Word Alignment

David Kauchak
CS159 – Spring 2023

Language translation

Yo quiero
Taco Bell

Word models: IBM Model 1

Mary did not slap the green witch

Maria no dio una botefada a la bruja verde

Each foreign word is aligned to exactly one English word

This is the ONLY thing we model!

\[ p(f_1, f_2, \ldots, f_F, a_1, a_2, \ldots, a_1, e_1, e_2, \ldots, e_E) = \prod_{i=1}^{F} p(f_i | e_{a_i}) \]
Training a word-level model

The old man is happy. He has fished many times.
His wife talks to him.
The sharks await.

\[ p(f_i | e_a) = \prod_{a} p(f_i | e_a) \]

\[ p(f_i | e_a) \]: probability that \( e \) is translated as \( f \)

How do we learn these?
What data would be useful?

Thought experiment

The old man is happy. He has fished many times.
His wife talks to him.
The sharks await.

\[ p(f_i | e_a) = ? \]

\[ p(\text{el | the}) = ? \]
\[ p(\text{Los | the}) = ? \]

Getting data like this is expensive!

Even if we had it, what happens when we switch to a new domain/corpus
Thought experiment #2
The old man is happy. He has fished many times.

\[ p(f|e_i) = \frac{\text{count}(f \text{ aligned-to } e)}{\text{count}(e)} \]

What do we do?

80 annotators
20 annotators

Use partial counts:
- count(viejo \rightarrow man) 0.8
- count(viejo \rightarrow old) 0.2

Thought experiment #2
The old man is happy. He has fished many times.

\[ p(f|e_i) = \frac{\text{count}(f \text{ aligned-to } e)}{\text{count}(e)} \]

What do we do?

Training without alignments

How should these be aligned?

There is some information!
(Think of the alien translation task last time)
Training without alignments

IBM model 1: Each foreign word is aligned to 1 English word (ignore NULL for now)

What are the possible alignments?

If I told you how likely each of these were, does that help us with calculating \( p(f \mid e) \)?

\[
p(f \mid e) = \frac{\text{count}(f \text{ aligned-to } e)}{\text{count}(e)}
\]

Use partial counts and sum:
- \( \text{count}(y \rightarrow a) \) 0.9+0.01
- \( \text{count}(x \rightarrow a) \) 0.01+0.01
One the one hand

<table>
<thead>
<tr>
<th>a b</th>
<th>a b</th>
<th>a b</th>
<th>a b</th>
</tr>
</thead>
<tbody>
<tr>
<td>x y</td>
<td>x y</td>
<td>x y</td>
<td>x y</td>
</tr>
</tbody>
</table>

0.01 0.9 0.08 0.01

If you had the likelihood of each alignment, you could calculate $p(f|e)$

$$p(f_i|e_i) = \frac{\text{count}(f \text{ aligned-to } e)}{\text{count}(e)}$$

One the other hand

We want to calculate the probability of the alignment, e.g.

$$p(\text{alignment1} \mid F, E) = p(A_1 \mid F, E)$$

We can calculate $p(A_i \mid F, E)$ using the word probabilities.

$$p(F, a_{i_1}, a_{i_2} \mid E) = \prod_{i} p(f_i \mid e_i)$$

One the other hand

We want to calculate the probability of the alignment, e.g.

$$p(\text{alignment1} \mid F, E) = p(A_1 \mid F, E)$$

We can calculate $p(A_i \mid F, E)$ using the word probabilities.

$$p(A_i \mid F, E) \neq p(A_1 \mid F, E)$$

How are these two probabilities related?
Our friend the chain rule

\[ p(A, F \mid E) = p(A \mid F, E) \cdot p(F \mid E) \]

\[ p(A \mid F, E) = \frac{p(A, F \mid E)}{p(F \mid E)} \]

What is \( p(F \mid E) \)?

Hint: how do we go from \( p(A, F \mid E) \) to \( p(F \mid E) \)?

---

One the other hand

Alignment 1  Alignment 2  Alignment 3  Alignment 4

\[ a b \quad a b \quad a b \quad a b \]

\[ \begin{array}{cccc}
  p(x_1 a) \cdot p(y_1 b) & p(x_1 b) \cdot p(y_1 a) & p(x_1 b) \cdot p(y_1 b) & p(x_1 a) \cdot p(y_1 a) \\
  p(F, a \mid E) & p(F, a \mid E) & p(F, a \mid E) & p(F, a \mid E)
  \end{array} \]

\[ p(F, a_1 a_2 \ldots a_n \mid E) = \prod_{i=1}^{n} p(f, x_{a_i}) \]

---

Our friend the chain rule

\[ p(A, F \mid E) = p(A \mid F, E) \cdot p(F \mid E) \]

\[ p(A \mid F, E) = \frac{p(A, F \mid E)}{p(F \mid E)} \]

\[ p(A \mid F, E) = \frac{p(A, F \mid E)}{\sum_{a} p(A, F \mid E)} \quad \text{sum over the variable!} \]

How likely is this alignment, compared to all other alignments under the model?

---

One the other hand

Alignment 1  Alignment 2  Alignment 3  Alignment 4

\[ a b \quad a b \quad a b \quad a b \]

\[ \begin{array}{cccc}
  p(x_1 a) \cdot p(y_1 b) & p(x_1 b) \cdot p(y_1 a) & p(x_1 b) \cdot p(y_1 b) & p(x_1 a) \cdot p(y_1 a) \\
  p(F, a \mid E) & p(F, a \mid E) & p(F, a \mid E) & p(F, a \mid E)
  \end{array} \]

Normalize

\[ p(a \mid E, F) = \frac{p(x \mid a) \cdot p(y \mid b)}{\sum_{a} p(F, a \mid E)} \]
Have we gotten anywhere?

Training without alignments

Initially assume a $p(f|e)$ are equally probable

Repeat:

– Enumerate all possible alignments
– Calculate how probable the alignments are under the current model (i.e. $p(f|e)$)
– Recalculate $p(f|e)$ using counts from all alignments, weighted by how probable they are

EM algorithm

*(something from nothing)*

General approach for calculating “hidden variables”, i.e. variables without explicit labels in the data

Repeat:

E-step: Calculate the expected probabilities of the hidden variables based on the current model

M-step: Update the model based on the expected counts/probabilities

EM alignment

E-step

– Enumerate all possible alignments
– Calculate how probable the alignments are under the current model (i.e. $p(f|e)$)

M-step

– Recalculate $p(f|e)$ using counts from all alignments, weighted by how probable they are
What are the different \( p(f|e) \) that make up my model?

Technically, all combinations of foreign and English words

Start with all \( p(f|e) \) equally probable

E-step: What are the probabilities of the alignments?
1. calculate: \( p(f_1, f_2, ..., f_n) \sum_{a_1, a_2, ..., a_n} p(f_1) p(a_1|f_1) p(a_2|a_1, f_2) ... p(a_n|a_{n-1}, f_n) \)

2. normalize: \( p(a|f, E) = \frac{p(f, a|E)}{\sum_{a'} p(f, a'|E)} \)
E-step: What are the probabilities of the alignments?
2. normalize:  \[
p(a | F, E) = \frac{p(F, a | E)}{\sum_{j=1}^{N} p(F, a_j | E)}
\]

M-step: What are the p(f|e) given the alignments?

Then, calculate the probabilities by normalizing the counts

First, calculate the partial counts

Then, calculate the probabilities by normalizing the counts
E-step: 1. what are the $p(A,F|E)$?

$p(a_i|E) = \frac{p(F,a_i|E)}{\sum_{a_j} p(F,a_j|E)}$

E-step: 2. what are the alignments, i.e. normalize?
### M-step: What are the $p(f|e)$ given the alignments?

| $f$   | $e$   | $p(f|e)$ |
|-------|-------|----------|
| casa  | green | 3/7      |
| verde | green | 4/7      |
| la    | green | 0        |
| casa  | house | 3/5      |
| verde | house | 1/5      |
| la    | house | 1/5      |
| casa  | the   | 3/7      |
| verde | the   | 0        |
| la    | the   | 4/7      |

Then, calculate the probabilities by normalizing the counts.
E-step: 1. what are the $p(A,F|E)$?

<table>
<thead>
<tr>
<th>Casa</th>
<th>Verde</th>
<th>La</th>
<th>Casa, Verde</th>
<th>Casa, Green</th>
<th>Casa, House</th>
<th>Casa, The</th>
</tr>
</thead>
<tbody>
<tr>
<td>0.108</td>
<td>0.432</td>
<td>0.151</td>
<td>0.309</td>
<td>0.108</td>
<td>0.69</td>
<td>0.36</td>
</tr>
</tbody>
</table>

E-step: 2. what are the alignments, i.e. normalize?

<table>
<thead>
<tr>
<th>Casa</th>
<th>Verde</th>
<th>La</th>
<th>Casa, Verde</th>
<th>Casa, Green</th>
<th>Casa, House</th>
<th>Casa, The</th>
</tr>
</thead>
<tbody>
<tr>
<td>0.36</td>
<td>0.64</td>
<td>0.15</td>
<td>0.15</td>
<td>0.36</td>
<td>0.69</td>
<td>0.36</td>
</tr>
</tbody>
</table>

\[
c(casa, green) = 0.108 \times 0.309 = 0.03282
\]
\[
c(casa, house) = 0.432 \times 0.69 = 0.29538
\]
\[
c(casa, the) = 0.151 \times 0.36 = 0.05436
\]
Iterate...

<table>
<thead>
<tr>
<th>5 iterations</th>
<th>10 iterations</th>
<th>100 iterations</th>
</tr>
</thead>
<tbody>
<tr>
<td>p( Casa</td>
<td>Green)</td>
<td>0.24</td>
</tr>
<tr>
<td>p( Verde</td>
<td>Green)</td>
<td>0.76</td>
</tr>
<tr>
<td>p( la</td>
<td>Green)</td>
<td>0</td>
</tr>
<tr>
<td>p( Casa</td>
<td>House)</td>
<td>0.84</td>
</tr>
<tr>
<td>p( Verde</td>
<td>House)</td>
<td>0.08</td>
</tr>
<tr>
<td>p( la</td>
<td>House)</td>
<td>0.08</td>
</tr>
<tr>
<td>p( Casa</td>
<td>The)</td>
<td>0.24</td>
</tr>
<tr>
<td>p( Verde</td>
<td>The)</td>
<td>0</td>
</tr>
<tr>
<td>p( la</td>
<td>The)</td>
<td>0.76</td>
</tr>
</tbody>
</table>

EM alignment

E-step
- Enumerate all possible alignments
- Calculate how probable the alignments are under the current model (i.e. p(f|e))

M-step
- Recalculate p(f|e) using counts from all alignments, weighted by how probable they are

Why does it work?

Intuitively:

M-step
- Recalculate p(f|e) using counts from all alignments, weighted by how probable they are

Things that co-occur will have higher probabilities

E-step
- Calculate how probable the alignments are under the current model (i.e. p(f|e))

Alignments that contain things with higher p(f|e) will be scored higher

Why does it work?
An aside: estimating probabilities

What is the probability of “the” occurring in a sentence?

\[
\frac{\text{number of sentences with "the"}}{\text{total number of sentences}}
\]

Is this right?

No. This is an estimate based on our data

This is called the maximum likelihood estimation. Why?

Maximum Likelihood Estimation (MLE)

Maximum likelihood estimation picks the values for the model parameters that maximize the likelihood of the training data

You flip a coin 100 times. 60 times you get heads.

What is the MLE for heads?

\[p(\text{head}) = 0.60\]

Maximum Likelihood Estimation (MLE)

Maximum likelihood estimation picks the values for the model parameters that maximize the likelihood of the training data

You flip a coin 100 times. 60 times you get heads.

What is the likelihood of the data under this model (each coin flip is a data point)?
MLE example
You flip a coin 100 times. 60 times you get heads.
MLE for heads: p(head) = 0.60
What is the likelihood of the data under this model (each coin flip is a data point)?

\[
\text{likelihood(data)} = \prod p(x_i) \\
\text{log}(0.60^{60} \times 0.40^{40}) = -67.3
\]

Can we do any better?
\[
p(\text{heads}) = 0.5 \\
\text{log}(0.50^{60} \times 0.50^{40}) = -69.3
\]
\[
p(\text{heads}) = 0.7 \\
\text{log}(0.70^{60} \times 0.30^{40}) = -69.5
\]

EM alignment: the math
The EM algorithm tries to find parameters of the model (p(f|e)) that maximize the likelihood of the data

In our case:

\[
p(f_1, f_2, ... f_F | e_1, e_2, ... e_E) = \sum_{e_1} \sum_{e_2} ... \sum_{e_E} p(f_i | e_i)
\]

Each iteration, we increase (or keep the same) the likelihood of the data

Implementation details

Any concerns/issues? Anything underspecified?

Repeat:

E-step
  • Enumerate all possible alignments
  • Calculate how probable the alignments are under the current model (i.e. p(f|e))

M-step
  • Recalculate p(f|e) using counts from all alignments, weighted by how probable they are
Implementation details

When do we stop?

Repeat:
E-step
• Enumerate all possible alignments
• Calculate how probable the alignments are under the current model (i.e. $p(f|e)$)
M-step
• Recalculate $p(f|e)$ using counts from all alignments, weighted by how probable they are

Implementation details

For $|E|$ English words and $|F|$ foreign words, how many alignments are there?

Repeat:
E-step
• Enumerate all possible alignments
• Calculate how probable the alignments are under the current model (i.e. $p(f|e)$)
M-step
• Recalculate $p(f|e)$ using counts from all alignments, weighted by how probable they are

Implementation details

Each foreign word can be aligned to any of the English words (or NULL)

$(|E|+1)^{|F|}$

Repeat:
E-step
• Enumerate all possible alignments
• Calculate how probable the alignments are under the current model (i.e. $p(f|e)$)
M-step
• Recalculate $p(f|e)$ using counts from all alignments, weighted by how probable they are
Thought experiment

The old man is happy. He has fished many times.

El viejo está feliz porque ha pescado muchos veces.

His wife talks to him.

Su mujer habla con él.

The sharks await.

Los tiburones esperan.

\[
p(f | e) = \frac{\text{count}(f \text{ aligned-to } e)}{\text{count}(e)}
\]

\[
p(\text{el } \leftrightarrow \text{the}) = 0.5
\]

\[
p(\text{Los } \leftrightarrow \text{the}) = 0.5
\]
### Thought experiment #2

The old man is happy. He has fished many times.

El viejo está feliz porque ha pescado muchos veces.

\[
p(f \mid e_i) = \frac{\text{count}(f \text{ aligned-to } e)}{\text{count}(e)}
\]

Use partial counts:
- \(\text{count(viejo \leftrightarrow man)} = 0.8\)
- \(\text{count(viejo \leftrightarrow old)} = 0.2\)

### Without the alignments

If \(f\) aligned to \(e\):
- \(\text{count}(e, f) += 1\)
- \(\text{count}(e) += 1\)

\[p(f \rightarrow e): \text{probability that } f \text{ is aligned to } e \text{ in this pair}\]

\[
\begin{align*}
\text{a} & \quad \text{b} & \quad \text{c} \\
\text{y} & \quad \text{z}
\end{align*}
\]

What is \(p(y \rightarrow a)\)?

Put another way, of all things that \(y\) could align to in this sentence, how likely is it to be \(a\)?

\[
\text{p(y | a)}
\]

Of all things that \(y\) could align to, how likely is it to be \(a\):
Without alignments

\[ p(f \rightarrow e) : \text{probability that } f \text{ is aligned to } e \text{ in this pair} \]

\[ \frac{a \quad b \quad c}{y \quad z} \]

Of all things that \( y \) could align to, how likely is it to be \( a \):

\[ p(y \mid a) \]

\[ \frac{p(y \mid a) + p(y \mid b) + p(y \mid c)}{\sum} \]

---

EM: without the alignments

Input: corpus of English/Foreign sentence pairs (no alignment)

for some number of iterations:

for \((E, F)\) in corpus:

for \(e \) in \(E\):

for \(f \) in \(F\):

\[ p(f \rightarrow e) = \frac{p(f \mid e) \sum_{e \in E} p(f \mid e)}{\sum_{e \in E} p(f \mid e)} \]

\[ \text{count}(e, f) += p(f \rightarrow e) \]

\[ \text{count}(e) += p(f \rightarrow e) \]

for all \((e, f)\) in count:

\[ p(f \mid e) = \frac{\text{count}(e, f)}{\text{count}(e)} \]

---

EM: without the alignments

Input: corpus of English/Foreign sentence pairs (no alignment)

for some number of iterations:

for \((E, F)\) in corpus:

for \(e \) in \(E\):

for \(f \) in \(F\):

\[ p(f \rightarrow e) = \frac{p(f \mid e) \sum_{e \in E} p(f \mid e)}{\sum_{e \in E} p(f \mid e)} \]

\[ \text{count}(e, f) += p(f \rightarrow e) \]

\[ \text{count}(e) += p(f \rightarrow e) \]

for all \((e, f)\) in count:

\[ p(f \mid e) = \frac{\text{count}(e, f)}{\text{count}(e)} \]
EM: without the alignments
Input: corpus of English/Foreign sentence pairs (no alignment)

for some number of iterations:
   for (E, F) in corpus:
      for e in E:
         for f in F:
            \( p(f \rightarrow e) = \frac{p(f|e)}{\sum_{e'} p(f|e')} \)
            \( \text{count}(e,f) += p(f \rightarrow e) \)
            \( \text{count}(e) += p(f \rightarrow e) \)

for all \((e,f)\) in count:
   \( p(f|e) = \frac{\text{count}(e,f)}{\text{count}(e)} \)

Where are the E and M steps?

EM: without the alignments
Input: corpus of English/Foreign sentence pairs (no alignment)

for some number of iterations:
   for (E, F) in corpus:
      for e in E:
         for f in F:
            \( p(f \rightarrow e) = \frac{p(f|e)}{\sum_{e'} p(f|e')} \)
            \( \text{count}(e,f) += p(f \rightarrow e) \)
            \( \text{count}(e) += p(f \rightarrow e) \)

for all \((e,f)\) in count:
   \( p(f|e) = \frac{\text{count}(e,f)}{\text{count}(e)} \)

Calculate how probable the alignments are under the current model (i.e. \( p(f|e) \))

Recalculate \( p(f|e) \) using counts from all alignments, weighted by how probable they are

NULL
Sometimes foreign words don’t have a direct correspondence to an English word

Adding a NULL word allows for \( p(f \mid \text{NULL}) \), i.e. words that appear, but are not associated explicitly with an English word

Implementation: add “NULL” (or some unique string representing NULL) to each of the English sentences, often at the beginning of the sentence

\begin{align*}
   p(\text{casa} \mid \text{NULL}) &= \frac{1}{3} \\
p(\text{verde} \mid \text{NULL}) &= \frac{1}{3} \\
p(\text{la} \mid \text{NULL}) &= \frac{1}{3}
\end{align*}
Benefits of word-level model

Rarely used in practice for modern MT system

Mary did not slap the green witch
Maria no dio una botefada a la bruja verde

Two key side effects of training a word-level model:
• Word-level alignment
• \( p(f | e) \): translation dictionary

How do I get this?

Word alignment

100 iterations

<table>
<thead>
<tr>
<th></th>
<th>green</th>
<th>casa</th>
<th>verde</th>
</tr>
</thead>
<tbody>
<tr>
<td>( p(\text{casa}</td>
<td>\text{green}) )</td>
<td>0.005</td>
<td></td>
</tr>
<tr>
<td>( p(\text{verde}</td>
<td>\text{green}) )</td>
<td>0.995</td>
<td></td>
</tr>
<tr>
<td>( p(\text{la}</td>
<td>\text{green}) )</td>
<td>0</td>
<td></td>
</tr>
</tbody>
</table>

How should these be aligned?

100 iterations

<table>
<thead>
<tr>
<th></th>
<th>the</th>
<th>casa</th>
</tr>
</thead>
<tbody>
<tr>
<td>( p(\text{casa}</td>
<td>\text{house}) )</td>
<td>1.0</td>
</tr>
<tr>
<td>( p(\text{verde}</td>
<td>\text{house}) )</td>
<td>0.0</td>
</tr>
<tr>
<td>( p(\text{la}</td>
<td>\text{house}) )</td>
<td>0.0</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th></th>
<th>la</th>
<th>casa</th>
</tr>
</thead>
<tbody>
<tr>
<td>( p(\text{casa}</td>
<td>\text{the}) )</td>
<td>0.005</td>
</tr>
<tr>
<td>( p(\text{verde}</td>
<td>\text{the}) )</td>
<td>0</td>
</tr>
<tr>
<td>( p(\text{la}</td>
<td>\text{the}) )</td>
<td>0.995</td>
</tr>
</tbody>
</table>

Word-level alignment

\[
\text{alignment}(E,F) = \arg \max \prod_{i=1}^{l} p(f_i | e_j)
\]

Which for IBM model 1 is:

\[
\text{alignment}(E,F) = \arg \max \prod_{i=1}^{l} p(f_i | e_j)
\]

Given a model (i.e. trained \( p(f|e) \)), how do we find this?

Align each foreign word \( f \) in \( F \) to the English word \( e \) in \( E \) with highest \( p(f|e) \)

\[
\alpha_i = \arg \max \prod_{i=1}^{l} p(f_i | e_j)
\]
Word-alignment Evaluation

The old man is happy. He has fished many times.

↓ ↓ ↓ ↓ ↓

El viejo está feliz porque ha pesca do muchos veces.

System:

The old man is happy. He has fished many times.

↓ ↓ ↓ ↓ ↓

El viejo está feliz porque ha pecado muchos veces.

Human

The old man is happy. He has fished many times.

↓ ↓ ↓ ↓ ↓

El viejo está feliz porque ha pesca do muchos veces.

Precision and recall!

Human

Precision: \( \frac{6}{7} \)

Recall: \( \frac{6}{10} \)