### Overview

- some syntactic phenomena that make (machine) translation difficult
- some solutions from MT research

### Word Order

- languages differ in word order:
  - SVO: English, French, Mandarin, Russian, ...
  - SOV: Hindi, Latin, Japanese, Korean, ...
  - VSO, VOS, OSV, OV exist, but less common
  - German is V2 in main clause, SOV in subordinate clause
  - word order more flexible when function is morphologically marked

example: German–English

```
| der Mann | der den letzten Marathon gewonnen hat |
```

the man who won the last marathon
translation units can be discontinuous

example: German separable verb prefixes are clause-final

he proposes a trade
er schlägt einen Handel vor
"The Awful German Language" by Mark Twain

The Germans have another kind of parenthesis, which they make by splitting a verb in two and putting half of it at the beginning of an exciting chapter and the other half at the end of it. Can any one conceive of anything more confusing than that? These things are called "separable verbs." The German grammar is blistered all over with separable verbs; and the wider the two portions of one of them are spread apart, the better the author of the crime is pleased with his performance.

source side pre-reordering

- preprocess the source text to better match target language word order
- standard for some language pairs for phrase-based SMT
- various approaches based on syntactic analysis of source sentence:
  - hand-written rules [Nießen and Ney, 2000, Collins et al., 2005]
  - neural pre-reordering [Miceli Barone and Attardi, 2015]
- negative results for neural MT [Du and Way, 2017]

target side pre-reordering?

- in principle, we can reorder target side before training
- need second step to restore original target language word order → this is hard
- some research, but never became standard

syntactic n-grams

- n-grams may not be meaningful if word order is flexible
- syntactic n-grams for evaluation: head-word-chain metric (HWCM) [Liu and Gildea, 2005]

syntactic n-grams
Non-projective Structures

classical example of context-sensitive structures
in Swiss German (Zurich dialect)

dass mer em Hans s’Huus hälfed aastriiche
that we help Hans paint the house

Non-projective Structures

non-projective German dependency tree

Wir müssen Systeme aufbauen, die sicher sind.
We have to build systems that are safe.

Subcategorization

words occur with specific syntactic arguments
complex mapping between syntactic arguments and meaning

example

- *remember* can have direct object or clausal object
  - semantic role: content of the memory
    - *he remembers his medical appointment.*
    - *he remembers that he has a medical appointment.*
  - *remind* can have direct object, and prep. or clausal object
    - direct object: recipient of information
    - prep. or clausal object: information
    - *I remind him of his medical appointment.*
    - *I remind him that he has a medical appointment.*
  - ungrammatical (or semantically nonsensical):
    - *he remembers of his medical appointment.*
    - *he reminds his medical appointment.*

- *he remembers the medical appointment.*
- *er erinnert den Arzttermin.*
- er erinnert sich an den Arzttermin.
- *he remembers himself to the medical appointment.*

for some translations, syntactic arguments swap semantic roles

- *he misses the cat*
  - *die Katze fehlt ihm*
  - (the cat is missing to him)
### Subcategorization

different meanings of words occur with different subcategories:

- *she applies for a job.*
  prep. object *for*: submit oneself as a candidate (German: “bewerben”)
- *this rule applies to everyone.*
  intransitive: be relevant (German: “gelten”)
- *he applies the wrong test.*
  transitive: use (German: “anwenden”)

### Syntax and Neural MT

attentional encoder–decoder is less limited than previous approaches

- reordering can be learned by attention model
- consequence of subcategorization constraints:
  you should not translate syntactic arguments independently
- recurrent model can handle discontiguous and non-projective structures

**recent research**
does neural MT benefit from syntactic structure/information?

### Linguistic Input Features

disambiguate words by POS

<table>
<thead>
<tr>
<th>English</th>
<th>German</th>
</tr>
</thead>
<tbody>
<tr>
<td>close&lt;sub&gt;verb&lt;/sub&gt;</td>
<td>schließen</td>
</tr>
<tr>
<td>close&lt;sub&gt;adj&lt;/sub&gt;</td>
<td>nah</td>
</tr>
<tr>
<td>close&lt;sub&gt;noun&lt;/sub&gt;</td>
<td>Ende</td>
</tr>
</tbody>
</table>

source

*We thought a win like this might be close<sub>adj</sub>.*

reference

*Wir dachten, dass ein solcher Sieg nah sein könnte.*

baseline NMT

*Wir dachten, ein Sieg wie dieser könnte schließen.*

source

*Dangerous is the route, however.*

reference

*However the route is dangerous.*

baseline NMT

*Gefährlich pred ist die Route subj aber dennoch.*

*Wir dachten, dass ein solcher Sieg nah sein könnte.*

source

*We thought a win like this might be close<sub>adj</sub>.*

reference

*Wir dachten, ein Sieg wie dieser könnte schließen.*

baseline NMT

*Gefährlich pred ist die Route subj aber dennoch.*
Neural Machine Translation: Multiple Input Features

Use separate embeddings for each feature, then concatenate (same method as for inclusion of lemma)

baseline: only word feature

\[
E(\text{close}) = \begin{bmatrix} 0.5 \\ 0.2 \\ 0.3 \\ 0.1 \end{bmatrix}
\]

\([F]\) input features

\[
E_1(\text{close}) = \begin{bmatrix} 0.4 \\ 0.1 \\ 0.2 \end{bmatrix}, \quad E_2(\text{adj}) = \begin{bmatrix} 0.1 \end{bmatrix}, \quad E_1(\text{close}) \parallel E_2(\text{adj}) = \begin{bmatrix} 0.4 \\ 0.1 \\ 0.2 \\ 0.1 \end{bmatrix}
\]

Experiments

Features

- lemmas
- morphological features
- POS tags
- dependency labels
- BPE tags

Data

- WMT16 training/test data
- English→German and English→Romanian

Results: BLEU ↑

- represent source sentence as binary tree
- leaf nodes: states of sequential RNN
- tree-based encoder computes state of \( k \)-th parent node \((h^p_k)\) as function of left and right child nodes \((h^l_k, h^r_k)\):

\[
h^p_k = f(h^l_k, h^r_k)
\]

- allow attention on original encoder states (leaves) and tree nodes

![Figure 5: Translation example of a long sentence and the attentional relations by our proposed model.](image)
Recurrent Neural Network Grammars

hypothesis:
- recurrent neural networks have recency bias
- instead, we want to induce syntactic bias

One theory of hierarchy

- Generate symbols sequentially using an RNN
- Add some “control symbols” to rewrite the history periodically
  - Periodically “compress” a sequence into a single “constituent”
  - Augment RNN with an operation to compress recent history into a single vector (-> “reduce”)
  - RNN predicts next symbol based on the history of compressed elements and non-compressed terminals (“shift” or “generate”)
  - RNN must also predict “control symbols” that decide how big constituents are
- We call such models recurrent neural network grammars.

Trees as sequences

The hungry cat meows.

S
  NP
  VP

S( NP( The hungry cat ) VP( meows ) . )
Recurrent Neural Network Grammars

Trees as sequences

\[
S( \text{NP( The hungry cat ) VP( meows ) . })
\]

Tree traversals correspond to stack operations

R. Sennrich  MT – 2018 – 13  26 / 30
Recurrent Neural Network Grammars

Trees as sequences

The hungry cat meows.

S( NP( The hungry cat ) VP( meows ) . )

Tree traversals correspond to stack operations

R. Sennrich
MT – 2018 – 13
26 / 30
Recurrent Neural Network Grammars

Trees as sequences

S( NP( The hungry cat ) VP( meows ) )

Tree traversals correspond to stack operations

Recurrent Neural Network Grammars

Trees as sequences

S( NP( The hungry cat ) VP( meows ) )

Tree traversals correspond to stack operations

Recurrent Neural Network Grammars

Trees as sequences

S( NP( The hungry cat ) VP( meows ) )

Tree traversals correspond to stack operations

Recurrent Neural Network Grammars

Trees as sequences

S( NP( The hungry cat ) VP( meows ) )

Tree traversals correspond to stack operations
<table>
<thead>
<tr>
<th>Terminals</th>
<th>Stack</th>
<th>Action</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td>NT(S)</td>
</tr>
</tbody>
</table>

R. Sennrich  MT – 2018 – 13  27/30

<table>
<thead>
<tr>
<th>Terminals</th>
<th>Stack</th>
<th>Action</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td>NT(S)</td>
</tr>
<tr>
<td></td>
<td></td>
<td>NT(NP)</td>
</tr>
</tbody>
</table>

R. Sennrich  MT – 2018 – 13  27/30

<table>
<thead>
<tr>
<th>Terminals</th>
<th>Stack</th>
<th>Action</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td>NT(S)</td>
</tr>
<tr>
<td></td>
<td></td>
<td>NT(NP)</td>
</tr>
<tr>
<td>GEN(The)</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

R. Sennrich  MT – 2018 – 13  27/30

<table>
<thead>
<tr>
<th>Terminals</th>
<th>Stack</th>
<th>Action</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td>NT(S)</td>
</tr>
<tr>
<td></td>
<td></td>
<td>NT(NP)</td>
</tr>
<tr>
<td>GEN(The)</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

R. Sennrich  MT – 2018 – 13  27/30
Recurrent Neural Network Grammars

<table>
<thead>
<tr>
<th>Terminals</th>
<th>Stack</th>
<th>Action</th>
</tr>
</thead>
<tbody>
<tr>
<td>(S)</td>
<td></td>
<td>NT(S)</td>
</tr>
<tr>
<td>(NT)</td>
<td></td>
<td></td>
</tr>
<tr>
<td>(NP)</td>
<td></td>
<td>GEN(The)</td>
</tr>
<tr>
<td>The</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Recurrent Neural Network Grammars

<table>
<thead>
<tr>
<th>Terminals</th>
<th>Stack</th>
<th>Action</th>
</tr>
</thead>
<tbody>
<tr>
<td>(S)</td>
<td></td>
<td>NT(S)</td>
</tr>
<tr>
<td>(NT)</td>
<td></td>
<td></td>
</tr>
<tr>
<td>(NP)</td>
<td></td>
<td>GEN(The)</td>
</tr>
<tr>
<td>The</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Recurrent Neural Network Grammars

<table>
<thead>
<tr>
<th>Terminals</th>
<th>Stack</th>
<th>Action</th>
</tr>
</thead>
<tbody>
<tr>
<td>(S)</td>
<td></td>
<td>NT(S)</td>
</tr>
<tr>
<td>(NT)</td>
<td></td>
<td></td>
</tr>
<tr>
<td>(NP)</td>
<td></td>
<td>GEN(hungry)</td>
</tr>
<tr>
<td>The</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Recurrent Neural Network Grammars

<table>
<thead>
<tr>
<th>Terminals</th>
<th>Stack</th>
<th>Action</th>
</tr>
</thead>
<tbody>
<tr>
<td>(S)</td>
<td></td>
<td>NT(S)</td>
</tr>
<tr>
<td>(NT)</td>
<td></td>
<td></td>
</tr>
<tr>
<td>(NP)</td>
<td></td>
<td>GEN(hungry)</td>
</tr>
<tr>
<td>The</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

R. Sennrich  MT – 2018 – 13  27/30

R. Sennrich  MT – 2018 – 13  27/30
### Recurrent Neural Network Grammars

<table>
<thead>
<tr>
<th>Terminals</th>
<th>Stack</th>
<th>Action</th>
</tr>
</thead>
<tbody>
<tr>
<td>NT(S)</td>
<td></td>
<td></td>
</tr>
<tr>
<td>NT(NP)</td>
<td></td>
<td></td>
</tr>
<tr>
<td>GEN(The)</td>
<td></td>
<td>REDUCE</td>
</tr>
<tr>
<td>GEN(hungry)</td>
<td></td>
<td>REDUCE</td>
</tr>
<tr>
<td>GEN(cat)</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Compress "The hungry cat" into a single composite symbol.
Recurrent Neural Network Grammars

<table>
<thead>
<tr>
<th>Terminals</th>
<th>Stack</th>
<th>Action</th>
</tr>
</thead>
<tbody>
<tr>
<td>NT(S)</td>
<td>(S)</td>
<td></td>
</tr>
<tr>
<td>NT(NP)</td>
<td>(S (NP The hungry cat)</td>
<td></td>
</tr>
<tr>
<td>GEN(The) GEN(hungry) GEN(cat)</td>
<td>The hungry cat</td>
<td>REDUCE</td>
</tr>
<tr>
<td>NT(VP)</td>
<td>(S (NP The hungry cat) (VP)</td>
<td></td>
</tr>
</tbody>
</table>

R. Sennrich  MT – 2018 – 13  27/30

Recurrent Neural Network Grammars
Recurrent Neural Network Grammars

### Syntactic Composition

Need representation for: \((\text{NP } \text{The hungry cat})\)

What head type? 

\begin{align*}
\text{The} & \\
\text{The hungry} & \\
\text{The hungry cat} & \\
\text{The hungry cat meows} & \\
\text{The hungry cat meows} & \\
\text{The hungry cat meows} & \\
\text{The hungry cat meows} & \\
\text{The hungry cat meows} & \\
\text{The hungry cat meows} & \\
\text{The hungry cat meows} & \\
\text{The hungry cat meows} & \\
\text{The hungry cat meows} & \\
\text{The hungry cat meows} & \\
\text{The hungry cat meows} & \\
\text{The hungry cat meows} & \\
\text{The hungry cat meows} & \\
\text{The hungry cat meows} & \\
\text{The hungry cat meows} & \\
\text{The hungry cat meows} & \\
\text{The hungry cat meows} & \\
\text{The hungry cat meows} & \\
\text{The hungry cat meows} & \\
\text{The hungry cat meows} & \\
\text{The hungry cat meows} & \\
\text{The hungry cat meows} & \\
\text{The hungry cat meows} & \\
\text{The hungry cat meows} & \\
\text{The hungry cat meows} & \\
\text{The hungry cat meows} & \\
\text{The hungry cat meows} & \\
\text{The hungry cat meows} & \\
\end{align*}

R. Sennrich  
MT – 2018 – 13  
27/30
Recurrent Neural Network Grammars

Syntactic Composition

Need representation for: (NP *The hungry cat*)

What head type?

Syntactic Composition

Need representation for: (NP *The hungry cat*)

What head type?
Syntactic Composition

Need representation for: (NP The hungry cat)
Recurrent Neural Network Grammars

Recursion

Need representation for: (NP The hungry cat)

(RP The (ADJP very hungry) cat)

RNNGs in MT

Eriguchi et al., Feb 2017

- Basic idea: learn decoder-encoder MT model and RNNG on parallel data with parsed target side, sharing target word embedding parameters. (multi-task learning). To translate, just use MT model.

<table>
<thead>
<tr>
<th></th>
<th>De-En</th>
<th>Ra-En</th>
<th>Cs-En</th>
<th>Jp-En</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>BLEU</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>NMT</td>
<td>16.61</td>
<td>12.03</td>
<td>11.22</td>
<td>17.88</td>
</tr>
<tr>
<td>NMT+RG</td>
<td>16.41</td>
<td>12.46</td>
<td>12.06</td>
<td>18.84</td>
</tr>
<tr>
<td>RIBES</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>NMT</td>
<td>73.75</td>
<td>69.56</td>
<td>69.59</td>
<td>71.27</td>
</tr>
<tr>
<td>NMT+RG</td>
<td>75.03</td>
<td>71.04</td>
<td>70.39</td>
<td>72.25</td>
</tr>
</tbody>
</table>

Table 2: BLEU and RIBES scores by the baseline and proposed models on the test set. We use the bootstrap resampling method from Koehn (2004) to compute the statistical significance. We use ↑ to mark those significant cases with \( p < 0.005 \).

<table>
<thead>
<tr>
<th></th>
<th>BLEU</th>
</tr>
</thead>
<tbody>
<tr>
<td>Jp-En (Dev)</td>
<td></td>
</tr>
<tr>
<td>NMT+RG</td>
<td>18.60</td>
</tr>
<tr>
<td>w/o Buffer</td>
<td>18.02</td>
</tr>
<tr>
<td>w/o Action</td>
<td>17.94</td>
</tr>
<tr>
<td>w/o Stack</td>
<td>17.58</td>
</tr>
<tr>
<td>NMT</td>
<td>17.75</td>
</tr>
</tbody>
</table>

Table 3: Effect of each component in RNNG.

slide credit for slides 2,4-6,24-27: Adam Lopez
### Bibliography I


### Bibliography II
