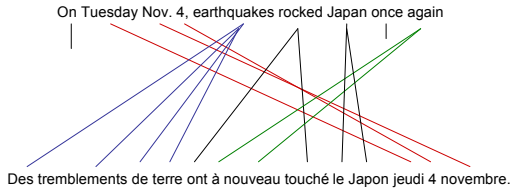
Statistical NLP
Spring 2008

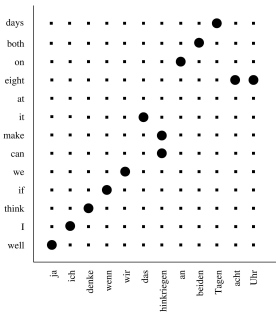
Lecture 12: Phrase Alignment

Dan Klein – UC Berkeley

Phrase Movement



Phrase Movement



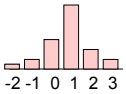
The HMM Model

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

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

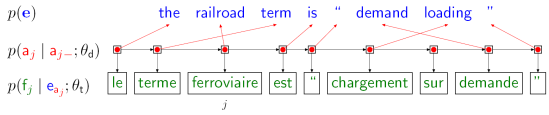
$$P(a_j - a_{j-1})$$

f	$t(f \mid e)$
nationale	0.469
national	0.418
nationaux	0.054
nationales	0.029



  - Re-estimate using the forward-backward algorithm
  - Handling nulls requires some care
- What are we still missing?

The HMM Model



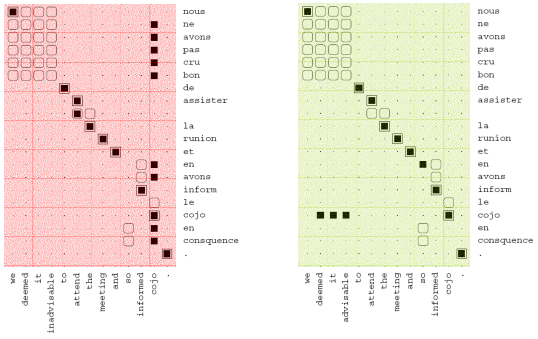
Distortion θ_d

$p(\quad) = 0.6$
$p(\quad) = 0.2$
$p(\quad) = \mathbf{0.1}$
...

Translation θ_t

$p(\text{ the} \rightarrow \text{le}) = 0.53$
$p(\text{ the} \rightarrow \text{la}) = 0.24$
$p(\text{ railroad} \rightarrow \text{ferroviaire}) = \mathbf{0.19}$
$p(\text{ NULL} \rightarrow \text{le}) = 0.12$
...

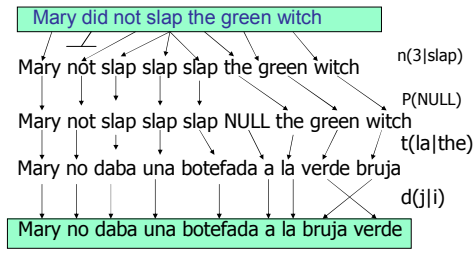
HMM Examples



AER for HMMs

Model	AER
Model 1 INT	19.5
HMM E→F	11.4
HMM F→E	10.8
HMM AND	7.1
HMM INT	4.7
GIZA M4 AND	6.9

IBM Models 3/4/5



[from Al-Onaizan and Knight, 1998]

Examples: Translation and Fertility

the

f	$t(f \mid e)$	ϕ	$n(\phi \mid e)$
le	0.497	1	0.746
la	0.207	0	0.254
les	0.155		
l′	0.086		
ce	0.018		
cette	0.011		

not

f	$t(f \mid e)$	ϕ	$n(\phi \mid e)$
ne	0.497	2	0.735
pas	0.442	0	0.154
non	0.029	1	0.107
rien	0.011		

farmers

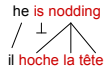
f	$t(f \mid e)$	ϕ	$n(\phi \mid e)$
agriculteurs	0.442	2	0.731
les	0.418	1	0.228
cultivateurs	0.046	0	0.039
producteurs	0.021		

Example: Morphology

should

f	$t(f \mid e)$	ϕ	$n(\phi \mid e)$
devrait	0.330	1	0.649
devraient	0.123	0	0.336
devrions	0.109	2	0.014
faudrait	0.073		
faut	0.058		
doit	0.058		
aurait	0.041		
doivent	0.024		
devons	0.017		
devrais	0.013		

Phrases in IBM Models



nodding

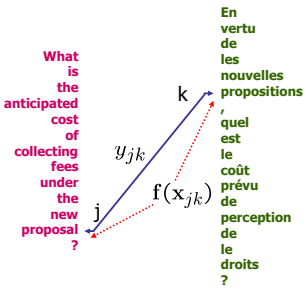
f	$t(f \mid e)$	ϕ	$n(\phi \mid e)$
signe	0.164	4	0.342
la	0.123	3	0.293
tête	0.097	2	0.167
oui	0.086	1	0.163
fait	0.073	0	0.023
que	0.073		
hoche	0.054		
hocher	0.048		
faire	0.030		
me	0.024		
approuve	0.019		
qui	0.019		
un	0.012		
faites	0.011		

Some Results

- [Och and Ney 03]

Model	Training scheme	0.5K	8K	128K	1.47M
Dice		50.9	43.4	39.6	38.9
Dice+C		46.3	37.6	35.0	34.0
Model 1	1^5	40.6	33.6	28.6	25.9
Model 2	$1^5 2^5$	46.7	29.3	22.0	19.5
HMM	$1^5 H^5$	26.3	23.3	15.0	10.8
Model 3	$1^5 2^5 3^3$	43.6	27.5	20.5	18.0
	$1^5 H^5 3^3$	27.5	22.5	16.6	13.2
Model 4	$1^5 2^5 3^3 4^3$	41.7	25.1	17.3	14.1
	$1^5 H^5 3^3 4^3$	26.1	20.2	13.1	9.4
	$1^5 H^5 4^3$	26.3	21.8	13.3	9.3
Model 5	$1^5 H^5 4^3 5^3$	26.5	21.5	13.7	9.6
	$1^5 H^5 3^3 4^3 5^3$	26.5	20.4	13.4	9.4
Model 6	$1^5 H^5 4^3 6^3$	26.0	21.6	12.8	8.8
	$1^5 H^5 3^3 4^3 6^3$	25.9	20.3	12.5	8.7

Feature-Based Alignment



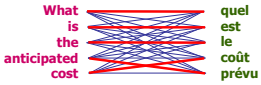
$\mathbf{f}(\mathbf{x}_{jk})$

Features:

- Association
- Lexical pair
- Position
- Orthography
- Neighborhood
- Resources
- IBM Models

$score(\mathbf{x}_{jk}) = \mathbf{w}^\top \mathbf{f}(\mathbf{x}_{jk})$

Finding Viterbi Alignments



$score(\mathbf{x}, \mathbf{y}) = \sum_{jk \in \mathbf{y}} score(\mathbf{x}_{jk})$

- Complete bipartite graph
- Maximum score matching with node degree ≤ 1

$$\mathbf{y} = \arg\max_{\mathbf{y}' \in \mathcal{Y}} score(\mathbf{x}, \mathbf{y}') = \arg\max_{\mathbf{y}' \in \mathcal{Y}} \mathbf{w}^\top \mathbf{f}(\mathbf{x}, \mathbf{y}')$$

⇒ Weighted bipartite matching problem

[Lacoste-Julien, Taskar, Jordan, and Klein, 05]

Learning **w**

- Supervised training data

$(\mathbf{x}^i, \mathbf{y}^i)$

- Training methods
  - Maximum likelihood/entropy
  - Perceptron
  - Maximum margin

[Lacoste-Julien, Taskar, Jordan, and Klein, 05]

Decoding

- In these word-to-word models
  - Finding best alignments is easy
  - Finding translations is hard (why?)

```
it is not clear .
|  \   |    \    \
|   \  +     \    \
|    \/ \     \    \
|    /\  \     \    \
CE NE EST PAS CLAIR  .
```

Bag "Generation" (Decoding)

soon me your as give possible please as response

the some let me disadvantages mention now of

missions research our has in two organization

Exact reconstruction (24 of 38)

Please give me your response as soon as possible.
⇒ Please give me your response as soon as possible.

Reconstruction preserving meaning (8 of 38)

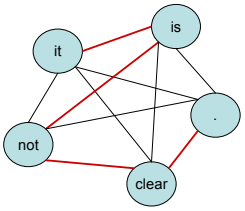
Now let me mention some of the disadvantages.
⇒ Let me mention some of the disadvantages now.

Garbage reconstruction (6 of 38)

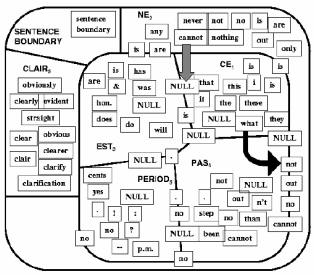
In our organization research has two missions.
⇒ In our missions research organization has two.

Bag Generation as a TSP

- Imagine bag generation with a bigram LM
  - Words are nodes
  - Edge weights are P(w|w')
  - Valid sentences are Hamiltonian paths
- Not the best news for word-based MT!



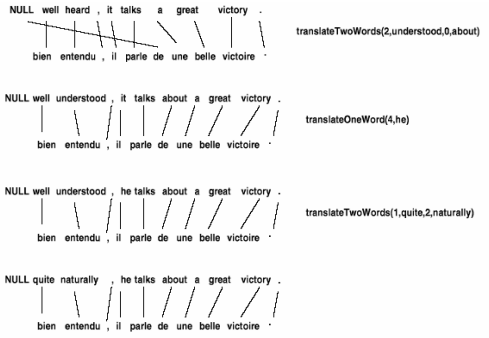
IBM Decoding as a TSP



Decoding, Anyway

- Simplest possible decoder:
  - Enumerate sentences, score each with TM and LM
- Greedy decoding:
  - Assign each French word it's most likely English translation
  - Operators:
    - Change a translation
    - Insert a word into the English (zero-fertile French)
    - Remove a word from the English (null-generated French)
    - Swap two adjacent English words
  - Do hill-climbing (or annealing)

Greedy Decoding

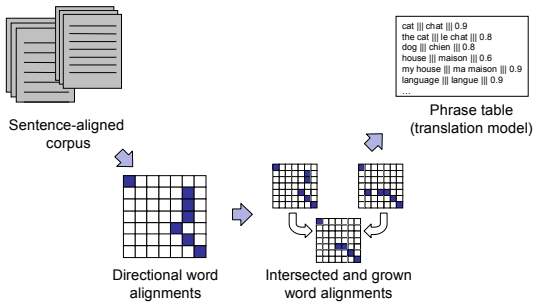


Stack Decoding

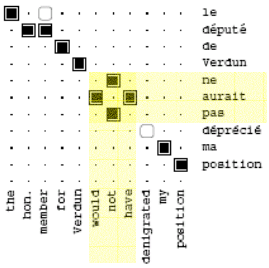
- Stack decoding:
  - Beam search
  - Usually A* estimates for completion cost
  - One stack per candidate sentence length
- Other methods:
  - Dynamic programming decoders possible if we make assumptions about the set of allowable permutations

sent length	decoder type	time (sec/sent)	search errors	translation errors (semantic and/or syntactic)	NE	PME	DSE	FSE	HSE	CE
6	IP	47.50	0	57	44	57	0	0	0	0
6	stack	0.79	5	58	43	53	1	0	0	4
6	greedy	0.07	18	60	38	45	5	2	1	10
8	IP	499.00	0	76	27	74	0	0	0	0
8	stack	5.67	20	75	24	57	1	2	2	15
8	greedy	2.66	43	75	20	38	4	5	1	33

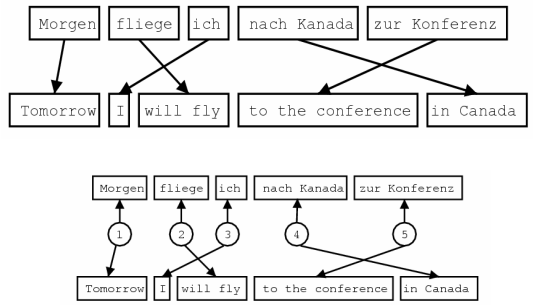
Overview: Extracting Phrases



Complex Configurations

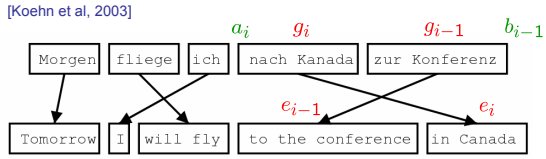


Phrase-Based Systems



Pharaoh's 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

Pharaoh's Model

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

$$\frac{1}{K} \qquad \frac{count(\bar{f}_i, \bar{e}_i)}{count(\bar{e}_i)} \qquad \alpha^{|a_i - b_{i-1}|}$$

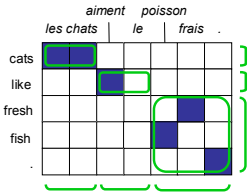
Where do we get these counts?

Phrase-Based Decoding

Decoder design is important: [Koehn et al. 03]

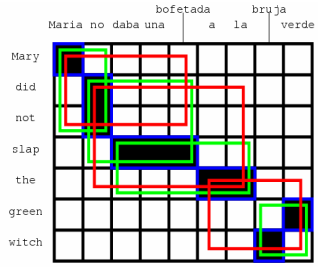
Phrase Scoring

$$\phi_{Marcu}(\bar{e}_i|\bar{f}_i) = \frac{c(\bar{f}_i, \bar{e}_i)}{c(\bar{f}_i)}$$



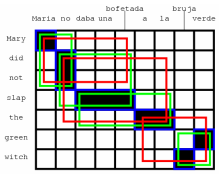
- Learning weights has been tried, several times:
  - [Marcu and Wong, 02]
  - [DeNero et al, 06]
  - … and others
- Seems not to work, for a variety of only partially understood reasons
- Main issue: big chunks get all the weight, obvious priors don't help

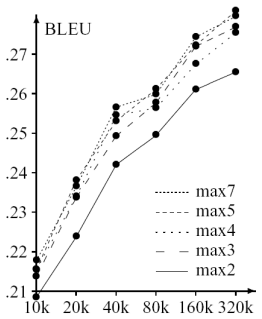
Extracting Phrases



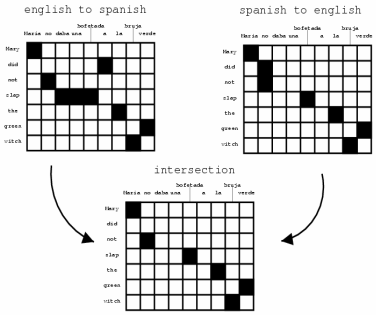
Phrase Size

- Phrases do help
  - But they don't need to be long
  - Why should this be?

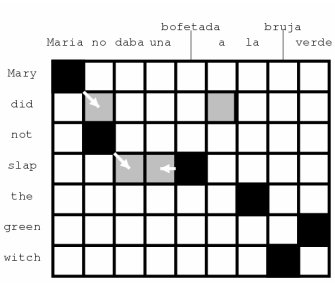


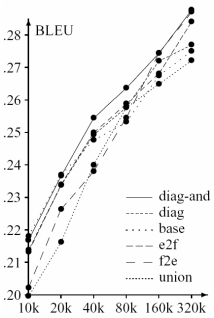


Bidirectional Alignment



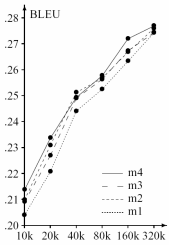
Alignment Heuristics

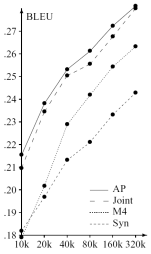




Sources of Alignments

Method	Training corpus size					
	10k	20k	40k	80k	160k	320k
AP	84k	176k	370k	736k	1536k	3152k
Joint	125k	220k	400k	707k	1254k	2214k
Syn	19k	24k	67k	105k	217k	373k



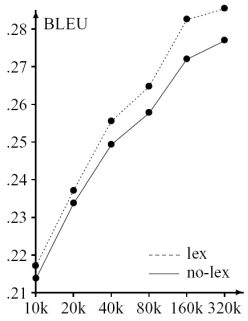


Lexical Weighting

$$\phi(\bar{f}_i|\bar{e}_i) = \frac{count(\bar{f}_i, \bar{e}_i)}{count(\bar{e}_i)} p_w(\bar{f}_i|\bar{e}_i)$$

```
        f1 f2 f3
NULL    -- -- ##
e1      ## -- --
e2      -- ## --
e3      -- ## --
```

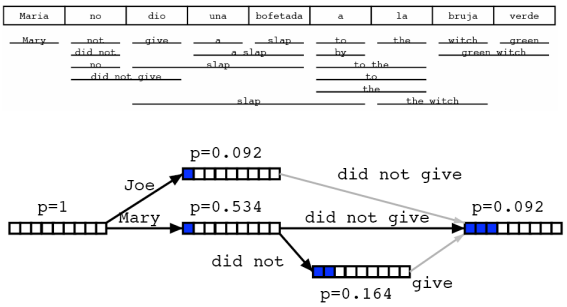
$$\begin{aligned} p_w(\bar{f}|\bar{e}, a) &= p_w(f_1 f_2 f_3 | e_1 e_2 e_3, a) \\ &= w(f_1|e_1) \\ &\quad \times \frac{1}{2}(w(f_2|e_2) + w(f_2|e_3)) \\ &\quad \times w(f_3|\text{NULL}) \end{aligned}$$



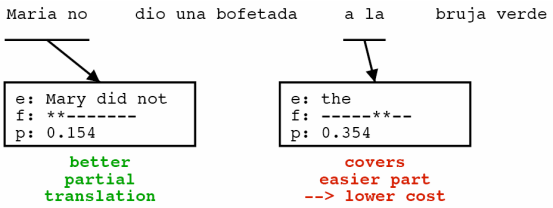
The Pharaoh Decoder

- Probabilities at each step include LM and TM

Hypotheis Lattices



Pruning



- Problem: easy partial analyses are cheaper
  - Solution 1: use beams per foreign subset
  - Solution 2: estimate forward costs (A*-like)

WSD?

- Remember when we discussed WSD?
  - Word-based MT systems rarely have a WSD step
  - Why not?

What's Next?

- Modeling syntax
  - PCFGs and phrase structure
  - Syntactic parsing
  - Grammar induction
  - Syntactic language and translation models
- Speech systems
  - Acoustics
  - Applications