Engineering Part IIB: Module 4F11
Speech and Language Processing
Lecture 12: Statistical Machine Translation – Alignment Models and Parameter Estimation

Bill Byrne

Lent 2014

Cambridge University Engineering Department
Word Alignment in Translation

Suppose we have a pair of English and French sentences that are known to be translations. How do the words in one sentence ‘generate’ the words that make up the other sentence?

Word Alignments

A link indicates that the words are translations of each other within these sentences.

Suppose the $i^{th}$ English word $e_i$ is associated with the $j^{th}$ French word $f_j$.

This is indicated by an alignment link $(i, j)$.

NULL links: used when simple word-for-word translation is inadequate.

The word alignment of a sentence pair can be specified by an Alignment Link Set.

Word alignment is also specified by the Alignment Process $a_j$.

Alignment Links and the Alignment Process are equivalent representations.

Alignment will play a crucial role in defining the Translation Probability.

---

1 If each French word is generated by a single English word.
Statistical Models for Word-Level Alignment in Translation

Suppose we have an English sentence $E = e^I_1$ and its French translation $F = f^J_1$.

How to define statistical models to assign likelihood to the sentences, i.e. $P(f^J_1|e^I_1)$?

Modeling Assumptions:

- Translation has ‘direction’: the sentence $E = e^I_0$ generates the sentence $F = f^J_1$.
  - $e_0$ is the ‘NULL’ word
- The alignment variable $a^J_1$ indicates the word pairs involved in translation: $e_{a_j} \rightarrow f_j$.
  - $a_j$ is an index of the English sentence $e^I_0$. Therefore $a_j$ takes values in $[0, 1, \ldots, I]$.
- Introduce the Alignment Process: $P(f^J_1, J|e^I_0) = \sum_{a^J_1} P(f^J_1, J, a^J_1|e^I_0)$
  - note that the French sentence length $J$ is a random variable
- Make simplifying conditional independence assumptions

\[
P(f^J_1, a^J_1, J|e^I_0) = P(f^J_1|J, a^J_1, e^I_0) \cdot P(a^J_1|J, e^I_0) \cdot P(J|I) = \prod_{j=1}^{J} p_T(f_j|e_{a_j}) \cdot P_A(a^J_1|J, I) \cdot p_L(J|I)
\]

Translation Model Component Distributions

- Sentence Length Distribution – $p_L(J|I)$
- Word Translation Distribution – $p_T(f|e)$
- Alignment Distribution – $P_A(a^J_1|J, I)$
Sentence Length and Word Translation Distributions

Sentence Length Distribution – $p_L(J|I)$
- Likelihood that an English sentence of $I$ words generates any Foreign sentence of $J$ words
- For specific language pairs, e.g. English $\rightarrow$ Chinese, English $\rightarrow$ French, ..... relative sentence lengths are fairly predictable. For instance, there tends to be approx. 1.5 Chinese words per English word in translated sentences. Distributions can be tuned to language pairs and translation domains.
- Very simple, but very informative!

Word Translation Distribution – $p_T(f|e)$
- Specifies the likelihood that an English word $e$ translates to a foreign word $f$
- Probabilities are maintained in (huge) tables defined for the words in both languages

<table>
<thead>
<tr>
<th>Foreign Vocabulary →</th>
<th>English Vocabulary →</th>
</tr>
</thead>
<tbody>
<tr>
<td>$p_T$(la</td>
<td>the) = 0.2</td>
</tr>
<tr>
<td>$p_T$(le</td>
<td>the) = 0.2</td>
</tr>
<tr>
<td>$p_T$(les</td>
<td>the) = 0.25</td>
</tr>
<tr>
<td>$p_T$(l'</td>
<td>the) = 0.25</td>
</tr>
</tbody>
</table>

- Entries are maintained for the likely foreign word translations and a back-off probability is distributed equally over the remaining foreign words
- Note that morphological forms are maintained, e.g. la, le, les, l’
Simple Alignment Distributions – IBM Model-1 and Model-2

Alignment Distribution – $P_A(a_1^J | J, I)$

- Assigns probability to the alignment links that describe word alignment
- Specifies the likelihood that any word in the $i^{th}$ position in the English sentence is aligned with any word appearing in the $j^{th}$ position in the foreign sentence.
- Does not depend on the words in either language (!)

Model-2 makes the following assumption about the alignment process:

$$P_A(a_1^J | J, I) = \prod_{j=1}^{J} p_{M2}(a_j | j, J, I)$$

- $p_{M2}(i | j, J, I)$ is stored in tables of dimension $I \times J \times J \times I$ (which can be simplified)
- alignment links are independent of each other
- the link distribution depends on the foreign word location $j$

Model-1 assumes that the component distribution is entirely flat: $p_{M2}(a_j | j, J, I) = \frac{1}{J}$

$$P_A(a_1^J | J, I) = \frac{1}{I^J}$$

- Special case of Model-2
- Likelihood of a collection of links $a_1^J$ depends only on the number of links, $J$
Incorporating Hidden Markov Models Into Word Alignment

One of the weaknesses of the Model-1 and Model-2 alignment process is due to the harsh conditional independence assumptions made within the alignment component:

\[
P(a_j^1 | J, l, e_0^J) \approx \prod_{j=1}^{J} P_A(a_j | J, l)
\]

A consequence of this assumption is that the alignment of individual foreign words is determined independently of the alignment of their neighbors, e.g. alignment of any Foreign word \( f_j \) is independent of the alignment of either \( f_{j-1} \) or \( f_{j+1} \).

The HMM Word Alignment Model strengthens the overall alignment model by introducing a first-order Markov process into the Alignment Distribution:

\[
P_A(a_j^1 | J, l) = \prod_{j=1}^{J} P(a_j | a_{j-1}^1, J, l) \approx \prod_{j=1}^{J} p_{HMM}(a_j | a_{j-1}, l)
\]

- **Markov Position Alignment Distribution**: \( p_{HMM}(a_j | a_{j-1}, l) \)
- Each link in the alignment sequence depends on the previous link.
- The overall process is an HMM
- Note that Model-1 and Model-2 are (almost) special cases of HMM Alignment
Example – Generation of HMM Word Alignments

States are associated with the words on the ‘generating’ side
Example – Generation of HMM Word Alignments

States are associated with the words on the ‘generating’ side
State sequences are generated and determine the word-to-word alignments
Words are generated one by one, one word with each state transition
States can occur multiple times in an alignment sequence
Summary of IBM Model-1, Model-2, and HMM Word Alignment

Step 1. Specify the foreign sentence length $J$ under $P_L(J|I)$

Step 2. Generate a word alignment sequence of length $J$

Model 1: $P_A(a^J_1 | e^I_0, J) = \frac{1}{(I+1)^J}$

Model 2: $P_A(a^J_1 | e^I_0, J) = \prod_{j=1}^{J} p_{M2}(a_j | J, I)$

HMM: $P_A(a^J_1 | e^I_0, J) = \prod_{j=1}^{J} p_{HMM}(a_j | a_{j-1}, I)$

Step 3. Generate the foreign sentence from the aligned English words:

$P(f^J_1 | a^J_1, e^I_0) = \prod_{j=1}^{J} p_T(f_j | e_{a_j})$

Overall Translation Probability Under Each Model:

$P(f^J_1, a^J_1, J|e^I_0) = \frac{1}{(I+1)^J} p_L(J|I) \prod_{j=1}^{J} p_T(f_j | e_{a_j}) - \text{Model-1}$

$P(f^J_1, a^J_1, J|e^I_0) = p_L(J|I) \prod_{j=1}^{J} p_T(f_j | e_{a_j}) p_{M2}(a_j | J, I) - \text{Model-2}$

$P(f^J_1, a^J_1, J|e^I_0) = p_L(J|I) \prod_{j=1}^{J} p_T(f_j | e_{a_j}) p_{HMM}(a_j | a_{j-1}, I) - \text{HMM}$
Automatic Alignments and Translation Probability

Suppose we have an alignment model - either Model-1, Model-2, or an HMM
Suppose we also have a pair of sentences \((e_1^l, f_1^l)\) which we know are translations.
We can use these models to find Automatic Alignments of the known translations.

\[
\hat{a}_1^j = \arg\max_{a_1^j} P(f_1^l, a_1^j, J, | e_1^l)
\]

We can also carry out efficient marginalization over all alignments to find the translation probability

\[
P(f_1^l | e_1^l) = \sum_{a_1^j} P(f_1^l, a_1^j, J, | e_1^l)
\]

- With Model-1, Model-2, and the HMM, these operations can be carried out either by the Viterbi procedure or the Forward-Backward Algorithm, respectively.
Parameter Estimation in Word Alignment Models

The basic models used in word alignment were defined in the previous lecture

- IBM Model-1, IBM Model-2, Alignment HMM

The general forms of the alignment models are specified by their component distributions

- Probability that a foreign sentence $F = f_1^J$ is a translation of an English sentence $E = e_0^I$:

$$P(f_1^J, a_1^J, J| e_0^I) = \underbrace{p_L(J| I)}_{\text{Sentence Length Distribution}} \underbrace{P_A(a_1^J| J, I)}_{\text{Alignment Distribution}} \prod_{j=1}^{J} \underbrace{p_T(f_j| e_{a_j})}_{\text{Word Translation Distribution}}$$

We have a collection of training sentences: $\{F^{(r)}, E^{(r)}\}_{r=1}^R \leftarrow \text{Sentence Aligned Parallel Text}$

How can the component models be estimated from the example translations?

**Sentence Length Distribution**:

Easy to estimate from sentence-level statistics:

$$P_L(J| I) = \frac{\sum_{r=1}^{R} 1(J = J^{(r)}, I = I^{(r)})}{\sum_{r=1}^{R} 1(I = I^{(r)})}$$

→ a distribution based on a simple sentence length ratio also works well

Estimation of other model components requires word alignment information...
Word Translation Distributions from Word Alignments

Recall that word aligned sentences can be represented by a set of Word Alignment Links

\[ e_i \leftrightarrow f_j \iff a_j = i \iff e_{a_j} \leftrightarrow f_j \]

NULL  Mr. Speaker, my question is directed to the Minister of Transport

Monsieur le Orateur, ma question se adresse à le ministre chargé de les transports

Word Translation Distribution:

How do we estimate \( P_T(f \mid e) \) from word-aligned sentences \( \{F^{(r)}, E^{(r)}, a^{(r)}\}_{r=1}^R \)?

\[ a^{(r)} = [a^{(r)}_1, \ldots, a^{(r)}_{J^{(r)}}] \leftarrow \text{word alignments for } F^{(r)}, E^{(r)} \]

Step 1: Count the number of times a foreign word \( f \) is aligned to an English word \( e \)

\[ \#_T(f \leftrightarrow e) = \sum_{r=1}^R \sum_{j=1}^{J^{(r)}} \sum_{i=1}^{I^{(r)}} 1(e = e_i^{(r)}) \cdot 1(f = f_j^{(r)}) \cdot 1(a_j^{(r)} = i) \]

- \( e \) is the \( i^{th} \) word of \( E^{(r)} \)
- \( f \) is the \( j^{th} \) word of \( F^{(r)} \)
- \( e_i^{(r)} \leftrightarrow f_j^{(r)} \)

Step 2: Compute the Word Translation Distribution:

\[ P_T(f \mid e) = \frac{\#_T(f \leftrightarrow e)}{\sum_{f'} \#_T(f' \leftrightarrow e)} \]
Word Translation Distributions from Word Alignments - An Example

Two aligned sentence pairs: \( R = 2, I^{(1)} = I^{(2)} = 5, J^{(1)} = J^{(2)} = 4 \)

| \( E^{(1)} \) | \( e_0^{(1)} e_1^{(1)} e_2^{(1)} e_3^{(1)} e_4^{(1)} e_5^{(1)} \) |
| \( F^{(1)} \) | \( f_1^{(1)} f_2^{(1)} f_3^{(1)} f_4^{(1)} \) |
| English Sentence | Foreign Sentence |
| NULL | a r p q r |
| e \( \gamma \) p e |
| a(1): a_1^{(1)} = 2, a_2^{(1)} = 1, a_3^{(1)} = 3, a_4^{(1)} = 5 |

| \( E^{(2)} \) | \( e_0^{(2)} e_1^{(2)} e_2^{(2)} e_3^{(2)} e_4^{(2)} e_5^{(2)} \) |
| \( F^{(2)} \) | \( f_1^{(2)} f_2^{(2)} f_3^{(2)} f_4^{(2)} \) |
| English Sentence | Foreign Sentence |
| NULL | o m r r a |
| \( \beta \) \( \alpha \) e \( \theta \) |
| a(2): a_1^{(2)} = 3, a_2^{(2)} = 0, a_3^{(2)} = 2, a_4^{(2)} = 4 |

\[
\#_T(\epsilon \leftrightarrow r) = 2 \\
p_T(\epsilon|r) = \frac{\#_T(\epsilon \leftrightarrow r)}{\#_T(r)} = \frac{2}{4} \\
\#_T(\theta \leftrightarrow r) = 1 \\
p_T(\theta|r) = \frac{\#_T(\theta \leftrightarrow r)}{\#_T(r)} = \frac{1}{4} \\
\#_T(\beta \leftrightarrow r) = 1 \\
p_T(\beta|r) = \frac{\#_T(\beta \leftrightarrow r)}{\#_T(r)} = \frac{1}{4} \\
\#_T(r) = 4
\]
Estimation of Model-2 Alignment Distribution

Model-2 Position Alignment Distribution: $P_A(a_i^j | J, I) = \prod_{j=1}^{J} p_{M2}(a_j | J, I)$

How do we estimate $p_{M2}(i | j, J, I)$ from word-aligned sentences?

Step 1: Accumulate position alignment statistics over the word-aligned sentences

$$\#_{M2}(i, j, J, I) = \sum_{r=1}^{R} 1(J = J^{(r)}, I = I^{(r)}) \sum_{j=1}^{J} 1(i = a_j^{(r)})$$

→ How often is position $i$ linked with position $j$ in sentences of length $J$ and $I$.

Step 2: Compute the Position Alignment Distribution

$$p_{M2}(i | j, J, I) = \frac{\#_{M2}(i, j, J, I)}{\sum_{i''=1}^{I} \#_{M2}(i'', j, J, I)}$$

→ Probability of linking position $j$ to position $i$ in sentences of length $J$ and $I$.
Estimation of Model-2 Alignment Distribution - An Example

Two aligned sentence pairs: \( R = 2, I^{(1)} = I^{(2)} = 5, J^{(1)} = J^{(2)} = 4 \)

\[
\begin{align*}
E^{(1)} &= e_0^{(1)} e_1^{(1)} e_2^{(1)} e_3^{(1)} e_4^{(1)} e_5^{(1)} \\
F^{(1)} &= f_1^{(1)} f_2^{(1)} f_3^{(1)} f_4^{(1)} \\
E^{(2)} &= e_0^{(2)} e_1^{(2)} e_2^{(2)} e_3^{(2)} e_4^{(2)} e_5^{(2)} \\
F^{(2)} &= f_1^{(2)} f_2^{(2)} f_3^{(2)} f_4^{(2)}
\end{align*}
\]

English Sentence

Foreign Sentence

\[
\begin{align*}
a^{(1)} &: a_1^{(1)} = 2, a_2^{(1)} = 1, a_3^{(1)} = 3, a_4^{(1)} = 5 \\
a^{(2)} &: a_1^{(2)} = 3, a_2^{(2)} = 0, a_3^{(2)} = 2, a_4^{(2)} = 4
\end{align*}
\]

\[
\begin{align*}
#_{M2}(i = 1, j = 2, J = 4, I = 5) &= 1 \\
p_{M2}(i = 1 | j = 2, J = 4, I = 5) &= \frac{#_{M2}(i=1,j=2,J=4,I=5)}{#_{M2}(j=2,J=4,I=5)} = \frac{1}{2} \\
#_{M2}(i = 2, j = 2, J = 4, I = 5) &= 1 \\
p_{M2}(i = 2 | j = 2, J = 4, I = 5) &= \frac{#_{M2}(i=2,j=2,J=4,I=5)}{#_{M2}(j=2,J=4,I=5)} = \frac{1}{2} \\
#_{M2}(j = 2, J = 4, I = 5) &= 2
\end{align*}
\]

Note that the alignment probabilities depend only on the word alignment information. The identity of the words in each position in the sentence does not play a role once the alignments are generated.
Estimation of HMM Alignment Distribution

**Markov Position Alignment Distribution** : \( P_A(a^j_1 | J, I) = \prod_{j=1}^J p_{\text{HMM}}(a_j | a_{j-1}, I) \)

**Step 1** : Accumulate position-to-position occurrence statistics over the alignments

\[
\#_{\text{HMM}}(i, i', I) = \sum_{r=1}^R 1(I = l^{(r)}) \sum_{j=1}^{J^{(r)}} 1(i' = a^{(r)}_{j-1}, i = a^{(r)}_j)
\]

\[
e_i^{(r)} \leftrightarrow f_{j-1}^{(r)} \text{ and } e_i^{(r)} \leftrightarrow f_j^{(r)}
\]

→ **Number of links to position \( i' \) followed by links to \( i \) in English sentences of length \( I \)

**Step 2** : Compute the Markov Position Alignment Distribution from these occurrence statistics

\[
p_{\text{HMM}}(i | i', I) = \frac{\#_{\text{HMM}}(i, i', I)}{\sum_{i''=1}^I \#_{\text{HMM}}(i'', i', I)}
\]

→ **Probability of linking to position \( i \) after a link to \( i' \) in an English sentence of length \( I \)**
Estimation HMM Alignment Distribution - *An Example*

Two aligned sentence pairs: \( R = 2 \), \( I^{(1)} = I^{(2)} = 5 \), \( J^{(1)} = J^{(2)} = 4 \)

English Sentence

\[
\begin{array}{cccccc}
\text{NULL} & a & r & p & q & r \\
0 & \varepsilon & \gamma & \rho & \varepsilon \\
1 & \alpha & \varepsilon & \theta \\
2 & \beta & \alpha & \varepsilon & \theta \\
3 & \beta & \alpha & \varepsilon & \theta \\
4 & \beta & \alpha & \varepsilon & \theta \\
5 & \beta & \alpha & \varepsilon & \theta \\
\end{array}
\]

Foreign Sentence

\[
\begin{array}{cccccc}
\text{NULL} & o & m & r & r & a \\
0 & \beta & \alpha & \varepsilon & \theta \\
1 & \beta & \alpha & \varepsilon & \theta \\
2 & \beta & \alpha & \varepsilon & \theta \\
3 & \beta & \alpha & \varepsilon & \theta \\
4 & \beta & \alpha & \varepsilon & \theta \\
5 & \beta & \alpha & \varepsilon & \theta \\
\end{array}
\]

\[E^{(1)}_e = e^{(1)}_0 e^{(1)}_1 e^{(1)}_2 e^{(1)}_3 e^{(1)}_4 e^{(1)}_5 \]

\[F^{(1)}_f = f^{(1)}_1 f^{(1)}_2 f^{(1)}_3 f^{(1)}_4 \]

\[a^{(1)}: a^{(1)}_1 = 2, a^{(1)}_2 = 1, a^{(1)}_3 = 3, a^{(1)}_4 = 5 \]

\[E^{(2)}_e = e^{(2)}_0 e^{(2)}_1 e^{(2)}_2 e^{(2)}_3 e^{(2)}_4 e^{(2)}_5 \]

\[F^{(2)}_f = f^{(2)}_1 f^{(2)}_2 f^{(2)}_3 f^{(2)}_4 \]

\[a^{(2)}: a^{(2)}_1 = 3, a^{(2)}_2 = 0, a^{(2)}_3 = 2, a^{(2)}_4 = 4 \]

\[\#_{HMM}(i, i', l) : \text{number of times } i \text{ follows } i' \text{ in the word alignments} \]

- easily computed from the reordered English word index sequences 2, 1, 3, 5 and 3, 0, 2, 4

\[
\begin{array}{c|c}
\#_{HMM}(i = 1, i' = 2, l = 5) & 1 \\
\#_{HMM}(i = 4, i' = 2, l = 5) & 1 \\
\#_{HMM}(i = 5, i' = 3, l = 5) & 1 \\
\#_{HMM}(i = 0, i' = 3, l = 5) & 1 \\
\end{array}
\]

\[
\begin{array}{c|c}
p_{HMM}(a_j = 1|a_{j-1} = 2, l = 5) & \frac{1}{2} \\
p_{HMM}(a_j = 4|a_{j-1} = 2, l = 5) & \frac{1}{2} \\
p_{HMM}(a_j = 5|a_{j-1} = 3, l = 5) & \frac{1}{2} \\
p_{HMM}(a_j = 0|a_{j-1} = 3, l = 5) & \frac{1}{2} \\
\end{array}
\]

\[
\sum_{i''=1}^{l=5} \#_{HMM}(i'', i' = 2, l = 5) = 2 \\
\sum_{i''=1}^{l=5} \#_{HMM}(i'', i' = 3, l = 5) = 2
\]
Automatic Word Alignments and Iterative Parameter Estimation

Viterbi Alignment can be performed for: IBM Model-1, IBM Model-2, and the HMM

\[ \hat{a}_i^j = \arg\max_{a_i^j} P(f_i^j, a_i^j | e_0^i) \]

Flat Start Training Procedure: Gradually increase the complexity of the alignment distribution

1. Model-1
   1.1 Model-1 Initialization – set \( p_T(f|e) \) to be uniform
   1.2 Perform initial Model-1 Viterbi alignment to generate word aligned parallel text
   1.3 Update \( p_T(f|e) \) from the word-aligned parallel text
   1.4 Perform Model-1 Viterbi alignment to generate word aligned parallel text
   1.5 Return to Step 1.3 until some stopping criteria is met
   1.6 Output: Model-1 word-aligned parallel text

2. Model-2
   2.1 Model-2 Initialization – find \( p_{M2}(i|j, J, I) \) and \( p_T(f|e) \) using the word-alignments from Step 1.6
   2.2 Perform Model-2 Viterbi alignment to generate word aligned parallel text
   2.3 Update \( p_{M2}(i|j, J, I) \) and \( p_T(f|e) \) from the word-aligned parallel text
   2.4 Return to Step 2.2 until some stopping criteria is met
   2.5 Output: Model-2 word-aligned parallel text

3. HMM
   3.1 HMM Initialization – find \( p_{HMM}(i|i', I, J) \) and \( p_T(f|e) \) using the word-alignments from Step 2.5
   3.2 Perform HMM Viterbi alignment to generate word aligned parallel text
   3.3 Update \( p_{HMM}(i|i', I, J) \) and \( p_T(f|e) \) from the word-aligned parallel text
   3.4 Return to Step 3.2 until some criteria is met
   3.5 Perform a final HMM Viterbi alignment to generate word aligned parallel text
   3.6 Output: HMM parameters and word-aligned parallel text
Iterative Alignment Model Parameter Estimation by EM

This estimation procedure uses ‘hard counts’ extracted from word alignments.

For example, the Word Translation Distribution relies on the function \( 1(a^{(r)}_j = i) \) which specifies that \( f_j^{(r)} \leftrightarrow e_i^{(r)} \).

\[
\#(f \leftrightarrow e) = \sum_{r=1}^{R} \sum_{j=1}^{J^{(r)}} \sum_{i=1}^{I^{(r)}} 1(e = e_i^{(r)}) 1(f = f_j^{(r)}) 1(a^{(r)}_j = i)
\]

EM replaces ‘hard counts’ by posterior probabilities

\[
\#(f \leftrightarrow e) = \sum_{r=1}^{R} \sum_{j=1}^{J^{(r)}} \sum_{i=1}^{I^{(r)}} 1(e = e_i^{(r)}) 1(f = f_j^{(r)}) P(a^{(r)}_j = i | E^{(r)}, F^{(r)})
\]

The Word Translation distribution is found as before, but with the posterior probabilities

\[
P_T(f|e) = \frac{\#(f \leftrightarrow e)}{\sum_{f' \neq f} \#(f' \leftrightarrow e)}
\]

and estimation formulae for the Model-2 and HMM alignment distributions are similar.

Posterior probabilities are easy to compute under the HMM – very similar to ASR.
IBM Model-4 for Word Alignment

Model 4 is a powerful model of word alignment within sentence pairs
Features:

- **Lexical translation model**: as in Model 1 and HMM alignment model
- **Fertility**: probability distribution over the number of French words each English word can generate
  - an approximation to *phrase modeling*
- **NULL Translation Model**: allows French words to be generated without a corresponding source on the English side
- **Distortion Model**: describes how French words are distributed throughout the French sentence when generated from a single English word
Model-4 Generation: Step 1 – Tablets

Create a tablet for each source word
Model-4 Generation: Step 2 – Fertility

Table lookup to decide fertility: # of target words connected

Chinese: 中国, 早日, 加入, 世贸组织
Model-4 Generation: Step 3 – Fill in Tablet Positions

Sample target words from translation table i.i.d.
Model-4 Generation: Step 4 – Fill in the English Sentence

The bitext word alignment process involves aligning words from different languages. In this example, the English sentence is:

`the China’s accession to the world trade organization at an early date`

The corresponding Chinese sentence is:

`中国早日加入世贸组织`

The alignment model depicted in the diagram shows how the English sentence is aligned with the Chinese sentence, with each word in the English sentence matched to its equivalent in the Chinese sentence. The alignment process is crucial for machine translation tasks, as it helps in mapping words from one language to another accurately.

The distortion model shown in the diagram helps in adjusting the alignments to minimize errors and improve the overall translation quality.
IBM Model-4 Summary

Very powerful model of word alignment in translated sentences

- Fertility and Distortion begin to capture generation of phrases in translation.

Model-4 has several shortcomings

- Deficiency - The Distortion Model used to move words from the tables into position in the sentence may place two words in one sentence position. This is a drawback – some probability mass under the model is assigned to what are effectively non-sentences. The probabilities over alignments and generated sentences do not sum to 1.0. The models are therefore called deficient.

- Parameter estimation and word alignment is difficult to implement
  - Unlike Model-1, Model-2, and HMM Alignment, dynamic programming based algorithms are not available for Model-4.

*It is difficult to balance modeling power against computational efficiency in alignment.*

Despite these difficulties, IBM Model-4 can be used to generate high-quality word alignments over parallel text, especially as implemented under the GIZA++ toolkit. However, it doesn’t parallelize as easily other models.

---

2 http://www.fjoch.com/GIZA++.html
Document Level Alignment

The definition of a document varies. Depending on the domain, a document could be a book, a news story, a numbered chapter, paragraph, or verse, ...

Parallel Documents are created purposefully by human translators, for example in creating a foreign language edition of a newspaper.

- correspondence between the translations and the original source language documents is maintained
- these are the most valuable, but are expensive to create and to obtain
- large collections of parallel documents are available from LDC

Some interest in searching for Parallel Documents ‘in the wild’ by crawling the web and looking for hints that documents encountered might be translations of each other.

... assume we have parallel documents
Document and Sentence Aligned Parallel Text

Parameter estimation procedures discussed so far require sentence-level translations for use as a training set. However, most parallel texts are available as documents.

The definition of a document varies. Depending on the domain, a document could be a book, a news story, a numbered chapter, paragraph, or verse, ...

English Document
[1] Perhaps the Commission or you could clarify a point for me. [2] It would appear that a speech made at the weekend by Mr Fischler indicates a change of his position. [3] I welcome this change because he has said that he will eat British beef [4] and that the ban was imposed specifically for economic and political reasons. [5]

French Document

Automatic Sentence Alignment:

Given a pair of documents, the goal of Sentence Alignment is to extract subsequences of text that are mutual translations. These may be sentences, sub-sentences, or multiple sentences.

- ‘Markers’ are inserted into the text to indicate possible translation boundaries
- Translation segments can be sub-sentence units, e.g. at [4]. As a result some aligned ‘sentences’ might be sub-sentence fragments.
- Segments can be discarded as spurious text without a corresponding translation

Aligned segments can be extracted and used for estimation of word alignment models.
Automatic Sentence Alignment Procedures

An Alignment Grid defines the possible segment alignments between two documents

- A path defines both segmentation and alignment

Segment Alignment Specified by Path (a)
- $E(1)E(2)$ are merged and aligned with $F(1)$
- $E(3)$ is deleted
- $E(4)$ is aligned to $F(2)$
- $F(3)F(4)$ are merged and aligned with $E(5)$

Alignment is based on two underlying processes:

1. A probability distribution defined over alignment paths.
   - Paths near the diagonal with shorter segments are preferred
2. A translation probability defined over aligned segment pairs
   - Employs a simple model such as IBM Model-1 with a sentence length component

The two scores are combined to assign a probability to a particular segmentation and alignment of the document pairs.

Monotone Segment Alignment

Dynamic programming search finds the most likely alignment of translation segments.

- Simple translation models (e.g. Model-1) are used so that large amounts of parallel documents can be sentence-aligned efficiently
- The alignment is monotone: segments may be deleted but not reordered. This prevents the search space from growing too large.
Sentence Alignment Goal: Better Word Translation Models

Benefits of good sentence alignment
- better training set alignment improves translation performance
- shorter aligned segments leads to faster translation model training

A good alignment algorithm should be ...
- fast, so multiple iterations are possible
- efficient: as little bitext should be discarded as possible
- initialized from a flat-start
- language independent
- require minimal linguistic knowledge
- able to work at the subsentence level
  - coarse monotonic alignment followed by fine divisive clustering works well
  - splitting points can depend on the language pairs