Natural Language Processing

Info 159/259
Lecture 25: Machine Translation (April 22, 2021)

David Bamman, UC Berkeley
Info 259
Project presentations

• 2-3:30pm Thursday 4/29

• Prepare a 5-minute presentation of your project to present to the class; be prepared to take questions from the audience.

• The project presentations won’t be recorded.
When I look at an article in Russian, I say: 'This is really written in English, but it has been coded in some strange symbols. I will now proceed to decode.'

Weaver 1955
# Machine Translation

<table>
<thead>
<tr>
<th>Task</th>
<th>X</th>
<th>Y</th>
</tr>
</thead>
<tbody>
<tr>
<td>Sentiment analysis</td>
<td>I hate this movie!</td>
<td>negative</td>
</tr>
<tr>
<td>POS tagging</td>
<td>I hate this movie!</td>
<td>PRP VB DT NN .</td>
</tr>
<tr>
<td>Parsing</td>
<td>I hate this movie!</td>
<td>[tree]</td>
</tr>
<tr>
<td>MT</td>
<td>Lasciate ogni speranza, voi ch'entrare</td>
<td>Abandon all hope, you who enter!</td>
</tr>
<tr>
<td>Conversational agent</td>
<td>How are you?</td>
<td>I’m great!</td>
</tr>
</tbody>
</table>
Lasciate ogni speranza, voi ch'entrate

Abandon all hope, ye who enter here
There are many English translations of this famous line. Some examples include:

- *All hope abandon, ye who enter here* – Henry Francis Cary (1805–1814)
- *All hope abandon, ye who enter in!* – Henry Wadsworth Longfellow (1882)
- *Leave every hope, ye who enter!* – Charles Eliot Norton (1891)
- *Leave all hope, ye that enter* – Carlyle Okey-Wicksteed (1932)
- *Lay down all hope, you that go in by me.* – Dorothy L. Sayers (1949)
- Abandon all hope, ye who enter here – John Ciardi (1954)
- Abandon every hope, you who enter. – Charles S. Singleton (1970)
- No room for hope, when you enter this place – C. H. Sisson (1980)
- Abandon every hope, who enter here. – Allen Mandelbaum (1982)
- Abandon every hope, all you who enter – Mark Musa (1995)

Verbatim, the line translates as "Leave (*lasciate*) every (*ogni*) hope (*speranza*), ye (*voi*) that (*ch") enter (*intrate)*."
Data

• Modern machine translation systems are learned from parallel texts: pairs of documents in two languages that have been aligned at the sentence level.
<table>
<thead>
<tr>
<th>Reprise de la session</th>
<th>Resumption of the session</th>
</tr>
</thead>
<tbody>
<tr>
<td>Je déclare reprise la session du Parlement européen qui avait été interrompue le vendredi 17 décembre dernier et je vous renouvelle tous mes vux en espérant que vous avez passé de bonnes vacances.</td>
<td>I declare resumed the session of the European Parliament adjourned on Friday 17 December 1999, and I would like once again to wish you a happy new year in the hope that you enjoyed a pleasant festive period.</td>
</tr>
<tr>
<td>Comme vous avez pu le constater, le grand &quot;bogue de l'an 2000&quot; ne s'est pas produit. En revanche, les citoyens d'un certain nombre de nos pays ont été victimes de catastrophes naturelles qui ont vraiment été terribles.</td>
<td>Although, as you will have seen, the dreaded 'millennium bug' failed to materialise, still the people in a number of countries suffered a series of natural disasters that truly were dreadful.</td>
</tr>
</tbody>
</table>
Data

- Europarl (proceedings of European parliament, 50M words/language)
  http://www.statmt.org/europarl/

- UN Corpus (United Nations documents, six languages, 300M words/language)
  http://www.euromatrixplus.net/multi-un/

- Common crawl (Web documents, long tail of language pairs)
Evaluation

Tell me Muse, of the man of many ways

Sing to me of the man, Muse, the man of twists and turns

Tell me about a complicated man
Evaluation

• BLEU (Papineni et al. 2002): what fraction of {1-4}-grams in the system translation appear in the reference translations?

\[ p_n = \frac{\text{Number of ngrams in system and reference translations}}{\text{Number of ngrams in system translation}} \]

\[ \text{BLEU} = BP \times \exp \left( \frac{1}{N} \sum_{n=1}^{N} \log p_n \right) \]
Orejuela appeared calm as he was led to the American plane that will take him to Miami, Florida.

Orejuela appeared calm while being escorted to the plane that would take him to Miami, Florida.

Orejuela appeared calm as he was being led to the American plane that was to carry him to Miami in Florida.

Orejuela seemed quite calm as he was being led to the American plane that would take him to Miami in Florida.

\[ p_1 = \frac{15}{18} = 0.833 \]

Ngrams appearing >1 time in the hypothesis can match up to the max number of times they appear in a single reference — e.g., two commas in hypothesis but one max in any single reference.

Callison-Burch et al. (2006), Re-evaluating the Role of BLEU in Machine Translation Research
<table>
<thead>
<tr>
<th>Hypothesis translation</th>
<th>Reference translations</th>
</tr>
</thead>
<tbody>
<tr>
<td>Appeared calm when he was taken to the American plane, which will to Miami, Florida.</td>
<td>Orejuela appeared calm as he was led to the American plane which will take him to Miami, Florida.</td>
</tr>
<tr>
<td></td>
<td>Orejuela appeared calm while being escorted to the plane that would take him to Miami, Florida.</td>
</tr>
<tr>
<td></td>
<td>Orejuela appeared calm as he was being led to the American plane that was to carry him to Miami in Florida.</td>
</tr>
<tr>
<td></td>
<td>Orejuela seemed quite calm as he was being led to the American plane that would take him to Miami in Florida.</td>
</tr>
</tbody>
</table>

\[
p_2 = \frac{10}{17} = 0.588
\]
We could optimize the score by minimizing the denominator (the number of ngrams generated).

Brevity penalty:

\[ p_n = \frac{\text{Number of ngrams in system and reference translations}}{\text{Number of ngrams in system translation}} \]

\[ \text{BLEU} = BP \times \exp \frac{1}{N} \sum_{n=1}^{N} \log p_n \]

- \( c = \) length of hypothesis translation
- \( r = \) length of closest reference translation

Brevity penalty:

\[ BP = \begin{cases} 1 & \text{if } c > r \\ e^{1-r/c} & \text{if } c \leq r \end{cases} \]
Statistical MT
# Noisy Channel

**Table:**

<table>
<thead>
<tr>
<th></th>
<th>X</th>
<th>Y</th>
</tr>
</thead>
<tbody>
<tr>
<td>ASR</td>
<td>speech signal</td>
<td>transcription</td>
</tr>
<tr>
<td>MT</td>
<td>target text</td>
<td>source text</td>
</tr>
<tr>
<td>OCR</td>
<td>pixel densities</td>
<td>transcription</td>
</tr>
</tbody>
</table>

Mathematical expression:

\[
P(Y \mid X) \propto P(X \mid Y) \quad \underbrace{P(Y)}_{\text{channel model}} \quad \underbrace{P(Y)}_{\text{source model}}
\]
Noisy Channel

\[ P(Y \mid X) \propto P(X \mid Y) \quad P(Y) \]

- channel model
- source model

• If we’re translating from English (X) into French (Y) we assume some true French sentence Y that was “corrupted” into English version X.
Noisy Channel

\[ P(Y \mid X) \propto P(X \mid Y) \]

channel model

source model

Estimate this from parallel texts

Estimate this from monolingual data
The statistical revolution in machine translation (1990) started by exploiting the structure of parallel sentences to learn the translation model.

Lasciate ogni speranza, voi ch'entrate

Abandon all hope, you who enter!

Brown et al. (1990), "A statistical approach to machine translation," Computational Linguistics
Lasciate ogni speranza, voi ch'entrate
Abandon all hope, you who enter!

mi lasciate in pace
Leave me in peace

Lasciate i monti
Leave the mountains
Lasciate ogni speranza, voi ch'entrate
Abandon all hope, you who enter!

mi lasciate in pace
Leave me in peace

Lasciate i monti
Leave the mountains
## Statistical MT

| Italian       | English            | P(English | Italian) |
|---------------|--------------------|------------|
| lasciate      | leave              | 0.67       |
| lasciate      | abandon            | 0.33       |

Translation table

| Italian       | English                  | P(English | Italian) |
|---------------|--------------------------|------------|
| Voi ch’entrate| you who enter            | 0.91       |
| Voi ch’entrate| you who are entering     | 0.09       |

Phrase translation table
IBM Alignment models

If we had explicit word alignments we could estimate translation tables directly from them.

mi lasciate in pace
Leave me in peace

Lasciate i monti
Leave the mountains

But we don’t have word alignments — just sentence alignments!
IBM Alignment models

Unsupervised models for aligning words and phrases in parallel sentences.

mi lasciate in pace
Leave me in peace

Lasciate i monti
Leave the mountains

## IBM Alignment models

<table>
<thead>
<tr>
<th>Model</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Model 1</td>
<td>Independent word translation (order doesn’t matter)</td>
</tr>
<tr>
<td>Model 2</td>
<td>Word translation + distance between source and target position</td>
</tr>
<tr>
<td>Model 3</td>
<td>Word translation + fertility (how many target words a source word can align to)</td>
</tr>
<tr>
<td>Model 4</td>
<td>Word translation + relative ordering among target words of same source</td>
</tr>
<tr>
<td>Model 5</td>
<td>(Fixes deficiency of model 4)</td>
</tr>
<tr>
<td>HMM</td>
<td>Word translation plus relative ordering</td>
</tr>
</tbody>
</table>

---

Neural MT

- Encoder-decoder
- Encoder-decoder + attention
- Transformer (Vaswani et al. 2018)
Sutskever et al. (2015);

Encoder-decoder framework

K-dimensional vector representing entire context

Condition on word generated in translation
Je suis heureux
Training

• As in other RNNs, we can train by minimizing the loss between what we predict at each time step and the truth.
I'm you are the ...

<table>
<thead>
<tr>
<th>I'm</th>
<th>you</th>
<th>are</th>
<th>the</th>
<th>...</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>I'm</th>
<th>you</th>
<th>are</th>
<th>the</th>
<th>...</th>
</tr>
</thead>
<tbody>
<tr>
<td>0.03</td>
<td>0.05</td>
<td>0.02</td>
<td>0.01</td>
<td>0.009</td>
</tr>
</tbody>
</table>

Training

Je

suis

heureux

EOS
Je suis heureux

truth

|   | happy | great | bad  | ok  | ...
|---|-------|-------|------|-----|------
| 1 | 0     | 0     | 0    | 0   | 0    |

predicted

|   | happy | great | bad  | ok  | ...
|---|-------|-------|------|-----|------
| 0.13 | 0.08 | 0.01 | 0.03 | 0.009 |
Encoder-decoder

• Sutskever et al. (2014) found better performance when the encoder reads the sentence in backwards, from right to left (increase in BLEU from 25.9 to 30.6)

Sutskever et al. (2014), “Sequence to Sequence Learning with Neural Networks”
The entire source sentence is summarized in this one vector.

The decoder state depends just on the previous state and the previous output.

\[ s_i = f(s_{i-1}, y_{i-1}) \]
Encoder-decoder with attention

The decoder state depends just the previous state, the previous output, and some context.

\[ s_i = f(s_{i-1}, y_{i-1}, c_i) \]

Je suis heureux

0.1
0.20
0.31
-1.4
0.8

0.8
-0.13
-0.78
1.78
3.2

0.5
0.3
-0.7
3.2
0.1
Encoder-decoder with attention

\[ c = h_1a_1 + h_2a_2 + h_3a_3 \]

\[ s_i = f(s_{i-1}, y_{i-1}, c_i) \]
Encoder-decoder with attention

\[ c = h_1a_1 + h_2a_2 + h_3a_3 \]

\[ s_i = f(s_{i-1}, y_{i-1}, c_i) \]
Encoder-decoder with attention

\[ c = h_1 a_1 + h_2 a_2 + h_3 a_3 \]
Encoder-decoder with attention

• Each time step in the decoder has its own weighted context vector

Je suis heureux

2.7 3.1 -1.4 -2.3

2.7 3.1 -1.4 -2.3

2.7 3.1 -1.4 -2.3

2.7 3.1 -1.4 -2.3

weighted sum

Je

suis

heureux
With document classification, we parameterized attention with a single vector \( v \) to be learned. Attention in an encoder-decoder network is a little different because we’re comparing a pair of vectors.

\[
\begin{align*}
  r_1 &= v^\top x_1 \\
  r_2 &= v^\top x_2 \\
  r_3 &= v^\top x_3 \\
  r_4 &= v^\top x_4 \\
  r_5 &= v^\top x_5
\end{align*}
\]
Encoder-decoder with attention

\[ c = h_1a_1 + h_2a_2 + h_3a_3 \]

\[ r_{1,j} = FFNN(h_1, s_{j-1}) \]
Feed-forward neural network

The feed-forward network here just takes the two vectors as input as outputs a single scalar. The parameters are all learned using backprop (just like every other parameter).
Encoder-decoder with attention

\[ a = \text{softmax}(r) \]

\[ r = [r_1, r_2, r_3] \]
Encoder-decoder with attention

\[ c = h_1a_1 + h_2a_2 + h_3a_3 \]

\[ a = \text{softmax}(r) \]

\[ r = [r_1, \ldots, r_5] \]

\[ s_i = f(s_{i-1}, y_{i-1}, c_i) \]

\[ h = \begin{bmatrix} 0.7 & 0.8 & 5.4 & 2.3 \\ \end{bmatrix} \]

\[ r_1 = \text{FFNN}(h_1, s_{j-1}) \]

\[ r_2 = \text{FFNN}(h_2, s_{j-1}) \]

\[ r_3 = \text{FFNN}(h_3, s_{j-1}) \]

h

Je

suis

heureux

EOS

happy

I’m

I’m

happy
Attention

• For text classification, attention helps decide which words in the text are important for the label; a document has a single attention vector.

• For MT, attention changes with each word being generated during decoding. Each subsequent word pays attention to different parts of the input.
Better performance on long sentences

The agreement on the European Economic Area was signed in August 1992.

It should be noted that the marine environment is the least known of environments.
Attention

- RNNs are hard to parallelize; important factor for long sequence lengths.
- Attention gives us access to an entire input sequence. Why do we need recurrence at all?
• Transformer network (Vaswani et al. 2017).
Self-attention

- Multiple layers of representations for an input sequence; each layer attends over the representations in the previous layer.
Self-attention

Je suis heureux

Layer 1

Embedding layer

weighted sum

Je  

suis  

heureux
Self-attention

Layer 1

Embedding layer

Je
suis
heureux
Self-attention

Layer 1

Weighted sum

Embedding layer

Je

suis

heureux
Self-attention

Layer 1

Layer 2

weighted sum

Embedding layer

Je

suis

heureux
Self-attention

Layer 1

Layer 2

Embedding layer

Je

suis

heureux
Self-attention

Layer 1

Layer 2

Embedding layer

Je

suis

heureux

weighted sum
Self-attention

- In the decoder, self-attention can only attend over words to the left of the position (since the right ones haven’t been generated yet).
Self-attention

• In the decoder, self-attention can only attend over words to the left of the position (since the right ones haven’t been generated yet).
Self-attention

- In the decoder, self-attention can only attend over words to the left of the position (since the right ones haven’t been generated yet).
Self-attention

- In the decoder, self-attention can only attend over words to the left of the position (since the right ones haven’t been generated yet).
Self-attention

- In the decoder, self-attention can only attend over words to the left of the position (since the right ones haven’t been generated yet).
Self-attention

- In the decoder, self-attention can only attend over words to the left of the position (since the right ones haven’t been generated yet).
Encoder-decoder attention

Je suis heureux

I am
Encoder-decoder attention

- Each layer in the decoder attends over the encoder output (as usual).
• Self-attention captures structure in the input (like coreference)
• Multiple heads capture different structure.
<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>ConvS2S [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></td>
</tr>
<tr>
<td>GNMT + RL Ensemble [38]</td>
<td>26.30</td>
</tr>
<tr>
<td>ConvS2S Ensemble [9]</td>
<td>26.36</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>