Projects and Literature Reviews

First report due Nov 15
(PDF written in LaTeX; no length restrictions; submission through Compass)

Purpose of this first report:
Check-in to make sure that you’re on track
(or, if not, that we can spot problems)
Get feedback from your peers

Rubrics for the final reports (due on Reading Day):
https://courses.engr.illinois.edu/CS447/LiteratureReviewRubric.pdf
https://courses.engr.illinois.edu/CS447/FinalProjectRubric.pdf

Projects and Literature Reviews

Guidelines for first Project Report:
What is your project about?
What are the relevant papers you are building on?
What data are you using?
What evaluation metric will you be using?
What models will you implement/evaluate?
What is your to-do list?

Guidelines for first Literature Review Report:
What is your literature review about?
(What task or what kind of models?)
Do you have any specific questions or focus?)
What are the papers you will review?
(If you already have it, give a brief summary of each of them)
What’s your to-do list?

Machine translation approaches
Direct translation
Maria non dió una bofetada a la bruja verde.

1. **Morphological analysis of source string**
   Maria nonNeg dar3sgF-Past una bofetada a la bruja verde
   (usually, a complete morphological analysis)

2. **Lexical transfer (using a translation dictionary):**
   Mary not slap3sgF-Past to the witch green.

3. **Local reordering:**
   Mary not slap3sgF-Past the green witch.

4. **Morphology:**
   Mary did not slap the green witch.

Syntactic transfer
Requires a syntactic parse of the source language, followed by reordering of the tree

**Local reordering:**

**Nonlocal reordering:**

Adverb placement in German:
The green witch is at home this week.

Diese Woche ist die grüne Hexe zuhause.

Japanese SOV order:
He adores listening to music
Kare ha ongaku wo kiku no ga daisuki desu

PPs in Chinese:
Jackie Cheng went to Hong Kong
Cheng Long dao Xianggang qu

Limits of direct translation:
Phrasal reordering

Noun
Adj
N
Noun

Adj
green witch
bruja verde

The green witch is
at home this week
diese Woche ist die grüne Hexe zuhause
Semantic transfer

Done at the level of **predicate-argument structure**
(some people call this syntactic transfer too...):

\[
\begin{align*}
\text{Hans kocht gerne} & \quad \rightarrow \quad \text{Hans likes cooking}
\end{align*}
\]

or at the level of **semantic representations** (e.g. DRSs):

\[
\begin{align*}
\text{Dorna et al. 1998}
\end{align*}
\]

Interlingua approaches

Based on the assumption that there is one **common meaning representation** (e.g. predicate logic) that abstracts away from any difference in surface realization.

Semantic transfer: each language produces its own meaning representation

Was thought useful for multilingual translation

\[
\begin{align*}
\text{Leavitt et al. 1994}
\end{align*}
\]

Statistical Machine Translation with the noisy channel model

**The noisy channel model**

Assume we want to **translate from Chinese to English**.

Chinese is the **source**, English is the **target**

The “noisy channel” metaphor assumes that the Chinese source text is English text that passed through a “noisy channel” which scrambled the English text such that it came out in Chinese.

... the **noisy channel** is a probability model \( P(\text{Chinese} \mid \text{English}) \)

... the target text is the input to the noisy channel

... the source text is the output of the noisy channel

... translation is the task of recovering the most likely original input (i.e. the target text) to the noisy channel, given the output (source text)

... this also requires a model \( P(\text{English}) \) over the possible input messages

\[
\begin{align*}
\text{argmax}_\text{Eng} P(\text{Eng} \mid \text{Chin}) & = \text{argmax}_\text{Eng} P(\text{Chin} \mid \text{Eng}) \times P(\text{Eng}) \\
\text{Translation Model} & \times \text{Language Model}
\end{align*}
\]
The noisy channel model

Translating from Chinese to English:
\[
\text{argmax}_{\text{Eng}} P(\text{Eng}|\text{Chin}) = \text{argmax}_{\text{Eng}} \left( \frac{P(\text{Chin}|\text{Eng}) \times P(\text{Eng})}{P(\text{Eng})} \right) \]

This is really just an application of Bayes' rule:
\[
\hat{E} = \arg\max_E P(E|F) = \arg\max_E \frac{P(F|E) \times P(E)}{P(F)} = \arg\max_E \left( \frac{P(F|E)}{P(F)} \right) \times P(E)
\]

The translation model \(P(F|E)\) is intended to capture the faithfulness of the translation.
It needs to be trained on a parallel corpus.

The language model \(P(E)\) is intended to capture the fluency of the translation.
It can be trained on a (very large) monolingual corpus.

Statistical MT with the noisy channel model

Parallel corpora

Monolingual corpora

Input

Decoding algorithm

Translation

President: Good morning, Honourable Members.
$n$-gram language models for MT

With training on data from the web and clever parallel processing (MapReduce/Bloom filters), $n$ can be quite large
- Google (2007) uses 5-grams to 7-grams,
- This results in huge models, but the effect on translation quality levels off quickly:

<table>
<thead>
<tr>
<th>Size of models</th>
<th>Effect on translation quality</th>
</tr>
</thead>
<tbody>
<tr>
<td>$n$-gram language models for MT</td>
<td>BLEU scores for varying amounts of data using Kneser-Ney (KN) and Simple Backoff (SB).</td>
</tr>
</tbody>
</table>

![Figure 3: Number of $n$-grams (sum of unigrams to 5-grams) for varying amounts of training data.](image1)

![Figure 5: BLEU scores for varying amounts of data using Kneser-Ney (KN) and Simple Backoff (SB).](image2)

This requires phrase alignment on a parallel corpus.

Translation probability $P(fp_i | ep_i)$

Phrase translation probabilities can be obtained from a phrase table:

<table>
<thead>
<tr>
<th>EP</th>
<th>FP</th>
<th>count</th>
</tr>
</thead>
<tbody>
<tr>
<td>green witch</td>
<td>grüne Hexe</td>
<td>...</td>
</tr>
<tr>
<td>at home</td>
<td>zuhause</td>
<td>10534</td>
</tr>
<tr>
<td>at home</td>
<td>daheim</td>
<td>9890</td>
</tr>
<tr>
<td>is</td>
<td>ist</td>
<td>598012</td>
</tr>
<tr>
<td>this week</td>
<td>diese Woche</td>
<td>...</td>
</tr>
</tbody>
</table>

Creating parallel corpora

A parallel corpus consists of the same text in two (or more) languages.
- Examples: Parliamentary debates: Canadian Hansards; Hong Kong Hansards, Europarl; Movie subtitles (OpenSubtitles)

In order to train translation models, we need to align the sentences (Church & Gale '93)

IBM models

First statistical MT models, based on noisy channel:
- Translate from source $f$ to target $e$ via a translation model $P(f | e)$ and a language model $P(e)$
- The translation model goes from target $e$ to source $f$ via word alignments $a$:

$$P(f | e) = \sum_a P(f, a | e)$$

Original purpose: Word-based translation models
Today: Can be used to obtain word alignments, which are then used to obtain phrase alignments for phrase-based translation models

Sequence of 5 translation models
- Model 1 is too simple to be used by itself, but can be trained very easily on parallel data.
IBM translation models: assumptions

The model “generates” the ‘foreign’ source sentence \( f \) conditioned on the ‘English’ target sentence \( e \) by the following stochastic process:

1. Generate the **length** of the source \( f \) with probability \( p = \ldots \)
2. Generate the **alignment** of the source \( f \) to the target \( e \) with probability \( p = \ldots \)
3. Generate the **words** of the source \( f \) with probability \( p = \ldots \)

Word alignment

<table>
<thead>
<tr>
<th>Jean aime Marie</th>
<th>... dass John Maria liebt.</th>
</tr>
</thead>
<tbody>
<tr>
<td>John</td>
<td>dass</td>
</tr>
<tr>
<td>loves</td>
<td>John</td>
</tr>
<tr>
<td>Mary</td>
<td>Maria</td>
</tr>
<tr>
<td></td>
<td>liebt</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Mary</th>
<th>no</th>
<th>dió</th>
<th>una</th>
<th>bofetada</th>
<th>a</th>
<th>la</th>
<th>bruja</th>
<th>verde</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Mary</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Word alignment

<table>
<thead>
<tr>
<th>Mary</th>
<th>no</th>
<th>dió</th>
<th>una</th>
<th>bofetada</th>
<th>a</th>
<th>la</th>
<th>bruja</th>
<th>verde</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Mary</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Word alignment

<table>
<thead>
<tr>
<th>Maria</th>
<th>no</th>
<th>dió</th>
<th>una</th>
<th>bofetada</th>
<th>a</th>
<th>la</th>
<th>bruja</th>
<th>verde</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Mary</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Word alignment

<table>
<thead>
<tr>
<th>Maria</th>
<th>no</th>
<th>dió</th>
<th>una</th>
<th>bofetada</th>
<th>a</th>
<th>la</th>
<th>bruja</th>
<th>verde</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Mary</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Word alignment

<table>
<thead>
<tr>
<th>Maria</th>
<th>no</th>
<th>dió</th>
<th>una</th>
<th>bofetada</th>
<th>a</th>
<th>la</th>
<th>bruja</th>
<th>verde</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Mary</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Word alignment

<table>
<thead>
<tr>
<th>Maria</th>
<th>no</th>
<th>dió</th>
<th>una</th>
<th>bofetada</th>
<th>a</th>
<th>la</th>
<th>bruja</th>
<th>verde</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Mary</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Word alignment

<table>
<thead>
<tr>
<th>Maria</th>
<th>no</th>
<th>dió</th>
<th>una</th>
<th>bofetada</th>
<th>a</th>
<th>la</th>
<th>bruja</th>
<th>verde</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Mary</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Word alignment

<table>
<thead>
<tr>
<th>Maria</th>
<th>no</th>
<th>dió</th>
<th>una</th>
<th>bofetada</th>
<th>a</th>
<th>la</th>
<th>bruja</th>
<th>verde</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Mary</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Word alignment

<table>
<thead>
<tr>
<th>Maria</th>
<th>no</th>
<th>dió</th>
<th>una</th>
<th>bofetada</th>
<th>a</th>
<th>la</th>
<th>bruja</th>
<th>verde</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Mary</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
Word alignment

<table>
<thead>
<tr>
<th>Target</th>
<th>Source</th>
</tr>
</thead>
<tbody>
<tr>
<td>Marie</td>
<td>a</td>
</tr>
<tr>
<td>a</td>
<td>traversé</td>
</tr>
<tr>
<td>traversé</td>
<td>le</td>
</tr>
<tr>
<td>le</td>
<td>lac</td>
</tr>
<tr>
<td>lac</td>
<td>à</td>
</tr>
<tr>
<td>à</td>
<td>la</td>
</tr>
<tr>
<td>la</td>
<td>nage</td>
</tr>
</tbody>
</table>

Mary a traversé le lac à la nage

swam
across
the
lake

One target word can be aligned to many source words.

But each source word can only be aligned to one target word.

Alignments are not symmetric!
Some source words may not align to any target words.
**Word alignment**

<table>
<thead>
<tr>
<th>Target</th>
<th>Source</th>
</tr>
</thead>
<tbody>
<tr>
<td>Marie</td>
<td>a</td>
</tr>
<tr>
<td>NULL</td>
<td>traversé</td>
</tr>
<tr>
<td>Mary</td>
<td>le</td>
</tr>
<tr>
<td>swam</td>
<td>lac</td>
</tr>
<tr>
<td>across</td>
<td>à</td>
</tr>
<tr>
<td>the</td>
<td>la</td>
</tr>
<tr>
<td>lake</td>
<td>nage</td>
</tr>
</tbody>
</table>

Variant: assume a NULL word in the target language.

**The IBM models**

Use the noisy channel (Bayes rule) to get the best (most likely) target translation \( e \) for source sentence \( f \):

\[
\arg \max_e P(e|f) = \arg \max_e P(f|e)P(e)
\]

The translation model \( P(f|e) \) requires alignments \( a \)

\[
P(f|e) = \sum_{a \in A(e, f)} P(f, a|e)
\]

Generate \( f \) and the alignment \( a \) with \( P(f, a|e) \):

\[
P(f, a|e) = \prod_{i=1}^{m} P(a_i|a_{i-1}, f_{i-1}, \ldots, f_1, e, m)P(f_i|a_{1..i}, \ldots, f_{i-1}, e, m)
\]

\( m = \#\text{words in } f \)  \( \text{probability of alignment } a_j \)  \( \text{probability of word } f_j \)
Model parameters

Length probability $P(m \mid n)$:
What’s the probability of generating a source sentence of length $m$ given a target sentence of length $n$?
Count in training data

Alignment probability: $P(a \mid m, n)$:
Model 1 assumes all alignments have the same probability:
For each position $a_1...a_m$, pick one of the $n+1$ target positions uniformly at random

Translation probability: $P(f_i = lac \mid a_i = i, e_i = lake)$:
In Model 1, these are the only parameters we have to learn.

---

IBM model 1: Generative process

For each target sentence $e = e_1...e_n$ of length $n$:

1. **Choose a length** $m$ for the source sentence (e.g. $m = 8$)
   
<table>
<thead>
<tr>
<th>Position</th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
<th>6</th>
<th>7</th>
<th>8</th>
</tr>
</thead>
<tbody>
<tr>
<td>Alignment</td>
<td>1</td>
<td>2</td>
<td>3</td>
<td>4</td>
<td>5</td>
<td>6</td>
<td>7</td>
<td>8</td>
</tr>
<tr>
<td>Translation</td>
<td>Marie</td>
<td>a</td>
<td>traversé</td>
<td>le</td>
<td>lac</td>
<td>à</td>
<td>la</td>
<td>nage</td>
</tr>
</tbody>
</table>

2. **Choose an alignment** $a = a_1...a_m$ for the source sentence
   Each $a_i$ corresponds to a word $e_i$ in $e$: $0 \leq a_i \leq n$

3. **Translate** each target word $e_i$ into the source language

---

Finding the best alignment

How do we find the best alignment between $e$ and $f$?

\[
\hat{a} = \arg \max_a P(f, a \mid e)
= \arg \max_a \frac{\varepsilon}{(n+1)^m} \prod_{j=1}^{m} P(f_j \mid e_{a_j})
= \arg \max_a \frac{\varepsilon}{(n+1)^m} \prod_{j=1}^{m} P(f_j \mid e_{a_j})
\]

\[
\hat{a}_j = \arg \max_{a_j} P(f_j \mid e_{a_j})
\]
Learning translation probabilities

The only parameters that need to be learned are the translation probabilities $P(f \mid e)$

$$P(f = lac \mid e = lake)$$

If the training corpus had word alignments, we could simply count how often ‘lake’ is aligned to ‘lac’:

$$P(lac \mid lake) = \frac{\text{count}(lac, lake)}{\sum_w \text{count}(w, lake)}$$

But we don’t have word alignments. So, instead of relative frequencies, we have to use expected relative frequencies:

$$P(lac \mid lake) = \frac{\langle \text{count}(lac, lake) \rangle}{\langle \sum_w \text{count}(w, lake) \rangle}$$

Expectation-Maximization (EM)

1. Initialize a first model, $M_0$

2. Expectation (E) step:
   Go through training data to gather expected counts
   $\langle \text{count}(lac, lake) \rangle$

3. Maximization (M) step:
   Use expected counts to compute a new model $M_{i+1}$
   $P_{i+1}(lac \mid lake) = \frac{\langle \text{count}(lac, lake) \rangle}{\sum_w \text{count}(w, lake)}$

4. Check for convergence:
   Compute log-likelihood of training data with $M_{i+1}$
   If the difference between new and old log-likelihood smaller than a threshold, stop. Else go to 2.

Training Model 1 with EM

The only parameters that need to be learned are the translation probabilities $P(f \mid e)$

We use the EM algorithm to estimate these parameters from a corpus with $S$ sentence pairs $s = (f^{(s)}, e^{(s)})$ with alignments $A(f^{(s)}, e^{(s)})$

- Initialization: guess $P(f \mid e)$
- Expectation step: compute expected counts
  $$\langle c(f, e) \rangle = \sum_{s \in S} \langle c(f, e | e^{(s)}, f^{(s)}) \rangle$$

- Maximization step: recompute probabilities $P(f \mid e)$
  $$P(f | e) = \frac{\langle c(f, e) \rangle}{\sum_{f'} \langle c(f', e) \rangle}$$

The E-step

Compute the expected count $\langle c(f, e | f, e) \rangle$:

$$\langle c(f, e | f, e) \rangle = \sum_{a \in A(f, e)} P(a | f, e) \cdot c(f, e | a, e, f)$$

$$P(a | f, e) = \frac{P(a, f | e)}{P(f | e) = \frac{P(a, f | e)}{\sum_{a'} P(a', f | e)}}$$

$$P(a, f | e) = \prod_j P(f_j | e_{a_j})$$

$$\langle c(f, e | f, e) \rangle = \sum_{a \in A(f, e)} \prod_j P(f_j | e_{a_j}) \cdot c(f, e | a, e, f)$$
Other translation models

Model 1 is a very simple (and not very good) translation model.

IBM models 2-5 are more complex. They take into account:
- “fertility”: the number of foreign words generated by each target word
- the word order and string position of the aligned words

Today’s key concepts

Why is machine translation hard?
Linguistic divergences: morphology, syntax, semantics

Different approaches to machine translation:
- Vauquois triangle
- Statistical MT: Noisy Channel, IBM Model 1 (more on this next time)