP$ ni $ADJP$ ;
- $NP$ isn’t $ADJP$ or $ADJP$ .
- $S \rightarrow NP \; VP ; \; NP \; VP$

Size of the source-side yield
Syntactic Decoding
Tree Transducer Grammars

S

| No se olvide de subir un canto rodado en Colorado |

Synchronous Grammar

NNP \rightarrow \text{Colorado}  ; \text{ Colorado}

NN \rightarrow \text{canto rodado}  ; \text{ boulder}

S \rightarrow \text{No se olvide de subir un NN en NNP}  ; \text{ Don’t forget to climb a NN in NNP}

Output

S

<table>
<thead>
<tr>
<th>NN</th>
<th>NNP</th>
</tr>
</thead>
<tbody>
<tr>
<td>Don’t forget to climb a boulder in Colorado</td>
<td></td>
</tr>
</tbody>
</table>
CKY-style Bottom-up Parsing

For each span length:
CKY-style Bottom-up Parsing

For each span length:  

For each span [i,j]:
For each span length:

For each span \([i,j]\):

Apply all grammar rules to \([i,j]\)
CKY-style Bottom-up Parsing

For each span length:

For each span [i,j]:

Apply all grammar rules to [i,j]

Binary rule: $X \rightarrow Y \ Z$
CKY-style Bottom-up Parsing

For each span length:

For each span \([i,j]\):

Apply all grammar rules to \([i,j]\)

Binary rule: \(X \rightarrow Y \: Z\)

Split points: \(i < k < j\)

Operations: \(O(j - i)\)

Time scales with: Grammar constant
CKY-style Bottom-up Parsing

For each span length:

For each span [i,j]:

Apply all grammar rules to [i,j]

No se olvide de subir un canto rodado en Colorado
CKY-style Bottom-up Parsing

For each span length:

For each span [i,j]: Apply all grammar rules to [i,j]

\[ S \rightarrow \text{No se } \text{VB } \text{de subir un } \text{NN } \text{en } \text{NNP} \]

No se olvide de subir un canto rodado en Colorado
For each span length: 

For each span \([i,j]\): 

Apply all grammar rules to \([i,j]\)

\[ S \rightarrow \text{No se} \quad \text{VB} \quad \text{de subir un} \quad \text{NN} \quad \text{en} \quad \text{NNP} \]

\[ \text{No se olvide de subir un canto rodado en Colorado} \]
CKY-style Bottom-up Parsing

For each span length:

For each span [i,j]: Apply all grammar rules to [i,j]

\[ S \rightarrow \text{No se} \quad \text{VB} \quad \text{de subir un} \quad \text{NN} \quad \text{en} \quad \text{NNP} \]

\[ \text{No se olvide de subir un canto rodado en Colorado} \]
CKY-style Bottom-up Parsing

For each span length:

For each span [i,j]:

Apply all grammar rules to [i,j]

\[ S \rightarrow \text{No se } \textbf{VB} \text{ de subir un } \textbf{NN} \text{ en } \textbf{NNP} \]

\[ i \text{ No se } \textit{olvide} \text{ de subir un canto rodado en Colorado } j \]
CKY-style Bottom-up Parsing

For each span length: For each span [i,j]: Apply all grammar rules to [i,j]

$S \rightarrow \text{No se } VB \text{ de subir un } NN \text{ en } NNP$

i No se olvide de subir un canto rodado en Colorado j
CKY-style Bottom-up Parsing

For each span length:

For each span [i,j]:

Apply all grammar rules to [i,j]

$S \rightarrow \text{No se } \text{VB} \text{ de subir un } \text{NN} \text{ en } \text{NNP}$

$i \text{ No se olvide de subir un canto rodado en Colorado } j$
CKY-style Bottom-up Parsing

For each span length:

For each span $[i,j]$:

Apply all grammar rules to $[i,j]$

$S \rightarrow$ No se $\mathbf{VB}$ de subir un $\mathbf{NN}$ en $\mathbf{NNP}$

Many untransformed lexical rules can be applied in linear time
CKY-style Bottom-up Parsing

For each span length:

For each span [i,j]:

Apply all grammar rules to [i,j]

\[ S \rightarrow \text{No se } VP \ NP \ PP \]

\[ \text{No se olvide de subir un canto rodado en Colorado} \]
CKY-style Bottom-up Parsing

For each span length:

For each span \([i,j]\):

Apply all grammar rules to \([i,j]\)

\[ S \rightarrow \text{No se} \quad \text{VP} \quad \text{NP} \quad \text{PP} \]

\[ \text{No se olvide de subir un canto rodado en Colorado} \]
CKY-style Bottom-up Parsing

For each span length:

For each span [i,j]: Apply all grammar rules to [i,j]

S → No se VP NP PP

No se olvide de subir un canto rodado en Colorado
For each span length:

For each span [i,j]: Apply all grammar rules to [i,j]

\[ S \rightarrow \text{No se} \quad \text{VP} \quad \text{NP} \quad \text{PP} \]

\( i \quad \text{No se olvide de subir un canto rodado en Colorado} \quad j \)

**Problem:** Applying adjacent non-terminals is slow
Eliminating Non-terminal Sequences

Lexical Normal Form (LNF)

(a) lexical rules have at most one adjacent non-terminal
(b) all unlexicalized rules are binary.

Original rule: \( S \rightarrow \text{No se} \text{ VB VB un NN PP} \)

Transformed rules: \( S \rightarrow \text{No se} \text{ VB~VB un NN~PP} \)
\( \text{VB~VB} \rightarrow \text{VB VB} \)
\( \text{NN~PP} \rightarrow \text{NN PP} \)

Parsing stages:
• Lexical rules are applied by matching
• Unlexicalized rules are applied by iterating over split points
Flexible Syntax
Soft Syntactic MT: From Chiang 2010

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</th>
<th>Reference</th>
</tr>
</thead>
<tbody>
<tr>
<td>string-to-string</td>
<td>ITG (Wu 1997)</td>
<td>Hiero (Chiang 2005)</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Y Liu et al., 2009</td>
</tr>
</tbody>
</table>
Hiero Rules

\[ S \rightarrow \langle S_1 X_2, S_1 X_2 \rangle \]
\[ S \rightarrow \langle X_1, X_1 \rangle \]
\[ X \rightarrow \langle yu X_1 you X_2, have X_2 with X_1 \rangle \]
\[ X \rightarrow \langle X_1 de X_2, the X_2 that X_1 \rangle \]
\[ X \rightarrow \langle X_1 zhiyi, one of X_1 \rangle \]
\[ X \rightarrow \langle Aozhou, Australia \rangle \]
\[ X \rightarrow \langle shi, is \rangle \]
\[ X \rightarrow \langle shaoshu guojia, few countries \rangle \]
\[ X \rightarrow \langle bangjiao, diplomatic relations \rangle \]
\[ X \rightarrow \langle Bei Han, North Korea \rangle \]

From [Chiang et al, 2005]
STSG extraction

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
STSG extraction

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

2. Phrase pairs form rules

3. Subtract phrases to form rules
Why is tree-to-tree hard?

too few rules

```
DT the a la bruja verde
    JJ green
    NN witch
```

too few derivations

```
NP
    QP
    JJR more
    IN than
    CD 20
    NP
        NN check
        NNS points
```
Extracting more rules

binarize head-out
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 \textit{subst}:\texttt{NP} \rightarrow \texttt{NP}
fire \textit{subst}:match

fire \textit{subst}:\texttt{NNS} \rightarrow \texttt{NP}
fire \textit{subst}:unmatch
Allow more derivations

Hiero-like decoding

STSG decoding

fuzzy STSG decoding
## Results

<table>
<thead>
<tr>
<th>Extraction</th>
<th>Chinese-English</th>
<th>Arabic-English</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>rules</td>
<td>feats</td>
</tr>
<tr>
<td>Hiero</td>
<td>440M</td>
<td>1k</td>
</tr>
<tr>
<td>Fuzzy STSG</td>
<td>50M</td>
<td>5k</td>
</tr>
<tr>
<td>Fuzzy STSG +binarize</td>
<td>64M</td>
<td>5k</td>
</tr>
<tr>
<td>Fuzzy STSG +SAMT</td>
<td>440M</td>
<td>160k</td>
</tr>
</tbody>
</table>
Example tree-to-tree translation

日本文部科学省官员表示，"亚伯拉罕的发言，令我们深感鼓舞"

Japan MEXT official said, "Abraham’s comment make us deeply feel courage"

*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.

*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."
Officials of the Japanese Ministry of Education, Culture, Science and Sports were very encouraged by the speeches of Abraham.