Algorithms for NLP

Machine Translation II

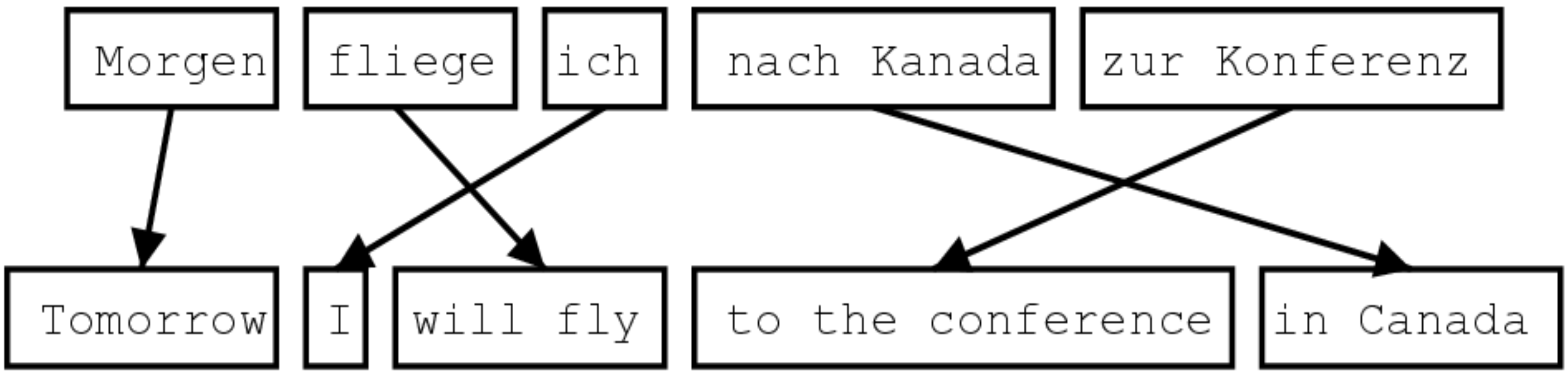
Taylor Berg-Kirkpatrick – CMU

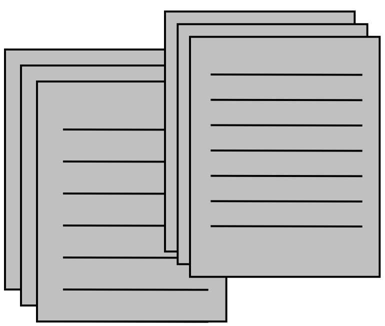
Slides: Dan Klein – UC Berkeley

Announcements

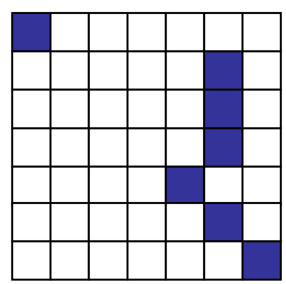
- Project 4: Word Alignment!
- Will be released soon! (~Monday)

Phrase-Based System Overview





Sentence-aligned corpus



Word alignments

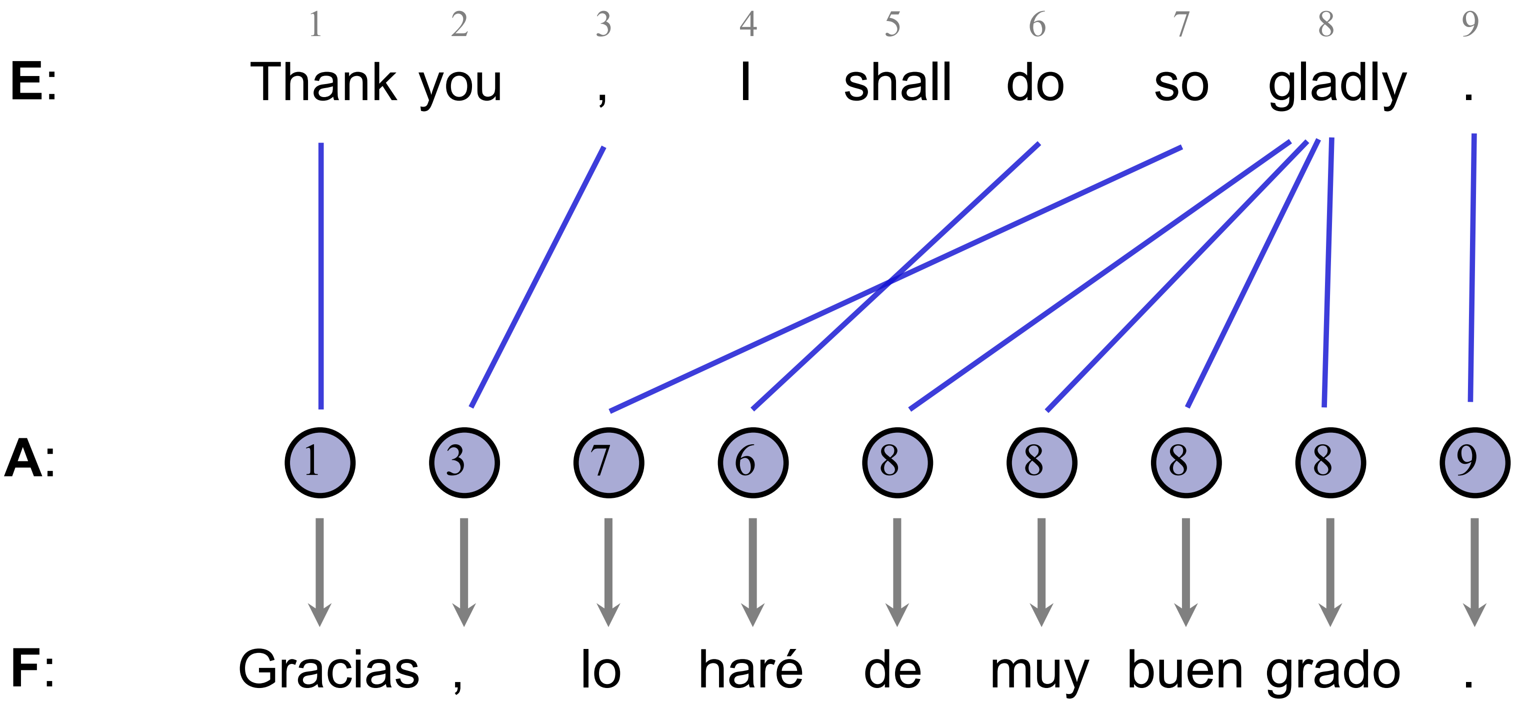
cat ||| chat ||| 0.9
the cat ||| le chat ||| 0.8
dog ||| chien ||| 0.8
house ||| maison ||| 0.6
my house ||| ma maison ||| 0.9
language ||| langue ||| 0.9
…

Phrase table (translation model)

Many slides and examples from Philipp Koehn or John DeNero

Word Alignment

IBM Models 1/2



Model Parameters

Emissions: $P(F_1 = \text{Gracias} \mid E_{A_1} = \text{Thank})$ *Transitions*: $P(A_2 = 3)$

EM for Models 1/2

- Model 1 Parameters:
  Translation probabilities (1+2) $P(f_j|e_i)$
  Distortion parameters (2 only) $P(a_j = i|j, I, J)$
- Start with $P(f_j|e_i)$ uniform, including $P(f_j|null)$
- 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

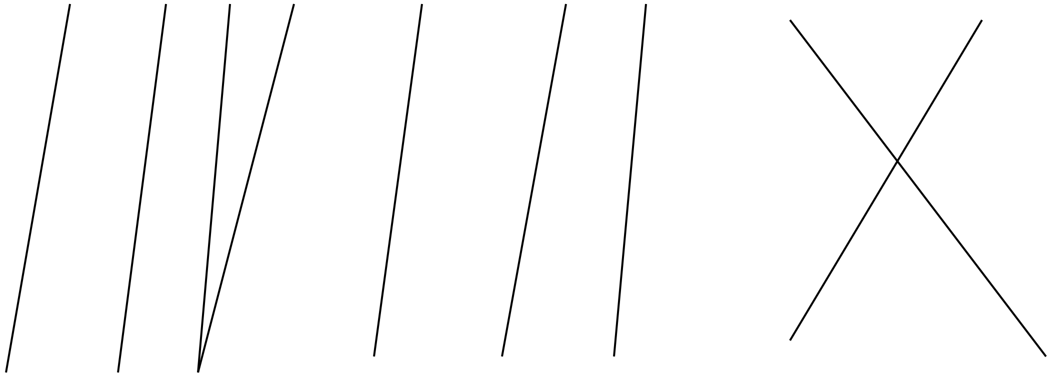
Monotonic Translation

Japan shaken by two new quakes

Le Japon secoué par deux nouveaux séismes

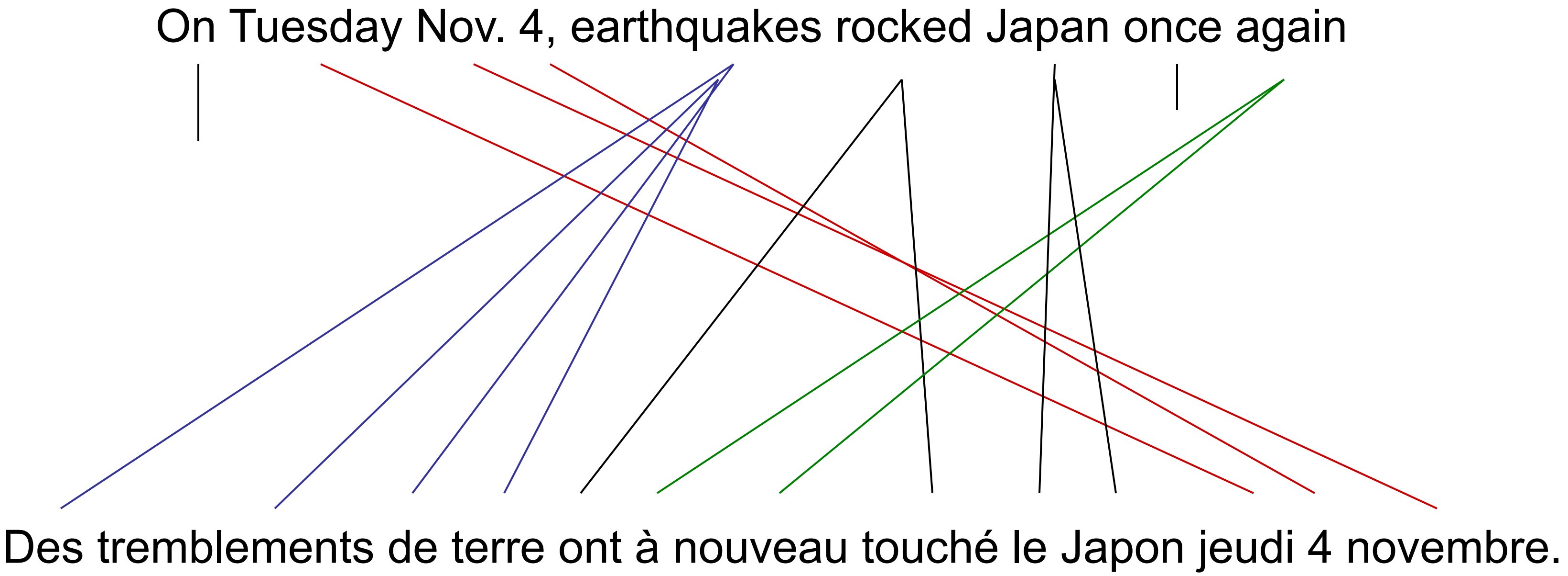
Local Order Change

Japan is at the junction of four tectonic plates

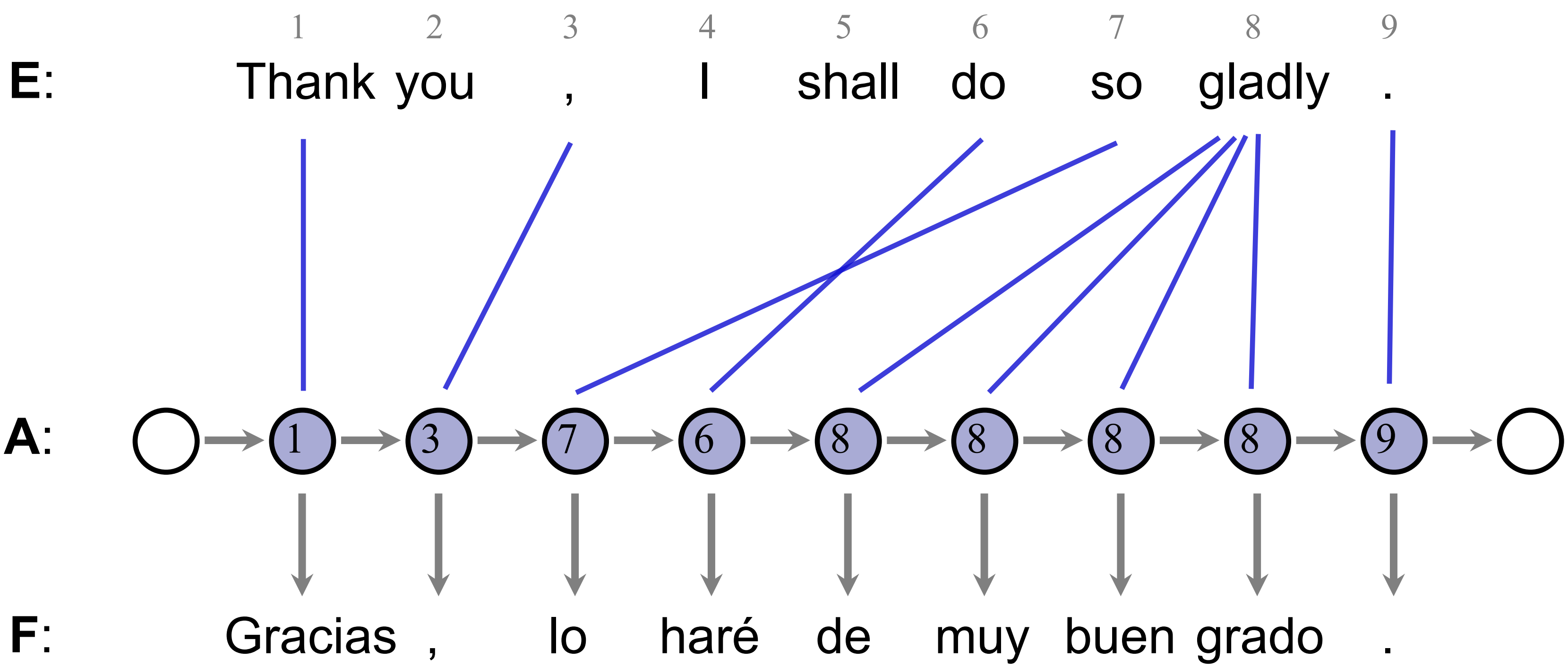


Le Japon est au confluent de quatre plaques tectoniques

Phrase Movement



The HMM Model



Model Parameters

Emissions: P(F_1 = Gracias | E_{A_1} = Thank)   *Transitions*: P(A_2 = 3 | A_1 = 1)

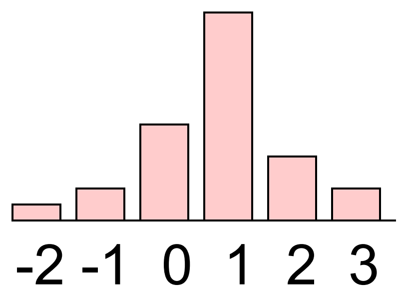
The HMM Model

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

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

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

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



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

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

Phrase-Based MT

Phrase-Based Translation Overview

The decoder...

tries different segmentations,

translates phrase by phrase,

and considers reorderings.

Input:	lo haré	rápidamente	.
Translations:	I'll do it	quickly	.
	quickly	I'll do it	.

Objective:

$$\arg\max_{\mathbf{e}} \left[P(\mathbf{f}|\mathbf{e}) \cdot P(\mathbf{e}) \right]$$

$$\arg\max_{\mathbf{e}} \left[\prod_{\langle \bar{e}, \bar{f} \rangle} P(\bar{f}|\bar{e}) \cdot \prod_{i=1}^{|\mathbf{e}|} P(e_i|e_{i-1}, e_{i-2}) \right]$$

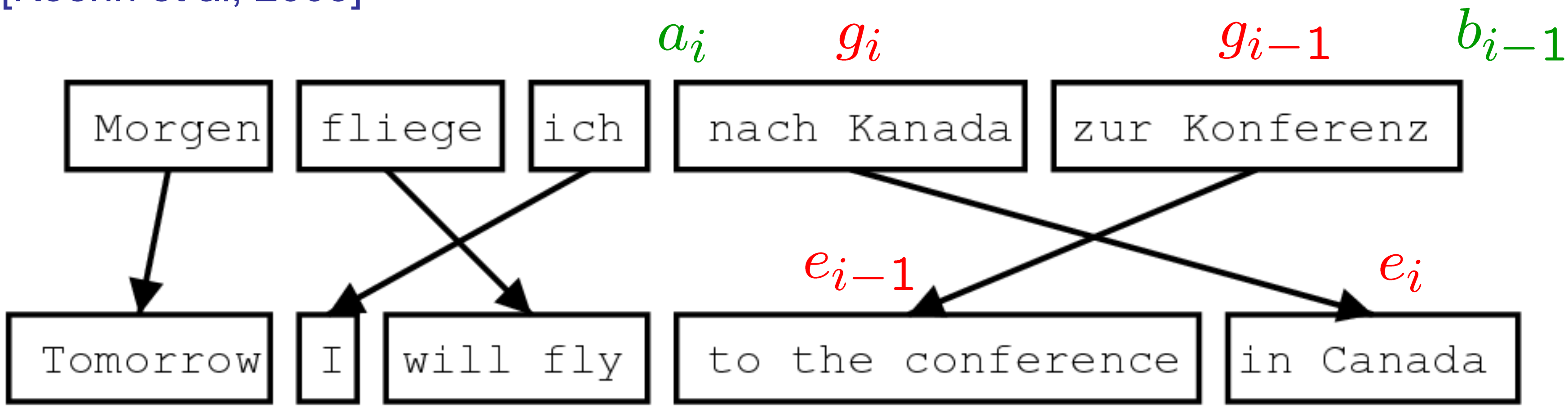
Phrase-Based Decoding

这	7人	中包括	来自	法国	和	俄罗斯	的	宇航	员	.
the	7 people	including	by some		**and**	the russian	**the**	the astronauts		,
it	7 people included		by france		and the	the russian		international astronautical	of rapporteur .	
this	7 out	including the	**from**	the french	and the russian		the fifth		.	
these	7 among	including from		the french and		of the russian	of	space	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	astronautical	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 representatives from		french	**and russia**			cosmonauts		
		include	came from france		and russia 's			cosmonauts .		
		includes	coming from	french and		russia 's		cosmonaut		
				french and russian			's	astronavigation	member .	
				french	**and russia**		**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"

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

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

Where do we get these counts?

Phrase 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 $\Longleftrightarrow$ 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 .

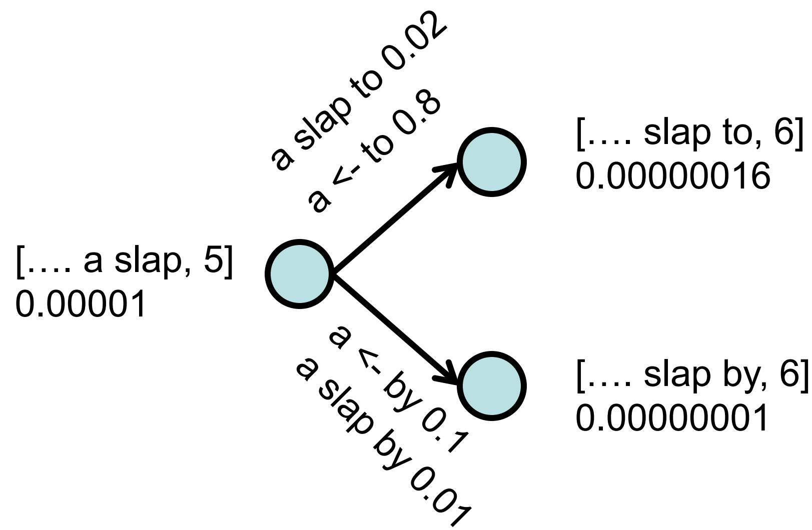
$$P(\bar{f}|\bar{e}) = \frac{\text{count}(\bar{f},\bar{e})}{\text{count}(\bar{e})}$$

Phrase-Based Decoding

Monotonic Word Translation

Maria	no	dio	una	bofetada	a	la	bruja	verde
Mary	not	give	a	slap	to	the	witch	green
	did not				by			
	no							

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



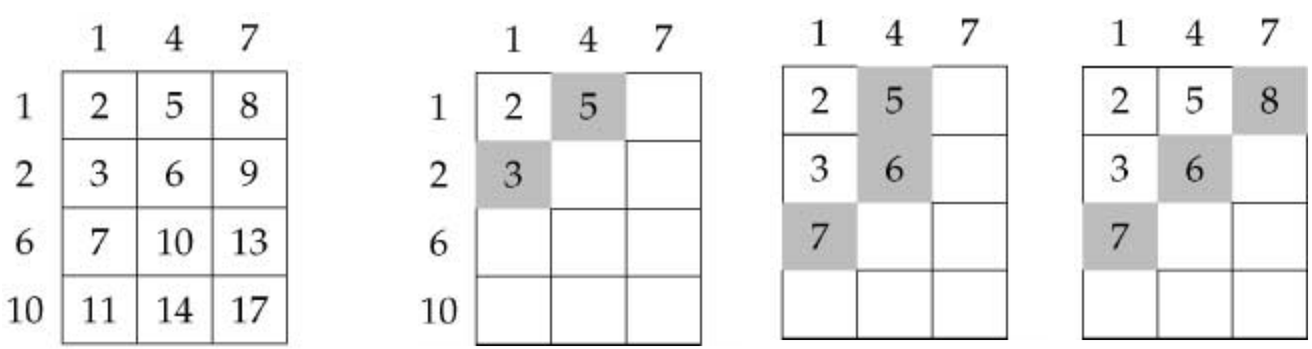

```
for (fPosition in 1…|f|)
  for (eContext in allEContexts)
    for (eOption in translations[fPosition])
      score = scores[fPosition-1][eContext] * LM(eContext+eOption) * TM(eOption, fWord[fPosition])
      scores[fPosition][eContext[2]+eOption] =max  score
```

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

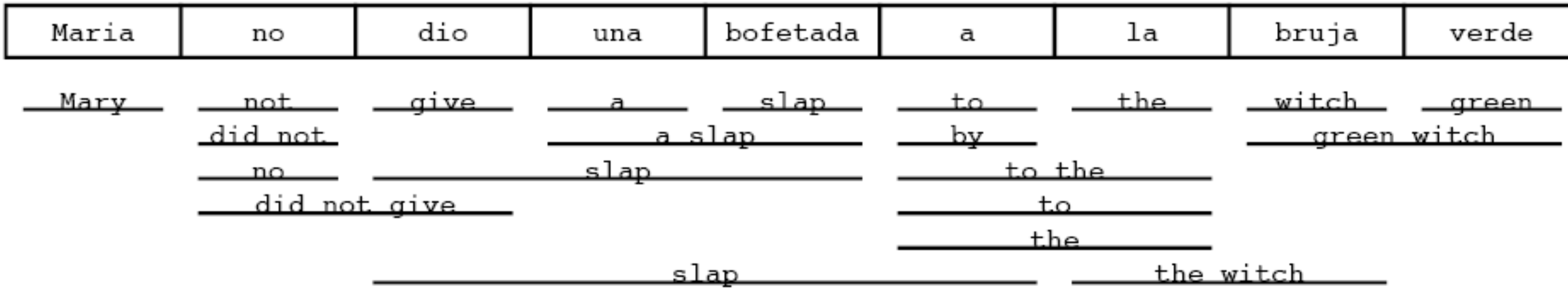
```
for (fPosition in 1…|f|)
  for (eContext in bestEContexts[fPosition])
    for (eOption in translations[fPosition])
      score = scores[fPosition-1][eContext] * LM(eContext+eOption) * TM(eOption, fWord[fPosition])
      bestEContexts.maybeAdd(eContext[2]+eOption, score)
```

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



Example from David Chiang

Phrase Translation

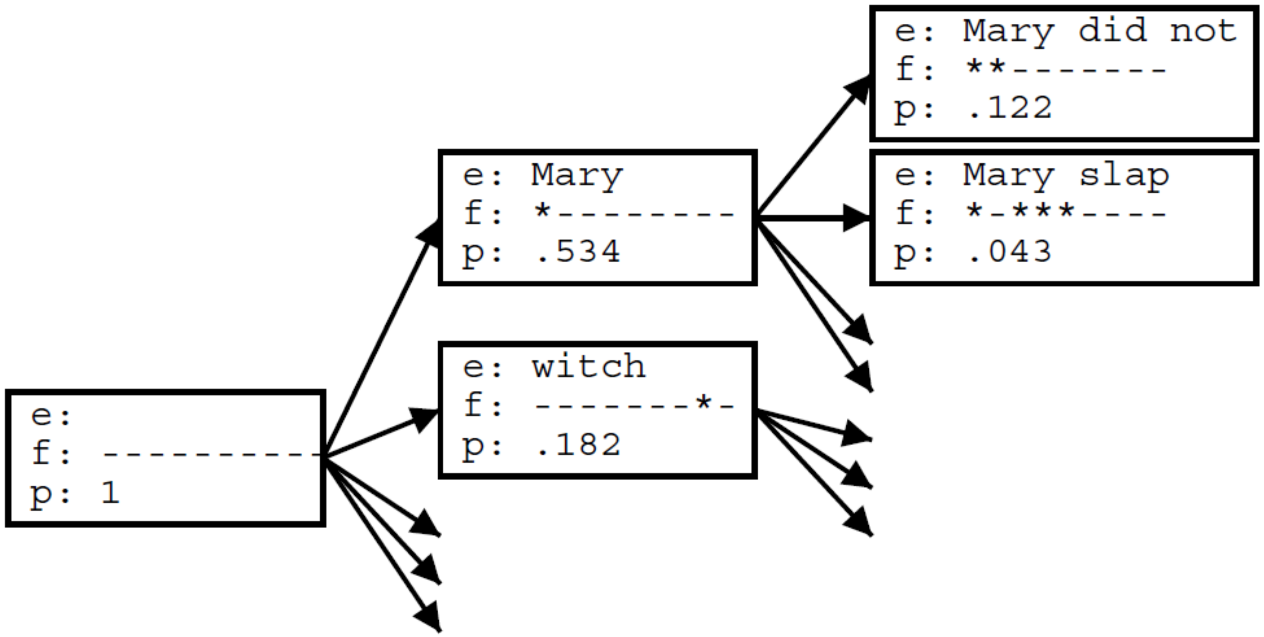


- If monotonic, almost an HMM; technically a semi-HMM

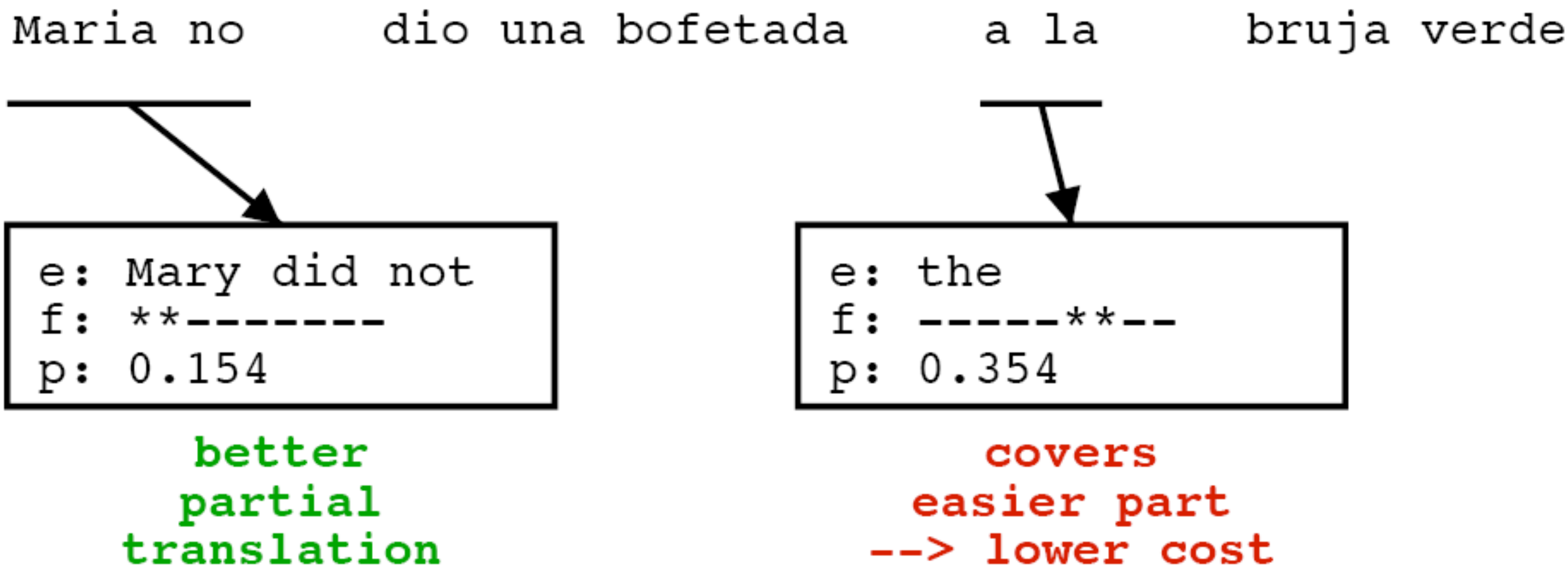
```
for (fPosition in 1…|f|)
  for (lastPosition < fPosition)
    for (eContext in eContexts)
      for (eOption in translations[fPosition])
        … combine hypothesis for (lastPosition ending in eContext) with eOption
```

- If distortion… now what?

Non-Monotonic Phrasal MT

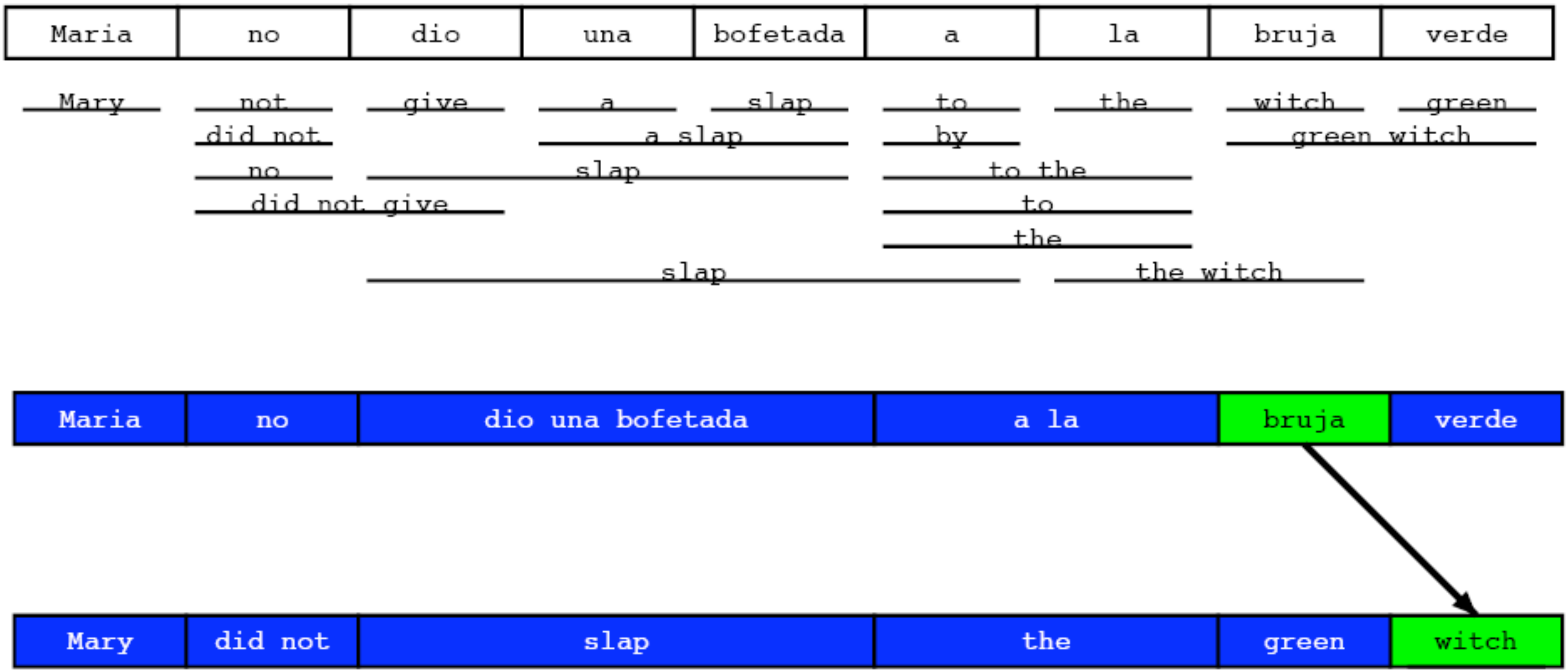


Pruning: Beams + Forward Costs

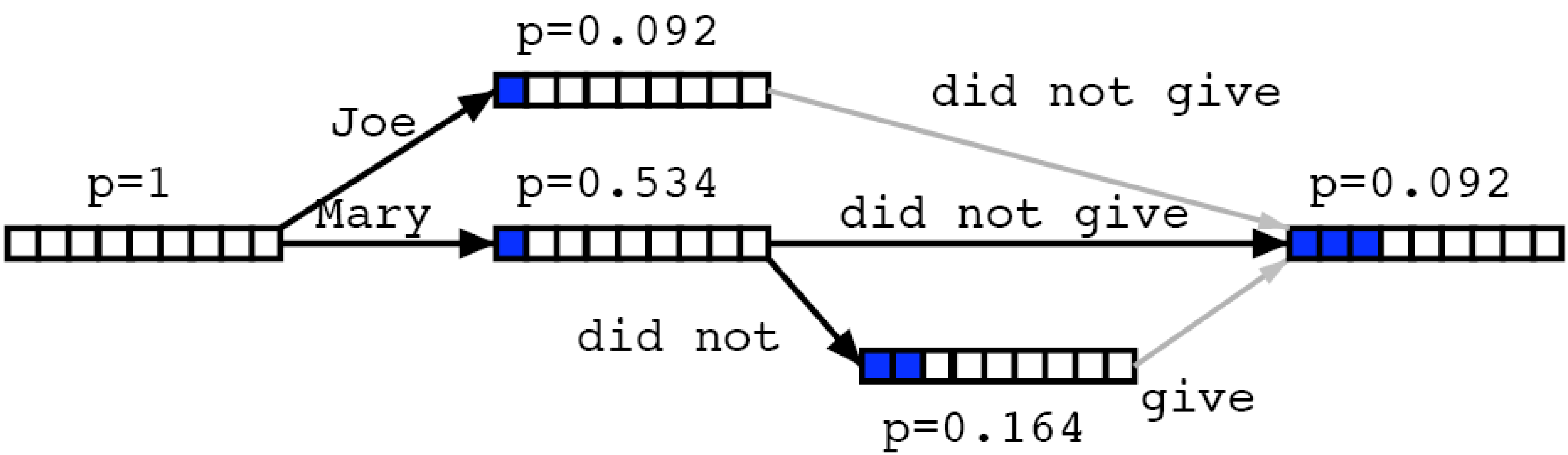


- Problem: easy partial analyses are cheaper
  - Solution 1: use beams per foreign subset
  - Solution 2: estimate forward costs (A*-like)

The Pharaoh Decoder



Hypotheis Lattices



Parameter Tuning

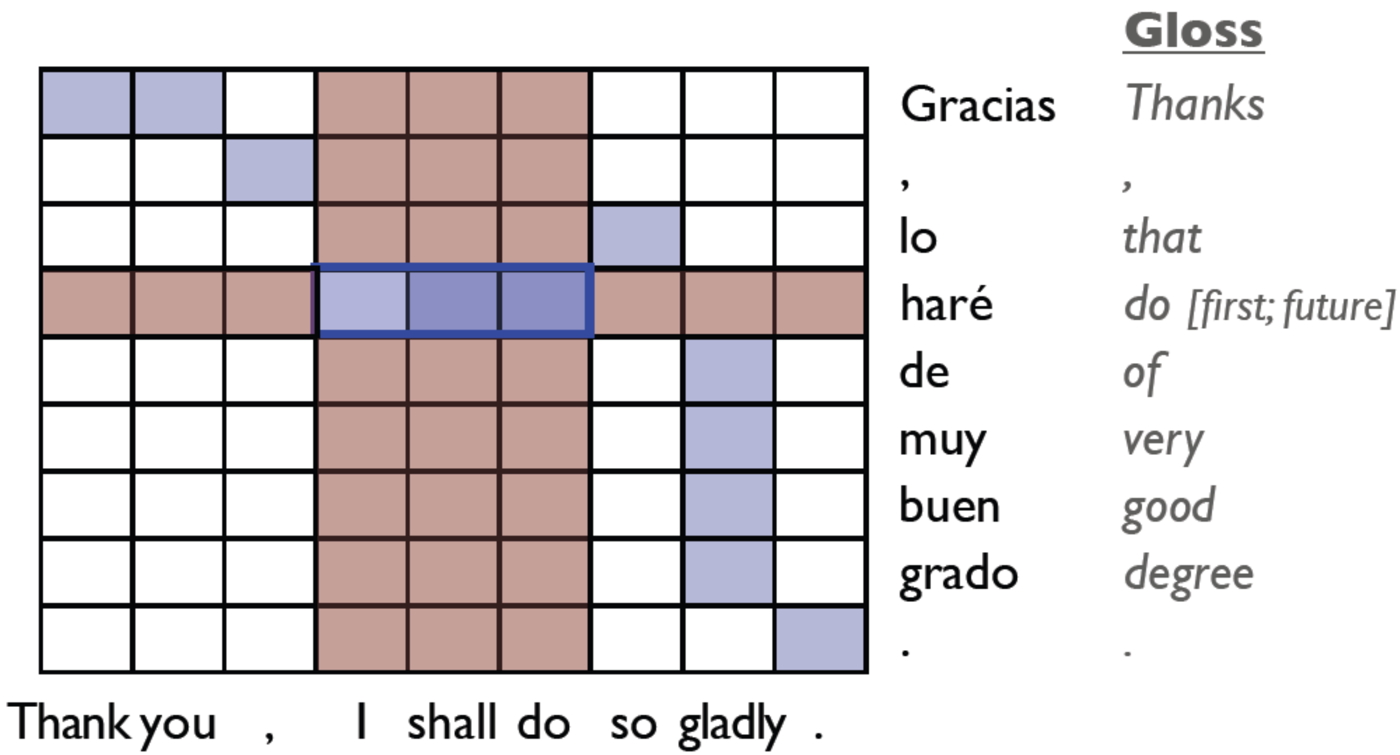
Counting Phrase Pairs

Input:

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

First, we learn word alignments,

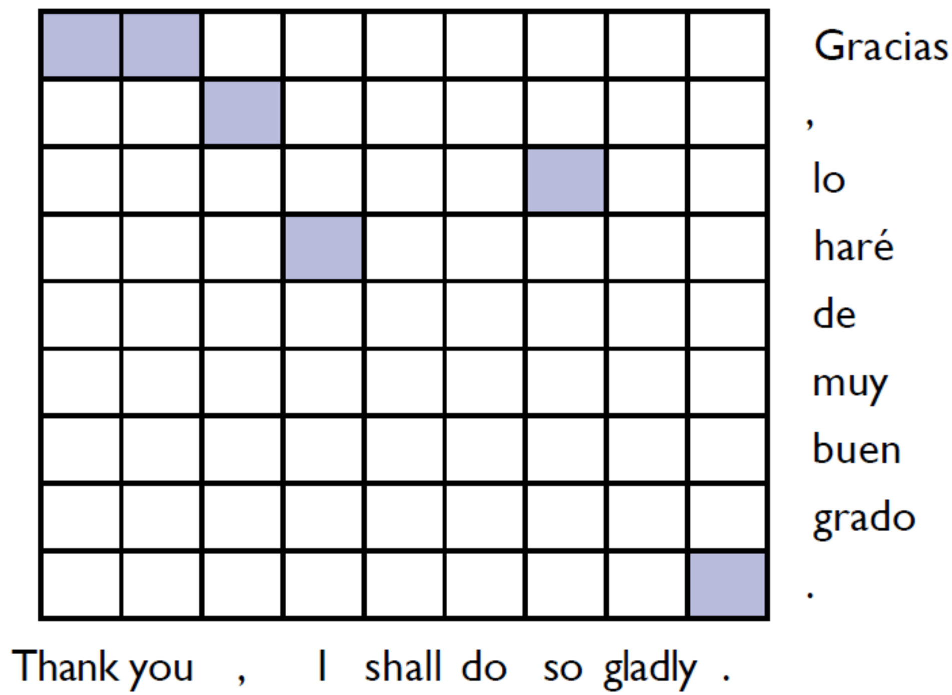
then we infer aligned phrases.



What Happens in Practice

A real word alignment

(GIZA++ Model 4 with
grow-diag-final combination)



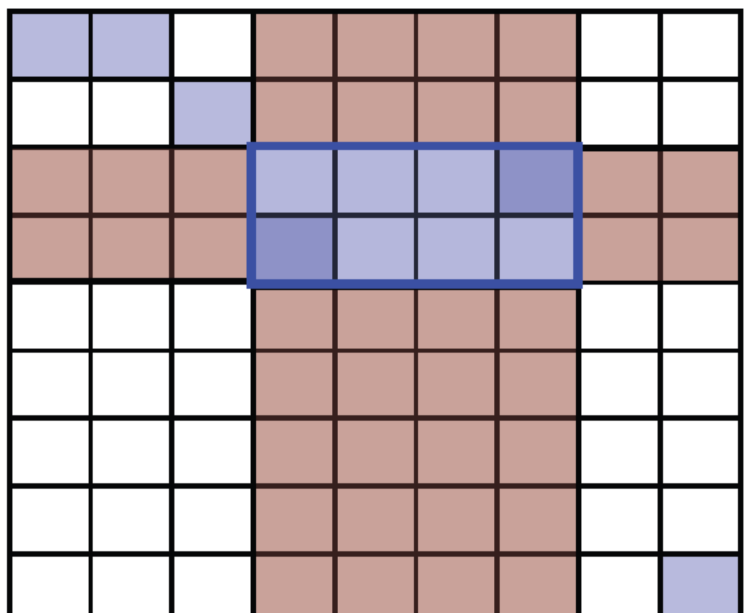
Gloss

Thanks
,
that
do [first; future]
of
very
good
degree
.

What Happens in Practice

A real word alignment

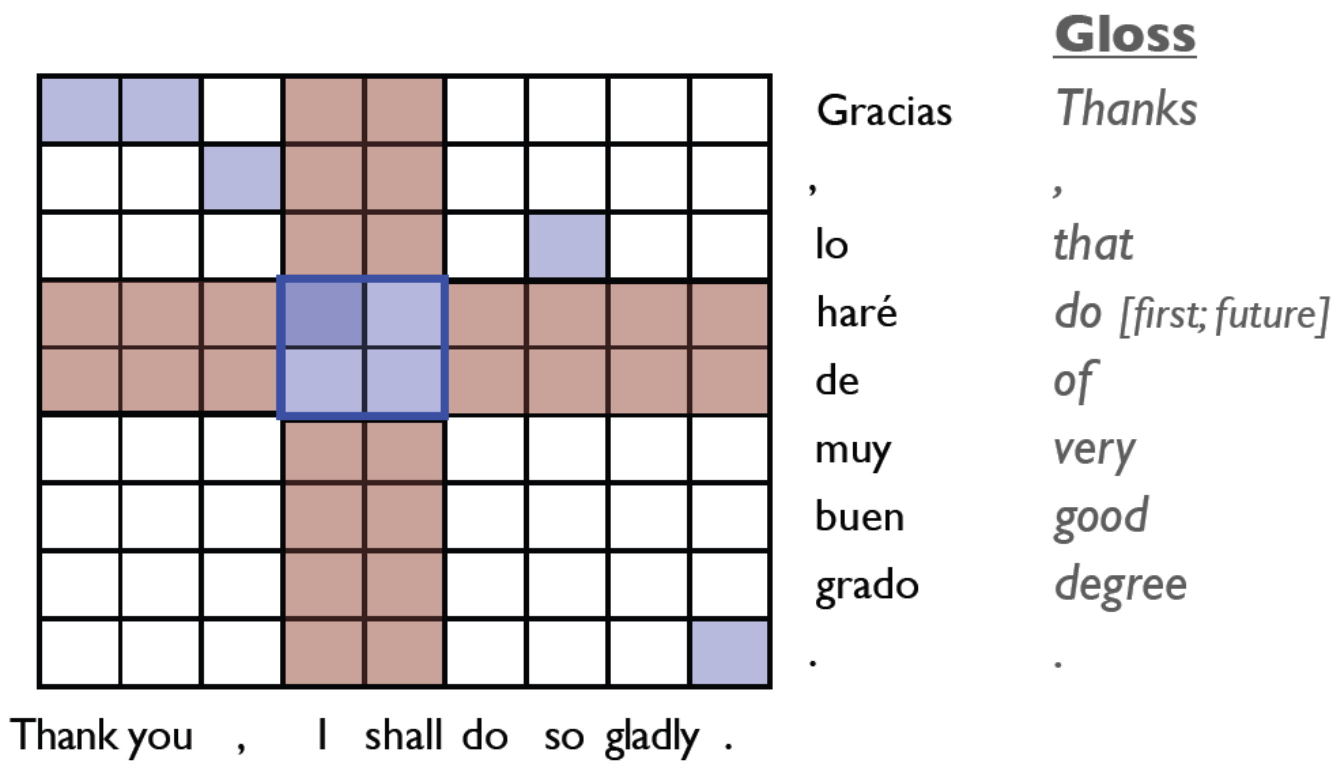
(GIZA++ Model 4 with
grow-diag-final combination)



What Happens in Practice

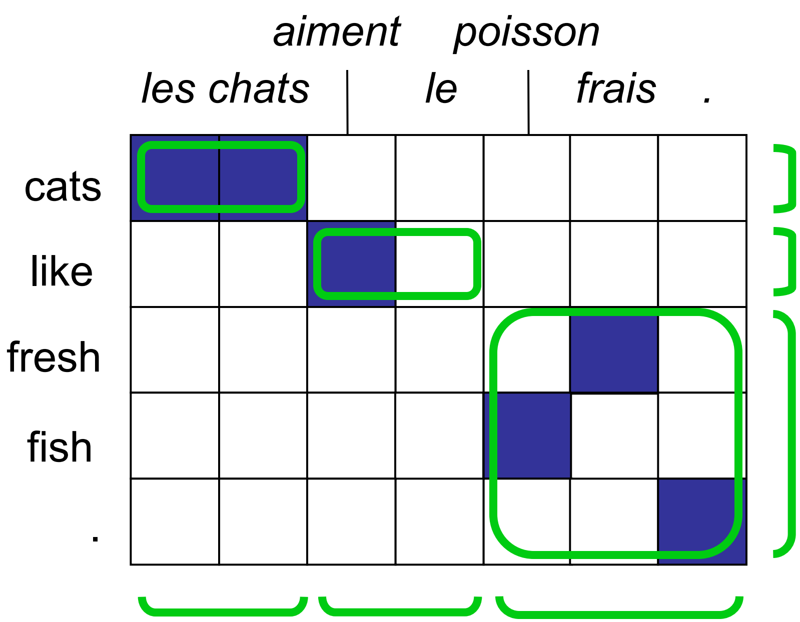
A real word alignment

(GIZA++ Model 4 with
grow-diag-final combination)



Phrase Scoring

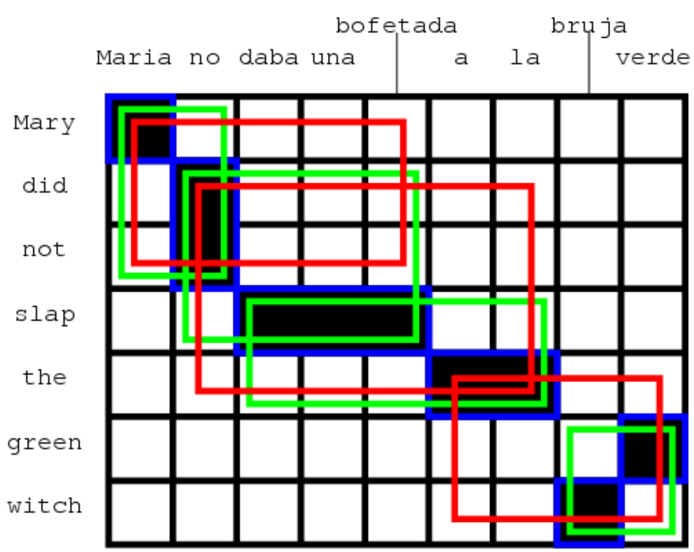
$$\phi_{new}(\bar{e}_j|\bar{f}_i) = \frac{c(\bar{f}_i, \bar{e}_j)}{c(\bar{f}_i)}$$

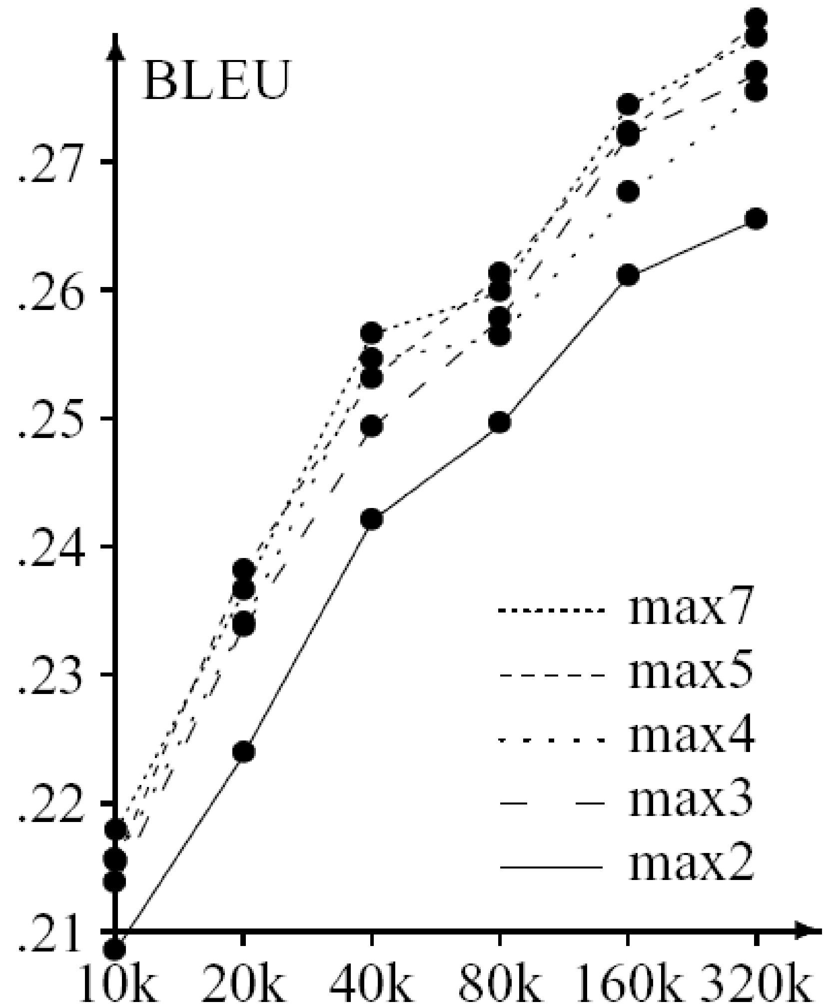


- Learning weights has been tried, several times:
  - [Marcu and Wong, 02]
  - [DeNero et al, 06]
  - … and others
- Seems not to work well, for a variety of partially understood reasons
- Main issue: big chunks get all the weight, obvious priors don't help
  - Though, [DeNero et al 08]

Phrase Size

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



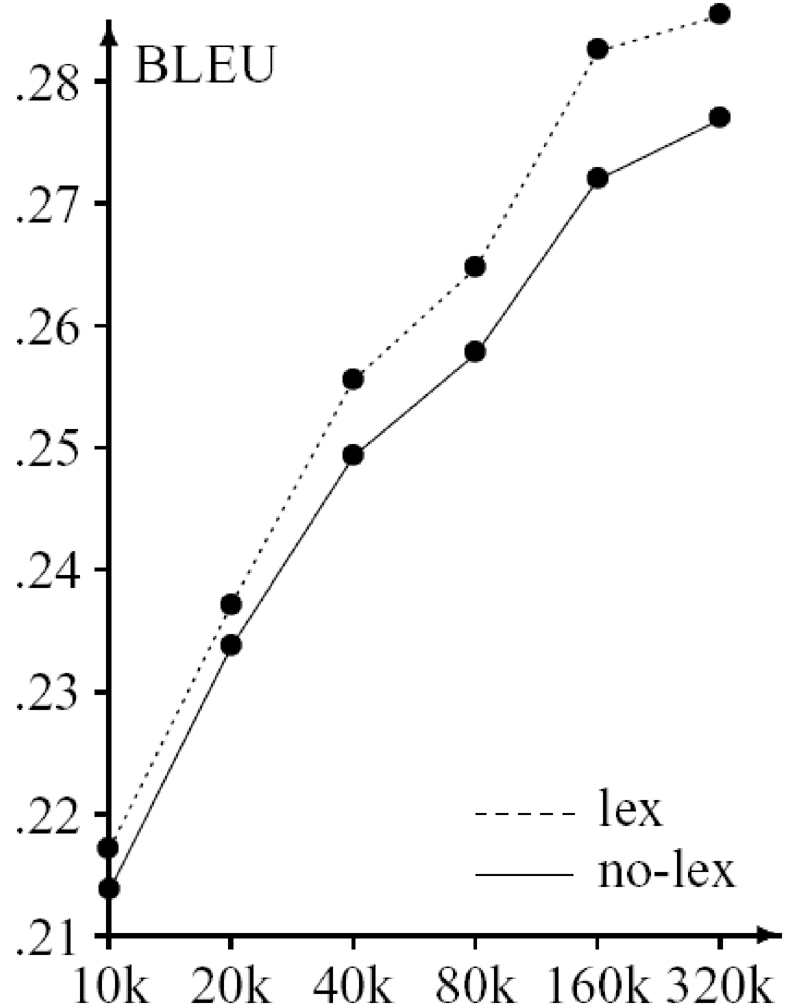


Lexical Weighting

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

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

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



Tuning for MT

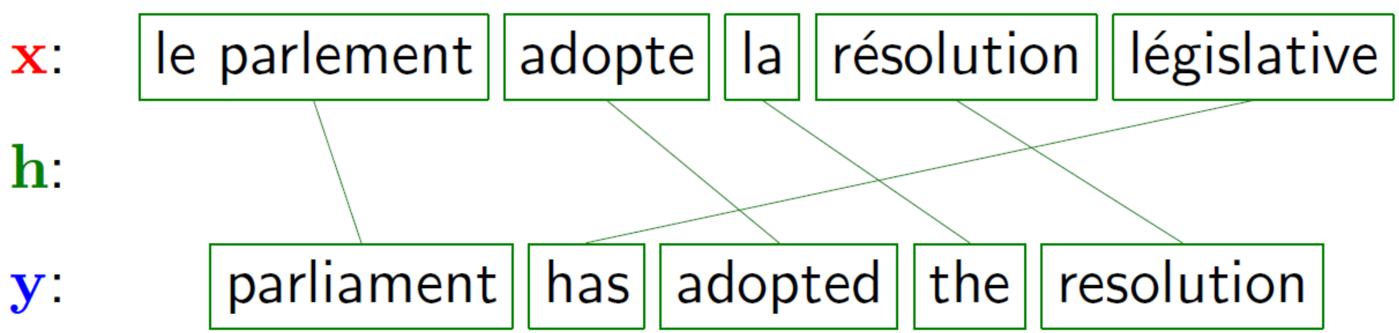
- Features encapsulate lots of information
  - Basic MT systems have around 6 features
  - P(e|f), P(f|e), lexical weighting, language model

- How to tune feature weights?

- Idea 1: Use your favorite classifier

Why Tuning is Hard

- Problem 1: There are latent variables
  - Alignments and segementations
  - Possibility: forced decoding (but it can go badly)

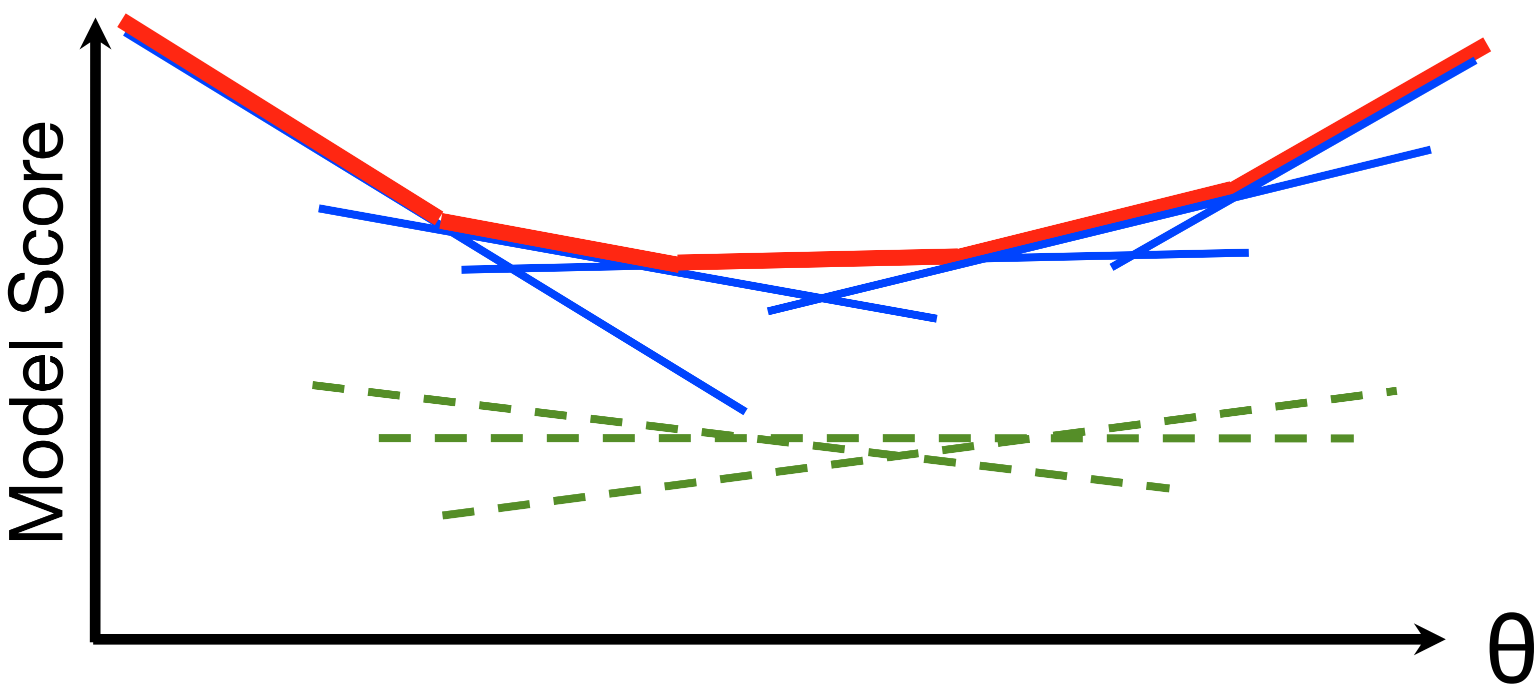


Why Tuning is Hard

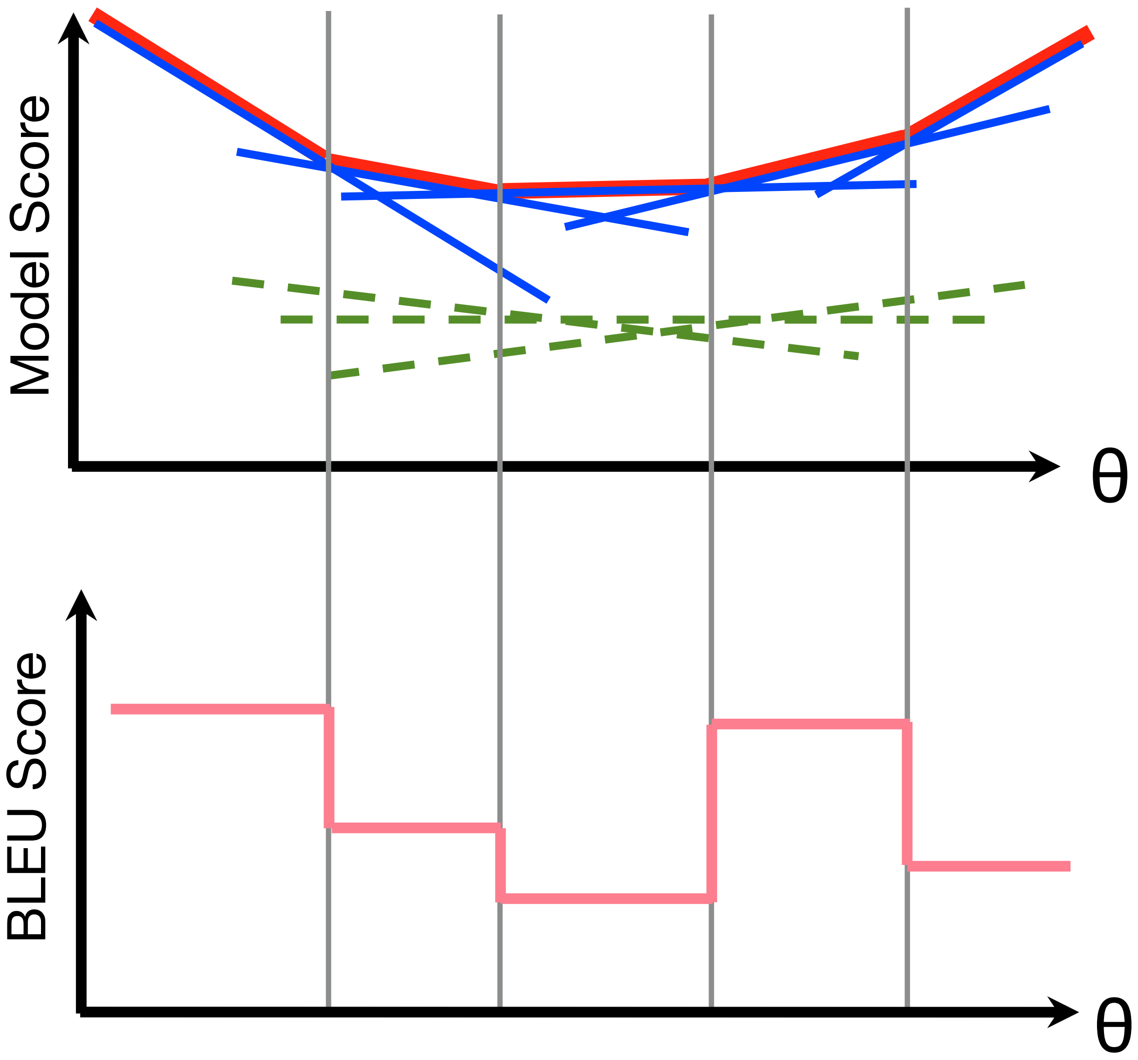
- Problem 3: Computational constraints
  - Discriminative training involves repeated decoding
  - Very slow! So people tune on sets much smaller than those used to build phrase tables

Minimum Error Rate Training

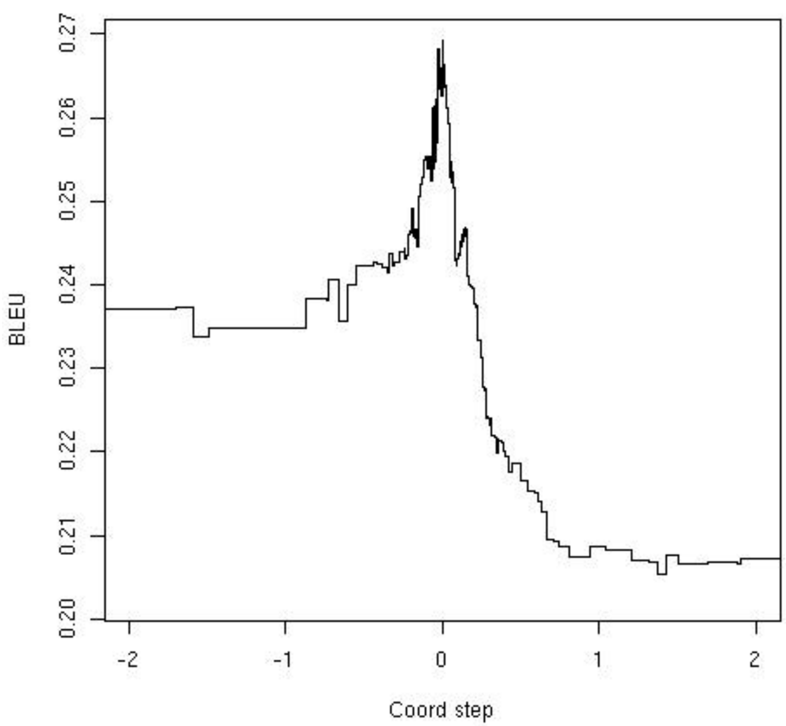
- Standard method: minimize BLEU directly (Och 03)
  - MERT is a discontinuous objective
  - Only works for max ~10 features, but works very well then
  - Here: k-best lists, but forest methods exist (Machery et al 08)
  - Recently, lots of alternatives being explored for more features



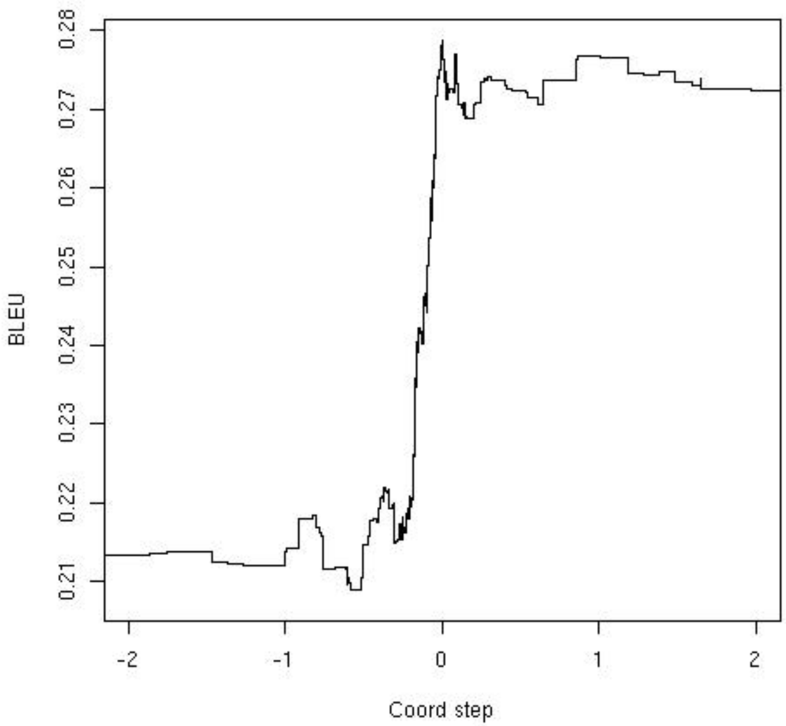
MERT



MERT

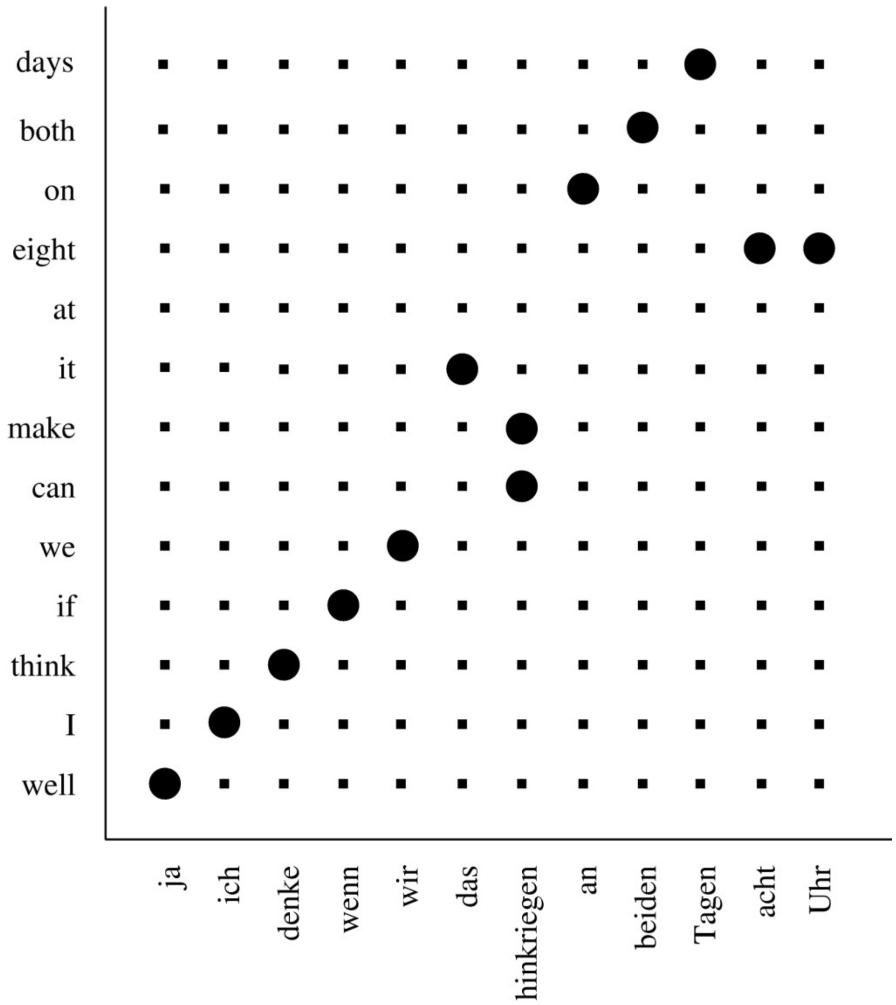


BLEU
0.20
0.21
0.22
0.23
0.24
0.25
0.26
0.27
-2
-1
0
1
2
Coord step

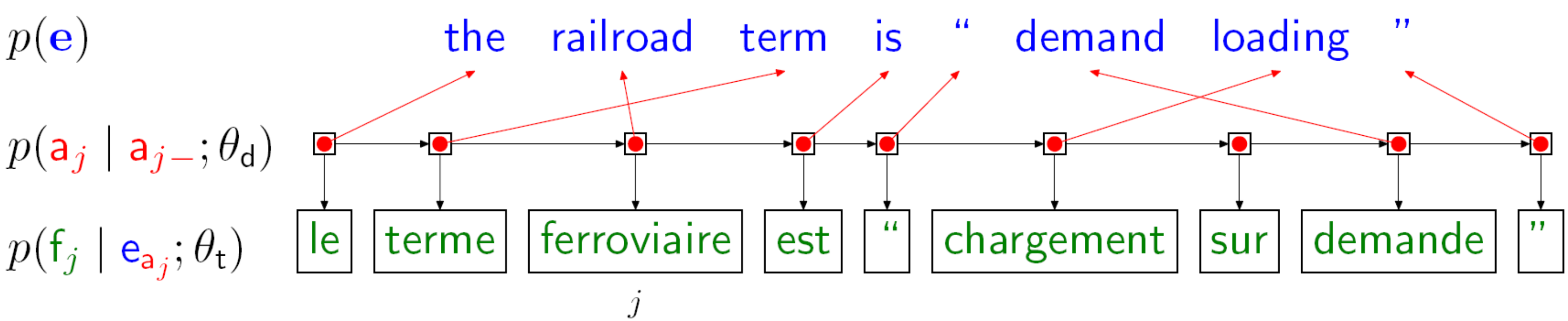


BLEU
0.21
0.22
0.23
0.24
0.25
0.26
0.27
0.28
-2
-1
0
1
2
Coord step

Phrase Movement



The HMM Model



Distortion θ_d

$p(\qquad) = 0.6$

$p(\qquad) = 0.2$

$p(\qquad) = \mathbf{0.1}$

$\cdots$

Translation θ_t

$p(\text{ the} \rightarrow \text{le }) = 0.53$

$p(\text{ the} \rightarrow \text{la }) = 0.24$

$p(\textbf{ railroad} \rightarrow \textbf{ferroviaire }) = \mathbf{0.19}$

$p(\text{ NULL} \rightarrow \text{le }) = 0.12$

$\cdots$