Recall: Phrase-Based MT

\[ P(e|f) \propto P(f|e)P(e) \]

Noisy channel model: combine scores from translation model + language model to translate foreign to English

Recall: HMM for Alignment

Sequential dependence between a’s to capture monotonicity

\[ P(f,a|e) = \prod_{i=1}^{n} P(f_i|a_i)P(a_i|a_{i-1}) \]

a. Thank you, I shall do so gladly.

f. Gracias, lo hare de muy buen grado.

Alignment dist parameterized by jump size: \( P(a_j - a_{j-1}) \)

\[ P(f_i|e_{a_i}) \]: word translation table

Brown et al. (1993)
Recall: Decoding

Scores from language model $P(e)$ + translation model $P(f|e)$

This Lecture

- Syntactic MT
- Neural MT details
- Dilated CNNs for MT
- Transformers for MT

Levels of Transfer: Vauquois Triangle

Is syntax a “better” abstraction than phrases?

Slide credit: Dan Klein
Syntactic MT

- Rather than use phrases, use a *synchronous context-free grammar*: constructs “parallel” trees in two languages simultaneously

\[
\begin{align*}
\text{NP} & \rightarrow [\text{DT}_1 \text{JJ}_2 \text{NN}_3; \text{DT}_1 \text{NN}_3 \text{JJ}_2] \\
\text{DT} & \rightarrow [\text{the}, \text{la}] \\
\text{NN} & \rightarrow [\text{car, voiture}] \\
\text{JJ} & \rightarrow [\text{yellow, jaune}]
\end{align*}
\]

- Assumes parallel syntax up to reordering
- Translation = parse the input with “half” the grammar, read off other half

Neural MT

- Relax this by using lexicalized rules, like “syntactic phrases”
- Leads to HUGE grammars, parsing is slow

Encoder-Decoder MT

- Sutskever seq2seq paper: first major application of LSTMs to NLP
- Basic encoder-decoder with beam search

<table>
<thead>
<tr>
<th>Method</th>
<th>test BLEU score (ntst14)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Single forward LSTM, beam size 12</td>
<td>26.17</td>
</tr>
<tr>
<td>Single reversed LSTM, beam size 12</td>
<td>30.59</td>
</tr>
<tr>
<td>Ensemble of 5 reversed LSTMs, beam size 1</td>
<td>33.00</td>
</tr>
<tr>
<td>Ensemble of 2 reversed LSTMs, beam size 12</td>
<td>33.27</td>
</tr>
<tr>
<td>Ensemble of 5 reversed LSTMs, beam size 2</td>
<td>34.50</td>
</tr>
<tr>
<td>Ensemble of 5 reversed LSTMs, beam size 12</td>
<td>34.81</td>
</tr>
</tbody>
</table>

- SOTA = 37.0 — not all that competitive…
**Results: WMT English-French**

- **12M sentence pairs**
- Classic phrase-based system: ~33 BLEU, uses additional target-language data
  - Rerank with LSTMs: 36.5 BLEU (long line of work here; Devlin+ 2014)
- Sutskever+ (2014) seq2seq single: 30.6 BLEU
- Sutskever+ (2014) seq2seq ensemble: 34.8 BLEU
- Luong+ (2015) seq2seq ensemble with attention and rare word handling: 37.5 BLEU
- But English-French is a really easy language pair and there’s tons of data for it! Does this approach work for anything harder?

---

**Results: WMT English-German**

- **4.5M sentence pairs**
- Classic phrase-based system: 20.7 BLEU
  - Luong+ (2014) seq2seq: 14 BLEU
  - Luong+ (2015) seq2seq ensemble with rare word handling: 23.0 BLEU
- BLEU isn’t comparable across languages, but this performance still isn’t as good
- French, Spanish = easiest
  - German, Czech, Chinese = harder
- Japanese, Russian = hard (grammatically different, lots of morphology...)

---

**MT Examples**

```
src In einem Interview sagte Bloom jedoch , dass er und Kerr sich noch immer lieben .
ref However , in an interview , Bloom has said that he and Kerr still love each other .
best In an interview , however , Bloom said that he and Kerr still love .
base However , in an interview , Bloom said that he and Tina were still <unk> .
```

- best = with attention, base = no attention
- NMT systems can hallucinate words, especially when not using attention — phrase-based doesn’t do this

Luong et al. (2015)
### MT Examples

**Source**

The austerity imposed by Berlin and the European Central Bank, coupled with the strict austerity measures imposed by Berlin and the European Central Bank in connection with the straitjacket in which the respective national economy is forced to adhere to the common currency has led many people to think that Project Europe has gone too far.

**Reference**

Because of the strict austerity measures imposed by Berlin and the Federal Central Bank with the strict austerity imposed on the national economy in the face of the single currency, many people believe that the European project has gone too far.

- best = with attention, base = no attention

Luong et al. (2015)

### MT Examples

**Source**

such changes in reaction conditions include , but are not limited to , an increase in temperature or change in pH.

**Reference**

<table>
<thead>
<tr>
<th>Source</th>
<th>Reference</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Zhang et al. (2017)

### Handling Rare Words

- Words are a difficult unit to work with: copying can be cumbersome, word vocabularies get very large
- Character-level models don’t work well
- Compromise solution: use thousands of “word pieces” (which may be full words but may also be parts of words)

Input: _the _eco tax _port i co _in _Po nt - de - Bui is _Portique _éco taxe _de Pont - de - Buis...

Output: _le _port ique _eco taxe _de _Pont - de - Buis

- Can achieve transliteration with this, subword structure makes some translations easier to achieve

Sennrich et al. (2016)

### Byte Pair Encoding (BPE)

- Start with every individual byte (basically character) as its own symbol

```python
for i in range(num_merges):
    pairs = get_stats(vocab)
    best = max(pairs, key=pairs.get)
    vocab = merge_vocabulary(best, vocab)
```

- Count bigram character cooccurrence
- Merge the most frequent pair of adjacent characters

- Do this either over your vocabulary (original version) or over a large corpus (more common version)
- Doing 8k merges => vocabulary of around 8000 word pieces. Includes many whole words
- Most SOTA NMT systems use this on both source + target

Sennrich et al. (2016)
### Word Pieces

while voc size < target voc size:
- Build a language model over your corpus
- Merge pieces that lead to highest improvement in language model perplexity

- Issues: what LM to use? How to make this tractable?
- SentencePiece library from Google: unigram LM
- Result: way of segmenting input appropriate for translation

Schuster and Nakajima (2012), Wu et al. (2016), Kudo and Richardson (2018)

### Google’s NMT System

8-layer LSTM encoder-decoder with attention, word piece vocabulary of 8k-32k

Google’s NMT System

Luong+ (2015) seq2seq ensemble with rare word handling: 37.5 BLEU
Google’s 32k word pieces: 38.95 BLEU

Google’s phrase-based system: 37.0 BLEU
Luong+ (2015) seq2seq ensemble with rare word handling: 23.0 BLEU
Google’s 32k word pieces: 24.2 BLEU

Human Evaluation (En-Es)

- Similar to human-level performance on English-Spanish

Wu et al. (2016)
Google’s NMT System

Source: She was spotted three days later by a dog walker trapped in the quarry
PBMT: Elle a été repérée trois jours plus tard par un promeneur de chiens piégé dans la carrière 6.0
GNMT: Elle a été repérée trois jours plus tard par un traîneur de chiens piégé dans la carrière. 2.0
Human: Elle a été repérée trois jours plus tard par une personne qui prononçait son chien coincée dans la carrière 5.0

Gender is correct in GNMT but not in PBMT
“sled” “walker”

Wu et al. (2016)

Backtranslation

• Classical MT methods used a bilingual corpus of sentences $B = (S, T)$ and a large monolingual corpus $T'$ to train a language model. Can neural MT do the same?

• Approach 1: force the system to generate $T'$ as targets from null inputs

• Approach 2: generate synthetic sources with a $T\rightarrow S$ machine translation system (backtranslation)

Sennrich et al. (2015)

<table>
<thead>
<tr>
<th>name</th>
<th>training data</th>
<th>BLEU</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>instances</td>
<td>tst2011</td>
</tr>
<tr>
<td>baseline (Güçlü et al., 2015)</td>
<td></td>
<td>18.4</td>
</tr>
<tr>
<td>deep fusion (Güçlü et al., 2015)</td>
<td></td>
<td>20.2</td>
</tr>
<tr>
<td>baseline</td>
<td>parallel</td>
<td>18.6</td>
</tr>
<tr>
<td>parallel.synth</td>
<td>parallel/parallel.synth</td>
<td>19.9</td>
</tr>
<tr>
<td>Gigaword mono</td>
<td>parallel/Gigaword mono</td>
<td>18.8</td>
</tr>
<tr>
<td>Gigaword synth</td>
<td>parallel/Gigaword synth</td>
<td>21.2</td>
</tr>
</tbody>
</table>

• Gigaword: large monolingual English corpus
• parallel.synth: backtranslate training data; makes additional noisy source sentences which could be useful

Sennrich et al. (2015)
Recall: Self-Attention

- Each word forms a "query" which then computes attention over each word
  \[
  \alpha_{i,j} = \text{softmax}(x_i^T x_j) \quad \text{scalar}
  \]
  \[
  x_i' = \sum_{j=1}^{n} \alpha_{i,j} x_j \quad \text{vector sum of scalar \times vector}
  \]

- Multiple "heads" analogous to different convolutional filters. Use parameters \( W_k \) and \( V_k \) to get different attention values + transform vectors
  \[
  \alpha_{k,i,j} = \text{softmax}(x_i^T W_k x_j) \quad x_{k,i} = \sum_{j=1}^{n} \alpha_{k,i,j} V_k x_j
  \]

Transformers

- Encoder and decoder are both transformers
  - Decoder consumes the previous generated token (and attends to input), but has no recurrent state

- Big = 6 layers, 1000 dim for each token, 16 heads,
  base = 6 layers + other params halved

Augment word embedding with position embeddings, each dim is a sine/cosine wave of a different frequency. Closer points = higher dot products

Works essentially as well as just encoding position as a one-hot vector

<table>
<thead>
<tr>
<th>Model</th>
<th>BLEU</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>EN-DE</td>
</tr>
<tr>
<td>ByteNet [18]</td>
<td>23.75</td>
</tr>
<tr>
<td>GNMT + RL [38]</td>
<td>24.6</td>
</tr>
<tr>
<td>ConvS25 [9]</td>
<td>25.16</td>
</tr>
<tr>
<td>MoE [32]</td>
<td>26.03</td>
</tr>
<tr>
<td>Deep-Att + PosUnk Ensemble [39]</td>
<td>40.4</td>
</tr>
<tr>
<td>GNMT + RL Ensemble [38]</td>
<td>26.30</td>
</tr>
<tr>
<td>Transformer (base model)</td>
<td>27.3</td>
</tr>
<tr>
<td>Transformer (big)</td>
<td><strong>28.4</strong></td>
</tr>
</tbody>
</table>
Takeaways

- Can build MT systems with LSTM encoder-decoders, CNNs, or transformers
- Word piece / byte pair models are really effective and easy to use
- State of the art systems are getting pretty good, but lots of challenges remain, especially for low-resource settings
- Next time: pre-trained transformer models (BERT), applied to other tasks