Computational Linguistics Seminar
LING-696G

Week 8
Last Time

- IBM Model 1 review
  - no alignment
- IBM Model 2
  - alignment probability distribution
  - *Python code on website*

Today's Topics

- IBM Model 3
- `nltk.translate`
- *Ideas for a group project*
Assume nltk 3.0 on python 3

• Python 3
  – download from python.org
  – non-standard on OS X (Apple: Python 2.7.11)
  – which python3
  – /Library/Frameworks/Python.framework/Versions/3.5/bin/python3
  – printenv PATH

• NLTK
  – [http://www.nltk.org/install.html](http://www.nltk.org/install.html)
  – nltk for 2.7 and 3.5 are separate...
  – pip targets my 2.7.11 install
  – which pip3
  – /Library/Frameworks/Python.framework/Versions/3.5/bin/pip3
  – sudo pip3 install –U nltk
nltk.translate

- [http://www.nltk.org/api/nltk.translate.html](http://www.nltk.org/api/nltk.translate.html)
- **Module:**
  - from nltk.translate import AlignedSent, Alignment, IBMMModel1
- **Data:**
  ```python
  bitext = []
  bitext.append(AlignedSent(['klein', 'ist', 'das', 'haus'], ['the', 'house', 'is', 'small']))
  bitext.append(AlignedSent(['das', 'haus', 'ist', 'ja', 'groß'], ['the', 'house', 'is', 'big']))
  bitext.append(AlignedSent(['das', 'buch', 'ist', 'ja', 'klein'], ['the', 'book', 'is', 'small']))
  bitext.append(AlignedSent(['das', 'haus'], ['the', 'house']))
  bitext.append(AlignedSent(['das', 'buch'], ['the', 'book']))
  bitext.append(AlignedSent(['ein', 'buch'], ['a', 'book']))
  ```
- **Run:**
  ```python
  ibm1 = IBMMModel1(bitext, 5)
  ```
nltk.translate

- **Translation probabilities:**
  
  ```
  print(ibm1.translation_table['buch']['book'])
  0.889...
  
  print(ibm1.translation_table['das']['book'])
  0.061...
  
  print(ibm1.translation_table['buch'][None])
  0.113...
  
  print(ibm1.translation_table['ja'][None])
  0.072...
  ```

- **Alignment:**
  
  ```python
  test_sentence = bitext[2]
  test_sentence.words
  ['das', 'buch', 'ist', 'ja', 'klein']
  
  test_sentence.mots
  ['the', 'book', 'is', 'small']
  
  test_sentence.alignment
  Alignment([(0, 0), (1, 1), (2, 2), (3, 2), (4, 3)])
  ```
**Model 2 training data:**

```python
bitext = []
bitext.append(AlignedSent(['klein', 'ist', 'das', 'haus'], ['the', 'house', 'is', 'small']))
bitext.append(AlignedSent(['das', 'haus', 'ist', 'ja', 'groß'], ['the', 'house', 'is', 'big']))
bitext.append(AlignedSent(['das', 'buch', 'ist', 'ja', 'klein'], ['the', 'book', 'is', 'small']))
bitext.append(AlignedSent(['das', 'haus'], ['the', 'house']))
bitext.append(AlignedSent(['das', 'buch'], ['the', 'book']))
bitext.append(AlignedSent(['ein', 'buch'], ['a', 'book']))
```
nltk.translate

- **Model 2 alignment table:**
  
  ```python
  print(ibm2.alignment_table[1][1][2][2])
  0.938...
  print(round(ibm2.alignment_table[1][2][2][2], 3))
  0.0
  print(round(ibm2.alignment_table[2][2][4][5], 3))
  1.0
  ```
We will describe Model 3 in a moment ...

- **Model 3 distortion table:**
  ```python
  print(round(ibm3.distortion_table[1][1][2][2], 3))
  1.0
  print(round(ibm3.distortion_table[1][2][2][2], 3))
  0.0
  print(round(ibm3.distortion_table[2][2][4][5], 3))
  0.75
  ```

- **Model 3 fertility table:**
  ```python
  print(round(ibm3.fertility_table[2]['summarize'], 3))
  1.0
  print(round(ibm3.fertility_table[1]['book'], 3))
  1.0
  ```

Some differences between the module and the book ...
Chapter 4 Errata for Model 3

• p. 103: The conditional fertility distribution should be instead of . This occurs in Equation 4.32 and 4.33 and in the mention inbetween.

• p. 103: Equation 4.33 should be

\[ p(e,f) = \cdots = \epsilon \sum_{a(1)=0}^{l_f} \cdots \sum_{a(l_e)=0}^{l_f} \left( \frac{l_e-\phi_0}{\phi_0} \right) \cdots \]

• instead of

\[ p(e,f) = \cdots = \epsilon \sum_{a(1)=0}^{l_f} \cdots \sum_{a(l_e)=0}^{l_f} \prod_{j=1}^{l_e} \left( \frac{l_e-\phi_0}{\phi_0} \right) \cdots \]

• p. 106: Line 13 of the algorithm should make reference to the term \( f_{a(j)} \) not the term \( f_{a}(j) \)

• p. 107: Line 3 of function neighboring should start with for instead of for for.
4.4.2 IBM Model 3

• Features:
  1. fertility: # of words per foreign word
  2. NULL word generation
  3. distortion
4.4.2 IBM Model 3

- **fertility** of input words:
  - \( n(\phi | f) = \) a probability
  - \( \phi = 0, 1, 2, \ldots \) output words  \hspace{1cm} (4.29)

- **Note:**
  - Fertility deals explicitly with dropping input words by allowing \( \phi = 0 \).

- \( \phi_i = \) number of words generated by \( f_i \)

- \( l_f (l_e) = \) total number of foreign (English) words (respectively)

- assume \( f_0 \) (NULL)

\[
\sum_{i=1}^{\hat{l}_f} \phi_i = l_e - \phi_0
\]
4.4.2 IBM Model 3

• **Fertility step** ($fs \rightarrow fs$):

```
1 2 3 4 5 6
ich gehe ja nicht zum haus
```

```
ich gehe nicht zum zum haus
```

• **Notes:**

1. **duplication** of $f_5$ zum.
2. $f_3$ ja maps to **zero words**.
   - some German words like zum typically translate to two English words, i.e., *to the*. Others, such as the flavoring particle ja, get dropped.
   - $n(1|haus) \approx 1$
   - $n(2|zum) \approx 1$
   - $n(0|ja) \approx 1$  \hspace{1cm} (4.30)
4.4.2 IBM Model 3

- **Null Generation (Insertion) step** (fs → fs):

  - # of inserted words *clearly* depends on the sentence length.
  - After the fertility step, introduce one NULL token with probability \( p_1 \) *after each generated word*, or none with probability \( p_0 = 1 - p_1 \).

Let \( \phi_0 \) be the # of NULLs generated.

By the Binomial Distribution \( B(# \text{ generated words}; p_0) \):

\[
p(\phi_0) = \binom{l_e - \phi_0}{\phi_0} p_1^{\phi_0} p_0^{l_e - 2\phi_0}
\]

(4.31)
4.4.2 IBM Model 3

• Fertility + NULL insertion:

\[
\left( l_e - \phi_0 \right) p_1^{\phi_0} p_0^{l_e - 2\phi_0} \prod_{i=1}^{l_f} \phi_i! \ n(\phi_i \ f_i) \quad (4.32)
\]
4.4.2 IBM Model 3

\[
\begin{pmatrix}
  l_e - \phi_0 \\ \phi_0
\end{pmatrix}
\begin{pmatrix}
  p_1 \\
  p_0
\end{pmatrix}
\begin{pmatrix}
  l_f \\
  \prod_{i=1}^{l_f} \phi_i!
\end{pmatrix}
\begin{pmatrix}
  n(\phi_i f_i) \\
  (f_i \text{fertility 3})
\end{pmatrix}
\]

(4.32)

- Multiple ways ($\phi_i!$):
  - 2! $f \rightarrow f \rightarrow e_2$ or $f \rightarrow f \times e_2$
  - 3! $e_1 e_2 e_3$ from $f_i f_i f_i$ in 6 ways
4.4.2 IBM Model 3

I do not go to the house.

I go.
4.4.2 IBM Model 3

• Recall Model 2 alignment:
  – $a(i \mid j, l_e, l_f)$ \hspace{1cm} (4.24)
  – foreign word $fs[i]$ paired with English word $es[j]$

• **Distortion** is modeled almost the same way as in IBM Model 2 with a probability distribution
  – $d(j \mid i, l_e, l_f)$
  which predicts the English output word position $j$ based on the foreign input word position $i$ and the respective sentence lengths.

• **Example:**
  – the placement of *go* as the 4th word of the seven-word English sentence as a translation of *gehe*, which was the 2nd word in the six-word German sentence, has probability $d(4 \mid 2, 7, 6)$
  – cf. $a(2 \mid 4,7,6)$
To finish up, let us combine the three elements of Model 3 – fertility, lexical translation, and distortion – in one formula:

\[
P(e|f) = \sum_a p(e, a|f) = \sum_{a(1)=0}^{l_f} \ldots \sum_{a(l_e)=0}^{l_f} \left( \frac{l_e - \phi_0}{\phi_0} \right)^{\phi_0} \left( \frac{l_e - 2\phi_0}{\phi_0} \right)^{l_e - 2\phi_0} \prod_{i=1}^{l_f} \phi_i! \cdot n(\phi_i|e_i) \times \prod_{j=1}^{l_e} t(e_j|f_{a(j)}) d(j|a(j), l_e, l_f)
\]
4.4.2 IBM Model 3

- **Alignment:**
  - use model 2 trained alignment as a starting point,
  - then do **hill-climbing**
    (sample: *don't look at all possible alignments*)

- **Neighboring alignments:**
  - **1 MOVE** difference: functions $a_1$ and $a_2$ are the same except in the case of a single word $j$, i.e. $a_1(j)$ and $a_2(j)$ differ
  - **1 SWAP** difference: functions $a_1$ and $a_2$ are the same except for *nontrivial* $j_1$ and $j_2$ (*nontrivial*: $j_1 \neq j_2$ and $a_1(j_1) \neq a_1(j_2)$) the assignments are swapped, i.e. $a_1(j_1) = a_2(j_1)$ and $a_1(j_2) = a_2(j_1)$
ibm_data2.txt

1. a the book house is small big
2. klein ist das Haus groß Buch ein ja
3. the house is small
4. klein ist das Haus
5. the house is big
6. das Haus ist ja groß
7. the book is small
8. das Buch ist ja klein
9. the house
10. das Haus
11. the book
12. das Buch
13. a book
14. ein Buch
• **tu.py** on website leveraging *nltk.translate*

  Usage:
  ```
  python3 tu.py -h
  ```

  ```
  python3 tu.py 5 ibm_data2.txt -m 3
  ```

  Using IBM Model 2
  Number of iterations: 5
  Read training data from file: ibm_data2.txt,
  pairs: 6, max word length: 5

  ist | is       | 0.60
  klein | small  | 1.00
  das | the     | 1.00
  das | None    | 1.00
  Haus | house   | 1.00
  groß | big     | 1.00
  ein  | a       | 1.00
  Buch | book    | 1.00
  ja   | is      | 0.40

  **Translation:**
  f_word e_word prob

  0<-1: 0.06 2,2
  1<-1: 0.94 2,2
  2<-2: 1.00 2,2
  0<-3: 0.47 4,4
  1<-3: 0.53 4,4
  2<-4: 1.00 4,4
  3<-2: 1.00 4,4
  4<-1: 1.00 4,4
  0<-1: 0.47 4,5
  1<-1: 0.53 4,5
  2<-2: 1.00 4,5
  3<-3: 1.00 4,5
  3<-4: 1.00 4,5
  4<-5: 1.00 4,5

  **Alignment:**
  j<-i: prob l_e,l_f

  Some differences between the module and the book ...


```python
python3 -i tu.py 20 ibm_data2.txt -m 3
```

Using IBM Model 3
Number of iterations: 20
Read training data from file: ibm_data2.txt,
pairs: 6, max word length: 5

<table>
<thead>
<tr>
<th>Word</th>
<th>Probability</th>
</tr>
</thead>
<tbody>
<tr>
<td>ist</td>
<td>0.60</td>
</tr>
<tr>
<td>ist</td>
<td>0.21</td>
</tr>
<tr>
<td>das</td>
<td>1.00</td>
</tr>
<tr>
<td>das</td>
<td>0.20</td>
</tr>
<tr>
<td>Haus</td>
<td>1.00</td>
</tr>
<tr>
<td>Haus</td>
<td>0.11</td>
</tr>
<tr>
<td>Buch</td>
<td>1.00</td>
</tr>
<tr>
<td>Buch</td>
<td>0.09</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Word</th>
<th>Probability</th>
</tr>
</thead>
<tbody>
<tr>
<td>groß</td>
<td>1.00</td>
</tr>
<tr>
<td>groß</td>
<td>0.09</td>
</tr>
<tr>
<td>ja</td>
<td>0.40</td>
</tr>
<tr>
<td>ja</td>
<td>0.19</td>
</tr>
<tr>
<td>klein</td>
<td>1.00</td>
</tr>
<tr>
<td>klein</td>
<td>0.11</td>
</tr>
<tr>
<td>ein</td>
<td>1.00</td>
</tr>
<tr>
<td>a</td>
<td>1.00</td>
</tr>
<tr>
<td>the</td>
<td>1.00</td>
</tr>
<tr>
<td>book</td>
<td>1.00</td>
</tr>
<tr>
<td>house</td>
<td>1.00</td>
</tr>
<tr>
<td>is</td>
<td>0.33</td>
</tr>
<tr>
<td>is</td>
<td>0.67</td>
</tr>
<tr>
<td>small</td>
<td>1.00</td>
</tr>
<tr>
<td>big</td>
<td>1.00</td>
</tr>
</tbody>
</table>

**Translation:**

<table>
<thead>
<tr>
<th>f_word</th>
<th>e_word</th>
<th>prob</th>
</tr>
</thead>
<tbody>
<tr>
<td>a</td>
<td>1.00</td>
<td></td>
</tr>
<tr>
<td>the</td>
<td>1.00</td>
<td></td>
</tr>
<tr>
<td>book</td>
<td>1.00</td>
<td></td>
</tr>
<tr>
<td>house</td>
<td>1.00</td>
<td></td>
</tr>
<tr>
<td>is</td>
<td>0.33</td>
<td></td>
</tr>
<tr>
<td>is</td>
<td>0.67</td>
<td></td>
</tr>
<tr>
<td>small</td>
<td>1.00</td>
<td></td>
</tr>
<tr>
<td>big</td>
<td>1.00</td>
<td></td>
</tr>
</tbody>
</table>

**Fertility:**

<table>
<thead>
<tr>
<th>e_word</th>
<th>fertility prob</th>
</tr>
</thead>
<tbody>
<tr>
<td>a</td>
<td></td>
</tr>
<tr>
<td>the</td>
<td></td>
</tr>
<tr>
<td>book</td>
<td></td>
</tr>
<tr>
<td>house</td>
<td></td>
</tr>
<tr>
<td>is</td>
<td></td>
</tr>
<tr>
<td>is</td>
<td></td>
</tr>
<tr>
<td>small</td>
<td></td>
</tr>
<tr>
<td>big</td>
<td></td>
</tr>
</tbody>
</table>

Some differences between the module and the book ...

tu.py

1<-0: 0.50 2,2
1<-1: 1.00 2,2
2<-0: 0.50 2,2
2<-2: 1.00 2,2

1<-0: 0.25 4,4
1<-4: 1.00 4,4
2<-0: 0.25 4,4
2<-3: 1.00 4,4
3<-0: 0.25 4,4
3<-3: 0.50 4,5
4<-0: 0.25 4,4
4<-2: 1.00 4,4

1<-0: 0.19 4,5
1<-1: 1.00 4,5
2<-0: 0.19 4,5
2<-2: 1.00 4,5
3<-0: 0.21 4,5
3<-3: 0.50 4,5
4<-0: 0.22 4,5
4<-3: 0.50 4,5
5<-0: 0.19 4,5
5<-4: 1.00 4,5

Distortion:
j <- i: prob |e|,|f|
Using IBM Model 1
Number of iterations: 16
Read training data from file: ibm1_datar.txt, pairs: 3, max word length: 5
das the : 1.00
das None : 0.50
Buch book : 1.00
Buch None : 0.50
Haus house: 1.00
ein a : 1.00

python3 tu.py 16
ibm1_datar.txt

Training data
ibm1_datar.txt:
the a book house
das ein Buch Haus
the house
Haus das
the book
Buch das
a book
Buch ein
tu.py

Training data ibm1_datar.txt:
the a book house
das ein Buch Haus
the house
Haus das
the book
Buch das
a book
Buch ein

python3 tu.py 16
ibm1_datar.txt -m 2
Using IBM Model 2
Number of iterations: 16
Read training data from file: ibm1_datar.txt, pairs:
3, max word length: 5
Buch book: 1.00
Buch None: 0.50
ein a: 1.00
das the: 1.00
das None: 0.50
Haus house: 1.00
1<-2: 1.00 2,2
2<-1: 1.00 2,2
Let's look at the source code...
The Bible

http://www.biblestudytools.com/genesis/1.html

• 1 In the beginning God created the heavens and the earth.
• 2 Now the earth was formless and empty, darkness was over the surface of the deep, and the Spirit of God was hovering over the waters.
• 3 And God said, “Let there be light,” and there was light.
• 4 God saw that the light was good, and he separated the light from the darkness.
• 5 God called the light “day,” and the darkness he called “night.” And there was evening, and there was morning—the first day.
• 6 And God said, “Let there be a vault between the waters to separate water from water.”

http://www.easyenglish.info/english-learners-bible/genesis-taw.htm

• 1 In the beginning, God began to make the earth and the heavens and everything in them.
• 2 The earth was without shape. Nothing was alive on it. Deep water covered the earth. There was no light. Everywhere was dark. The Spirit of God moved above the water.
• 3 God said ‘I want light to appear.’ And there was light.
• 4 God looked at the light. And he saw that it was good. He separated the light from the dark.
• 5 God called the light ‘day’. He called the dark ‘night’. Evening passed and morning came. This was the first (1st) day.
• 6 Then God said ‘A wide space will appear between the waters. So the waters will separate. Some of the water will stay above the space. The rest of the water will stay below the space.’
• **ROMEO**
  • A sick man in sadness makes his will,
  • A word ill urged to one that is so ill.
  • In sadness, cousin, I do love a woman.

• **BENVOLIO**
  • I aimed so near when I supposed you loved.

• **ROMEO**
  • A right good markman! And she’s fair I love.

• **BENVOLIO**
  • A right fair mark, fair coz, is soonest hit.

• **ROMEO**
  • Well, in that hit you miss. She’ll not be hit
  • With Cupid’s arrow. She hath Dian’s wit.

• **ROMEO**
  • You wouldn’t tell a sick man he “seriously” has to make his will—it would just make him worse. Seriously, cousin, I love a woman.

• **BENVOLIO**
  • I guessed that already when I guessed you were in love.

• **ROMEO**
  • Then you were right on target. The woman I love is beautiful.

• **BENVOLIO**
  • A beautiful target is the one that gets hit the fastest.

• **ROMEO**
  • Well, you’re not on target there. She refuses to be hit by Cupid’s arrow. She’s as clever as Diana

http://nfs.sparknotes.com/romeoandjuliet/page_24.html
Romeo and Juliet is a play written by William Shakespeare.

It is set in Italy and is about the love between two young people from noble families that are enemies.

Romeo and Juliet has always been one of Shakespeare's most popular plays. It has been adapted to opera, ballet, television productions, and movies.

Romeo and Juliet is a tragedy written by William Shakespeare early in his career about two young star-crossed lovers whose deaths ultimately reconcile their feuding families.

It was among Shakespeare's most popular plays during his lifetime and, along with Hamlet, is one of his most frequently performed plays.

Romeo and Juliet has been adapted numerous times for stage, film, musical and opera venues.