Efficient solutions for word reordering in German-English phrase-based SMT

Arianna Bisazza & Marcello Federico – FBK (Italy)
Outline

• Why German-English?
• Why phrase-based SMT?
• Goal of this work
• Techniques to achieve it:
  1. early distortion cost
  2. word-after-word reordering pruning
• Experiments & discussion
Jedoch **konnten** sie Kinder in Teilen von Helmand und Kandahar im Süden aus Sicherheitsgrund **nicht erreichen**.

But they **could not reach** children in parts of Helmand and Kandahar in the south for security reasons.
Why German-English?

Jedoch *konnten* sie Kinder in Teilen von Helmand und Kandahar im Süden aus Sicherheitsgrund *nicht erreichen*.

But they *could not reach* children in parts of Helmand and Kandahar in the south for security reasons.

### German word order

- Discontinuous verb phrases, main verb far from inflected auxiliary or modal
- Verb-second order VS English SVO
- Clause-final verb in subordinate clauses

Long-range reordering of isolated words or short phrases is frequent and important for translation quality!
Why phrase-based SMT?

- Shallow modeling: learns direct correspondences between surface forms in two languages
- Versatile, cost-effective
- Wrh hierarchical SMT: smaller models, faster decoding, very competitive for translating between similar languages

Most popular framework in SMT production scenarios today

Problem: doesn’t handle well long-range reordering!

(cf. typical configurations use DL=6 up to 10)
Most popular framework in SMT production scenarios today

Problem: doesn’t handle well long-range reordering!
Most popular framework in SMT production scenarios today
Problem: doesn’t handle well long-range reordering!

Goal of this work

Improve handling of large reordering search spaces in PSMT.

How?
1. anticipate payment of distortion penalty for long backward jumps
2. dynamically prune unlikely long jumps before they are performed

Better translation quality and faster decoding at high distortion limits
How (1): Early Distortion Cost

[Moore & Quirk 2007]

**Standard**: pay jump cost *when* jumping

**Early**: accumulate cost gradually *before* jumping
How (1): Early Distortion Cost

[Moore & Quirk 2007]

Standard: pay jump cost \textit{when} jumping

\begin{itemize}
  \item [\textbullet{}\textbullet{}\textbullet{}\textbullet{}\textbullet{}\textbullet{}\textbullet{}\textbullet{}\textbullet{}\textbullet{}\textbullet{}\textbullet{}] \text{TotDisto}=0
  \item [\textbullet{}\textbullet{}\textbullet{}\textbullet{}\textbullet{}\textbullet{}\textbullet{}\textbullet{}\textbullet{}\textbullet{}\textbullet{}\textbullet{}] \text{1}
  \item [\textbullet{}\textbullet{}\textbullet{}\textbullet{}\textbullet{}\textbullet{}\textbullet{}\textbullet{}\textbullet{}\textbullet{}\textbullet{}\textbullet{}] \text{1}
  \item [\textbullet{}\textbullet{}\textbullet{}\textbullet{}\textbullet{}\textbullet{}\textbullet{}\textbullet{}\textbullet{}\textbullet{}\textbullet{}\textbullet{}] \text{1}
  \item [\textbullet{}\textbullet{}\textbullet{}\textbullet{}\textbullet{}\textbullet{}\textbullet{}\textbullet{}\textbullet{}\textbullet{}\textbullet{}\textbullet{}] \text{1}
  \item [\textbullet{}\textbullet{}\textbullet{}\textbullet{}\textbullet{}\textbullet{}\textbullet{}\textbullet{}\textbullet{}\textbullet{}\textbullet{}\textbullet{}] \text{1}
  \item [\textbullet{}\textbullet{}\textbullet{}\textbullet{}\textbullet{}\textbullet{}\textbullet{}\textbullet{}\textbullet{}\textbullet{}\textbullet{}\textbullet{}] \text{1}
  \item [\textbullet{}\textbullet{}\textbullet{}\textbullet{}\textbullet{}\textbullet{}\textbullet{}\textbullet{}\textbullet{}\textbullet{}\textbullet{}\textbullet{}] \text{1}
  \item [\textbullet{}\textbullet{}\textbullet{}\textbullet{}\textbullet{}\textbullet{}\textbullet{}\textbullet{}\textbullet{}\textbullet{}\textbullet{}\textbullet{}] \text{14}
  \item [\textbullet{}\textbullet{}\textbullet{}\textbullet{}\textbullet{}\textbullet{}\textbullet{}\textbullet{}\textbullet{}\textbullet{}\textbullet{}\textbullet{}] \text{14}
\end{itemize}

Early: accumulate cost gradually \textit{before} jumping

\begin{itemize}
  \item [\textbullet{}\textbullet{}\textbullet{}\textbullet{}\textbullet{}\textbullet{}\textbullet{}\textbullet{}\textbullet{}\textbullet{}\textbullet{}\textbullet{}] \text{TotDisto}=0
  \item [\textbullet{}\textbullet{}\textbullet{}\textbullet{}\textbullet{}\textbullet{}\textbullet{}\textbullet{}\textbullet{}\textbullet{}\textbullet{}\textbullet{}] \text{4}
  \item [\textbullet{}\textbullet{}\textbullet{}\textbullet{}\textbullet{}\textbullet{}\textbullet{}\textbullet{}\textbullet{}\textbullet{}\textbullet{}\textbullet{}] \text{6}
  \item [\textbullet{}\textbullet{}\textbullet{}\textbullet{}\textbullet{}\textbullet{}\textbullet{}\textbullet{}\textbullet{}\textbullet{}\textbullet{}\textbullet{}] \text{8}
  \item [\textbullet{}\textbullet{}\textbullet{}\textbullet{}\textbullet{}\textbullet{}\textbullet{}\textbullet{}\textbullet{}\textbullet{}\textbullet{}\textbullet{}] \text{10}
  \item [\textbullet{}\textbullet{}\textbullet{}\textbullet{}\textbullet{}\textbullet{}\textbullet{}\textbullet{}\textbullet{}\textbullet{}\textbullet{}\textbullet{}] \text{12}
  \item [\textbullet{}\textbullet{}\textbullet{}\textbullet{}\textbullet{}\textbullet{}\textbullet{}\textbullet{}\textbullet{}\textbullet{}\textbullet{}\textbullet{}] \text{14}
  \item [\textbullet{}\textbullet{}\textbullet{}\textbullet{}\textbullet{}\textbullet{}\textbullet{}\textbullet{}\textbullet{}\textbullet{}\textbullet{}\textbullet{}] \text{14}
\end{itemize}

Very important for handling long backward jumps

Implemented in Moses
• New reordering models are designed every year, but problem of long reordering is still unsolved

• Existing word reordering models are not perfect, but they are expected to guide search over huge search spaces

... then...

... let’s refine the reordering search space!
How (2): Word-after-word reordering pruning

Standard search: explore all jumps within fixed DL

Our method: only explore long reorderings that are likely according to the reordering model
How (2): Word-after-word reordering pruning

Standard search: explore all jumps within fixed DL

Our method: only explore long reorderings that are likely according to the reordering model
How (2): Word-after-word reordering pruning

Standard search: explore all jumps within fixed DL

Our method: only explore long reorderings that are likely according to the reordering model

“Safe zone” always explored
How (2): Word-after-word reordering pruning

Standard search: explore all jumps within fixed DL

Our method: only explore long reorderings that are likely according to the reordering model

Rationale:
- don’t waste time exploring unlikely long jumps
- less hypo’s in stack => less risk of search/model errors

“Safe zone” always explored

Reo. scores

DL=6

0.2

0.2

0.4

0.6

0.6

0.7

DL=6

ϑ=2
Reordering **model** ad hoc:

max-ent binary classifier predicting whether a given input word should be translated *right after* another

Binary **features**, extracted from local context of starting/landing positions using surface form, POS or chunk labels

\[ \text{POS}(w_i) = \text{adj} \land \text{POS}(w_j) = \text{noun} \]

\[ w_b = \text{‘jedoch’} \]

*(Model details & features in the TACL paper)*
Experiments

• Experiments on WMT-tests (09-11) using WMT-10 training data

• Systems based on Moses, include state-of-the-art hierarchical lexicalized reordering models [Koehn & al 05; Galley & Manning 08]

• Contrastive experiment with hierarchical SMT:
  - standard Moses configuration
  - decoding-time span constraint = 10 or 20

• Evaluation by:
  - BLEU for lexical match & local order
  - KRS Kendall Reordering Score for global order
    same as LRscore with $\alpha=1$ [Birch & al. 2010]
Exps (1): Early Distortion Cost

[Moore & Quirk 2007]

Large improvement in reordering under high DL, but loss is still there

Included in following exp’s
Exps (2): Word-after-word reordering pruning

- WaWprune: non-prunable zone of width $\delta=5$
- More metrics in the paper
Exps (2): Word-after-word reordering pruning

- **WaWprune**: non-prunable zone of width $\theta=5$
- More metrics in the paper
**What’s going on?**

Hypotheses created: scored by all models and added to stack

Long-range phrase-to-phrase jumps performed for every 100 sentences of the test

<table>
<thead>
<tr>
<th>System</th>
<th>DL</th>
<th>#hyp/sent</th>
<th>(#jumps/sent)$\times$100</th>
</tr>
</thead>
<tbody>
<tr>
<td>baseline</td>
<td>8</td>
<td>600K</td>
<td>90</td>
</tr>
<tr>
<td>baseline</td>
<td>18</td>
<td>1278K</td>
<td>88  61  48</td>
</tr>
<tr>
<td>+WaW r.prun.</td>
<td>18</td>
<td>364K</td>
<td>52  29  17</td>
</tr>
</tbody>
</table>
**SRC**  Jedoch *konnten* sie Kinder in Teilen von Helm. und Kand. im Süden aus Sicherheitsgründen *nicht erreichen*.

**REF**  But they *could not reach* children in parts of Helm. and Kand. in the south for security reasons.

**DL8**  However, they *were* children in parts of Helm. and Kand. in the south, for security reasons.

**DL18**  However, they *were* children in parts of Helm. and Kand. in the south *do not reach* for security reasons.

**+WaW**  However, they *could not reach* children in parts of Helm. and Kand. in the south for security reasons.

**H10**  However, they *were* children in parts of Helm. and Kand. in the south *not reach* for security reasons.

**H20**  However, they *were* children in parts of Helm. and Kand. in the south *not reach* for security reasons.
Conclusions

• Long-range reordering in PSMT can be made possible by:
  - using a better distortion cost function (Moore & Quirk 2007)
  - dynamically refining the reordering search space, i.e. only exploring long jumps that are “promising”

• Results:
  - long jumps captured, similar BLEU, higher KRS
  - faster decoding

• Narrowed the gap between PSMT and hiero, with faster decoding and smaller models

• Reordering pruning can be tried with other kinds of reo. models

• Can benefit other language pairs with isolated long-range reorderings (e.g. Arabic-English)

Thanks for your attention!
Other slides...
Part 2: Word-after-word reordering modeling and pruning

[Bisazza & Federico 2013]

Idea: dynamically prune unlikely long reordering steps *before* performing them

Method: train a binary classifier to learn if an input word should be translated *right after* another

At decoding time, only explore long reorderings that are likely according to this model

*Jedoch konnten sie Kinder in Teilen von ... nicht erreichen.*
Part 2: Word-after-word reordering modeling and pruning

[Bisazza & Federico 2013]

Idea: dynamically prune unlikely long reordering steps before performing them

Method: train a binary classifier to learn if an input word should be translated right after another

At decoding time, only explore long reorderings that are likely according to this model
Part 2: Word-after-word reordering modeling and pruning

[Bisazza & Federico 2013]

Idea: dynamically prune unlikely long reordering steps before performing them

Method: train a binary classifier to learn if an input word should be translated right after another

At decoding time, only explore long reorderings that are likely according to this model