Algorithms for NLP

Neural Machine Translation

Yulia Tsvetkov – CMU
Slides: Chris Dyer – DeepMind
<table>
<thead>
<tr>
<th>Languages</th>
<th>Languages</th>
<th>Languages</th>
<th>Languages</th>
<th>Languages</th>
<th>Languages</th>
</tr>
</thead>
<tbody>
<tr>
<td>Afrikaans</td>
<td>Czech</td>
<td>Hebrew</td>
<td>Latin</td>
<td>Portuguese</td>
<td>Tajik</td>
</tr>
<tr>
<td>Albanian</td>
<td>Danish</td>
<td>Hindi</td>
<td>Latvian</td>
<td>Punjabi</td>
<td>Tamil</td>
</tr>
<tr>
<td>Amharic</td>
<td>Dutch</td>
<td>Hmong</td>
<td>Lithuanian</td>
<td>Romanian</td>
<td>Telugu</td>
</tr>
<tr>
<td>Arabic</td>
<td>English</td>
<td>Hungarian</td>
<td>Luxembourg</td>
<td>Russian</td>
<td>Thai</td>
</tr>
<tr>
<td>Armenian</td>
<td>Esperanto</td>
<td>Icelandic</td>
<td>Macedonian</td>
<td>Samoan</td>
<td>Turkish</td>
</tr>
<tr>
<td>Azerbaijani</td>
<td>Estonian</td>
<td>Igbo</td>
<td>Malagasy</td>
<td>Scots Gaelic</td>
<td>Ukrainian</td>
</tr>
<tr>
<td>Basque</td>
<td>Filipino</td>
<td>Indonesian</td>
<td>Malay</td>
<td>Serbian</td>
<td>Urdu</td>
</tr>
<tr>
<td>Belarusian</td>
<td>Finnish</td>
<td>Irish</td>
<td>Malayalam</td>
<td>Sesotho</td>
<td>Uzbek</td>
</tr>
<tr>
<td>Bengali</td>
<td>French</td>
<td>Italian</td>
<td>Maltese</td>
<td>Shona</td>
<td>Vlaxianese</td>
</tr>
<tr>
<td>Bosnian</td>
<td>Frisian</td>
<td>Japanese</td>
<td>Maori</td>
<td>Sindhi</td>
<td>Welsh</td>
</tr>
<tr>
<td>Bulgarian</td>
<td>Galician</td>
<td>Javanese</td>
<td>Marathi</td>
<td>Sinhala</td>
<td>Xhosa</td>
</tr>
<tr>
<td>Catalan</td>
<td>Georgian</td>
<td>Kannada</td>
<td>Mongolian</td>
<td>Slovak</td>
<td>Yiddish</td>
</tr>
<tr>
<td>Cebuano</td>
<td>German</td>
<td>Kazakh</td>
<td>Myanmar (Burmese)</td>
<td>Slovenian</td>
<td>Yoruba</td>
</tr>
<tr>
<td>Chichewa</td>
<td>Greek</td>
<td>Khmer</td>
<td>Nepali</td>
<td>Somali</td>
<td>Zulu</td>
</tr>
</tbody>
</table>
Noisy Channel Model

\[ \hat{e} = \arg \max_{e} p_{\varphi}(e) \times p_{\theta}(f \mid e) \]

- Sent transmission: English
- Received transmission: “French”
- Decoder: Recovered message English’
- Language model
- Translation model
Phrase-Based MT

**Translation Model** \( P(f|e) \)

**Language Model** \( P(e) \)

**Reranking Model**

\[
\arg \max_e P(f|e)P(e)
\]
Two Views of MT

- **Code breaking (aka the noisy channel, Bayes rule)**
  - I know the target language
  - I have example translations texts (example enciphered data)
    \[ \hat{e} = \arg \max_{\hat{e}} p_\phi(e) \times p_\theta(f \mid e) \]

- **Direct modeling (aka pattern matching)**
  - I have really good learning algorithms and a bunch of example inputs (source language sentences) and outputs (target language translations)
    \[ \hat{e} = \arg \max_{e} p_\lambda(e \mid f) \]
Two Views of MT

- **Code breaking** (aka the noisy channel, Bayes rule)
  - I know the *target language*
  - I have example *translations texts* (example enciphered data)

  ![Statistical Machine Translation (SMT)](image)

- **Direct modeling** (aka pattern matching)
  - I have *really good learning algorithms* and a bunch of *example inputs* (source language sentences) and *outputs* (target language translations)

  ![Neural Machine Translation (NMT)](image)
MT as Direct Modeling

\[ \hat{e} = \arg \max_e p_x(e \mid f) \]

- one model does everything
- trained to reproduce a corpus of translations
In der Innenstadt explodierte eine Autobombe

A car bomb exploded downtown
Я увидела кОшку

I saw a cat

<EOS>
Я увидела кОшку

I saw a cat

Ilya Sutskever, Oriol Vinyals, Quoc V. Le. 2014. Sequence to Sequence Learning with Neural Networks. *Proc. NIPS*

I saw a cat.

Я увидела кОшку.

(sentence representation)
“You can’t cram the meaning of a whole %&!$ing sentence into a single %&!*$ing vector!”
— Ray Mooney
- Fixed sized representation degrades as sentence length increases
  - Reversing source brings some improvement (Sutskever et al., 2014)
Cho’s question: does a translator read and memorize the input sentence/document and then generate the output?

- Compressing the entire input sentence into a vector basically says “memorize the sentence”
- Common sense experience says translators refer back and forth to the input. (also backed up by eyetracking studies)
Sequence-to-Sequence Models for NMT

- By far the most widely used architecture is **Bidirectional RNN with Attention** due to Bahdanau et al (2015)

- **One column per word**
- Each column (word) has two halves concatenated together:
  - a “forward representation”, i.e., a word and its left context
  - a “reverse representation”, i.e., a word and its right context
- **Implementation**: bidirectional RNNs (GRUs or LSTMs) to read $f$ from left to right and right to left, concatenate representations
Encoder: Bidirectional RNN

\[
\begin{align*}
x_1 & \quad x_2 & \quad x_3 & \quad x_4 \\
\text{Ich} & \quad \text{möchte} & \quad \text{ein} & \quad \text{Bier}
\end{align*}
\]
Encoder: Bidirectional RNN

\[ \begin{array}{c}
\hat{h}_1 \rightarrow \hat{h}_2 \rightarrow \hat{h}_3 \rightarrow \hat{h}_4 \\
x_1 \rightarrow x_2 \rightarrow x_3 \rightarrow x_4 \\
\text{Ich} \rightarrow \text{möchte} \rightarrow \text{ein} \rightarrow \text{Bier}
\end{array} \]
Encoder: Bidirectional RNN
Encoder: Bidirectional RNN

\[ f_t = [\vec{h}_t; \vec{h}_t] \]

Input sequences:
- Ich
- möchte
- ein
- Bier
Encoder: Bidirectional RNN

\[ f_i = [\vec{h}_i; \vec{h}_i] \]

\[ x_1 \quad x_2 \quad x_3 \quad x_4 \]

Ich möchte ein Bier
Encoder: Bidirectional RNN

\[ f_i = [\overrightarrow{h_i}; \overrightarrow{h_i}] \]
Encoder: Bidirectional RNN

\[ f_i = [\vec{h}_i; \vec{h}_i] \]
Matrix Sentence Encoding

\[ f_i = [\vec{h}_i; \vec{h}_i] \]

- matrix-encoded sentence

\( \vec{h}_1 \rightarrow \vec{h}_2 \rightarrow \vec{h}_3 \rightarrow \vec{h}_4 \)

\( x_1 \rightarrow x_2 \rightarrow x_3 \rightarrow x_4 \)

\( F \in \mathbb{R}^{2n \times |f|} \)

Ich möchte ein Bier
Decoder: RNN + Attention

Ich möchte ein Bier
Attention history:

\[ a_i \]
Attention history:

$\alpha_j \begin{bmatrix} \bullet & \bullet & \bullet & \bullet \end{bmatrix}$
Ich möchte ein Bier
Ich möchte ein Bier
Attention history:

Ich möchte ein Bier
- Bahdanau et al. (2015) were the first to propose using **attention** for translating from matrix-encoded sentences
- **High-level idea**
  - Generate the output sentence word by word using an RNN
  - At each output position $t$, the RNN receives **two** inputs (in addition to any recurrent inputs)
    - a fixed-size vector embedding of the previously generated output symbol $e_{t-1}$
    - a fixed-size vector encoding a “view” of the input matrix
  - How do we get a fixed-size vector from a matrix that changes over time?
    - Bahdanau et al: do a weighted sum of the columns of input words based on how important they are at the **current time step**.
    - The weighting of the input columns at each time-step is called attention
Attention

- How do we know what to attend to at each timestep?
Minh-Thang Luong, Hieu Pham, Christopher D. Manning. 2015. Effective Approaches to Attention-based Neural Machine Translation. *Proc. EMNLP*

\[
\text{score}(h_t, \bar{h}_s) = \begin{cases} 
  h_t^\top \bar{h}_s \\
  h_t^\top W_a \bar{h}_s \\
  v_a^\top \tanh (W_a[h_t; \bar{h}_s])
\end{cases}
\]

(Luong et al. ‘15)

(Bahdanau et al.’15)
Minh-Thang Luong, Hieu Pham, Christopher D. Manning. 2015. Effective Approaches to Attention-based Neural Machine Translation. *Proc. EMNLP*
Attention vs Alignment

- Attention is similar to alignment, but there are important differences

(Cho et al. 2015)
Attention vs Alignment

- Attention is similar to alignment, but there are important differences
  - alignment makes stochastic but hard decisions: the model picks one word or phrase at a time
  - attention is “soft” (you add together all the words)
  - there is no guarantee that attention corresponds to alignment since information can also flow along recurrent connections

(Cho et al. 2015)
Conditional Language Models

- Speech recognition
- Vision
  - Image captioning
- NLP
  - NMT
  - Summarization
  - QA
  - Dialogue

Examples of attending to the correct object (white indicates the attended regions, underlines indicated the corresponding word)

- A woman is throwing a frisbee in a park.
- A dog is standing on a hardwood floor.
- A stop sign is on a road with a mountain in the background.

Show, Attend and Tell: Neural Image Caption Generation with Visual Attention (Xu et al. 2016)

(Chan et al. 2015)
Decoding

- **Exact search**
  - generate every possible sentence $T$ in target language
  - compute score $p(T|S)$ for each
  - pick best one
Decoding

- **Exact search**
  - generate every possible sentence $T$ in target language
  - compute score $p(T|S)$ for each
  - pick best one

  $\rightarrow$ intractable: $|\text{vocab}|^N$ translations for output length $N$
Decoding

- **Exact search**
  - generate every possible sentence $T$ in target language
  - compute score $p(T|S)$ for each
  - pick best one
  
  \[\rightarrow\text{intractable: }|\text{vocab}|^N\text{ translations for output length }N\]

- **Greedy search**
  - at each time stamp pick the most likely word
  \[
  \text{argmax } \log p(y_i|S, y<i)
  \]
  - until $<$EOS$>$
Decoding

- **Exact search**
  - generate every possible sentence $T$ in target language
  - compute score $p(T|S)$ for each
  - pick best one

  $\rightarrow$ intractable: $|\text{vocab}|^N$ translations for output length $N$

- **Greedy search**
  - at each time stamp pick the most likely word
    $\operatorname{argmax} \log p(y_i|S, y_{<i})$
  - until $<\text{EOS}>$

  $\rightarrow$ efficient, but heavily suboptimal
Decoding

- **Beam search**
  - maintain list of $K$ hypotheses (beam)
  - at each time step, expand each hypothesis
  - select hypotheses with highest total probability

$K = 3$

Image thanks to Rico Sennrich
Decoding with Beam Search

<table>
<thead>
<tr>
<th>Strategy</th>
<th># Chains</th>
<th>Valid Set NLL</th>
<th>Valid Set BLEU</th>
<th>Test Set NLL</th>
<th>Test Set BLEU</th>
</tr>
</thead>
<tbody>
<tr>
<td>Ancestral Sampling</td>
<td>50</td>
<td>22.98</td>
<td>15.64</td>
<td>26.25</td>
<td>16.76</td>
</tr>
<tr>
<td>Greedy Decoding</td>
<td>-</td>
<td>27.88</td>
<td>15.50</td>
<td>26.49</td>
<td>16.66</td>
</tr>
<tr>
<td>Beamsearch</td>
<td>5</td>
<td>20.18</td>
<td>17.03</td>
<td>22.81</td>
<td>18.56</td>
</tr>
<tr>
<td>Beamsearch</td>
<td>10</td>
<td>19.92</td>
<td>17.13</td>
<td>22.44</td>
<td>18.59</td>
</tr>
</tbody>
</table>

(Cho 2016)

(Koehn & Knowles 2017)
Dealing with Very Large Vocabulary

This is a <unk> sentence with very <unk> <unk> and <unk>.

\[ P(w_i|h) = \frac{e^{s_i}}{\sum_j e^{s_j}} \]
Dealing with Very Large Vocabulary

- **Sampling-based approximations**
  - Importance Sampling: evaluate the denominator over a subset
  - Noise Contrastive Estimation: convert to a proxy binary classification problem
- **Structure-based approximations**
  - Class-based Softmax: divide the vocabulary to multiple classes; first predict a class, then predict a word of the class
  - Hierarchical Softmax: binary tree with words as leaves
- **Self normalization** (Devlin et al. ‘2014, Andreas et al. 2015)
- **Subword Units**
  - Byte Pair Encoding (BPE) (Sennrich et al. ‘2016)
  → current standard
**Byte pair encoding for word segmentation**

- Repeatedly replace most frequent symbol pair (‘A’, ‘B’) with ‘AB’

<table>
<thead>
<tr>
<th>system</th>
<th>sentence</th>
</tr>
</thead>
<tbody>
<tr>
<td>source</td>
<td>health research institutes</td>
</tr>
<tr>
<td>reference</td>
<td>Gesundheitsforschungsinstitute</td>
</tr>
<tr>
<td>word-level (with back-off)</td>
<td>Forschungsinstitute</td>
</tr>
<tr>
<td>character bigrams</td>
<td>Forscher</td>
</tr>
<tr>
<td>BPE</td>
<td>Gesundheits</td>
</tr>
<tr>
<td>source</td>
<td>rakfisk</td>
</tr>
<tr>
<td>reference</td>
<td>ракфиска (rakfiska)</td>
</tr>
<tr>
<td>word-level (with back-off)</td>
<td>rakfisk → UNK → rakfisk</td>
</tr>
<tr>
<td>character bigrams</td>
<td>ra</td>
</tr>
<tr>
<td>BPE</td>
<td>rak</td>
</tr>
</tbody>
</table>
## Alternative to Softmax

<table>
<thead>
<tr>
<th></th>
<th>Sampling Based</th>
<th>Structure Based</th>
<th>Subword Units</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Training Time</strong></td>
<td>😊</td>
<td>😊</td>
<td>😊</td>
</tr>
<tr>
<td><strong>Test Time</strong></td>
<td>😞</td>
<td>😞</td>
<td>😊</td>
</tr>
<tr>
<td><strong>Accuracy</strong></td>
<td>😞</td>
<td>😞</td>
<td>😊</td>
</tr>
<tr>
<td><strong>Memory</strong></td>
<td>😞</td>
<td>😞</td>
<td>😊</td>
</tr>
<tr>
<td><strong>Handle Very Large Vocab</strong></td>
<td>😞</td>
<td>😞</td>
<td>😊</td>
</tr>
</tbody>
</table>

- 😊: Similar
- 😞: Worse
- 😎: Better
- 😃: Much Better (>2X)
SMT vs NMT
SMT vs NMT

Pros of NMT

- simpler end-to-end pipeline
- output conditioned on full source text and target history
- continuous word representations better exploit similarities
- smaller model
- more fluent outputs

Cons of NMT

- poor interpretability
- hard to integrate knowledge
- data hungry, underperform in low-resource settings
PBMT vs NMT

### PBMT vs NMT


<table>
<thead>
<tr>
<th>System</th>
<th>Law</th>
<th>Medical</th>
<th>IT</th>
<th>Koran</th>
<th>Subtitles</th>
</tr>
</thead>
<tbody>
<tr>
<td>All Data</td>
<td>30.5</td>
<td>32.8</td>
<td>45.1</td>
<td>42.2</td>
<td>35.3</td>
</tr>
<tr>
<td>Law</td>
<td>31.1</td>
<td>34.4</td>
<td>12.1</td>
<td>18.2</td>
<td>3.5</td>
</tr>
<tr>
<td>Medical</td>
<td>3.9</td>
<td>10.2</td>
<td>39.4</td>
<td>43.5</td>
<td>2.0</td>
</tr>
<tr>
<td>IT</td>
<td>1.9</td>
<td>3.7</td>
<td>6.5</td>
<td>5.3</td>
<td>42.1</td>
</tr>
<tr>
<td>Koran</td>
<td>0.4</td>
<td>1.8</td>
<td>0.0</td>
<td>2.1</td>
<td>0.0</td>
</tr>
<tr>
<td>Subtitles</td>
<td>7.0</td>
<td>9.9</td>
<td>9.3</td>
<td>17.8</td>
<td>9.2</td>
</tr>
</tbody>
</table>

Figure 1: Quality of systems (BLEU), when trained on one domain (rows) and tested on another domain (columns). Comparably, NMT systems (left bars) show more degraded performance out of domain.
Case Studies
Google's Multilingual Neural Machine Translation System: Enabling Zero-Shot Translation

Melvin Johnson, Mike Schuster, Quoc V. Le, Maxim Krikun, Yonghui Wu, Zhifeng Chen, Nikhil Thorat, Fernanda Viégas, Martin Wattenberg, Greg Corrado, Macduff Hughes, Jeffrey Dean
Multilingual NMT
Artificial token in the beginning of the input sentence to indicate the target language:

<2es> Hello, how are you? -> ¿Hola como estás?
## Table 5: Portuguese→Spanish BLEU scores using various models.

<table>
<thead>
<tr>
<th>Model</th>
<th>Zero-shot</th>
<th>BLEU</th>
</tr>
</thead>
<tbody>
<tr>
<td>(a) PBMT bridged</td>
<td>no</td>
<td>28.99</td>
</tr>
<tr>
<td>(b) NMT bridged</td>
<td>no</td>
<td>30.91</td>
</tr>
<tr>
<td>(c) NMT Pt→Es</td>
<td>no</td>
<td>31.50</td>
</tr>
<tr>
<td>(d) Model 1 (Pt→En, En→Es)</td>
<td>yes</td>
<td>21.62</td>
</tr>
<tr>
<td>(e) Model 2 (En↔{Es, Pt})</td>
<td>yes</td>
<td>24.75</td>
</tr>
<tr>
<td>(f) Model 2 + incremental training</td>
<td>no</td>
<td>31.77</td>
</tr>
</tbody>
</table>
Transformers

Attention Is All You Need


- SOTA results on WMT datasets
- Fast: only matrix multiplications
- stack of N self-attention layers
- self-attention in decoder is masked
- decoder also attends to encoder states
- RNN can learn to count raw text
- Transformer needs positional encoding

Figure 1: The Transformer - model architecture.
Multi-Head Attention
Unsupervised Machine Translation Using Monolingual Corpora Only

Guillaume Lample, Alexis Conneau, Ludovic Denoyer, Marc'Aurelio Ranzato
http://www.statmt.org/wmt18/

WMT competitions

THIRD CONFERENCE ON MACHINE TRANSLATION (WMT18)

October 31 — November 1, 2018
Brussels, Belgium

Home

[HOME] [SCHEDULE] [PAPERS]

TRANSLATION TASKS: [NEWS] [BIOMEDICAL] [MULTIMODAL]
EVALUATION TASKS: [METRICS] [QUALITY ESTIMATION]
OTHER TASKS: [AUTOMATIC POST-EDITING] [PARALLEL CORPUS FILTERING]

This conference builds on a series of annual workshops and conferences on statistical machine translation, going back to 2006:

- the NAACL-2006 Workshop on Statistical Machine Translation,
- the ACL-2007 Workshop on Statistical Machine Translation,
- the ACL-2008 Workshop on Statistical Machine Translation,
- the EACL-2009 Workshop on Statistical Machine Translation,
- the ACL-2010 Workshop on Statistical Machine Translation,
- the EMNLP-2011 Workshop on Statistical Machine Translation,
- the NAACL-2012 Workshop on Statistical Machine Translation,
- the ACL-2013 Workshop on Statistical Machine Translation,
- the ACL-2014 Workshop on Statistical Machine Translation,
- the EMNLP-2015 Workshop on Statistical Machine Translation,
- the First Conference on Machine Translation (at ACL-2016),
WMT competitions

http://matrix.statmt.org/