Announcements
- A4 due tonight
- A5 out tonight, due next Tuesday

Recap Phrase-based MT
1. Word alignment (src → trg / trg → src, intersect)
2. Phrase extraction
3. Decoding: search over phrase lattice + LM

Today Seq-to-seq models / Neural MT
1. Connections to RNN LMs
2. Training
3. Problems
4. Attention
5. Mechanism
6. Connections to alignment

Seq-to-seq models for MT
Sutskever et al. NeurIPS 2014
RNN "encoder" $\overline{h}$, RNN "decoder" $v$

Je vais le faire

Whole input sent is "encoded" into $\overline{h}/c$ here

NMT: encode src w/RNN, "decode" trg
with another RNN (diff params)

$$P(t | \overline{s}) = P(t_1 | \overline{s}) P(t_2 | t_1, \overline{s}) P(t_3 | t_2, \overline{s}) \ldots$$

$$P(t_i | t_{i-1}, \overline{s}) = \text{Softmax} \left( W^{(d)} \overline{h}_i \right)$$

$W^{(d)}$: trg vocab x hidden size mat

hidden state at $i$th step of decoder

Like an RNN LM additionally conditioned on $\overline{s}$

Decoder: feed the previous predicted word into next decoder timestep
Decoding: $t_0 = \langle s \rangle$
while $t_i \neq \text{STOP}$

$$t_i = \text{arg max}_t P(\{ t_1 \cdots t_{i-1}, s \})$$

Training looks like LM training

Recall: $w_1, w_2, w_3$  \[ \text{LM: } \]  \[ \text{NMT: } \]

Encoder trained too!
Teaching the model to predict till the correct \( t_i \) up to this point

What do we really want? Model’s predicted sequence \( \hat{t}_1, \ldots, \hat{t}_n \) has high BLEU score.

Requires reinforcement learning.

Details: Lengths can vary, need to pad. Lengths in batch:

- LM: \[ \text{20 chars} \]
- MT: \[ \text{src trg} \]

need padding + smart handling.
What goes wrong?

1) Repetition

Je vais le faire → I am going to do it going to going to going to...

Won’t happen in PBMT (every word translated as part of one phase)

Need a notion of coverage in NMT

2) Unknown words

Decoder has a fixed vocab |V_t|

PBMT: “copy” rules

Pont-de-Buis → copy to output

NMT: produce UNK
1. Repetition (going to going to)
2. UNKs
3. Poor performance on long sents

\[ \text{BLEU} \quad \text{PBMT} \quad \text{PBMT} \quad \text{NMT} \quad \text{NMT} \]

Bahdanau et al. (2015) (attention)

NMT: fixed size $h/c$ vectors don't scale arbitrarily
LSTMs can't remember for too long

Sutskever: reverse input as a trick to address this

\[ s_m \ s_{m-1} \rightarrow s_1 \rightarrow t_1 \ t_2 \cdots \]

\[ \text{at least remember the first few s} \]
Attention: Je fais un bureau → I make a desk
What if we know translation should be word-by-word?

Hack

\[
\begin{array}{cccc}
\mathbf{J} & \mathbf{f} & \mathbf{u} & \mathbf{b} \\
\end{array}
\] → \[
\begin{array}{c}
\mathbf{I} \\
\end{array}
\] \text{make} a
\[
\begin{array}{c}
\mathbf{J} \\
\end{array}
\] → \[
\begin{array}{c}
\mathbf{s} \rightarrow \text{Je fais un} \\
\end{array}
\] to \[
\begin{array}{c}
\mathbf{s}_1, \mathbf{s}_2 \\
\end{array}
\]

Manually feel \( s_i \) into \( i \)-th decoder step (along with \( t_{i-1} \))

Q: What problems can this fix? (1-3)

1+3  ✓ 1 Less likely to repeat if we see explicit words

✓ 3 Anchored more to input

✗ 2 Requires changing output layer
I am going to do it

```
je vais le faire
```

- Run out of s before end of t?
- Ordering is inflexible

**Attention** Target decoder picks where to look in source

```
\[ \text{ATTN} = \text{inputs} \]
\[ \frac{h(e)}{h_j}, \frac{h(e)}{h_i} \]
\[ i = 1 \]
\[ \alpha = \text{softmax} \]
\[ \frac{h(e)}{h_1}, \frac{h(e)}{h_2}, \ldots, \frac{h(e)}{h_m} \]
\[ \sum_{j=v}^f \alpha_j \]
```

\( \alpha \) distribution over source posns
\[ \overline{C}_i = \sum_{j=1}^{m} \alpha_j \cdot \overline{h}_j \quad \text{weighted average of} \quad \overline{h}_j \quad \text{with} \quad \alpha \quad \text{as weights} \]

\[ P(t_i | s, t_{i-1}) = \text{softmax} \left( W^{(d)} \begin{bmatrix} \overline{h}_i^{(d)} \\ \overline{h}_i^{(e)} \end{bmatrix} \right) \]

\[ \alpha \text{ can also use other funs to compare } \overline{h}_i^{(d)} / \overline{h}_i^{(e)} \text{ besides dot} \]

\[ \alpha_i = [0.99, 0.003, 0.003, 0.003] \]

Output: \[ \text{softmax} \left( W^{(d)} \begin{bmatrix} \overline{h}_i^{(d)} \\ \overline{h}_i^{(e)} \end{bmatrix} \right) \]

Think about Je fais un bureau

\[ \nabla \text{ Je} \]

No attn \[ \Rightarrow \] w/ attn

\[ \text{gradient} \]
Je vais le faire  
I am going to do it

Learning mapping  1 2 2 4 4 3

- Not input at train time!

Je vais le faire  →  1 2 2 4 4 3

- easier than translation

- regular mapping

- This is learnable!