L12: Machine Translation

Spring 2022
• One of the “holy grail” problems in artificial intelligence
• Practical use case: Facilitate communication between people in the world
• Extremely challenging (especially for low-resource languages)
How many languages do you speak?
A) 1
B) 2
C) 3
D) 4+
Some translations

• Easy:
  • I like apples ↔ ich mag Äpfel (German)

• Not so easy:
  • I like apples ↔ J'aime les pommes (French)
  • I like red apples ↔ J'aime les pommes rouges (French)
  • les ↔ the but les pommes ↔ apples
Basics of machine translation

• **Goal:** Translate a sentence $w^{(s)}$ in a **source language (input)** to a sentence in the **target language (output)**

• Can be formulated as an optimization problem:
  - Most likely translation, $\hat{w}^{(t)} = \arg\max_{w^{(t)}} \psi(w^{(s)}, w^{(t)})$
  - where $\psi$ is a scoring function over source and target sentences

• Requires **two** components:
  - **Learning algorithm** to compute parameters of scoring fn. $\psi$
  - **Decoding algorithm** for computing the best translation $\hat{w}^{(t)}$
Why is MT challenging?

• Single words may be replaced with multi-word phrases
  
  • I like apples $\leftrightarrow$ J'aime les pommes

• Reordering of phrases
  
  • I like red apples $\leftrightarrow$ J'aime les pommes rouges

• Contextual dependence
  
  • les $\leftrightarrow$ the but les pommes $\leftrightarrow$ apples

Extremely large output space $\implies$ Decoding is NP-hard
Vauquois Pyramid

- Hierarchy of concepts and distances between them in different languages
- Lowest level: individual words/characters
- Higher levels: syntax, semantics
- Interlingua: Generic language-agnostic representation of meaning
Evaluating machine translation

- Two main criteria:
  - **Adequacy**: Translation $w^{(t)}$ should adequately reflect the linguistic content of $w^{(s)}$
  - **Fluency**: Translation $w^{(t)}$ should be fluent text in the target language

Different translations of “A Vinay le gusta Python”

<table>
<thead>
<tr>
<th>First</th>
<th>Second</th>
<th>Third</th>
</tr>
</thead>
<tbody>
<tr>
<td>To Vinay it like Python</td>
<td>Vinay debugs memory leaks</td>
<td>Vinay likes Python</td>
</tr>
</tbody>
</table>

Which of these translations is both adequate and fluent?

A) first  
B) second  
C) third  
D) none of them
Evaluating machine translation

- Two main criteria:
  - **Adequacy**: Translation $w^{(t)}$ should adequately reflect the linguistic content of $w^{(s)}$
  - **Fluency**: Translation $w^{(t)}$ should be fluent text in the target language

Different translations of "A Vinay le gusta Python"

<table>
<thead>
<tr>
<th>Translation</th>
<th>Adequate?</th>
<th>Fluent?</th>
</tr>
</thead>
<tbody>
<tr>
<td>To Vinay it like Python</td>
<td>yes</td>
<td>no</td>
</tr>
<tr>
<td>Vinay debugs memory leaks</td>
<td>no</td>
<td>yes</td>
</tr>
<tr>
<td>Vinay likes Python</td>
<td>yes</td>
<td>yes</td>
</tr>
</tbody>
</table>

Which of these translations is both adequate and fluent?

A) first
B) second
C) third
D) none of them
Evaluation metrics

• Manual evaluation: ask a native speaker to verify the translation
  • Most accurate, but expensive

• Automated evaluation metrics:
  • Compare system hypothesis with reference translations
  • BiLingual Evaluation Understudy (BLEU) (Papineni et al., 2002):
    • Modified n-gram precision

\[ p_n = \frac{\text{number of } n\text{-grams appearing in both reference and hypothesis translations}}{\text{number of } n\text{-grams appearing in the hypothesis translation}} \]
BLEU

\[
\text{BLEU} = \exp \left( \frac{1}{N} \sum_{n=1}^{N} \log p_n \right)
\]

- To avoid \( \log 0 \), all precisions are smoothed
- Each n-gram in reference can be used at most once
- Ex. **Hypothesis**: to to to to vs **Reference**: to be or not to be should not get a unigram precision of 1
- BLEU-k: average of BLEU scores computed using 1-gram through k-gram.

**Problem**: Precision-based metrics favor short translations

- Solution: Multiply score with a brevity penalty for translations shorter than reference, \( e^{1-r/h} \)
BLEU

- Correlates with human judgements

(G. Doddington, NIST)
BLEU scores

Sample BLEU scores for various system outputs

<table>
<thead>
<tr>
<th>Translation</th>
<th>p₁</th>
<th>p₂</th>
<th>p₃</th>
<th>p₄</th>
<th>BP</th>
</tr>
</thead>
<tbody>
<tr>
<td>Reference</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Vinay likes programming in Python</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>1.0</td>
</tr>
<tr>
<td>Sys1: To Vinay it like to program Python</td>
<td>²/₇</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>1.0</td>
</tr>
<tr>
<td>Sys2: Vinay likes Python</td>
<td>³/₃</td>
<td>½</td>
<td>0</td>
<td>0</td>
<td>0.51</td>
</tr>
<tr>
<td>Sys3: Vinay likes programming in his pajamas</td>
<td>⁴/₆</td>
<td>³/₅</td>
<td>²/₄</td>
<td>¹/₃</td>
<td>1.0</td>
</tr>
</tbody>
</table>

• Alternatives have been proposed:
  • METEOR: weighted F-measure
  • Translation Error Rate (TER): Edit distance between hypothesis and reference

Which of these translations do you think will have the highest BLEU-4 score?
A) sys1
B) sys2
C) sys3

BP: brevity penalty
Data

- Statistical MT relies requires **parallel corpora (bilingual)**

<table>
<thead>
<tr>
<th>1. Chapter 4, Koch (DE)</th>
<th>de</th>
<th>es</th>
</tr>
</thead>
<tbody>
<tr>
<td>context</td>
<td>Wir möchten sicherstellen, daß hierauf bereits in den Erwägungsgründen hingewiesen wird und die uneindeutig formulierte Frist, innerhalb der der Rat eine Entscheidung treffen muß, auf maximal drei Monate fixiert wird.</td>
<td>Quiséräms asegurar que se aluda ya a esto en los considerandos y que el plazo, imprecisamente formulado, dentro del cual el Consejo ha de adoptar una decisión, se fije en tres meses como máximo.</td>
</tr>
<tr>
<td>as early as the recitals and that the period within which the Council has to make a decision - which is not clearly worded - is set at a maximum of three months.</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>2. Chapter 3, FÅrm (SV)</th>
<th>de</th>
<th>es</th>
</tr>
</thead>
<tbody>
<tr>
<td>context</td>
<td>Wir möchten sicherstellen, daß hierauf bereits in den Erwägungsgründen hingewiesen wird und die uneindeutig formulierte Frist, innerhalb der der Rat eine Entscheidung treffen muß, auf maximal drei Monate fixiert wird.</td>
<td>Quiséräms asegurar que se aluda ya a esto en los considerandos y que el plazo, imprecisamente formulado, dentro del cual el Consejo ha de adoptar una decisión, se fije en tres meses como máximo.</td>
</tr>
<tr>
<td>as effective as detailed bureaucratic supervision.</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

(Europarl, Koehn, 2005)

- And lots of it!

- Not easily available for many low-resource languages in the world
Statistical MT

\[ \hat{w}^{(t)} = \arg \max_{w^{(t)}} \psi (w^{(s)}, w^{(t)}) \]

- We can break down the scoring function \( \psi \) as:
  \[ \psi (w^{(s)}, w^{(t)}) = \psi_A (w^{(s)}, w^{(t)}) + \psi_F (w^{(t)}) \]
  \( (\text{adequacy}) \quad (\text{fluency}) \)

- Allows us to estimate parameters of \( \psi \) on separate data
  - \( \psi_A \) from aligned bilingual corpora
  - \( \psi_F \) from monolingual corpora
Noisy channel model

- Generative process for source sentence
- Use Bayes rule to recover $w^{(t)}$ that is maximally likely under the conditional distribution $p_{T|S}$ (which is what we want)

$$
\Psi_A(w^{(s)}, w^{(t)}) \triangleq \log p_{S|T}(w^{(s)} | w^{(t)}) \quad \text{(adequacy)}
$$

$$
\Psi_F(w^{(t)}) \triangleq \log p_T(w^{(t)}) \quad \text{(fluency)}
$$

$$
\Psi(w^{(s)}, w^{(t)}) = \log p_{S|T}(w^{(s)} | w^{(t)}) + \log p_T(w^{(t)}) = \log p_{S,T}(w^{(s)}, w^{(t)}). \quad \text{(overall)}
$$

$$
\arg \max_T p_{T|S} = \arg \max_T \frac{p_T}{p_S} p_{S|T}
$$
Noisy channel model

- Use Bayes rule to recover $w^{(t)}$ that is maximally likely under the conditional distribution $p_{T|S}$ (which is what we want)

\[
\Psi_A(w^{(s)}, w^{(t)}) \triangleq \log p_{S|T}(w^{(s)} | w^{(t)})
\]
\[
\Psi_F(w^{(t)}) \triangleq \log p_T(w^{(t)})
\]
\[
\Psi(w^{(s)}, w^{(t)}) = \log p_{S|T}(w^{(s)} | w^{(t)}) + \log p_T(w^{(t)}) = \log p_{S,T}(w^{(s)}, w^{(t)}).
\]

Allows us to use a standalone language model $p_T$ to improve fluency
IBM Models

• Early approaches to statistical MT

• Key questions:
  • How do we define the translation model $p_{S|T}$?
  • How can we estimate the parameters of the translation model from parallel training examples?
  • Make use of the idea of alignments
Alignments

How should we align words in source to words in target?

\[
\begin{array}{cccc}
\text{A} & \text{Vinay} & \text{le} & \text{gusta} & \text{python} \\
\text{Vinay} & & & & \\
\text{likes} & & & & \\
\text{python} & & & & \\
\end{array}
\]

- **good** \[\mathcal{A}(w^{(s)}, w^{(t)}) = \{(A, \emptyset), (\text{Vinay}, \text{Vinay}), (\text{le}, \text{likes}), (\text{gusta}, \text{likes}), (\text{Python}, \text{Python})\} .\]

- **bad** \[\mathcal{A}(w^{(s)}, w^{(t)}) = \{(A, \text{Vinay}), (\text{Vinay}, \text{likes}), (\text{le}, \text{Python}), (\text{gusta}, \emptyset), (\text{Python}, \emptyset)\} .\]
Incorporating alignments

Let us define the joint probability of alignment and translation as:

$$p(w^{(s)}, A \mid w^{(t)}) = \prod_{m=1}^{M^{(s)}} p(w^{(s)}_{m}, a_{m} \mid w^{(t)}_{a_{m}}, m, M^{(s)}, M^{(t)})$$

$$= \prod_{m=1}^{M^{(s)}} p(a_{m} \mid m, M^{(s)}, M^{(t)}) \times p(w^{(s)}_{m} \mid w^{(t)}_{a_{m}}).$$

- $M^{(s)}, M^{(t)}$ are the number of words in source and target sentences
- $a_{m}$ is the alignment of the $m^{th}$ word in the source sentence
- i.e. it specifies that the $m^{th}$ word in source is aligned to the $a_{m}^{th}$ word in target
- Translation probability for word in source to be a translation of its alignment word
Independence assumptions

\[ p(w^{(s)}, A | w^{(t)}) = \prod_{m=1}^{M^{(s)}} p(w_m^{(s)}, a_m | w_{a_m}^{(t)}, m, M^{(s)}, M^{(t)}) \]

\[ = \prod_{m=1}^{M^{(s)}} p(a_m | m, M^{(s)}, M^{(t)}) \times p(w_m^{(s)} | w_{a_m}^{(t)}). \]

- Two independence assumptions:
  - Alignment probability factors across tokens:
    \[ p(A | w^{(s)}, w^{(t)}) = \prod_{m=1}^{M^{(s)}} p(a_m | m, M^{(s)}, M^{(t)}). \]
  - Translation probability factors across tokens:
    \[ p(w^{(s)} | w^{(t)}, A) = \prod_{m=1}^{M^{(s)}} p(w_m^{(s)} | w_{a_m}^{(t)}). \]
Limitations

Multiple source words may align to the same target word!

Or a source word may not have any corresponding target.
Reordering and word insertion

\( \mathbf{a} = (3, 4, 2, 1)^\top \)

\( \mathbf{a} = (1, 2, 3, 0, 4)^\top \)

Assume extra NULL token

(Slide credit: Brendan O’Connor)
IBM Model 1

- Assume \( p(a_m \mid m, M^{(s)}, M^{(t)}) = \frac{1}{M^{(t)}} \)

- Is this a good assumption?

Every alignment is equally likely!
IBM Model 1

- Assume \( p(a_m | m, M^{(s)}, M^{(t)}) = \frac{1}{M^{(t)}} \)

- We then have (for each pair of words in source and target):

\[
p(w^{(s)}, w^{(t)}) = p(w^{(t)}) \sum_A \left( \frac{1}{M^{(t)}} \right)^{M^{(s)}} p(w^{(s)} | w^{(t)})
\]

- How do we estimate \( p(w^{(s)} = v | w^{(t)} = u) \)?
If we have word-to-word alignments, we can compute the probabilities using the MLE:

\[ p(v | u) = \frac{\text{count}(u, v)}{\text{count}(u)} \]

where \( \text{count}(u, v) = \# \text{instances where target word } u \text{ was aligned to source word } v \) in the training set.

However, word-to-word alignments are often hard to come by.

Solution: Unsupervised learning
Expectation Maximization (advanced)

• **(E-Step)** If we had an accurate translation model, we can estimate likelihood of each alignment as:

\[
q_m(a_m \mid w^{(s)}, w^{(t)}) \propto p(a_m \mid m, M^{(s)}, M^{(t)}) \times p(w_m^{(s)} \mid w_{a_m}^{(t)}),
\]

Remember these are fixed

• **(M Step)** Use expected count to re-estimate translation parameters:

\[
p(v \mid u) = \frac{E_q[\text{count}(u, v)]}{\text{count}(u)}
\]

\[
E_q[\text{count}(u, v)] = \sum_m q_m(a_m \mid w^{(s)}, w^{(t)}) \times \delta(w_m^{(s)} = v) \times \delta(w_{a_m}^{(t)} = u).
\]
How do we translate?

- We want: \(\arg \max_{w(t)} p(w(t) | w(s)) = \arg \max_{w(t)} \frac{p(w(s), w(t))}{p(w(s))}\)

- Sum over all possible alignments:

\[
p(w^{(s)}, w^{(t)}) = \sum_{A} p(w^{(s)}, w^{(t)}, A) = p(w^{(t)}) \sum_{A} p(A) \times p(w^{(s)} | w^{(t)}, A)
\]

- Alternatively, take the max over alignments

- Decoding: Greedy/beam search
Model 1: Decoding

At every step $m$, pick target word $w^{(t)}_m$ to maximize product of:

1. Language model: $p_{LM}(w^{(t)}_m | w^{(t)}_{<m})$
2. Translation model: $p(w^{(s)}_{b_m} | w^{(t)}_m)$

where $b_m$ is the inverse alignment from target to source
**IBM Model 1**

- Assume \( p(a_m | m, M^{(s)}, M^{(t)}) = \frac{1}{M^{(t)}} \)

- Each source word is aligned to at most one target word

- We then have:

\[
p(w^{(s)}, w^{(t)}) = p(w^{(t)}) \sum_A \left( \frac{1}{M^{(t)}} \right)^{M^{(s)}} p(w^{(s)} | w^{(t)})
\]
IBM Model 2

• Slightly relaxed assumption:

• \( p(a_m | m, M^{(s)}, M^{(t)}) \) is also estimated/learned, not set to constant

• Some independence assumptions from Model 1 still required:

• Alignment probability factors across tokens:

\[
p(A | w^{(s)}, w^{(t)}) = \prod_{m=1}^{M^{(s)}} p(a_m | m, M^{(s)}, M^{(t)}).
\]

• Translation probability factors across tokens:

\[
p(w^{(s)} | w^{(t)}, A) = \prod_{m=1}^{M^{(s)}} p(w^{(s)}_m | w^{(t)}_{a_m}).
\]
Other IBM models

Model 1: lexical translation
Model 2: additional absolute alignment model
Model 3: extra fertility model
Model 4: added relative alignment model
Model 5: fixed deficiency problem.
Model 6: Model 4 combined with a HMM alignment model in a log linear way

- Models 3 - 6 make successively weaker assumptions
- But get progressively harder to optimize
- Simpler models are often used to ‘initialize’ complex ones
- e.g train Model 1 and use it to initialize Model 2 translation parameters
Phrase-based MT

• Word-by-word translation is not sufficient in many cases

Nous allons prendre un verre

(literal)  We will take a glass

(actual)  We’ll have a drink

• Solution: build alignments and translation tables between multiword spans or “phrases”
Phrase-based MT

- Solution: build alignments and translation tables between multiword spans or “phrases”
- Translations condition on multi-word units and assign probabilities to multi-word units
- Alignments map from spans to spans

\[
p(w^{(s)} | w^{(t)}, A) = \prod_{((i,j), (k,l)) \in A} P_{w^{(s)} | w^{(t)}} (\{w_{i+1}^{(s)}, w_{i+2}^{(s)}, \ldots, w_j^{(s)}\} | \{w_{k+1}^{(t)}, w_{k+2}^{(t)}, \ldots, w_l^{(t)}\})
\]
Vauquois Pyramid

- Hierarchy of concepts and distances between them in different languages
- Lowest level: individual words/characters
- Higher levels: syntax, semantics
- Interlingua: Generic language-agnostic representation of meaning
Rather than use phrases, use a **synchronous context-free grammar**: constructs “parallel” trees in two languages simultaneously.

\[
NP \rightarrow [DT_1 \ JJ_2 \ NN_3; \ DT_1 \ NN_3 \ JJ_2]
\]

- \( DT \rightarrow \) [the, la]
- \( DT \rightarrow \) [the, le]
- \( NN \rightarrow \) [car, voiture]
- \( JJ \rightarrow \) [yellow, jaune]

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

(Slide credit: Greg Durrett)
Syntactic MT

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

Grammar:

\[
\begin{align*}
S & \rightarrow \langle \text{VP} . ; \text{I VP} . \rangle \quad \text{OR} \quad S \rightarrow \langle \text{VP} . ; \text{you VP} . \rangle \\
\text{VP} & \rightarrow \langle \text{lo haré ADV ; will do it ADV} \rangle \\
S & \rightarrow \langle \text{lo haré ADV . ; I will do it ADV .} \rangle \\
\text{ADV} & \rightarrow \langle \text{de muy buen grado ; gladly} \rangle
\end{align*}
\]

Slide credit: Dan Klein

Next time: Neural machine translation