More on Syntax in MT

Ondřej Bojar
Institute of Formal and Applied Linguistics
Faculty of Mathematics and Physics
Charles University, Prague

Thu Sep 12, 2013
Outline

- Refresher: Motivation to go beyond phrases.
- Constituency vs. dependency trees.
- Tree vs. linear context.
- Non-projectivity and why it matters in MT.
Refresher: Prove Google is Phrase-Based

Natáhnout bačkory.  Kick the bucket.  ✓
Refresher: Prove Google is Phrase-Based

Natáhnout bačkory.  Kick the bucket.  
Proč musel natáhnout bačkory?  Why did he kick the bucket?
Refresher: Prove Google is Phrase-Based

Word form variations:

Natáhnout bačkory.  Kick the bucket.  ✓
Proč musel natáhnout bačkory?  Why did he kick the bucket?  ✓
Proč natáhl bačkory?  Why stretched slippers?  ✗
Refresher: Prove Google is Phrase-Based

Word form variations:

Natáhnout bačkory. Kick the bucket. ✓
Proč musel natáhnout bačkory? Why did he kick the bucket? ✓
Proč natáhl bačkory? Why stretched slippers? ✗

Pumping words into phrases:

Jan s Marií se vzali. John and Mary were married. ✓
Refresher: Prove Google is Phrase-Based

Word form variations:

Natáhnout bačkory. Kick the bucket. ✓
Proč musel natáhnout bačkory? Why did he kick the bucket? ✓
Proč natáhl bačkory? Why stretched slippers? x

Pumping words into phrases:

Jan s Marií se vzali. John and Mary were married. ✓
Jan s Marií se včera vzali. John and Mary married yesterday. ✓
Refresher: Prove Google is Phrase-Based

Word form variations:

Natáhnout bačkory.  Kick the bucket.  ✓
Proč musel natáhnout bačkory?  Why did he kick the bucket?  ✓
Proč natáhl bačkory?  Why stretched slippers?  ×

Pumping words into phrases:

Jan s Marií se vzali.  John and Mary were married.  ✓
Jan s Marií se včera vzali.  John and Mary married yesterday.  ✓
Jan s Marií se včera v kostele vzali.  John and Mary are married in church yesterday.  ~
Refresher: Prove Google is Phrase-Based

Word form variations:

- Natáhnout bačkory.  Kick the bucket. ✓
- Proč musel natáhnout bačkory?  Why did he kick the bucket? ✓
- Proč natáhl bačkory?  Why stretched slippers? ×

Pumping words into phrases:

- Jan s Marií se vzali.  John and Mary were married. ✓
- Jan s Marií se včera vzali.  John and Mary married yesterday. ✓
- Jan s Marií se včera v kostele vzali.  John and Mary are married in church yesterday. ~
- Jan s Marií se včera v kostele svatého Ducha vzali.  John and Mary yesterday in the Church of the Holy Spirit took. ×
PBMT vs. RBMT

(Prove Systran is not phrase-based.)
PBMT vs. RBMT

(Prove Systran is not phrase-based.)

Stell dir das vor.
Google  Imagine that. ✓
Systran  Imagine. ✓
PBMT vs. RBMT

(Prove Systran is not phrase-based.)

Stell dir das vor.

Google  Imagine that. ✓
Systran  Imagine. ✓

Stell dir ein Haus vor.

Google  Imagine a house before. ×
Systran  Imagine a house. ✓
### PBMT vs. RBMT

(Prove Systran is not phrase-based.)

<table>
<thead>
<tr>
<th>German</th>
<th>Google</th>
<th>Systran</th>
</tr>
</thead>
<tbody>
<tr>
<td>Stell dir das vor.</td>
<td>Imagine that.√</td>
<td>Imagine.√</td>
</tr>
<tr>
<td>Google Imagine that.</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Systran Imagine.</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Stell dir ein Haus vor.</td>
<td>❌</td>
<td>✓</td>
</tr>
<tr>
<td>Google Imagine a house before.</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Systran Imagine a house.</td>
<td>✓</td>
<td></td>
</tr>
<tr>
<td>Stell dir ein kleines Haus vor.</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Google Imagine a small house in front.</td>
<td></td>
<td>❌</td>
</tr>
<tr>
<td>Systran Imagine a small house.</td>
<td></td>
<td>✓</td>
</tr>
</tbody>
</table>
PBMT vs. RBMT

(Prove Systran is not phrase-based.)

<table>
<thead>
<tr>
<th>German Text</th>
<th>Google Translation</th>
<th>Correct</th>
</tr>
</thead>
<tbody>
<tr>
<td>Stell dir das <strong>vor</strong>.</td>
<td>Imagine that.</td>
<td>✓</td>
</tr>
<tr>
<td>Google</td>
<td>Imagine.</td>
<td>✓</td>
</tr>
<tr>
<td>Systran</td>
<td>Imagine.</td>
<td>✓</td>
</tr>
<tr>
<td>Stell dir ein Haus <strong>vor</strong>.</td>
<td>Imagine a house <strong>before</strong>.</td>
<td>✗</td>
</tr>
<tr>
<td>Google</td>
<td>Imagine a house.</td>
<td>✓</td>
</tr>
<tr>
<td>Systran</td>
<td>Imagine a house.</td>
<td>✓</td>
</tr>
<tr>
<td>Stell dir ein kleines Haus <strong>vor</strong>.</td>
<td>Imagine a small house <strong>in front</strong>.</td>
<td>✗</td>
</tr>
<tr>
<td>Google</td>
<td>Imagine a small house.</td>
<td>✓</td>
</tr>
<tr>
<td>Systran</td>
<td>Imagine a small house.</td>
<td>✓</td>
</tr>
<tr>
<td>Stell dir ein kleines Haus mit vierzehn Fenster <strong>vor</strong>.</td>
<td>Imagine a small house with fourteen windows <strong>in front</strong>.</td>
<td>✗</td>
</tr>
<tr>
<td>Google</td>
<td>Imagine a small house with fourteen windows.</td>
<td>✓</td>
</tr>
<tr>
<td>Systran</td>
<td>Imagine a small house with fourteen windows.</td>
<td>✓</td>
</tr>
</tbody>
</table>
Constituency vs. Dependency

Constituency trees (CFG) represent only bracketing: which adjacent constituents are glued together.

Dependency trees represent which words depend on which. Usually, some agreement/conditioning along the edge.
What Dependency Trees Tell Us

Input: The grass around your house should be cut soon.

Google: Trávu kolem vašeho domu by se měl snížit brzy.

- Bad lexical choice for cut = sekat/snížit/krájet/řezat/…
  - Due to long-distance lexical dependency with grass.
  - One can “pump” many words in between.
  - Could be handled by full source-context (e.g. maxent) model.

- Bad case of tráva.
  - Depends on the chosen active/passive form:

<table>
<thead>
<tr>
<th>active =&gt; accusative</th>
<th>passive =&gt; nominative</th>
</tr>
</thead>
<tbody>
<tr>
<td>trávu ... by ste se/ měl posekat</td>
<td>tráva ... by se měla posekat</td>
</tr>
<tr>
<td>tráva ... by měla být posekána</td>
<td></td>
</tr>
</tbody>
</table>

Examples by Zdeněk Žabokrtský, Karel Oliva and others.
Tree vs. Linear Context

- Tree context (neighbours in the dependency tree):
  - is better at predicting lexical choice than $n$-grams.
  - often equals linear context:
    - Czech manual trees: 50% of edges link neighbours,
    - 80% of edges fit in a 4-gram.

- Phrase-based MT is a very good approximation.
- Hierarchical MT can even capture the dependency in one phrase:
  \[ X \rightarrow < \text{the grass } X \text{ should be cut}, \text{trávu } X \text{ byste měl posekat} > \]
“Crossing Brackets”

- Constituent outside its father’s span causes “crossing brackets.”
  - Linguists use “traces” (□) to represent this.
- Sometimes, this is not visible in the dependency tree:
  - There is no “history of bracketing”.
  - See Holan et al. (1998) for dependency trees including derivation history.
Non-Projectivity

= a gap in a subtree span, filled by a node higher in the tree. Ex. Dutch “cross-serial” dependencies, a non-projective tree with one gap caused by *saw* within the span of *swim*.

0 gaps = projective tree ⇒ representable in CFG.

Why Non-Projectivity Matters?

- CFGs cannot handle non-projective constructions:
  
  Imagine John **grass saw being cut**!

- No way to glue these crossing dependencies together:
  - Lexical choice:
    
    \[ X \rightarrow < \text{grass} X \text{ being cut}, \text{trávu} X \text{ sekat} > \]
  - Agreement in gender:
    
    \[ X \rightarrow < \text{John} X \text{ saw}, \text{Jan} X \text{ viděl} > \]
    \[ X \rightarrow < \text{Mary} X \text{ saw}, \text{Marie} X \text{ viděla} > \]

- Phrases can memorize **fixed** sequences containing:
  - the non-projective construction
  - and all the words in between! (⇒ extreme sparseness)
Is Non-Projectivity Severe?

Depends on the language.

In principle unlimited:

- Czech allows long gaps as well as many gaps in a tree.

In treebank data:

- 23% of Czech sentences contain a non-projectivity.
- 99.5% of Czech sentences are well nested with ≤ 1 gap.

In parallel data:

- ~3–15% English-Czech sents beyond ITG reordering.
Summary

- Limitations of phrase-based MT:
  - Little or no dependencies across phrases.
  - Practice: dependencies are often local enough.
- Limitations of hierarchical/constituency-based MT:
  - Non-projective constructions are bound to fail.

→ deep-syntactic (dependency) translation as a solution.
References
