Algorithms for NLP

Machine Translation II

Taylor Berg-Kirkpatrick – CMU

Slides: Dan Klein – UC Berkeley
Announcements

- Project 4: Word Alignment!
- Will be released soon! (~Monday)
Phrase-Based System Overview

Morgen | fliege | ich | nach Kanada | zur Konferenz

Tomorrow | I will fly | to the conference | in Canada

Sentence-aligned corpus

Word alignments

Phrase table (translation model)

Many slides and examples from Philipp Koehn or John DeNero
Word Alignment
IBM Models 1/2

E: Thank you, I shall do so gladly.

A: 1 3 7 6 8 8 8 8 9

F: Gracias, lo haré de muy buen grado.

Model Parameters

*Emissions:* $P(F_1 = \text{Gracias} | E_{A_1} = \text{Thank})$

*Transitions:* $P(A_2 = 3)$
EM for Models 1/2

- **Model 1 Parameters:**
  - Translation probabilities (1+2)
  - Distortion parameters (2 only)
  
  \[
P(f_j|e_i) \quad P(a_j = i|j, I, J)\]

- Start with \(P(f_j|e_i)\) uniform, including \(P(f_j|\text{null})\)
- For each sentence:
  - For each French position \(j\)
    - Calculate posterior over English positions

  \[
P(a_j = i|f, e) = \frac{P(a_j = i|j, I, J)P(f_j|e_i)}{\sum_{i'} P(a_j = i'|j, I, J)P(f_j|e_{i'})}\]

  (or just use best single alignment)
  - Increment count of word \(f_j\) with word \(e_i\) by these amounts
  - Also re-estimate distortion probabilities for model 2

- Iterate until convergence
Japan shaken by two new quakes

Le Japon secoué par deux nouveaux séismes
Japan is at the junction of four tectonic plates

Le Japon est au confluent de quatre plaques tectoniques
On Tuesday Nov. 4, earthquakes rocked Japan once again.

Des tremblements de terre ont à nouveau touché le Japon jeudi 4 novembre.
The HMM Model

E: Thank you, I shall do so gladly.

F: Gracias, lo haré de muy buen grado.

Model Parameters

Emissions: \( P(F_1 = \text{Gracias} | E_{A_1} = \text{Thank}) \)

Transitions: \( P(A_2 = 3 | A_1 = 1) \)
The HMM Model

- Model 2 preferred global monotonicity
- We want local monotonicity:
  - Most jumps are small
- HMM model (Vogel 96)

\[ P(f, a|e) = \prod_j P(a_j|a_{j-1})P(f_j|e_i) \]

- Re-estimate using the forward-backward algorithm
- Handling nulls requires some care
AER for HMMs

<table>
<thead>
<tr>
<th>Model</th>
<th>AER</th>
</tr>
</thead>
<tbody>
<tr>
<td>Model 1 INT</td>
<td>19.5</td>
</tr>
<tr>
<td>HMM E→F</td>
<td>11.4</td>
</tr>
<tr>
<td>HMM F→E</td>
<td>10.8</td>
</tr>
<tr>
<td>HMM AND</td>
<td>7.1</td>
</tr>
<tr>
<td>HMM INT</td>
<td>4.7</td>
</tr>
<tr>
<td>GIZA M4 AND</td>
<td>6.9</td>
</tr>
</tbody>
</table>
Phrase-Based MT
Phrase-Based Translation Overview

Input: \( \text{lo haré rápidamente} \).

Translations: I’ll do it quickly.

The decoder...
tries different segmentations,
translates phrase by phrase,
and considers reorderings.

Objective: \[
\arg \max_e [P(f|e) \cdot P(e)]
\]

\[
\arg \max_e \left[ \prod_{\langle e,f \rangle} P(f|e) \cdot \prod_{i=1}^{|e|} P(e_i|e_{i-1}, e_{i-2}) \right]
\]
Phrase-Based Decoding

This 7 people include 7 people included by some the russian the the astronauts international astronomical of rapporteur.

These 7 among including from the french and the russian the fifth members.

That 7 persons including from the of france and to russian of the aerospace members.

7 include from the of france and russian astronauts. the

7 numbers include from france and russian of astronauts who.

7 populations include those from france and russian astronauts.

7 deportees included come from france and russia in astronomical personnel;

7 philtrum including those from france and russia a space member

including representatives from france and the russia astronaut

include came from france and russia by cosmonauts

include came from france and russia’s cosmonauts.

includes coming from france and russia’s cosmonaut

french and russian ’s astronaut navigation member.

french and russia’s astronauts

and russia’s special rapporteur

, and russia rapporteur

, and russia rapporteur.

, and russia or russia’s

Decoder design is important: [Koehn et al. 03]
The Pharaoh “Model”

[Koehn et al, 2003]

\[ P(e|g) = P(\{\bar{g}_i\}|g) \prod_i \phi(\bar{e}_i|\bar{g}_i) d(a_i - b_{i-1}) \]

Segmentation  Translation  Distortion
The Pharaoh “Model”

\[ P(f|e) = P(\{\bar{e}_i\}|e) \prod_i \phi(f_i|\bar{e}_i) d(a_i - b_{i-1}) \]

\[ \frac{1}{K} \quad \frac{\text{count}(f_i, \bar{e}_i)}{\text{count}(\bar{e}_i)} \quad \alpha |a_i - b_{i-1}| \]

*Where do we get these counts?*
Phrase Weights

How the MT community estimates $P(\bar{f} | \bar{e})$

Parallel training sentences provide phrase pair counts.

Gracias, lo haré de muy buen grado. Thank you, I shall do so gladly.

lo haré $\leftrightarrow$ I shall do so
44 times in the corpus

All phrase pairs are counted, and counts are normalized.

Gracias, lo haré de muy buen grado.
Thank you, I shall do so gladly.

$$P(\bar{f} | \bar{e}) = \frac{\text{count}(\bar{f}, \bar{e})}{\text{count}(\bar{e})}$$
Phrase-Based Decoding

<table>
<thead>
<tr>
<th>Maria</th>
<th>no</th>
<th>dio</th>
<th>una</th>
<th>bofetada</th>
<th>a</th>
<th>la</th>
<th>bruja</th>
<th>verde</th>
</tr>
</thead>
<tbody>
<tr>
<td>Mary</td>
<td>not</td>
<td>give</td>
<td>a</td>
<td>slap</td>
<td>to</td>
<td>the</td>
<td>witch</td>
<td>green</td>
</tr>
<tr>
<td>did not</td>
<td>a slap</td>
<td>by</td>
<td>green witch</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>no</td>
<td>slap</td>
<td>to the</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>did not give</td>
<td></td>
<td>to</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>the</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>slap</td>
<td>the witch</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
Monotonic Word Translation

- Cost is \( LM \times TM \)
- It’s an HMM?
  - \( P(e|e_{-1}, e_{-2}) \)
  - \( P(f|e) \)
- State includes
  - Exposed English
  - Position in foreign
- Dynamic program loop?

```plaintext
for (fPosition in 1…|f|)
  for (eContext in allEContexts)
    for (eOption in translations[fPosition])
      score = scores[fPosition-1][eContext] * LM(eContext+eOption) * TM(eOption, fWord[fPosition])
      scores[fPosition][eContext][2+eOption] = max score
```
Beam Decoding

- For real MT models, this kind of dynamic program is a disaster (why?)
- Standard solution is beam search: for each position, keep track of only the best k hypotheses

\[
\text{for } (f\text{Position} \text{ in } 1\ldots|f|) \\
\quad \text{for } (e\text{Context} \text{ in } \text{bestEContexts}[f\text{Position}]) \\
\quad \quad \text{for } (e\text{Option} \text{ in } \text{translations}[f\text{Position}]) \\
\quad \quad \quad \text{score} = \text{scores}[f\text{Position}-1][e\text{Context}] \times \text{LM(eContext+eOption)} \times \text{TM(eOption, fWord}[f\text{Position}]) \\
\quad \quad \quad \text{bestEContexts.maybeAdd(eContext}[2]+e\text{Option}, \text{score})
\]

- Still pretty slow... why?
- Useful trick: cube pruning (Chiang 2005)

Example from David Chiang
If monotonic, almost an HMM; technically a semi-HMM

for (fPosition in 1…|f|)
  for (lastPosition < fPosition)
    for (eContext in eContexts)
      for (eOption in translations[fPosition])
        … combine hypothesis for (lastPosition ending in eContext) with eOption

If distortion... now what?
Non-Monotonic Phrasal MT

e: Mary
f: *--------
p: .534

e: Mary did not
f: **-------
p: .122

e: Mary slap
f: *---**----
p: .043

e: witch
f: ----***--
p: .182
**Problem:** easy partial analyses are cheaper

- **Solution 1:** use beams per foreign subset
- **Solution 2:** estimate forward costs (A*-like)
Maria no dio una bofetada a la bruja verde

Mary did not give a slap to the witch green

Mary did not give a slap by the green witch

Mary did not give a slap to the witch
<table>
<thead>
<tr>
<th>Maria</th>
<th>no</th>
<th>dio</th>
<th>una</th>
<th>bofetada</th>
<th>a</th>
<th>la</th>
<th>bruja</th>
<th>verde</th>
</tr>
</thead>
<tbody>
<tr>
<td>Mary</td>
<td>not give</td>
<td>a slap</td>
<td>to the witch</td>
<td>green</td>
<td>green witch</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>did not</td>
<td>a slap</td>
<td>by</td>
<td>to the</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>no</td>
<td>slap</td>
<td>to</td>
<td>the</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>did not give</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>slap</td>
<td></td>
<td>the witch</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

**Diagram:**
- **Joe**: Probability $p=1$
- **Mary**: Probability $p=0.534$
- **p=0.92**: Probability $p=0.092$
- **p=0.92**: Probability $p=0.92$
- **p=0.164**: Probability $p=0.164$
Parameter Tuning
**Counting Phrase Pairs**

**Input:**

Gracias, lo haré de muy buen grado.
Thank you, I shall do so gladly.

**First, we learn word alignments,**

**then we infer aligned phrases.**

**Gracias**

Thanks

,**

that

do [first; future]

of

very

good

degree

,**

lo

haré

de

muy

buen

grado

,**

Thank you, I shall do so gladly.
What Happens in Practice

A real word alignment
(GIZA++ Model 4 with
grow-diag-final combination)

Gracias
,  
lo  
haré  
de  
muy  
buen  
grado  
.

Thank you, I shall do so gladly.

Gloss
Thanks
,  
that
do [first; future]
of
very
good
degree
What Happens in Practice

A real word alignment
(GIZA++ Model 4 with
grow-diag-final combination)

Gracias
,

lo
haré
de
muy
buen
grado
.

Thank you, I shall do so gladly.

Gloss

Thanks
,
that
do [first; future]
of
very
good
degree
.
What Happens in Practice

A real word alignment
(GIZA++ Model 4 with grow-diag-final combination)

Gracias
, that
lo
do [first; future]
haré
of
muy
very
buen
good
grado
degree

Thank you , I shall do so gladly .

Gloss

Thanks
Learning weights has been tried, several times:
- [Marcu and Wong, 02]
- [DeNero et al, 06]
- ... and others

Seems not to work well, for a variety of partially understood reasons

Main issue: big chunks get all the weight, obvious priors don’t help
- Though, [DeNero et al 08]
Phrases do help
- But they don’t need to be long
- Why should this be?
Lexical Weighting

\[ \phi(f_i|e_i) = \frac{\text{count}(f_i, e_i)}{\text{count}(e_i)} p_w(f_i|e_i) \]

\[
\begin{align*}
  f_1 & \quad f_2 & \quad f_3 \\
  \text{NULL} & \quad -- & \quad -- & \quad ## \\
  e_1 & \quad ## & \quad -- & \quad -- \\
  e_2 & \quad -- & \quad ## & \quad -- \\
  e_3 & \quad -- & \quad ## & \quad -- \\
\end{align*}
\]

\[ p_w(f|e, a) = p_w(f_1 f_2 f_3 | e_1 e_2 e_3, a) = w(f_1 | e_1) \times \frac{1}{2} (w(f_2 | e_2) + w(f_2 | e_3)) \times w(f_3 | \text{NULL}) \]

---

--- lex

--- no-lex
Tuning for MT

- Features encapsulate lots of information
  - Basic MT systems have around 6 features
    - $P(e|f)$, $P(f|e)$, lexical weighting, language model

- How to tune feature weights?

- Idea 1: Use your favorite classifier
Why Tuning is Hard

- Problem 1: There are latent variables
  - Alignments and segmentations
  - Possibility: forced decoding (but it can go badly)
Why Tuning is Hard

- **Problem 3: Computational constraints**
  - Discriminative training involves repeated decoding
  - Very slow! So people tune on sets much smaller than those used to build phrase tables
Minimum Error Rate Training

- Standard method: minimize BLEU directly (Och 03)
  - MERT is a discontinuous objective
  - Only works for max ~10 features, but works very well then
  - Here: k-best lists, but forest methods exist (Machery et al 08)
  - Recently, lots of alternatives being explored for more features
MERT
Phrase Movement
The HMM Model

\[ p(e) \]

\[ p(a_j | a_{j-1}; \theta_d) \]

\[ p(f_j | e_{a_j}; \theta_t) \]

**Distortion** \( \theta_d \)

- \( p(\uparrow \uparrow) = 0.6 \)
- \( p(\uparrow \downarrow) = 0.2 \)
- \( p(\downarrow \downarrow) = 0.1 \)
- \( \ldots \)

**Translation** \( \theta_t \)

- \( p(\text{the} \rightarrow \text{le}) = 0.53 \)
- \( p(\text{the} \rightarrow \text{la}) = 0.24 \)
- \( p(\text{railroad} \rightarrow \text{ferroviaire}) = 0.19 \)
- \( p(\text{NULL} \rightarrow \text{le}) = 0.12 \)
- \( \ldots \)