Machine Translation as Sequence Modelling

Philipp Koehn

23 July 2016
Sequence Model

• Input sentence

Eu quero ouvir uma apresentação muito interessante.

• Output

I want to listen to a very interesting presentation.

• Idea: produce output one word at a time
N-Gram Model

• Input sentence

Eu quero ouvir uma apresentação muito interessante.

• Output

  – \( p(I) \)
  – \( p(\text{want}|I) \)
  – \( p(\text{to}|\text{I want}) \)

• We learned how to do this today

• Major flaw: Output is not conditioned on input
Conditioning on Input

• Input sentence

Eu quero ouvir uma apresentação muito interessante.

• Output

- $p(\text{I} | \text{Eu quero ouvir uma apresentação muito interessante.})$
- $p(\text{want} | \text{I, Eu quero ouvir uma apresentação muito interessante.})$
- $p(\text{to} | \text{I want, Eu quero ouvir uma apresentação muito interessante.})$

• Conditioning on entire source sentence too sparse to estimate
  (unlikely that we have seen input sentence before)
### 1-1 Alignment to Input

<table>
<thead>
<tr>
<th>Input</th>
<th>Eu</th>
<th>quero</th>
<th>ouvir</th>
<th>uma</th>
<th>...</th>
</tr>
</thead>
<tbody>
<tr>
<td>Output</td>
<td>I</td>
<td>want</td>
<td>hear</td>
<td>a</td>
<td></td>
</tr>
<tr>
<td>Model</td>
<td>$p(I</td>
<td>Eu)$</td>
<td>$p(\text{want}</td>
<td>quero)$</td>
<td>$p(\text{hear}</td>
</tr>
</tbody>
</table>

- We are slowly getting somewhere

- Open problems
  - we need to move beyond 1-1 alignments
  - where do we get the probabilities from?
ibm model 1
Lexical Translation

• How to translate a word → look up in dictionary
  
  **Haus** — house, building, home, household, shell.

• Multiple translations
  
  – some more frequent than others
  – for instance: *house*, and *building* most common
  – special cases: *Haus* of a *snail* is its *shell*

• Note: In all lectures, we translate from a foreign language into English
Collect Statistics

Look at a parallel corpus (German text along with English translation)

<table>
<thead>
<tr>
<th>Translation of Haus</th>
<th>Count</th>
</tr>
</thead>
<tbody>
<tr>
<td>house</td>
<td>8,000</td>
</tr>
<tr>
<td>building</td>
<td>1,600</td>
</tr>
<tr>
<td>home</td>
<td>200</td>
</tr>
<tr>
<td>household</td>
<td>150</td>
</tr>
<tr>
<td>shell</td>
<td>50</td>
</tr>
</tbody>
</table>
Maximum likelihood estimation

\[ p_f(e) = \begin{cases} 
0.8 & \text{if } e = \text{house}, \\
0.16 & \text{if } e = \text{building}, \\
0.02 & \text{if } e = \text{home}, \\
0.015 & \text{if } e = \text{household}, \\
0.005 & \text{if } e = \text{shell}.
\end{cases} \]
Alignment

- In a parallel text (or when we translate), we align words in one language with the words in the other

<table>
<thead>
<tr>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
</tr>
</thead>
<tbody>
<tr>
<td>das</td>
<td>Haus</td>
<td>ist</td>
<td>klein</td>
</tr>
<tr>
<td>the</td>
<td>house</td>
<td>is</td>
<td>small</td>
</tr>
</tbody>
</table>

- Word positions are numbered 1–4
Alignment Function

• Formalizing alignment with an alignment function

• Mapping an English target word at position $i$ to a German source word at position $j$ with a function $a : i \rightarrow j$

• Example

$$a : \{1 \rightarrow 1, 2 \rightarrow 2, 3 \rightarrow 3, 4 \rightarrow 4\}$$
Reordering

Words may be reordered during translation

\[ a : \{1 \rightarrow 3, 2 \rightarrow 4, 3 \rightarrow 2, 4 \rightarrow 1\} \]
IBM Model 1

- Generative model: break up translation process into smaller steps
  - IBM Model 1 only uses lexical translation

- Translation probability
  - for a foreign sentence \( f = (f_1, ..., f_{l_f}) \) of length \( l_f \)
  - to an English sentence \( e = (e_1, ..., e_{l_e}) \) of length \( l_e \)
  - with an alignment of each English word \( e_j \) to a foreign word \( f_i \) according to the alignment function \( a : j \rightarrow i \)

\[
p(e, a|f) = \frac{\epsilon}{(l_f + 1)^{l_e}} \prod_{j=1}^{l_e} t(e_j|f_{a(j)})
\]

- parameter \( \epsilon \) is a normalization constant
Example

\[
p(e, a | f) = \frac{\epsilon}{4^3} \times t(\text{the} | \text{das}) \times t(\text{house} | \text{Haus}) \times t(\text{is} | \text{ist}) \times t(\text{small} | \text{klein})
\]

\[
= \frac{\epsilon}{4^3} \times 0.7 \times 0.8 \times 0.8 \times 0.4
\]

\[
= 0.0028\epsilon
\]
• We would like to estimate the lexical translation probabilities $t(e|f)$ from a parallel corpus

• ... but we do not have the alignments

• Chicken and egg problem
  – if we had the alignments,
    → we could estimate the parameters of our generative model
  – if we had the parameters,
    → we could estimate the alignments
EM Algorithm

• Incomplete data
  – if we had complete data, would could estimate model
  – if we had model, we could fill in the gaps in the data

• Expectation Maximization (EM) in a nutshell
  1. initialize model parameters (e.g. uniform)
  2. assign probabilities to the missing data
  3. estimate model parameters from completed data
  4. iterate steps 2–3 until convergence
EM Algorithm

... la maison ... la maison blue ... la fleur ...  

... the house ... the blue house ... the flower ...  

• Initial step: all alignments equally likely  

• Model learns that, e.g., la is often aligned with the
EM Algorithm

... la maison ... la maison blue ... la fleur ...

... the house ... the blue house ... the flower ...

- After one iteration

- Alignments, e.g., between la and the are more likely
EM Algorithm

... la maison ... la maison bleu ... la fleur ...

... the house ... the blue house ... the flower ...

• After another iteration

• It becomes apparent that alignments, e.g., between fleur and flower are more likely (pigeon hole principle)
EM Algorithm

... la maison ... la maison bleu ... la fleur ...

/ / / / X / /

... the house ... the blue house ... the flower ...

• Convergence

• Inherent hidden structure revealed by EM
EM Algorithm

... la maison ... la maison bleu ... la fleur ...

/ / / X /

... the house ... the blue house ... the flower ...

\[ p(\text{la} \mid \text{the}) = 0.453 \]
\[ p(\text{le} \mid \text{the}) = 0.334 \]
\[ p(\text{maison} \mid \text{house}) = 0.876 \]
\[ p(\text{bleu} \mid \text{blue}) = 0.563 \]

- Parameter estimation from the aligned corpus
IBM Model 1 and EM

- EM Algorithm consists of two steps

- Expectation-Step: Apply model to the data
  - parts of the model are hidden (here: alignments)
  - using the model, assign probabilities to possible values

- Maximization-Step: Estimate model from data
  - take assign values as fact
  - collect counts (weighted by probabilities)
  - estimate model from counts

- Iterate these steps until convergence
IBM Model 1 and EM

• We need to be able to compute:

  – Expectation-Step: probability of alignments
  – Maximization-Step: count collection
IBM Model 1 and EM

- **Probabilities**
  
  \[
  p(\text{the}|\text{la}) = 0.7 \quad p(\text{house}|\text{la}) = 0.05 \\
  p(\text{the}|\text{maison}) = 0.1 \quad p(\text{house}|\text{maison}) = 0.8
  \]

- **Alignments**
  
  \[
  \begin{array}{c}
  \text{la} \xrightarrow{\text{the}} \text{house} \\
  \text{maison} \xrightarrow{\text{the}} \text{house}
  \end{array}
  \]
  \[
  p(\text{e}, \text{a}|\text{f}) = 0.56 \quad p(\text{e}, \text{a}|\text{f}) = 0.035 \quad p(\text{e}, \text{a}|\text{f}) = 0.08 \quad p(\text{e}, \text{a}|\text{f}) = 0.005
  \]
  \[
  p(\text{a}|\text{e}, \text{f}) = 0.824 \quad p(\text{a}|\text{e}, \text{f}) = 0.052 \quad p(\text{a}|\text{e}, \text{f}) = 0.118 \quad p(\text{a}|\text{e}, \text{f}) = 0.007
  \]

- **Counts**
  
  \[
  c(\text{the}|\text{la}) = 0.824 + 0.052 \quad c(\text{house}|\text{la}) = 0.052 + 0.007 \\
  c(\text{the}|\text{maison}) = 0.118 + 0.007 \quad c(\text{house}|\text{maison}) = 0.824 + 0.118
  \]
hmm model
Modeling Alignment

• IBM Model 1 uses alignments to identify conditioning context

• But: does not model alignment itself

• Is it better to start translating the 1st input word or 10th input word?
HMM Model

- Condition word movements on previous word

- HMM alignment model:

\[ p(a(j)|a(j-1), l_f) \]
Decoding

- Input sentence

Eu quero ouvir uma apresentação muito interessante.

- Translation

<table>
<thead>
<tr>
<th>Input</th>
<th>Eu</th>
<th>quero</th>
<th>ouvir</th>
<th>uma</th>
</tr>
</thead>
<tbody>
<tr>
<td>Output</td>
<td>I</td>
<td>want</td>
<td>to</td>
<td>hear</td>
</tr>
</tbody>
</table>

| Translation | p(I|Eu) | p(want|quero) | p(to|quero) | p(hear|ouvir) | p(a|uma) |
|-------------|-------|------------|----------|-------------|----------|
| Alignment   | p(1|0, 7) | p(2|1, 7)   | p(2|2, 7)  | p(3|2, 7)   | p(4|3, 7) |
| Language Model | p(I|START) | p(want|I) | p(to|want) | p(hear|to) | p(a|hear) |
phrase-based model
Motivation

• Word-Based Models translate *words* as atomic units

• Phrase-Based Models translate *phrases* as atomic units

• Advantages:
  – many-to-many translation can handle non-compositional phrases
  – use of local context in translation
  – the more data, the longer phrases can be learned

• "Standard Model", used by Google Translate and others
Phrase-Based Model

- Foreign input is segmented in phrases
- Each phrase is translated into English
- Phrases are reordered

German: natuerlich, hat, john, spass am, spiel
English: of course, john, has, fun with the, game
Phrase Translation Table

- Main knowledge source: table with phrase translations and their probabilities

- Example: phrase translations for *natuerlich*

| Translation      | Probability $\phi(\overline{e}|f)$ |
|------------------|-----------------------------------|
| of course        | 0.5                               |
| naturally        | 0.3                               |
| of course ,      | 0.15                              |
| , of course ,    | 0.05                              |
**Real Example**

- Phrase translations for *den Vorschlag* learned from the Europarl corpus:

| English              | \( \phi(\bar{e} | f) \) | English             | \( \phi(\bar{e} | f) \) |
|----------------------|---------------------------|---------------------|---------------------------|
| the proposal         | 0.6227                    | the suggestions     | 0.0114                    |
| ’s proposal          | 0.1068                    | the proposed        | 0.0114                    |
| a proposal           | 0.0341                    | the motion          | 0.0091                    |
| the idea             | 0.0250                    | the idea of         | 0.0091                    |
| this proposal        | 0.0227                    | the proposal ,      | 0.0068                    |
| proposal             | 0.0205                    | its proposal        | 0.0068                    |
| of the proposal      | 0.0159                    | it                  | 0.0068                    |
| the proposals        | 0.0159                    | ...                 | ...                       |

- lexical variation (*proposal* vs *suggestions*)
- morphological variation (*proposal* vs *proposals*)
- included function words (*the*, *a*, ...)
- noise (*it*)
Learning a Phrase Translation Table

- Task: learn the model from a parallel corpus

- Three stages:
  - word alignment: using IBM models or other method
  - extraction of phrase pairs
  - scoring phrase pairs
Philipp Koehn  
Machine Translation  
23 July 2016
extract phrase pair consistent with word alignment:

assumes that / geht davon aus, dass
All words of the phrase pair have to align to each other.
### Phrase Pair Extraction

<table>
<thead>
<tr>
<th>michael</th>
<th>geht</th>
<th>davon</th>
<th>aus</th>
<th>dass</th>
<th>er</th>
<th>im</th>
<th>haus</th>
<th>bleibt</th>
</tr>
</thead>
<tbody>
<tr>
<td>michael</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>assumes</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>that</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>he</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>will</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>stay</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>in</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>the</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>house</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Smallest phrase pairs:

- michael — michael
- assumes — geht davon aus / geht davon aus ,
- that — dass / , dass
- he — er
- will stay — bleibt
- in the — im
- house — haus

unaligned words (here: German comma) lead to multiple translations
Larger Phrase Pairs

<table>
<thead>
<tr>
<th>michael</th>
<th>geht</th>
<th>davon</th>
<th>aus</th>
<th>dass</th>
<th>er</th>
<th>im</th>
<th>haus</th>
<th>bleibt</th>
</tr>
</thead>
<tbody>
<tr>
<td>assumes</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>that</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>he</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>will</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>stay</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>in</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>the</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>house</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

michael assumes — michael geht davon aus / michael geht davon aus,
assumes that — geht davon aus, dass ; assumes that he — geht davon aus, dass er
that he — dass er / , dass er ; in the house — im haus
michael assumes that — michael geht davon aus, dass
michael assumes that he — michael geht davon aus , dass er
michael assumes that he will stay in the house — michael geht davon aus , dass er im haus bleibt
assumes that he will stay in the house — geht davon aus , dass er im haus bleibt
that he will stay in the house — dass er im haus bleibt ; dass er im haus bleibt,
he will stay in the house — er im haus bleibt ; will stay in the house — im haus bleibt
Scoring Phrase Translations

- Phrase pair extraction: collect all phrase pairs from the data
- Phrase pair scoring: assign probabilities to phrase translations
- Score by relative frequency:

\[ \phi(\bar{f}|\bar{e}) = \frac{\text{count}(\bar{e}, \bar{f})}{\sum_{\bar{f}_i} \text{count}(\bar{e}, \bar{f}_i)} \]
Decoding

• We have a mathematical model for translation

\[ p(e|f) \]

• Task of decoding: find the translation \( e_{\text{best}} \) with highest probability

\[ e_{\text{best}} = \arg\max_e p(e|f) \]

• Two types of error

  – the most probable translation is bad → fix the model
  – search does not find the most probably translation → fix the search

• Decoding is evaluated by search error, not quality of translations (although these are often correlated)
Translation Process

- Task: translate this sentence from German into English

er geht ja nicht nach hause
Translation Process

• Task: translate this sentence from German into English

er geht ja nicht nach hause

• Pick phrase in input, translate
Translation Process

• Task: translate this sentence from German into English

er geht ja nicht nach hause

he does not

• Pick phrase in input, translate
  – it is allowed to pick words out of sequence reordering
  – phrases may have multiple words: many-to-many translation
Translation Process

- Task: translate this sentence from German into English

```
er geht ja nicht nach hause
```

- Pick phrase in input, translate

```
er geht ja nicht
he does not go
```

Philipp Koehn  Machine Translation  23 July 2016
Translation Process

- Task: translate this sentence from German into English

- Pick phrase in input, translate

```
er geht ja nicht nach hause
he does not go home
```
Many translation options to choose from

- in Europarl phrase table: 2727 matching phrase pairs for this sentence
- by pruning to the top 20 per phrase, 202 translation options remain
The machine translation decoder does not know the right answer
- picking the right translation options
- arranging them in the right order

→ Search problem solved by heuristic beam search
consult phrase translation table for all input phrases
Decoding: Start with Initial Hypothesis

initial hypothesis: no input words covered, no output produced
Decoding: Hypothesis Expansion

pick any translation option, create new hypothesis
Decoding: Hypothesis Expansion

create hypotheses for all other translation options
Decoding: Hypothesis Expansion

er geht ja nicht nach hause

also create hypotheses from created partial hypothesis
Decoding: Find Best Path

backtrack from highest scoring complete hypothesis
Computational Complexity

- The suggested process creates exponential number of hypothesis

- Machine translation decoding is NP-complete

- Reduction of search space:
  - recombination (risk-free)
  - pruning (risky)
Recombination

• Two hypothesis paths lead to two matching hypotheses
  – same number of foreign words translated
  – same English words in the output
  – different scores

• Worse hypothesis is dropped
Recombination

• Two hypothesis paths lead to hypotheses indistinguishable in subsequent search
  – same number of foreign words translated
  – same last two English words in output (assuming trigram language model)
  – same last foreign word translated
  – different scores

• Worse hypothesis is dropped
Pruning

- Recombination reduces search space, but not enough  
  (we still have a NP complete problem on our hands)

- Pruning: remove bad hypotheses early
  - put comparable hypothesis into stacks  
    (hypotheses that have translated same number of input words)
  - limit number of hypotheses in each stack
- Hypothesis expansion in a stack decoder
  - translation option is applied to hypothesis
  - new hypothesis is dropped into a stack further down
Stack Decoding Algorithm

1: place empty hypothesis into stack 0
2: for all stacks 0...n – 1 do
3:   for all hypotheses in stack do
4:     for all translation options do
5:       if applicable then
6:         create new hypothesis
7:         place in stack
8:         recombine with existing hypothesis if possible
9:       prune stack if too big
10:      end if
11:     end for
12:   end for
13: end for
Pruning

• Pruning strategies
  – histogram pruning: keep at most $k$ hypotheses in each stack
  – stack pruning: keep hypothesis with score $\alpha \times$ best score ($\alpha < 1$)

• Computational time complexity of decoding with histogram pruning

  $O(\text{max stack size} \times \text{translation options} \times \text{sentence length})$

• Number of translation options is linear with sentence length, hence:

  $O(\text{max stack size} \times \text{sentence length}^2)$

• Quadratic complexity
operation sequence model
A Critique: Phrase Segmentation is Arbitrary

- If multiple segmentations possible - why chose one over the other?
  
  spass am spiel vs. spass am spiel

- When choose larger phrase pairs or multiple shorter phrase pairs?
  
  spass am spiel vs. spass am spiel vs. spass am spiel

- None of this has been properly addressed
A Critique: Strong Independence Assumptions

- Lexical context considered only within phrase pairs

\[ \text{spass am} \rightarrow \text{fun with} \]

- No context considered between phrase pairs

\[ ? \text{spass am} ? \rightarrow ? \text{fun with} ? \]

- Some phrasal context considered in lexicalized reordering model
  ... but not based on the identity of neighboring phrases
Segmentation? Minimal Phrase Pairs

natürlich hat John Spaß am Spiel

⇓

of course John has fun with the game

natürlich hat John Spaß am Spiel

⇓

of course John has fun with the game
Independence?
Consider Sequence of Operations

<table>
<thead>
<tr>
<th>Step</th>
<th>Operation</th>
<th>Inserted Words</th>
<th>Result</th>
</tr>
</thead>
<tbody>
<tr>
<td>01</td>
<td>Generate(natürlich, of course)</td>
<td>natürlich ↓ of course</td>
<td>of course</td>
</tr>
<tr>
<td>02</td>
<td>Insert Gap</td>
<td>natürlich ↓</td>
<td>John</td>
</tr>
<tr>
<td>03</td>
<td>Generate (John, John)</td>
<td>natürlich ↓</td>
<td>of course John</td>
</tr>
<tr>
<td>04</td>
<td>Jump Back (1)</td>
<td>natürlich hat ↓</td>
<td>John</td>
</tr>
<tr>
<td>05</td>
<td>Generate (hat, has)</td>
<td>natürlich hat ↓</td>
<td>of course John has</td>
</tr>
<tr>
<td>06</td>
<td>Jump Forward</td>
<td>natürlich hat John ↓</td>
<td>of course John has</td>
</tr>
<tr>
<td>07</td>
<td>Generate(natürlich, of course)</td>
<td>natürlich hat John Spaß ↓</td>
<td>of course John has fun</td>
</tr>
<tr>
<td>08</td>
<td>Generate(am, with)</td>
<td>natürlich hat John Spaß am ↓</td>
<td>of course John has fun with the</td>
</tr>
<tr>
<td>09</td>
<td>GenerateTargetOnly(the)</td>
<td>natürlich hat John Spaß am Spiel ↓</td>
<td>of course John has fun with the game</td>
</tr>
<tr>
<td>10</td>
<td>Generate(Spiel, game)</td>
<td>natürlich hat John Spaß am Spiel ↓</td>
<td>of course John has fun with the game</td>
</tr>
</tbody>
</table>
Operation Sequence Model

• Operations
  – generate (phrase translation)
  – generate target only
  – generate source only
  – insert gap
  – jump back
  – jump forward

• N-gram sequence model over operations, e.g., 5-gram model:

\[ p(o_1) p(o_2|o_1) p(o_3|o_1, o_2) \ldots p(o_{10}|o_6, o_7, o_8, o_9) \]
In Practice

• Operation Sequence Model used as additional feature function

• Significant improvements over phrase-based baseline

→ State-of-the-art systems include such a model
syntax
Sequence Model — Really?

- Different languages have different word order
- Language is recursive → tree formalisms
- Need to translate meaning, not words
A Vision

Lexical Transfer
Syntactic Transfer
Semantic Transfer
Interlingua Analysis
Generation

Source
Interlingua
Target
Phrase Structure Grammar

• Phrase structure
  – noun phrases: the big man, a house, ...
  – prepositional phrases: at 5 o’clock, in Edinburgh, ...
  – verb phrases: going out of business, eat chicken, ...
  – adjective phrases, ...

• Context-free Grammars (CFG)
  – non-terminal symbols: phrase structure labels, part-of-speech tags
  – terminal symbols: words
  – production rules: \( NT \rightarrow [NT,T]^+ \)
    example: \( NP \rightarrow \text{DET NN} \)
Phrase structure grammar tree for an English sentence
(as produced Collins’ parser)
Synchronous Phrase Structure Grammar

- English rule

\[ NP \rightarrow \text{DET} \text{JJ} \text{NN} \]

- French rule

\[ NP \rightarrow \text{DET} \text{NN} \text{JJ} \]

- Synchronous rule (indices indicate alignment):

\[ NP \rightarrow \text{DET}_1 \text{NN}_2 \text{JJ}_3 \mid \text{DET}_1 \text{JJ}_3 \text{NN}_2 \]
Synchronous Grammar Rules

• Nonterminal rules

\[ NP \rightarrow \text{DET}_1 \text{NN}_2 \text{JJ}_3 \mid \text{DET}_1 \text{JJ}_3 \text{NN}_2 \]

• Terminal rules

\[ N \rightarrow \text{maison} \mid \text{house} \]
\[ NP \rightarrow \text{la maison bleue} \mid \text{the blue house} \]

• Mixed rules

\[ NP \rightarrow \text{la maison JJ}_1 \mid \text{the JJ}_1 \text{house} \]
Tree-Based Translation Model

- Translation by parsing
  - synchronous grammar has to parse entire input sentence
  - output tree is generated at the same time
  - process is broken up into a number of rule applications

- Translation probability

\[
\text{SCORE}(\text{TREE}, E, F) = \prod_{i} \text{RULE}_{i}
\]

- Many ways to assign probabilities to rules
Aligned Tree Pair

Phrase structure grammar trees with word alignment
(German–English sentence pair.)
Reordering Rule

- Subtree alignment

\[
\begin{array}{c}
\text{VP} \\
\text{PPER} \\
\text{NP} \\
\text{VVFIN} \\
\end{array}
\leftrightarrow
\begin{array}{c}
\text{VP} \\
\text{VBG} \\
\text{RP} \\
\text{PP} \\
\text{NP} \\
\end{array}
\]

- Synchronous grammar rule

\[
\text{VP} \rightarrow \text{PPER}_1 \text{ NP}_2 \text{ aushändigen} \mid \text{passing on PP}_1 \text{ NP}_2
\]

- Note:
  - one word \text{aushändigen} mapped to two words \text{passing on} ok
  - but: fully non-terminal rule not possible
    (one-to-one mapping constraint for nonterminals)
Learning Syntactic Translation Rules

Ich werde Ihnen die entspr. Anm. aushänd.

<table>
<thead>
<tr>
<th>PRP</th>
<th>MD</th>
<th>VB</th>
<th>VBG</th>
<th>DT</th>
<th>RP</th>
<th>PP</th>
<th>TO</th>
<th>PRP</th>
<th>NNS</th>
<th>VP</th>
<th>S</th>
</tr>
</thead>
<tbody>
<tr>
<td>I</td>
<td>shall</td>
<td>be</td>
<td>passing</td>
<td>some comments</td>
<td>to you</td>
<td>Ihnen</td>
<td>to you</td>
<td>PP</td>
<td>PRO</td>
<td>Ich werde Ihnen die entspr. Anm. aushänd.</td>
<td></td>
</tr>
</tbody>
</table>
I shall be passing on to you some comments.

Ich werde Ihnen die entsprechenden Anmerkungen aushändigen.

Align each node in the parse tree.
Syntax Decoding

German input sentence with tree

Sie
PPER
will
VAFIN
eine
ART
Tasse
NN
Kaffee
NN
trinken
VVINF

S
VP
NP
Purely lexical rule: filling a span with a translation (a constituent in the chart)
Syntax Decoding

Purely lexical rule: filling a span with a translation (a constituent in the chart)
Syntax Decoding

Purely lexical rule: filling a span with a translation (a constituent in the chart)
Complex rule: matching underlying constituent spans, and covering words
Syntax Decoding

Complex rule with reordering
Sie will eine Tasse Kaffee trinken.

Translation: She wants to drink a cup of coffee.
Syntactic Decoding

Inspired by monolingual syntactic chart parsing:

During decoding of the source sentence, a chart with translations for the $O(n^2)$ spans has to be filled.
• Syntax-based models proven to work well for German, Chinese

• Decoding more complex and slower

• Needed: syntactic parser and hand-holding for each language pair
in defense of sequence models
Evidence from Human Translators

- Translation process studies (e.g., in CASMACAT)
- Humans start translating after reading a few words
Left-to-Right Parsing

Push Down Automaton

The interesting lecture ends soon
Left-to-Right Parsing

Push Down Automaton

look up POS tag

The interesting lecture ends soon
DET
Left-to-Right Parsing

Push Down Automaton

look up POS tag

The interesting lecture ends soon
DET JJ DET
Left-to-Right Parsing

Push Down Automaton

look up POS tag

The interesting lecture ends soon
DET JJ N
DET JJ DET
Left-to-Right Parsing

Push Down Automaton

apply rule

The interesting lecture ends soon
DET JJ NP
DET
Left-to-Right Parsing

Push Down Automaton

look up POS tag

The interesting lecture ends soon
DET JJ NP VB
DET NP

Philipp Koehn
Machine Translation
23 July 2016
Left-to-Right Parsing

Push Down Automaton

look up POS tag

<table>
<thead>
<tr>
<th>The</th>
<th>interesting</th>
<th>lecture</th>
<th>ends</th>
<th>soon</th>
</tr>
</thead>
<tbody>
<tr>
<td>DET</td>
<td>JJ</td>
<td>NP</td>
<td>VB</td>
<td>RB</td>
</tr>
<tr>
<td>DET</td>
<td></td>
<td>NP</td>
<td>VB</td>
<td></td>
</tr>
</tbody>
</table>
Left-to-Right Parsing

Push Down Automaton

apply rule

The DET interesting JJ lecture NP ends VB VP soon NP NP
Push Down Automaton

apply rule

The DET interesting JJ lecture NP ends VB soon S
neural translation
Neural Networks

- Real valued vector representations
- Multiple layers of computation
- Non-linear functions

\[ \vec{h} = \text{sigmoid}(W \vec{x}) \]
\[ \vec{y} = \text{sigmoid}(V \vec{h}) \]
Word Embeddings
Why Neural Machine Translation?

- Word embeddings allow learning from similar examples
- Condition on a lot of context without backoff schemes
- Maybe there is something to non-linearity
Neural N-Gram Language Model
Recurrent Neural Networks

Word 1: C

Word 2: copy values

Word 3: copy values

Word 4:
Encoder-Decoder Translation Model

Input Word Embeddings

Recurrent NN

Recurrent NN

Output Words
Attention Translation Model

Input Word Embeddings

Left-to-Right Recurrent NN

Right-to-Left Recurrent NN

Alignment

Input Context

Hidden State

Output Words

Input Word
Embeddings

Left-to-Right
Recurrent NN

Right-to-Left
Recurrent NN

Alignment

Input Context

Hidden State

Output Words
practical matters
How Good is MT?

Portuguese:
A seleção portuguesa de futebol, que se sagrou no domingo pela primeira vez campeã europeia, ao vencer por 1-0 a França na final, foi hoje recebida em euforia por milhares de pessoas no aeroporto Humberto Delgado, em Lisboa.

O avião Eusbio, que foi escoltado por dois aviões da Força Aérea Portuguesa desde a entrada em território português, aterrou em Lisboa às 12:40, tendo passado por um improvisado ‘arco do triunfo’, formado por dois jatos de água com as duas cores principais da bandeira nacional.

Google Translate:
The Portuguese national soccer team, which won on Sunday for the first time European champions by winning 1-0 to France in the final, was received today in euphoria by thousands of people at the airport Humberto Delgado in Lisbon.

The plane Eusebius, who was escorted by two aircraft of the Portuguese Air Force since the entry into Portuguese territory, landed in Lisbon at 12:40, having gone through a makeshift ‘triumphal arch’, formed by two water jets with two colors main national flag.
How Good is MT?

Portuguese:
A seleção portuguesa de futebol, que se sagrou no domingo pela primeira vez campeã europeia, ao vencer por 1-0 a França na final, foi hoje recebida em euforia por milhares de pessoas no aeroporto Humberto Delgado, em Lisboa.

O avião Eusbio, que foi escoltado por dois aviões da Força Área Portuguesa desde a entrada em território português, aterrou em Lisboa às 12:40, tendo passado por um improvisado ‘arco do triunfo’, formado por dois jatos de água com as duas cores principais da bandeira nacional.

Google Translate:
The Portuguese national soccer team, which won on Sunday for the first time European champions by winning 1-0 to France in the final, was received today in euphoria by thousands of people at the airport Humberto Delgado in Lisbon.

The plane Eusebius, who was escorted by two aircraft of the Portuguese Air Force since the entry into Portuguese territory, landed in Lisbon at 12:40, having gone through a makeshift ‘triumphal arch’, formed by two water jets with two colors main national flag.
What Works Best?

- WMT evaluation campaign

- Winner English–German (with official ties)

<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>rule</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>phrase</td>
<td></td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>syntax</td>
<td></td>
<td></td>
<td></td>
<td>X</td>
<td></td>
<td></td>
<td>X</td>
<td>X</td>
<td></td>
</tr>
<tr>
<td>neural</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>X</td>
</tr>
</tbody>
</table>

- For other language pairs, phrase-based systems dominated longer
Software

- **Moses** statistical machine translation toolkit
  - developed since 2006
  - reference implementation of state-of-the-art methods
  - used in academia as benchmark and testbed
  - extensive commercial deployment
  - [http://www.statmt.org/moses/](http://www.statmt.org/moses/)

- **DL4MT** (or **Nematus**) neural translation toolkit
  - developed since 2016
  - state-of-the-art performance in 2016
  - [https://github.com/rsennrich/nematus](https://github.com/rsennrich/nematus)
New chapter on neural machine translation:
Thank You

questions?