Estimation Problems in Machine Translation
(learning to translate)

John DeNero

Some slides borrowed from Dan Klein and David Chiang

Parameter Estimation

- So far, we've seen formalisms and search techniques for translation
- Now, we need to assign features and scores to translations so that we can pick one
- Machine translation systems typically incorporate multiple estimation problems

Derivations, Features and Models

Synchronous Derivation

lo haré de muy buen grado .

Grammar

X → ⟨ lo haré X . ; I will do it X . ⟩

X → ⟨ de muy buen grado ; gladly ⟩

Derivations, Features and Models

Synchronous Derivation

X

lo haré de muy buen grado .

X

gladly

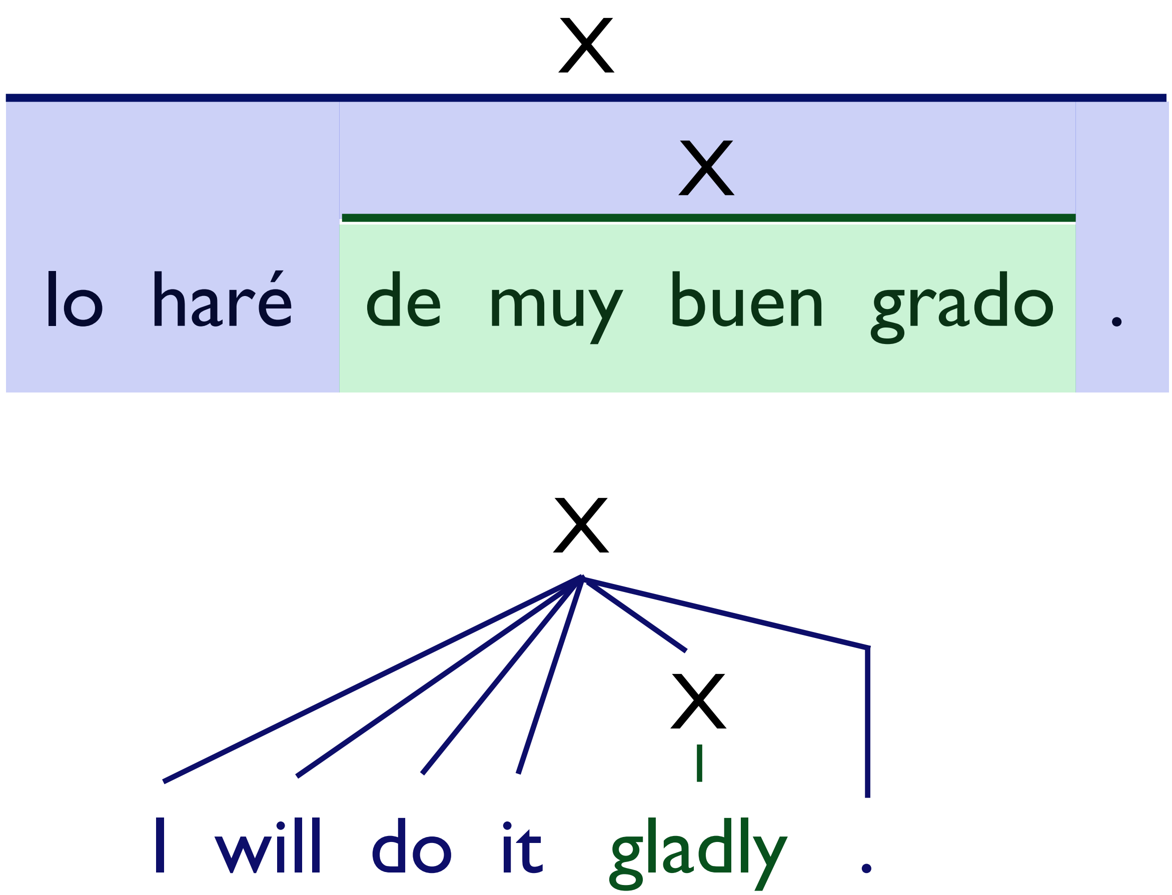
Grammar

X → ⟨ lo haré X . ; I will do it X . ⟩

X → ⟨ de muy buen grado ; gladly ⟩

Derivations, Features and Models

Synchronous Derivation



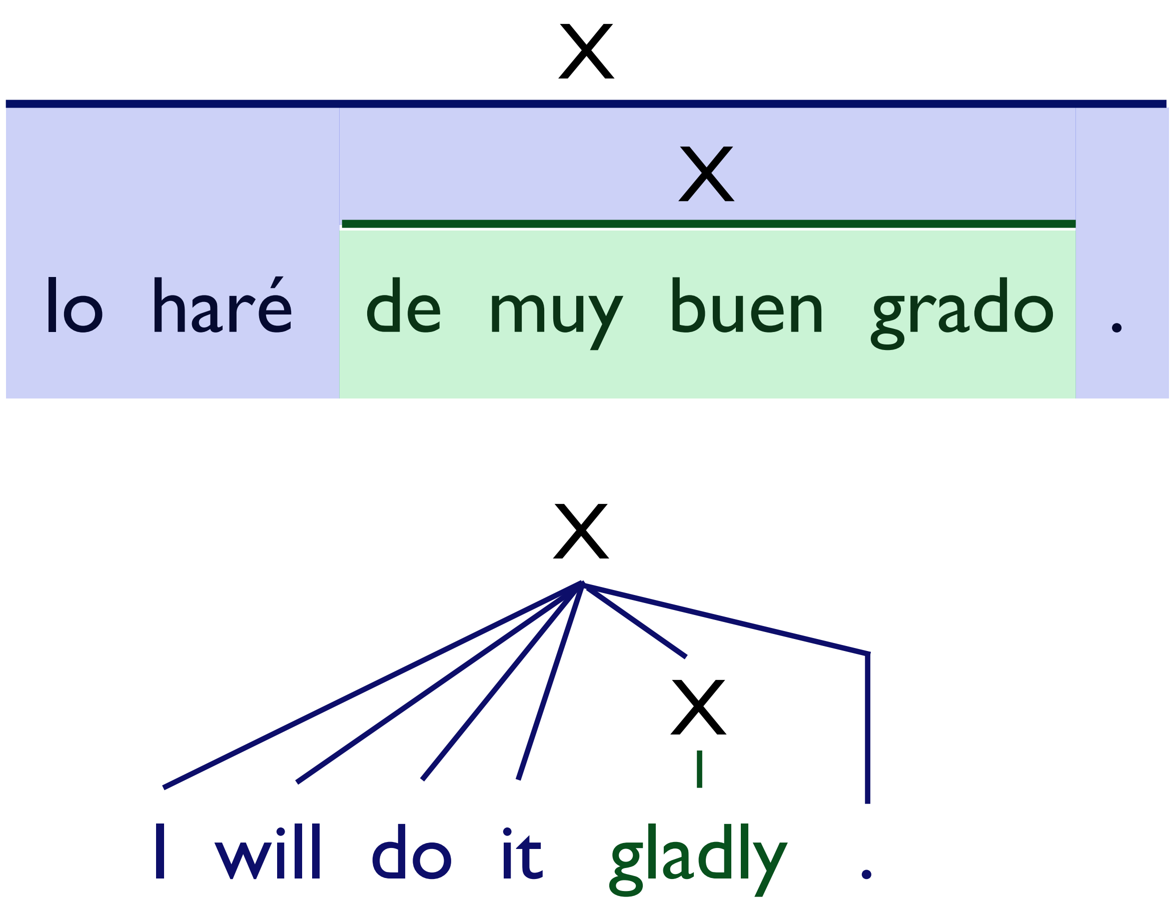
Grammar

X → ⟨ lo haré X . ; I will do it X . ⟩

X → ⟨ de muy buen grado ; gladly ⟩

Derivations, Features and Models

Synchronous Derivation



Grammar

X → ⟨ lo haré X . ; I will do it X . ⟩

X → ⟨ de muy buen grado ; gladly ⟩

Features

Language model

Translation models

Simple features

Score

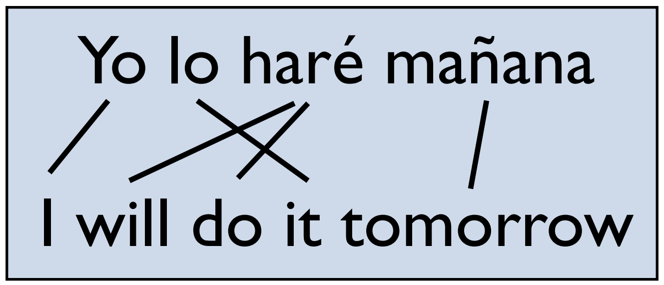
$$\prod_{i=1}^{I} P(e_i | e_{i-1}, \ldots, e_1)^{\lambda_1} \cdot$$

$$\prod_{r} P(e_r | f_r)^{\lambda_2} P(f_r | e_r)^{\lambda_3} \ldots$$

Learn all these from data

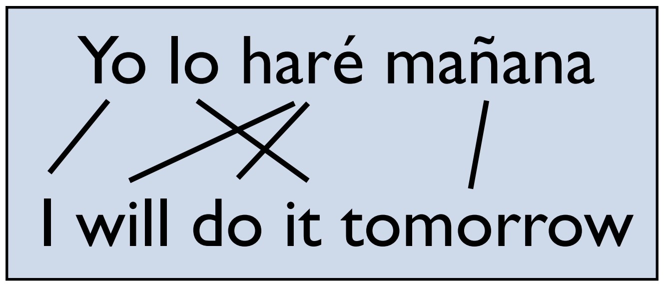
Features Match Model Structure

In 1993, we aligned words



Features Match Model Structure

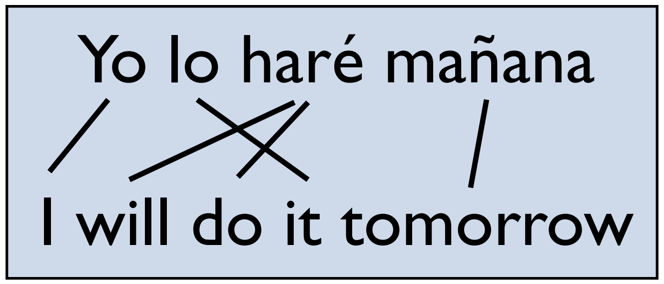
In 1993, we aligned words



English (E)	*P*(E | mañana)
tomorrow	0.7
morning	0.3

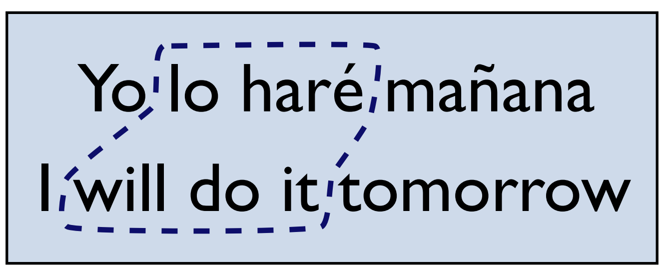
Features Match Model Structure

In 1993, we aligned words



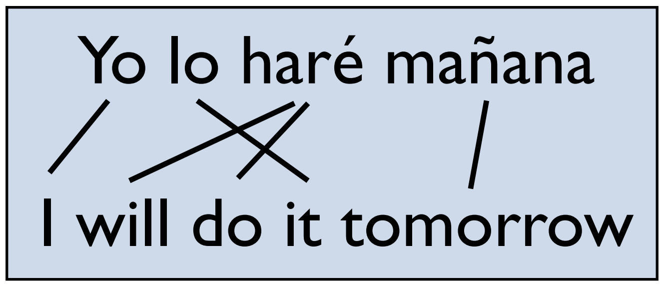
English (E)	*P*(E | mañana)
tomorrow	0.7
morning	0.3

In 1999, we aligned phrases



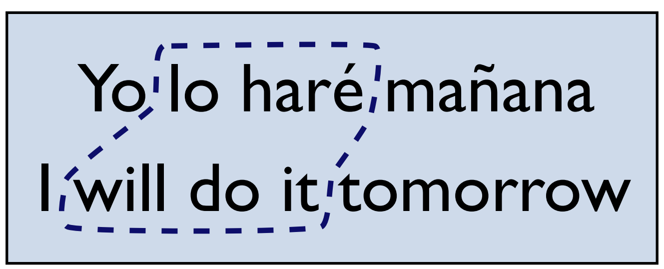
Features Match Model Structure

In 1993, we aligned words



English (E)	*P*(E | mañana)
tomorrow	0.7
morning	0.3

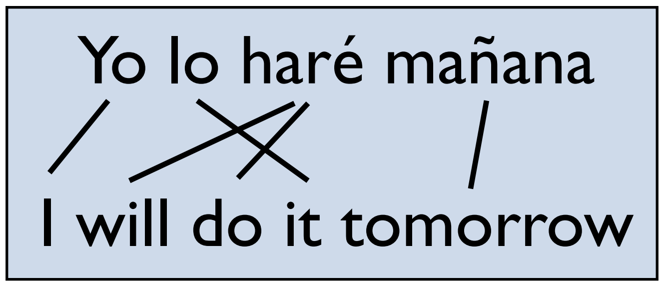
In 1999, we aligned phrases



English (E)	*P*(E | lo haré)
will do it	0.8
will do so	0.2

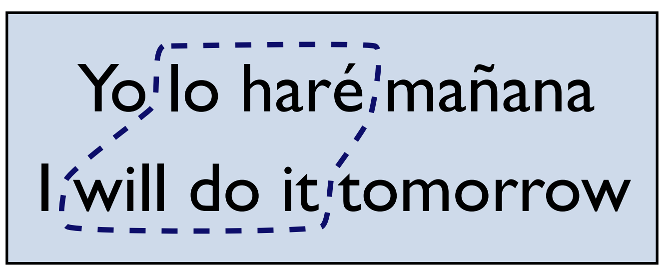
Features Match Model Structure

In 1993, we aligned words



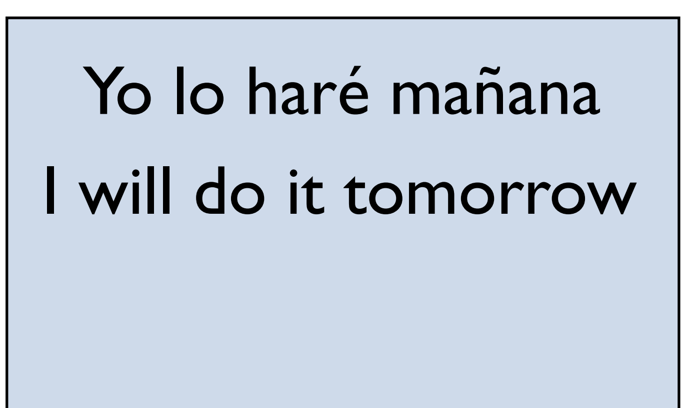
English (E)	*P*(E | mañana)
tomorrow	0.7
morning	0.3

In 1999, we aligned phrases



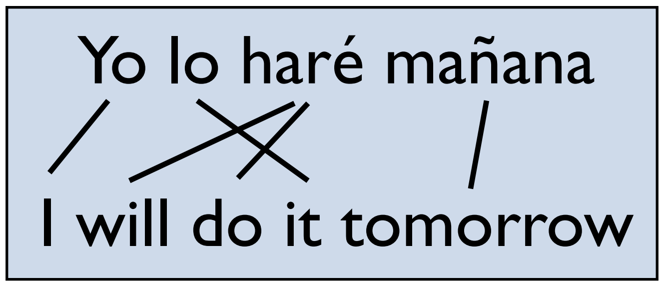
English (E)	*P*(E | lo haré)
will do it	0.8
will do so	0.2

In 2004, we aligned trees



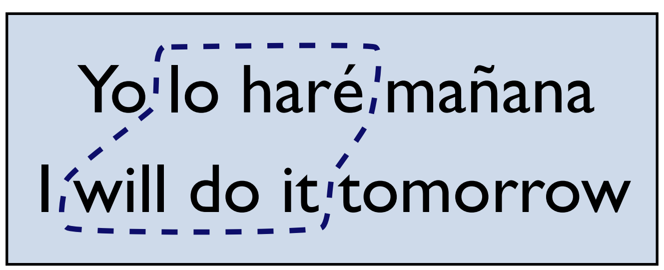
Features Match Model Structure

In 1993, we aligned words



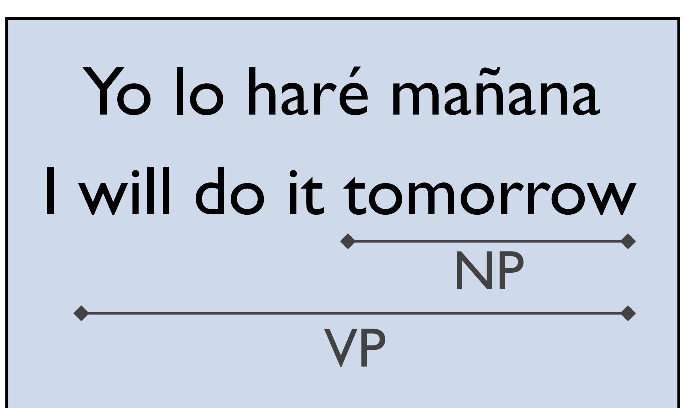
English (E)	*P*(E | mañana)
tomorrow	0.7
morning	0.3

In 1999, we aligned phrases



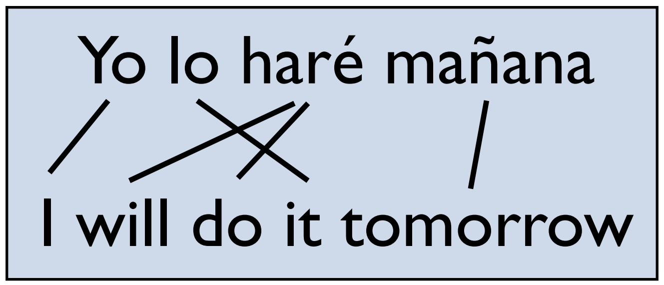
English (E)	*P*(E | lo haré)
will do it	0.8
will do so	0.2

In 2004, we aligned trees



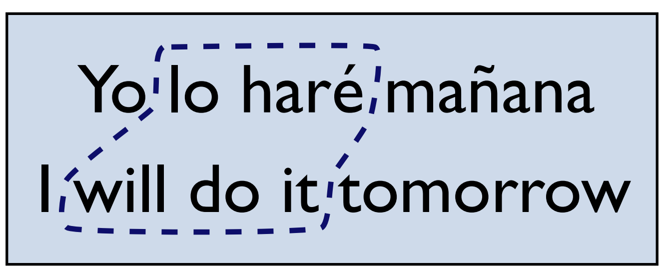
Features Match Model Structure

In 1993, we aligned words



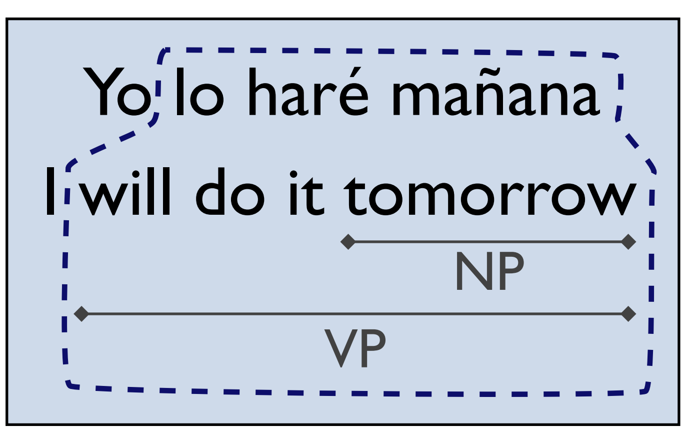
English (E)	*P*(E | mañana)
tomorrow	0.7
morning	0.3

In 1999, we aligned phrases



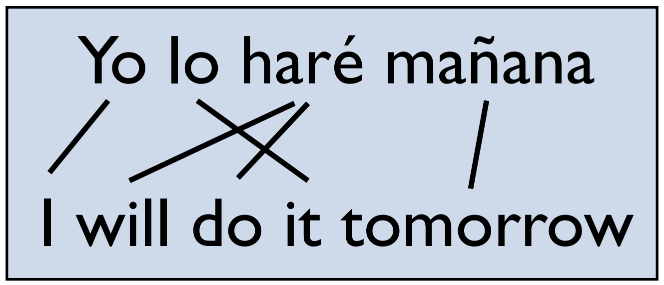
English (E)	*P*(E | lo haré)
will do it	0.8
will do so	0.2

In 2004, we aligned trees



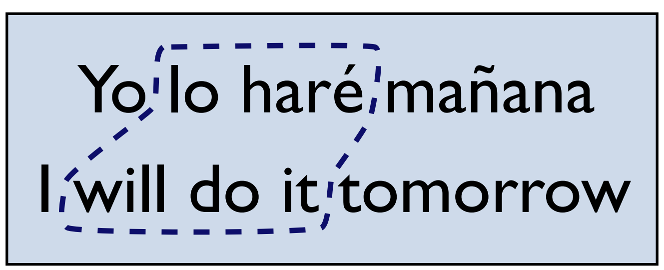
Features Match Model Structure

In 1993, we aligned words



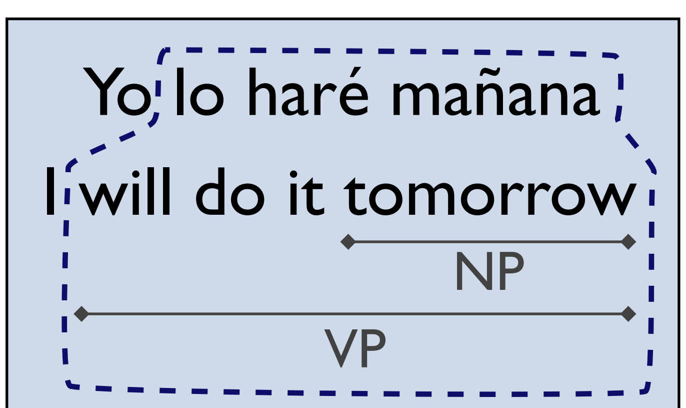
English (E)	*P*(E | mañana)
tomorrow	0.7
morning	0.3

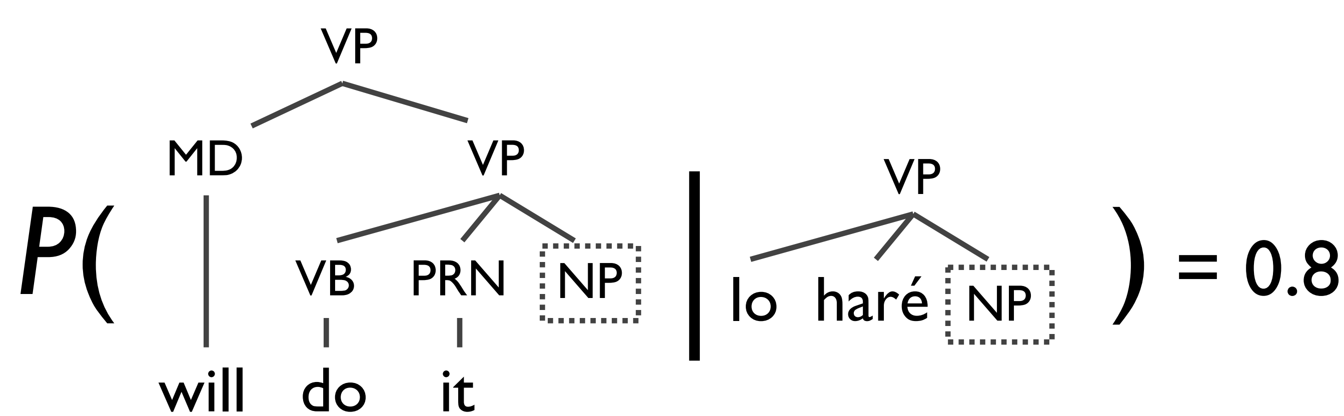
In 1999, we aligned phrases



English (E)	*P*(E | lo haré)
will do it	0.8
will do so	0.2

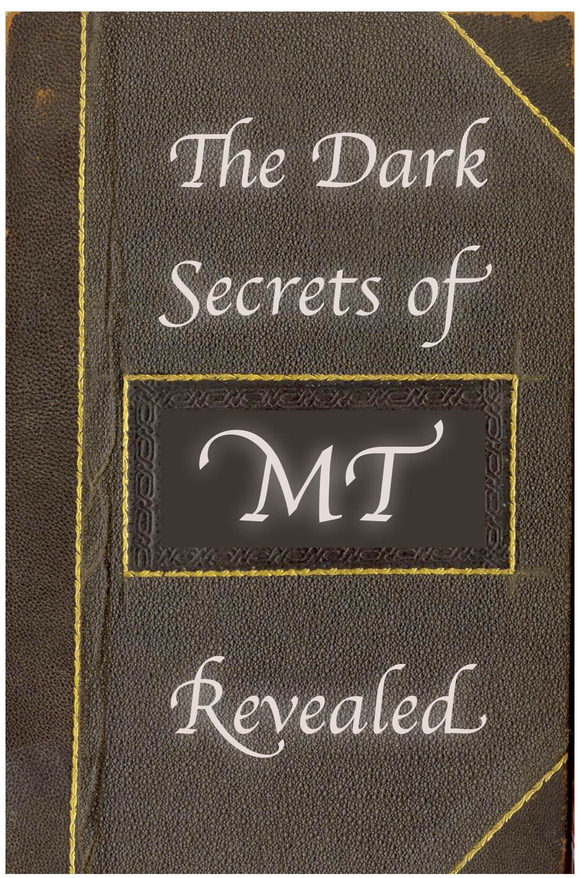
In 2004, we aligned trees





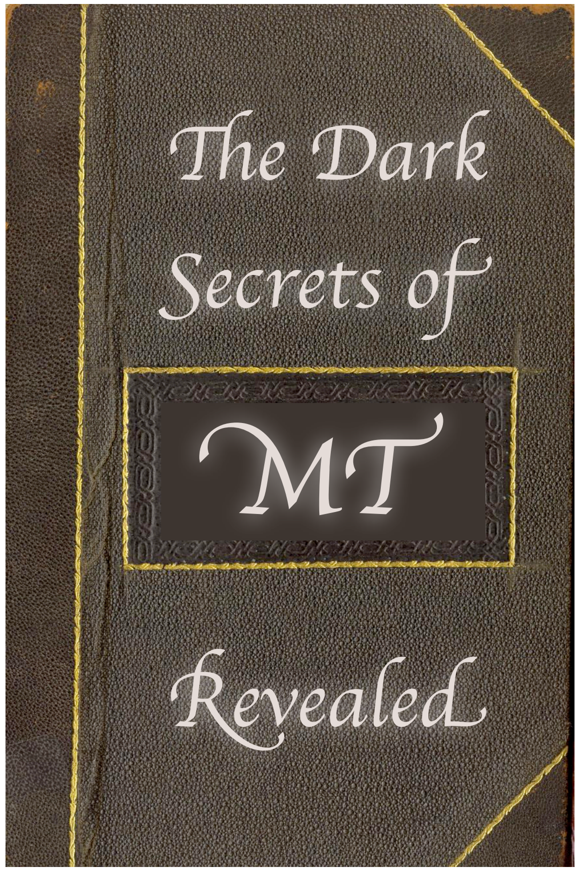
Aligning Structural Components

Today, we actually still align words



Aligning Structural Components

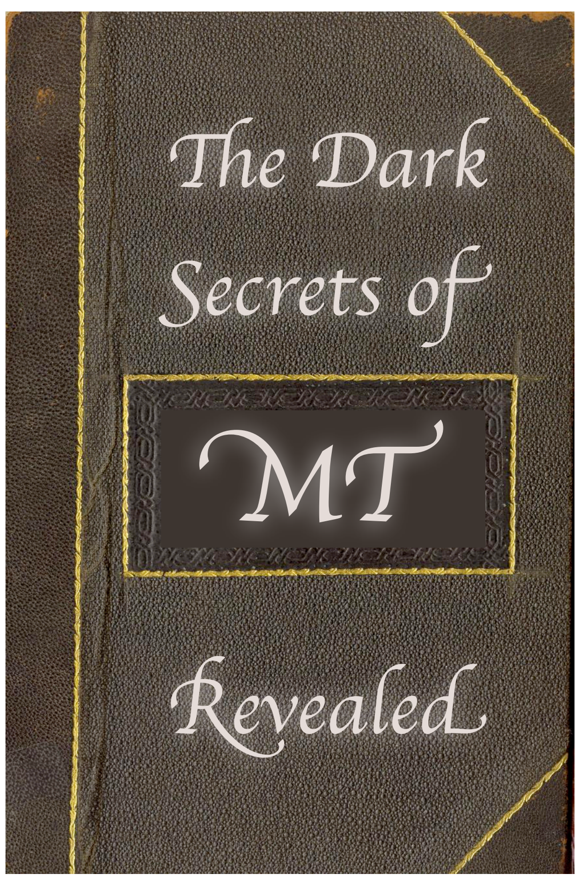
Today, we actually still align words



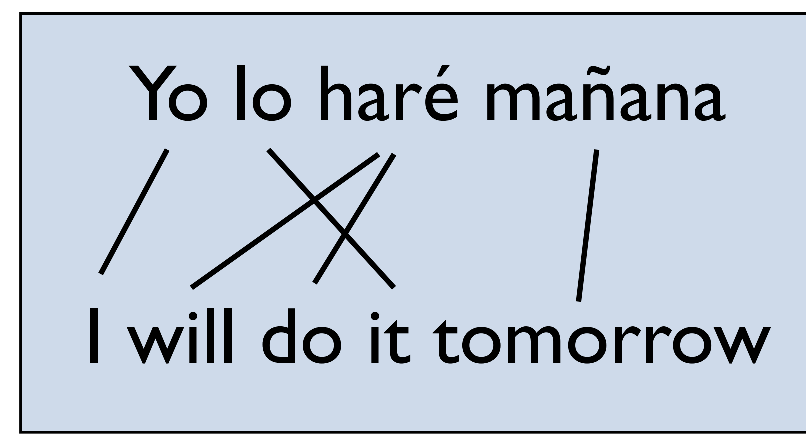
1. *Align words with a probabilistic model*

Aligning Structural Components

Today, we actually still align words

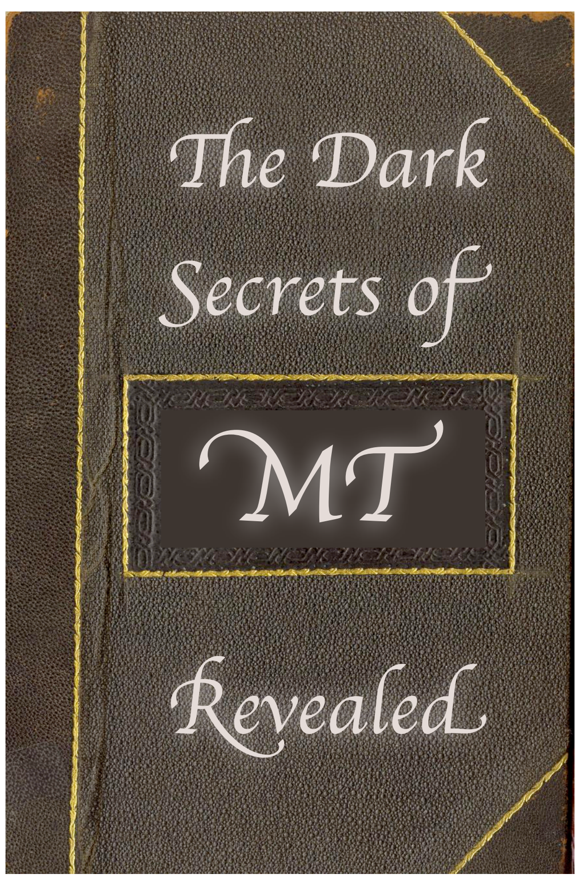


1 *Align words with a probabilistic model*

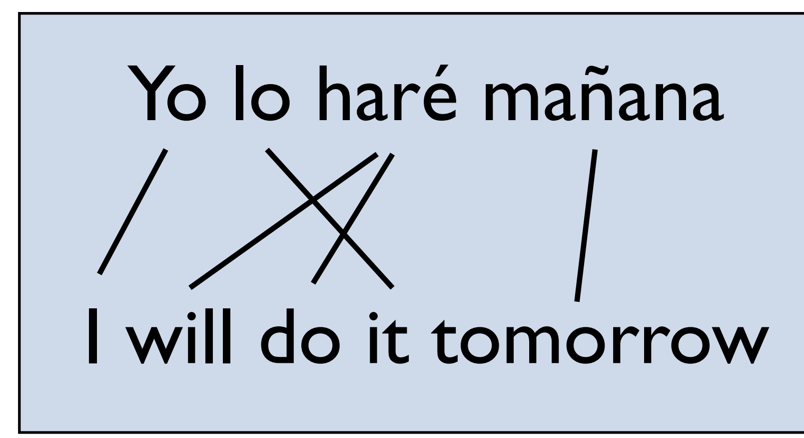


Aligning Structural Components

Today, we actually still align words

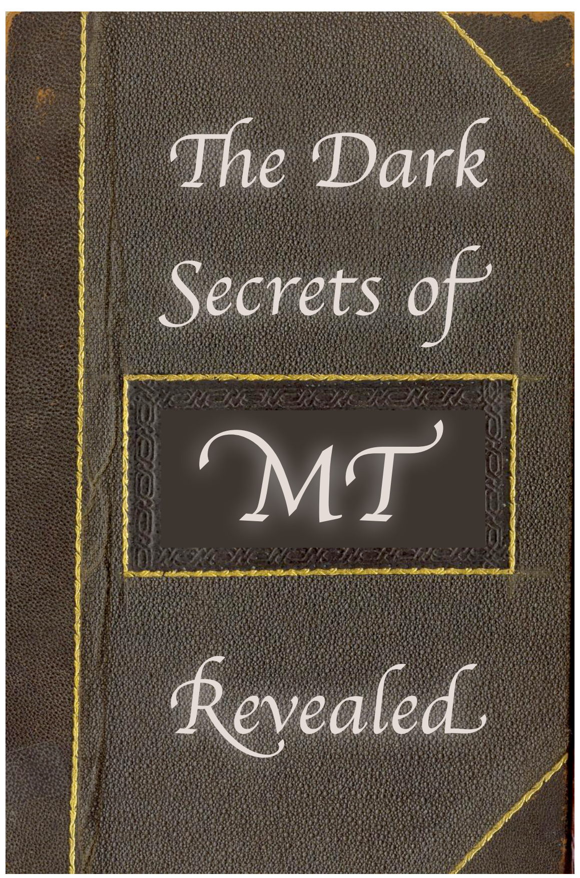


1. *Align words with a probabilistic model*
2. *Infer presence of larger structures from this alignment*



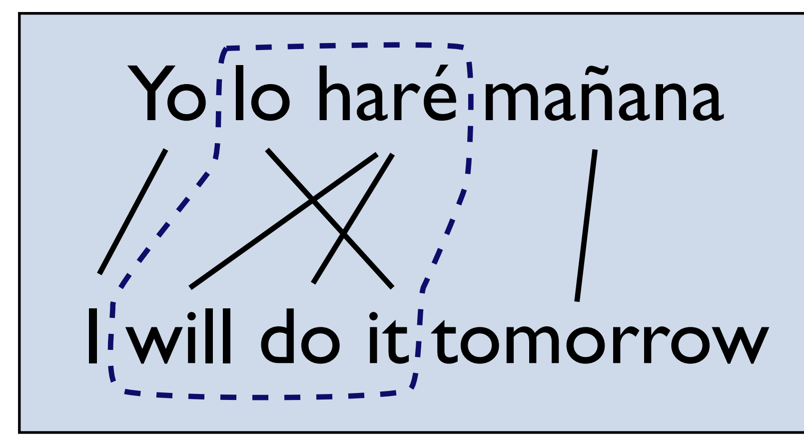
Aligning Structural Components

Today, we actually still align words



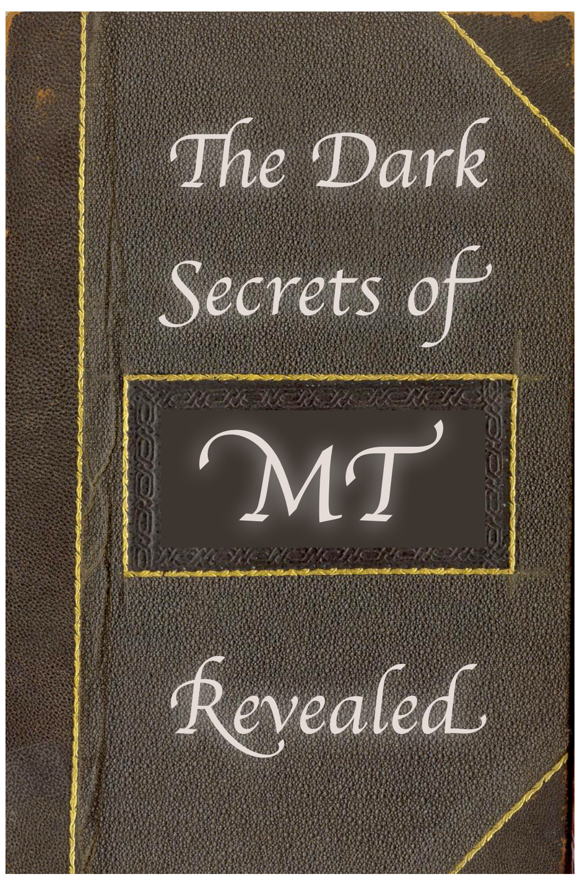
1. *Align words with a probabilistic model*

2. *Infer presence of larger structures from this alignment*

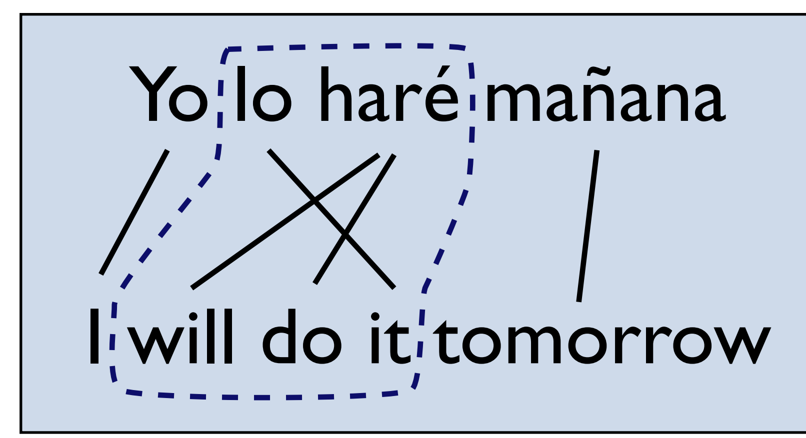


Aligning Structural Components

Today, we actually still align words



1. *Align words with a probabilistic model*
2. *Infer presence of larger structures from this alignment*
3. *Translate with the larger structures*

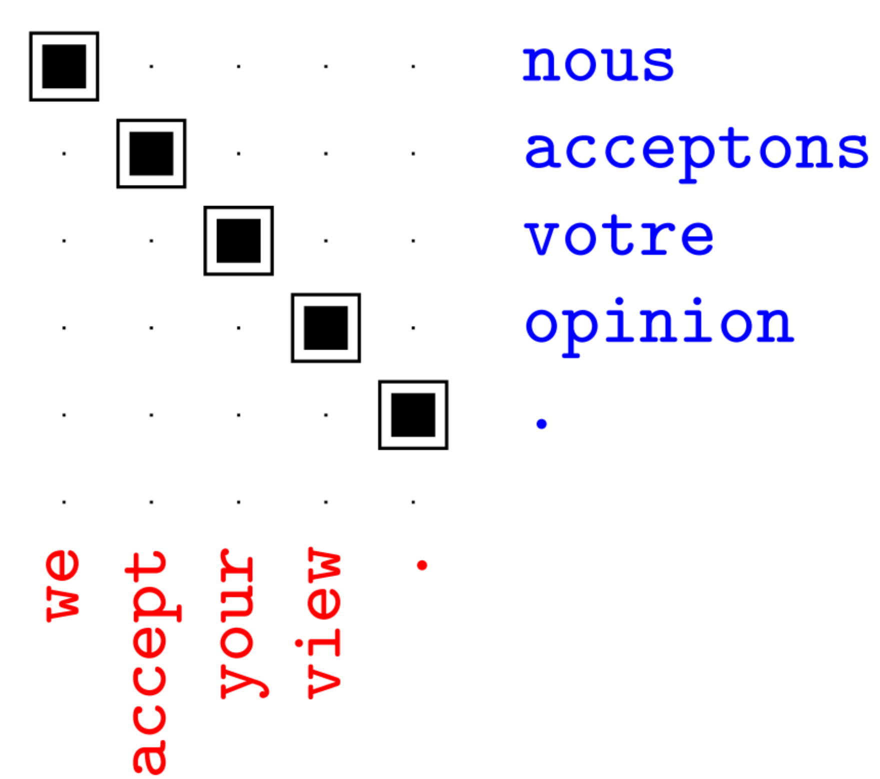


Unsupervised Word Alignment

- **Input**: A large *bitext* of sentences and their translations
- **Approach**: Using what we know about the problem and corpus statistics, align automatically
- **Exciting fact**: Unsupervised methods perform well enough that very few systems use supervised word alignment

Unsupervised Word Alignment

- **Input**: A large *bitext* of sentences and their translations
- **Approach**: Using what we know about the problem and corpus statistics, align automatically
- **Exciting fact**: Unsupervised methods perform well enough that very few systems use supervised word alignment



Properties of Word Alignments

I declare resumed the session of the european parliament

Declaro reanudado el periodo de sesiones del parlamento europeo

adjourned on Friday 17 December 1999 , ...

interrumpido el Viernes 17 de Diciembre pasado , ...

Properties of Word Alignments

I declare resumed the session of the european parliament

Declaro reanudado el periodo de sesiones del parlamento europeo

adjourned on Friday 17 December 1999 , ...

interrumpido el Viernes 17 de Diciembre pasado , ...

Properties of Word Alignments

I declare resumed the session of the european parliament

Declaro reanudado el periodo de sesiones del parlamento europeo

adjourned on Friday 17 December 1999 , ...

interrumpido el Viernes 17 de Diciembre pasado , ...

Properties of Word Alignments

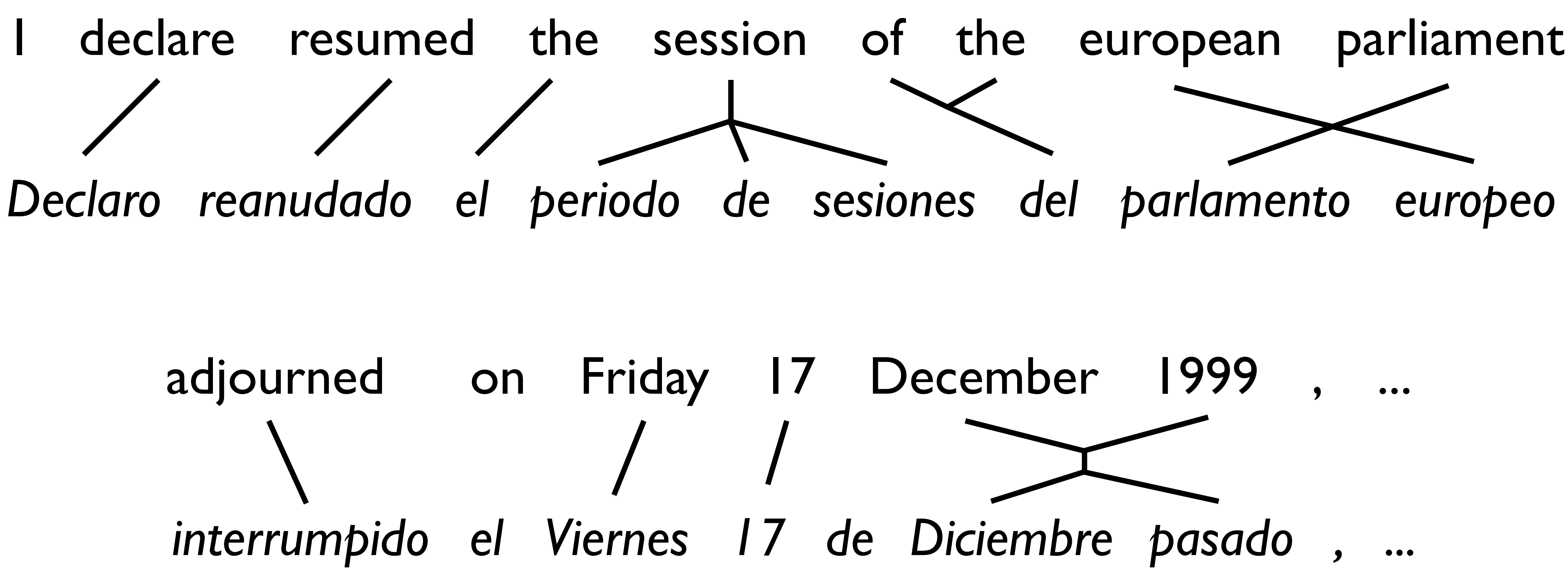
I declare resumed the session of the european parliament

Declaro reanudado el periodo de sesiones del parlamento europeo

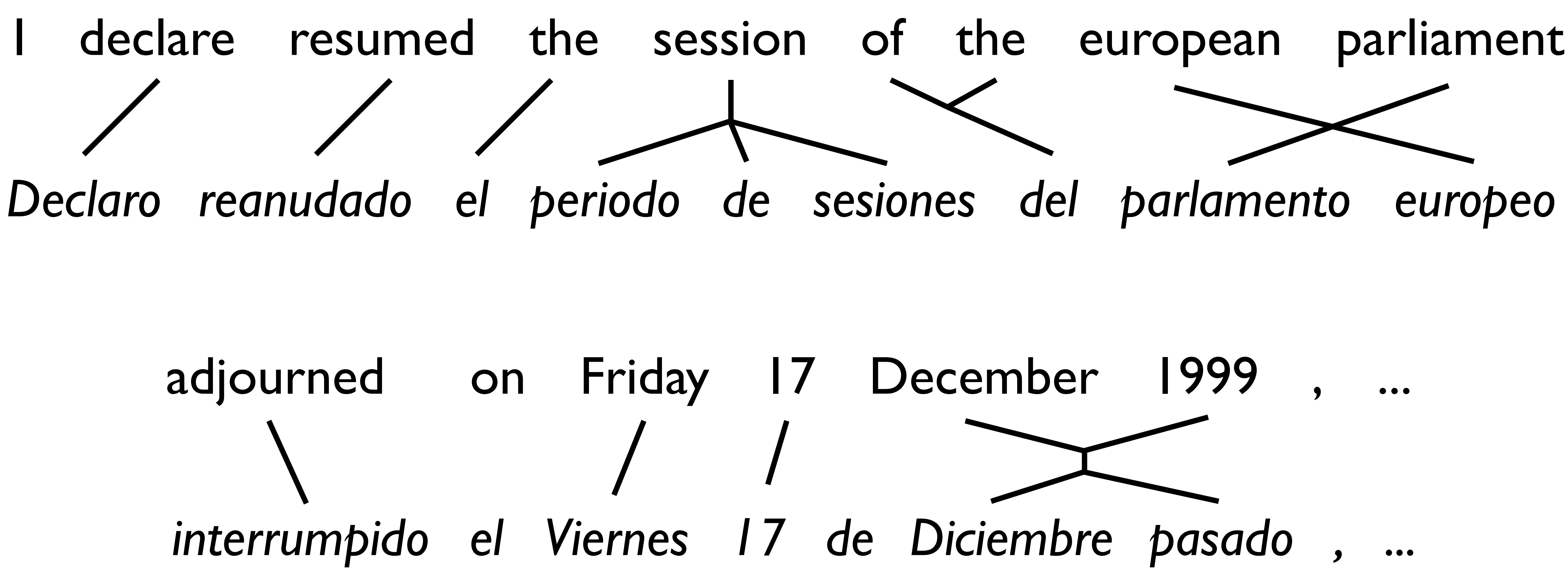
adjourned on Friday 17 December 1999 , ...

interrumpido el Viernes 17 de Diciembre pasado , ...

Properties of Word Alignments



Properties of Word Alignments



- Often one-to-one or many-to-one (usually over contiguous phrases)
- Occasionally many-to-many, driven by non-literal translations

Heuristic Estimation

- Two words that co-occur regularly are translations

$c(e, f)$ *The number of times e and f appear together*

Heuristic Estimation

- Two words that co-occur regularly are translations

  $c(e, f)$ *The number of times e and f appear together*

- Normalize by the word frequencies

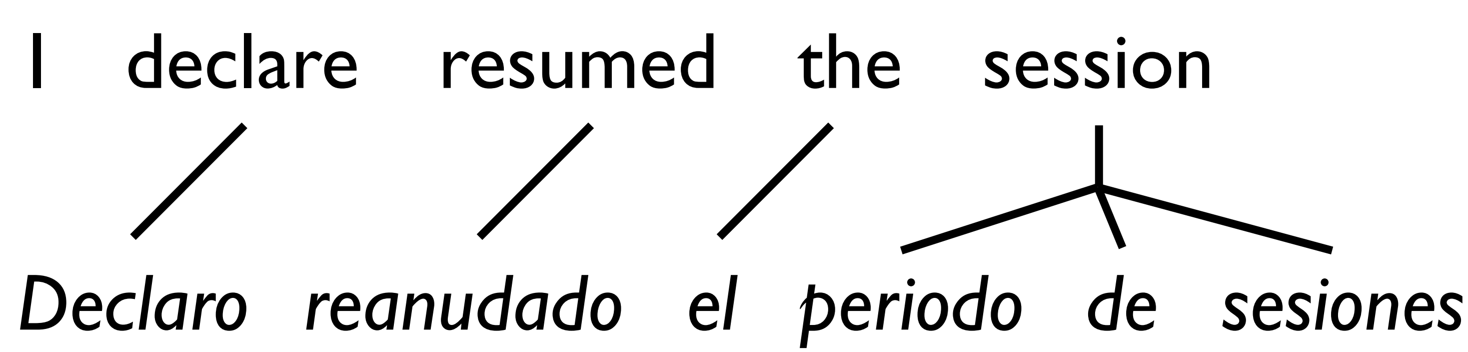
  $c(f)$ *Count of word f* $c(e)$ *Count of word f*

  $$\frac{2 \cdot c(e, f)}{c(e) + c(f)}$$ *Dice coefficient*

- Enforcing competition across words (e.g., finding a one-to-one or many-to-one mapping) is a good idea

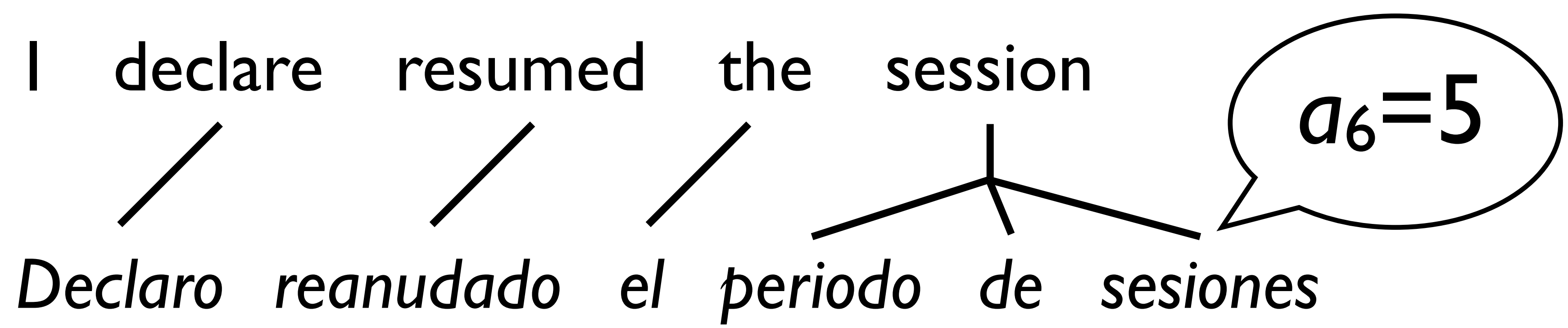
IBM Model 1 (Brown '93)

- Probabilistic models naturally impose competition
- Assume that foreign words are generated independently
- Assume a hidden alignment vector *a* encoding which English word generates each foreign word



IBM Model 1 (Brown '93)

- Probabilistic models naturally impose competition
- Assume that foreign words are generated independently
- Assume a hidden alignment vector *a* encoding which English word generates each foreign word



IBM Model 1 (Brown '93)

- Probabilistic models naturally impose competition
- Assume that foreign words are generated independently
- Assume a hidden alignment vector *a* encoding which English word generates each foreign word

I declare resumed the session

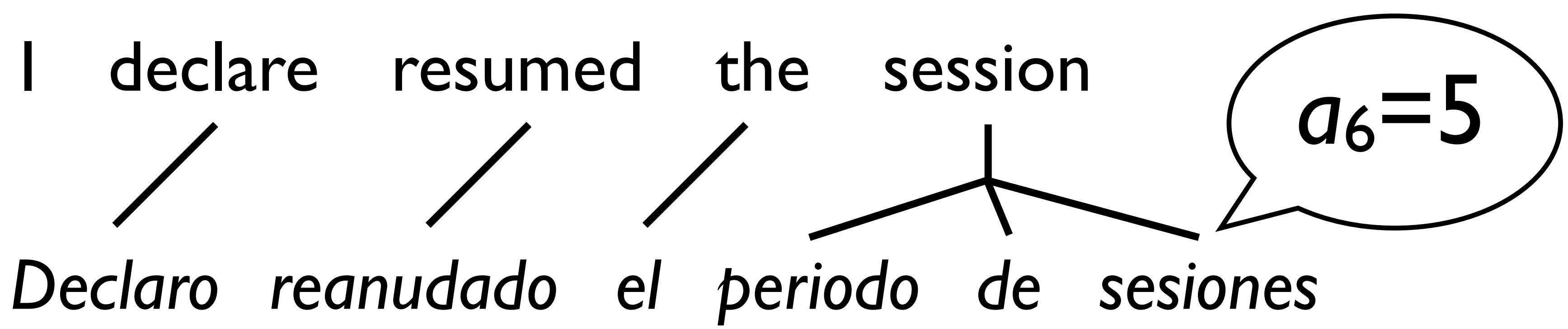
$a_6=5$

Declaro reanudado el periodo de sesiones

$$P(f, a|e) \quad = \quad \prod_{j=1}^{J} P(a_j = i|I, J) P(f_j|e_i)$$

IBM Model 1 (Brown '93)

- Probabilistic models naturally impose competition
- Assume that foreign words are generated independently
- Assume a hidden alignment vector *a* encoding which English word generates each foreign word



$$P(f, a|e) = \prod_{j=1}^{J} P(a_j = i|I, J) P(f_j|e_i)$$

$$= \frac{1}{I+1} P(f_j|e_i)$$

Estimating Model 1 Parameters

$$P(f|e)$$

Estimating Model 1 Parameters

- Free parameters in the model: $P(f|e)$
- Goal is to maximize the data likelihood

Estimating Model 1 Parameters

- Free parameters in the model: $P(f|e)$
- Goal is to maximize the data likelihood
- E-step computes expected alignments (posteriors)

Estimating Model 1 Parameters

- Free parameters in the model: $P(f|e)$

- Goal is to maximize the data likelihood

- E-step computes expected alignments (posteriors)

$$P(a_j = i|\mathbf{e}, \mathbf{f}) = \frac{\frac{1}{I+1} P(f_j|e_i)}{\sum_{i'} \frac{1}{I+1} P(f_j|e_{i'})}$$

Estimating Model 1 Parameters

- Free parameters in the model: $P(f|e)$
- Goal is to maximize the data likelihood
- E-step computes expected alignments (posteriors)

$$P(a_j = i|\mathbf{e}, \mathbf{f}) = \frac{\frac{1}{I+1} P(f_j|e_i)}{\sum_{i'} \frac{1}{I+1} P(f_j|e_{i'})}$$

- M-step computes ratios of expected counts

Estimating Model 1 Parameters

- Free parameters in the model: $P(f|e)$
- Goal is to maximize the data likelihood
- E-step computes expected alignments (posteriors)

$$P(a_j = i|\mathbf{e},\mathbf{f}) = \frac{\frac{1}{I+1}P(f_j|e_i)}{\sum_{i'} \frac{1}{I+1}P(f_j|e_{i'})}$$

- M-step computes ratios of expected counts

$$P(f|e) = \frac{\text{sum of posteriors for } f \text{ aligned to } e}{\text{sum of posteriors of any } f' \text{ aligned to } e}$$

Estimating Model 1 Parameters

- Free parameters in the model: $P(f|e)$
- Goal is to maximize the data likelihood
- E-step computes expected alignments (posteriors)

$$P(a_j = i|\mathbf{e}, \mathbf{f}) = \frac{\frac{1}{I+1} P(f_j|e_i)}{\sum_{i'} \frac{1}{I+1} P(f_j|e_{i'})}$$

- M-step computes ratios of expected counts

$$P(f|e) = \frac{\text{sum of posteriors for } f \text{ aligned to } e}{\text{sum of posteriors of any } f' \text{ aligned to } e}$$

- Repeat e- and m-step many times (like 5 or 10)

Aligning Words Under the Model

- **Viterbi**: For every j, select i that maximizes

$$P(a_j = i|\mathbf{e}, \mathbf{f})$$

Gives competition among explanations

- **Posterior**: Align every (i,j) that has

$$P(a_j = i|\mathbf{e}, \mathbf{f}) > \tau$$

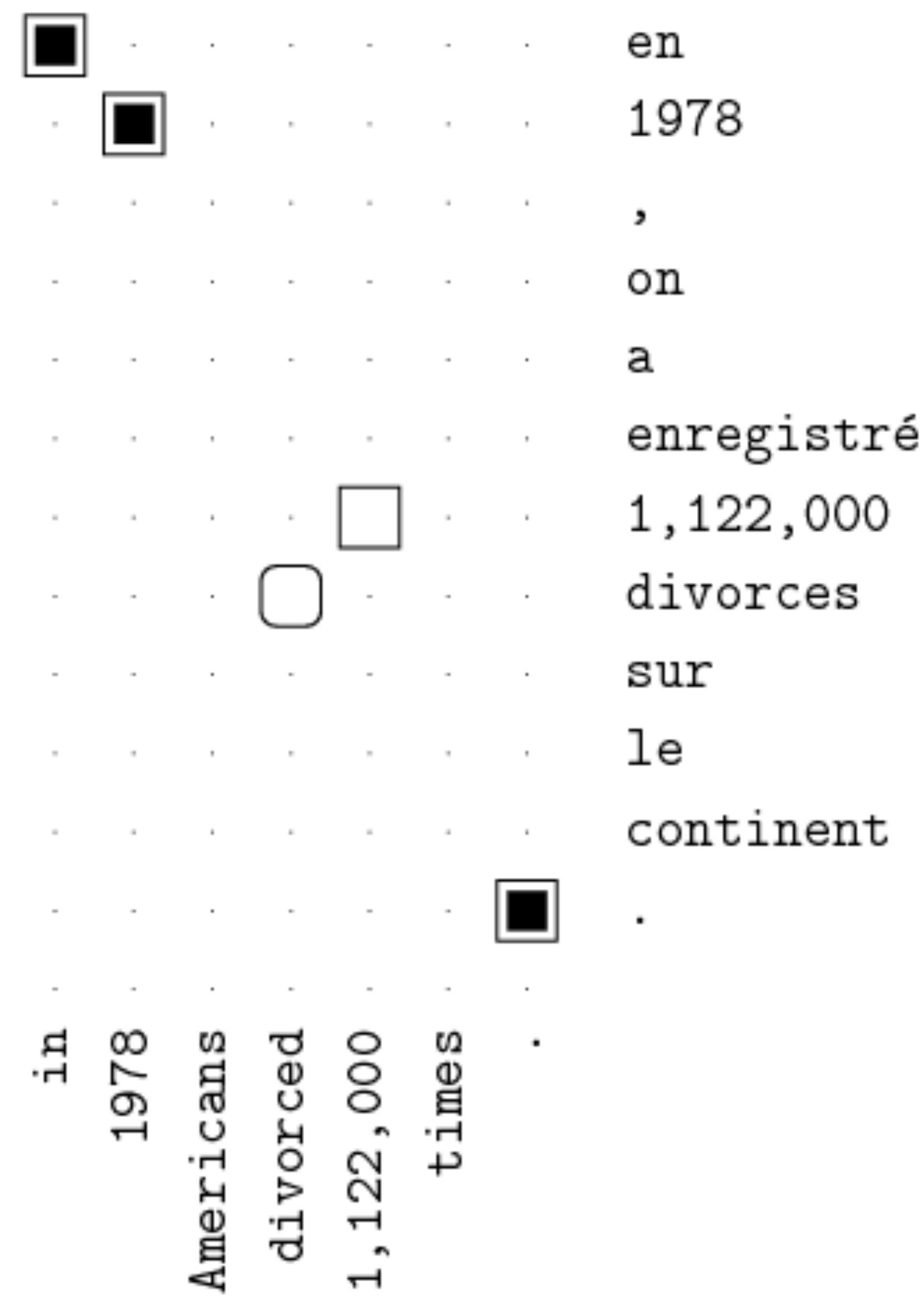
Gives control over how many alignment links to posit

Evaluation: Alignment Error Rate

= Sure

= Possible

= Predicted



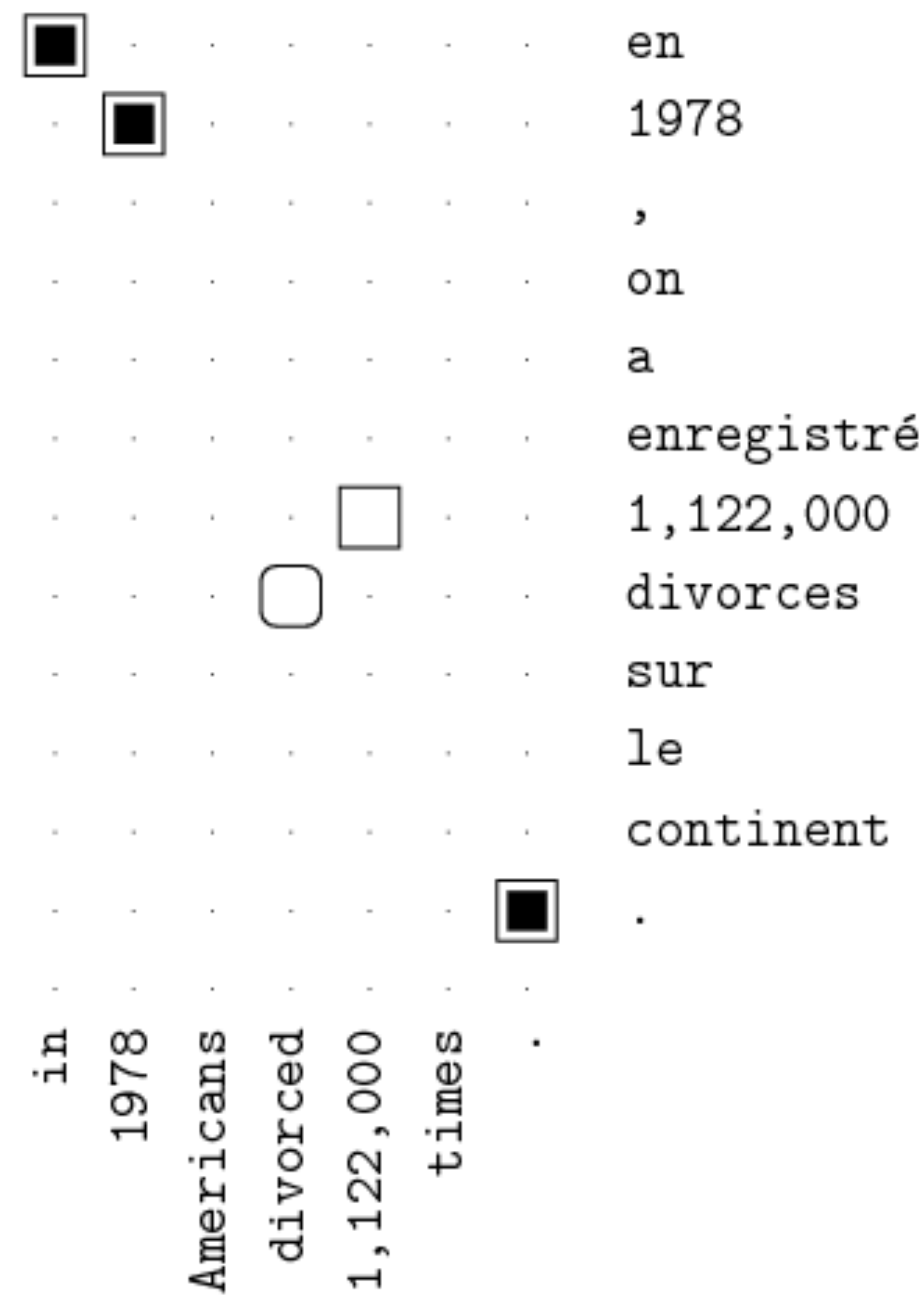
Evaluation: Alignment Error Rate

= Sure

= Possible

= Predicted

$$AER(A, S, P) = \left(1 - \frac{|A \cap S| + |A \cap P|}{|A| + |S|}\right)$$



Evaluation: Alignment Error Rate

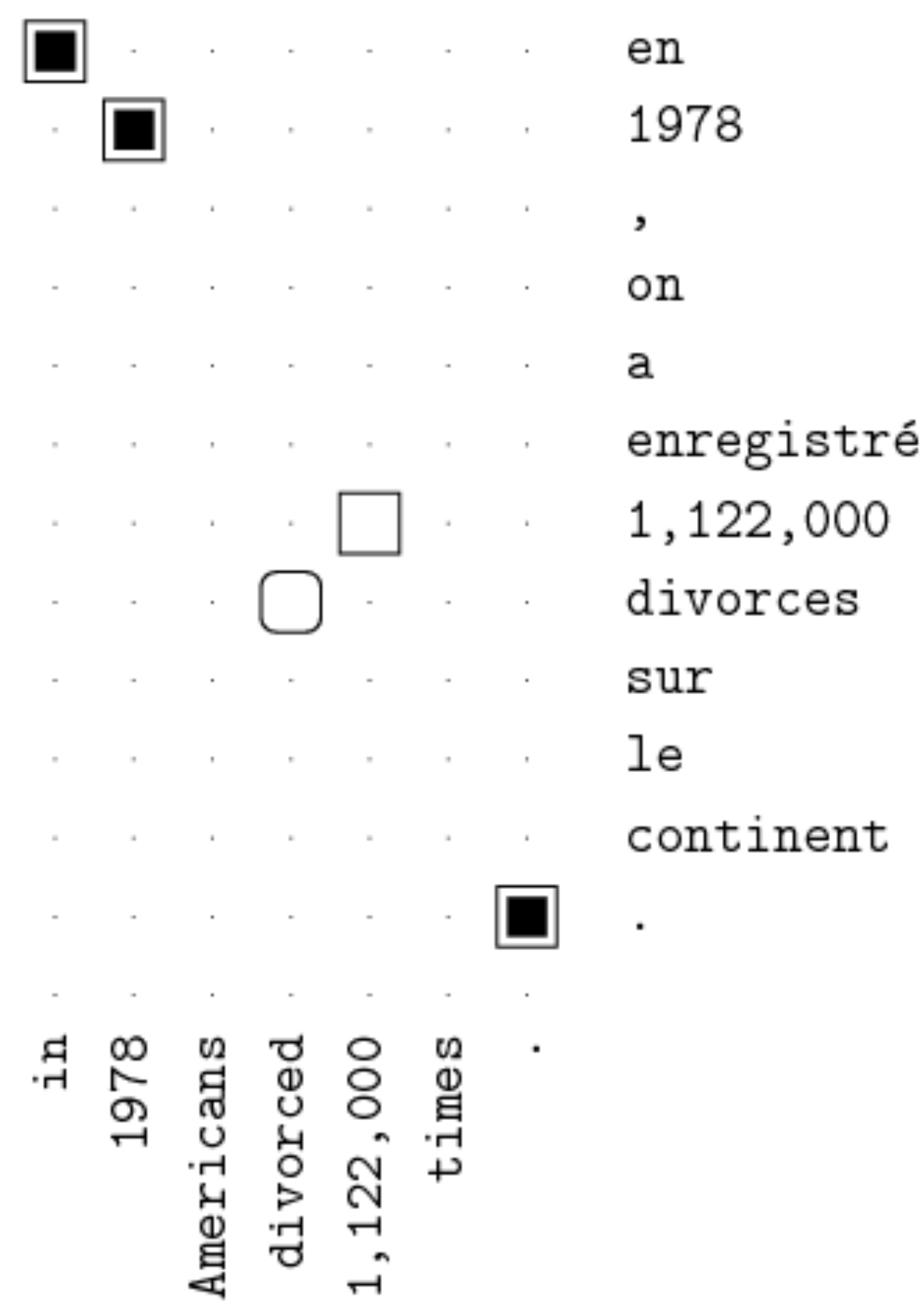
□ = Sure

▢ = Possible

■ = Predicted

$$AER(A, S, P) = \left(1 - \frac{|A \cap S| + |A \cap P|}{|A| + |S|}\right)$$

$$= \left(1 - \frac{3+3}{3+4}\right) = \frac{1}{7}$$



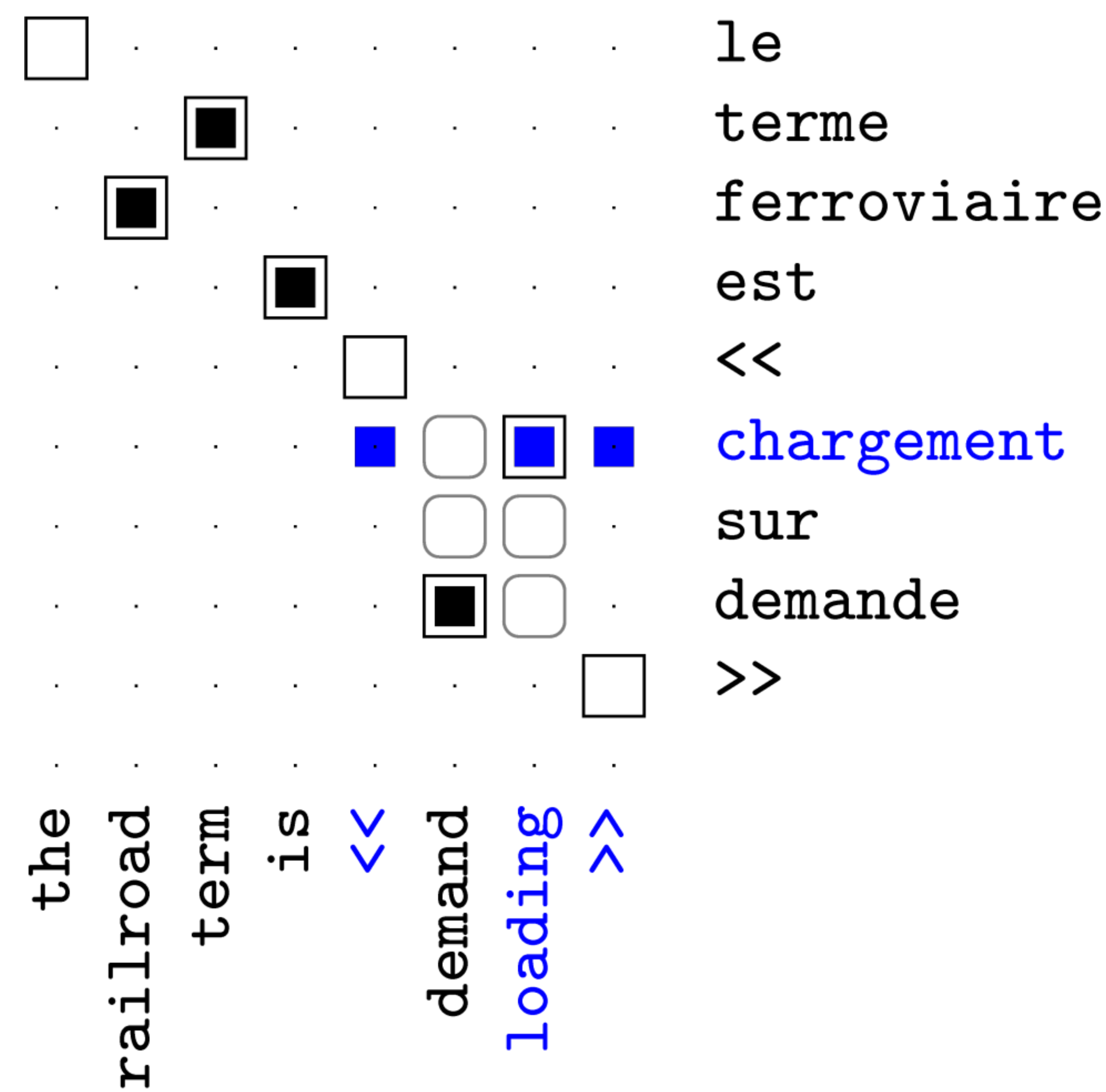
Problems with IBM Model 1

- Too many alignments to rare words (garbage collection)
- Alignments jump around all over the sentence

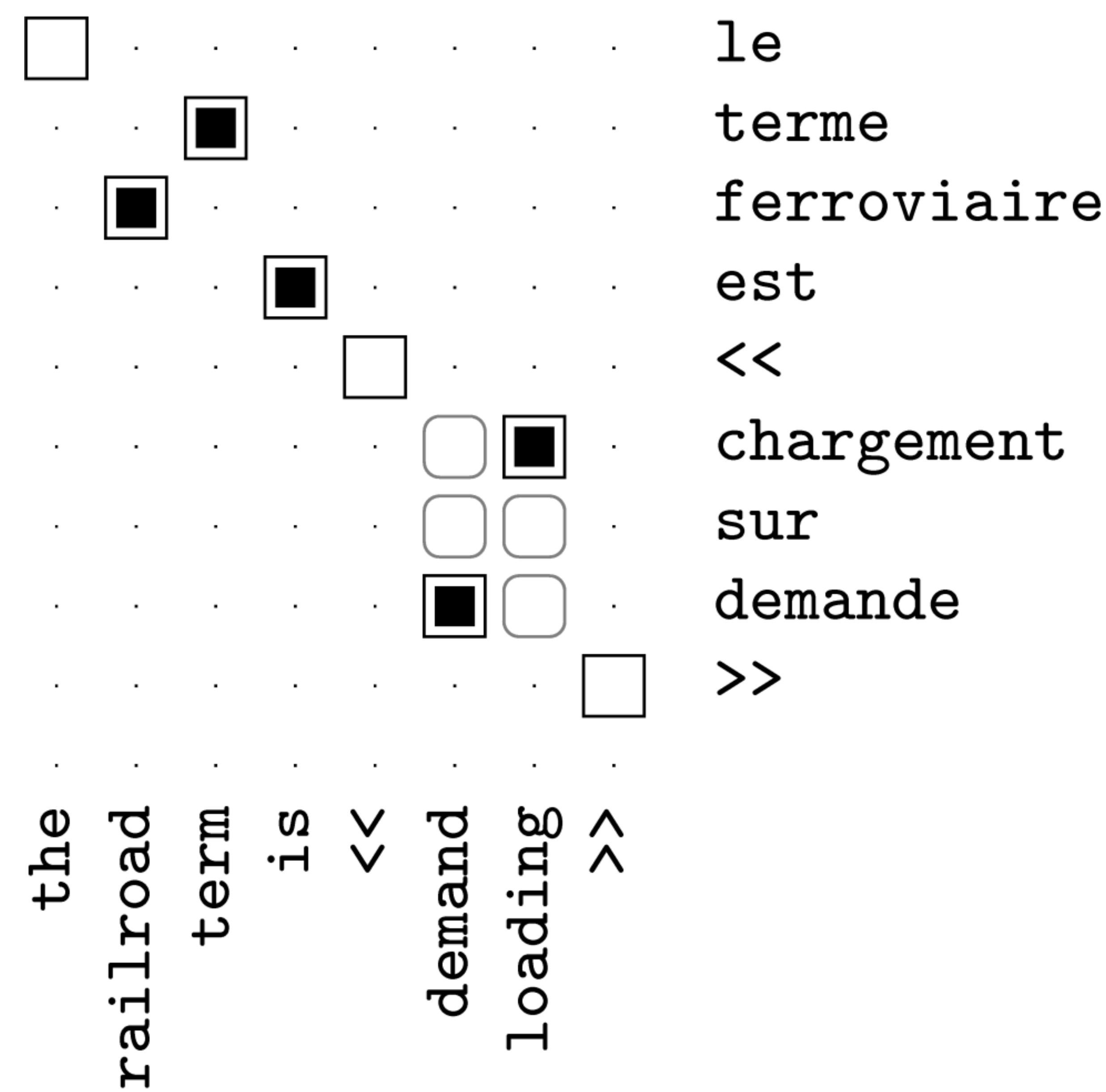
Problems with IBM Model 1

- Too many alignments to rare words (garbage collection)

- Alignments jump around all over the sentence

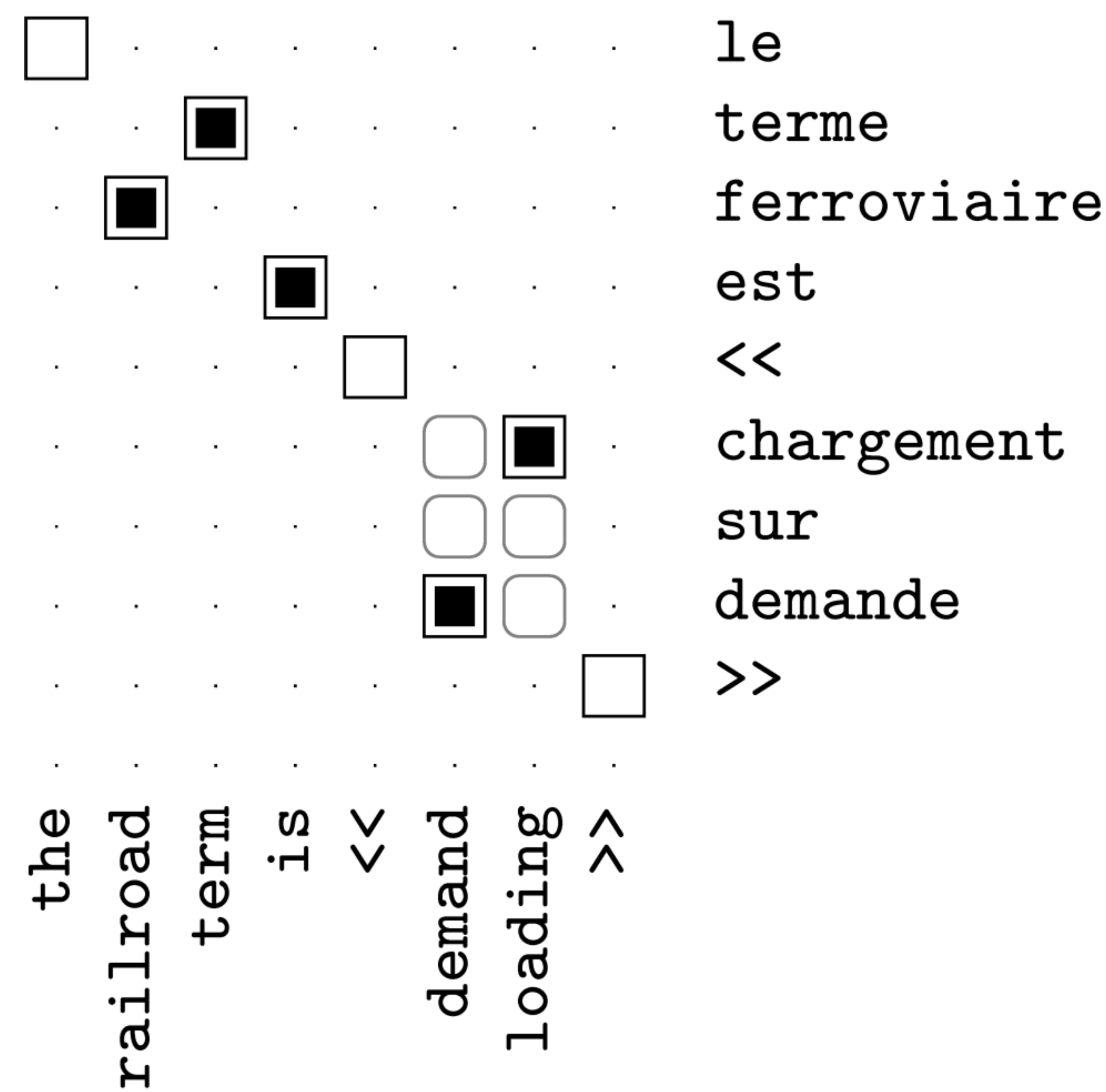


Intersected IBM Model 1



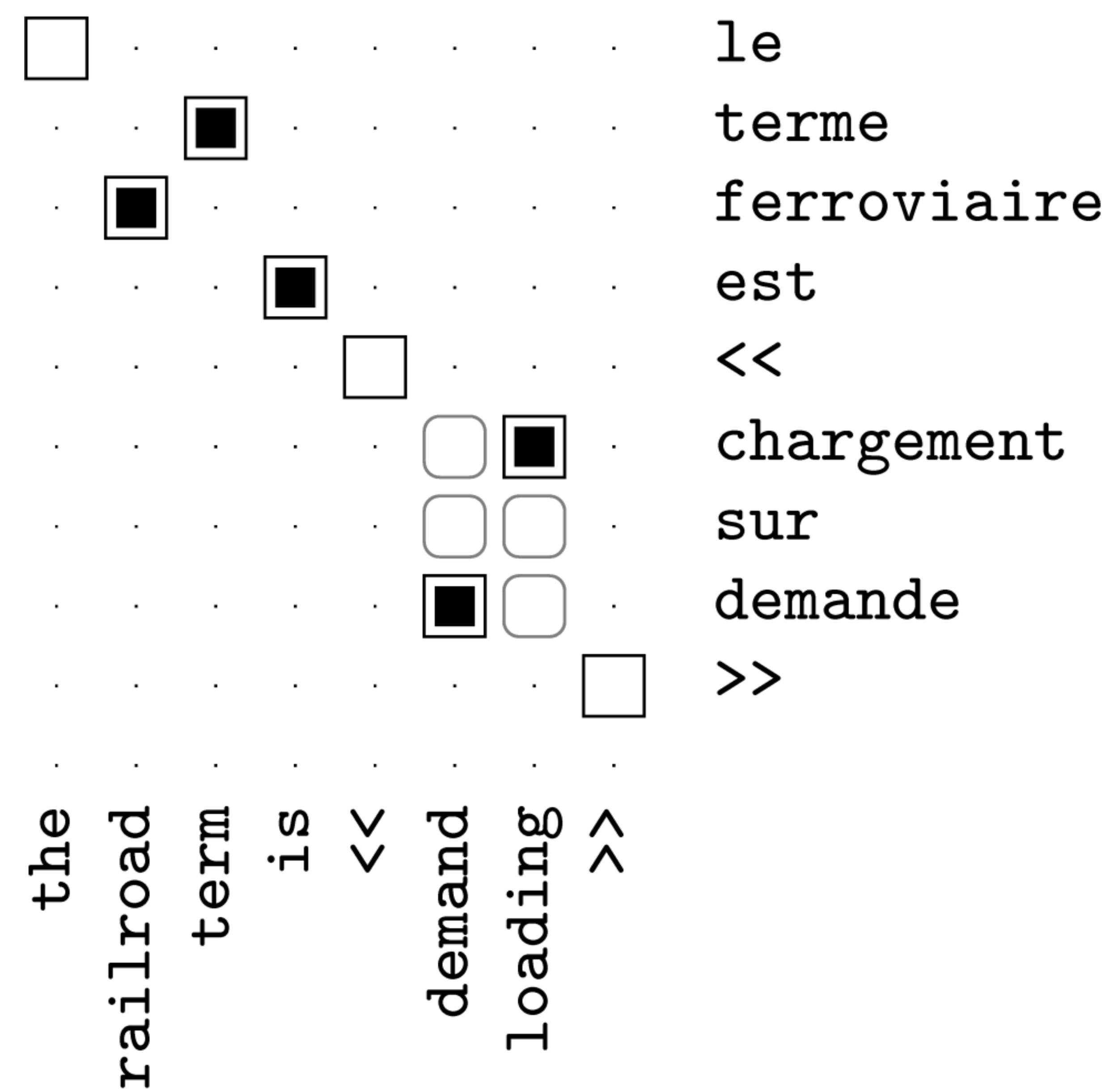
Intersected IBM Model 1

- Train Model 1 in both directions, align with each, then intersect the output (Och and Ney, ’03)
- Result is one-to-one with Viterbi alignments
- Second model filters the first, eliminating mistakes



Intersected IBM Model 1

- Train Model 1 in both directions, align with each, then intersect the output (Och and Ney, '03)
- Result is one-to-one with Viterbi alignments
- Second model filters the first, eliminating mistakes



Model	P/R	AER
Model 1 E→F	82/58	30.6
Model 1 F→E	85/58	28.7
Model 1 AND	96/46	34.8

Joint Training for IBM Model 1

- We can intersect model predictions during training as well
- Modified alignment posterior: $P_{e \to f}(a_j = i|\mathbf{e}, \mathbf{f}) \cdot P_{f \to e}(a_i = j|\mathbf{e}, \mathbf{f})$
- Models are forced to agree as they select parameters
- Same precision benefits, but higher recall from more agreement

Model	P/R	AER
Model 1 E→F	82/58	30.6
Model 1 F→E	85/58	28.7
Model 1 AND	96/46	34.8
Model 1 INT	93/69	19.5

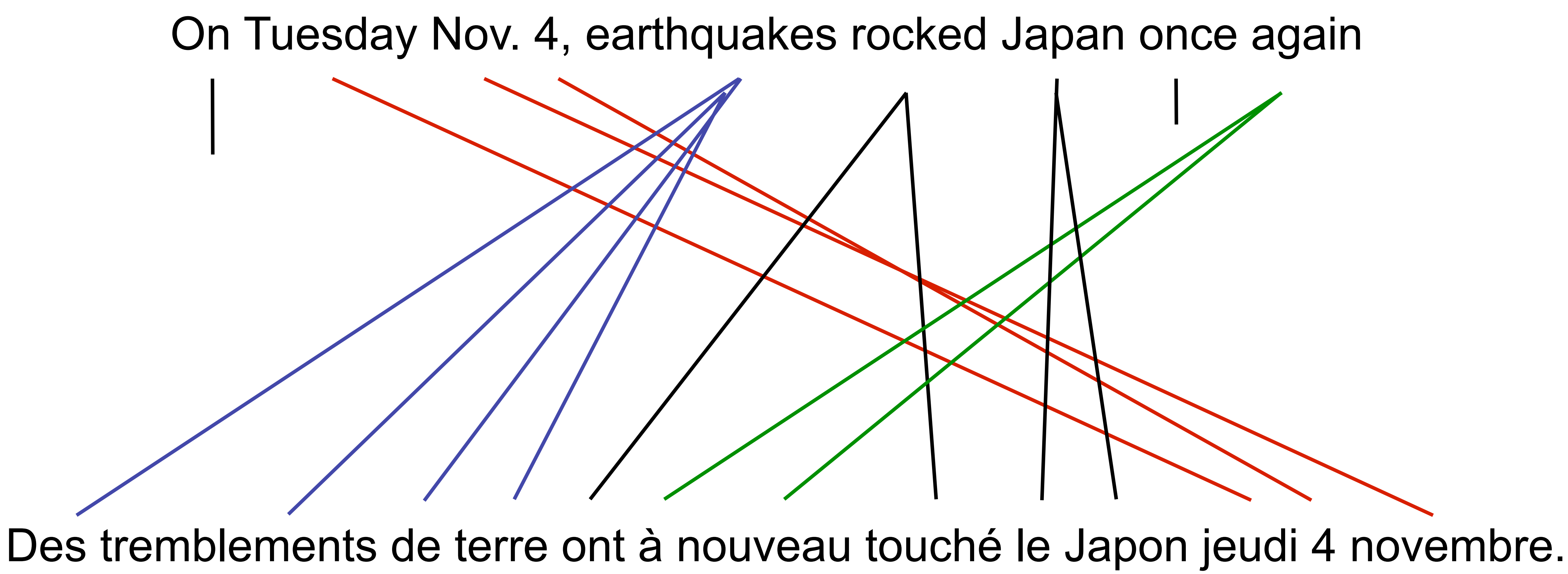
IBM Model 2

- Words at the beginning of sentences should align
- Words at the end of sentences should align
- Alignment probability depends on position

$$
\begin{aligned}
P(f,a|e) &= \prod_{j=1}^{J} P(a_j = i|I,J) \cdot P(f_j|e_i) \\
&\propto exp(-\alpha \left| a_i - i\frac{I}{J} \right|) \cdot P(f_j|e_i)
\end{aligned}
$$

Phrase Movement

Absolute position distortion isn't quite right

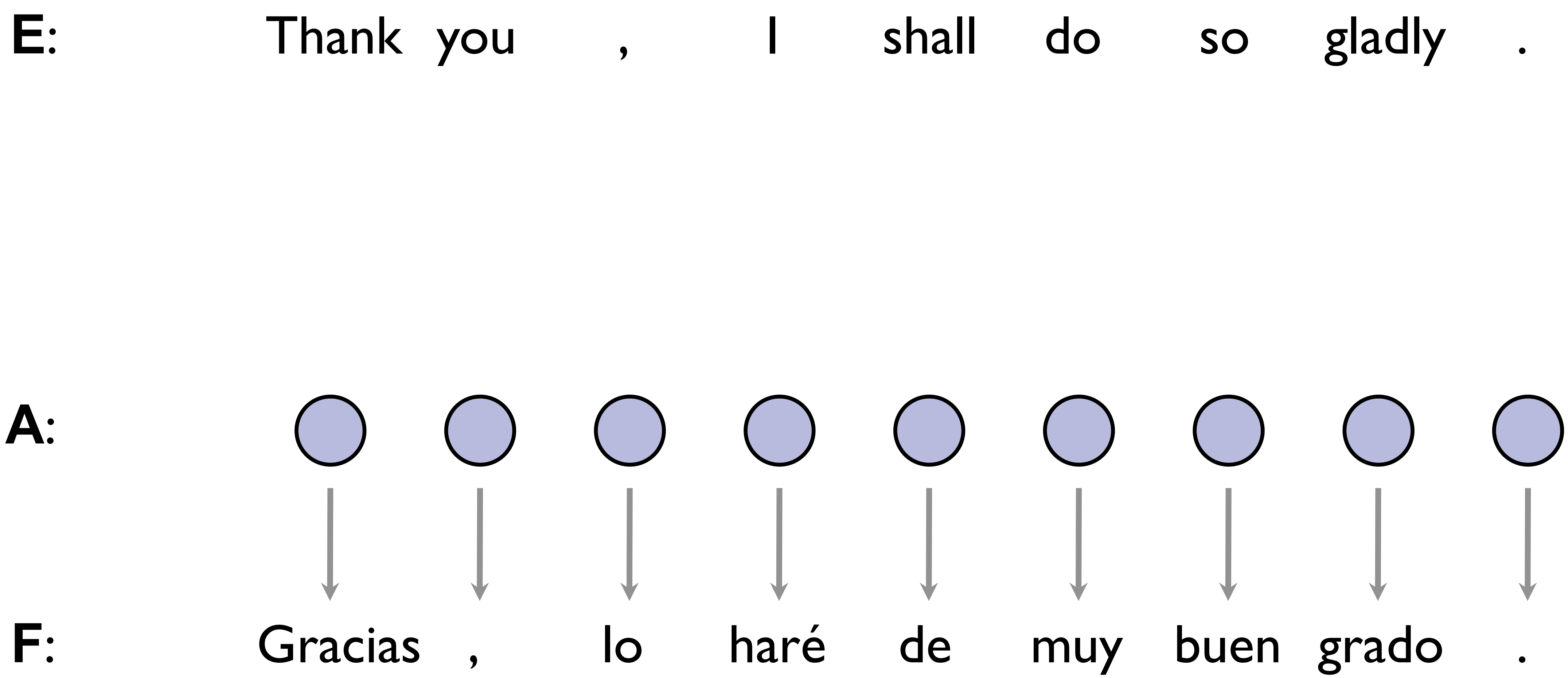


IBM Models 1/2

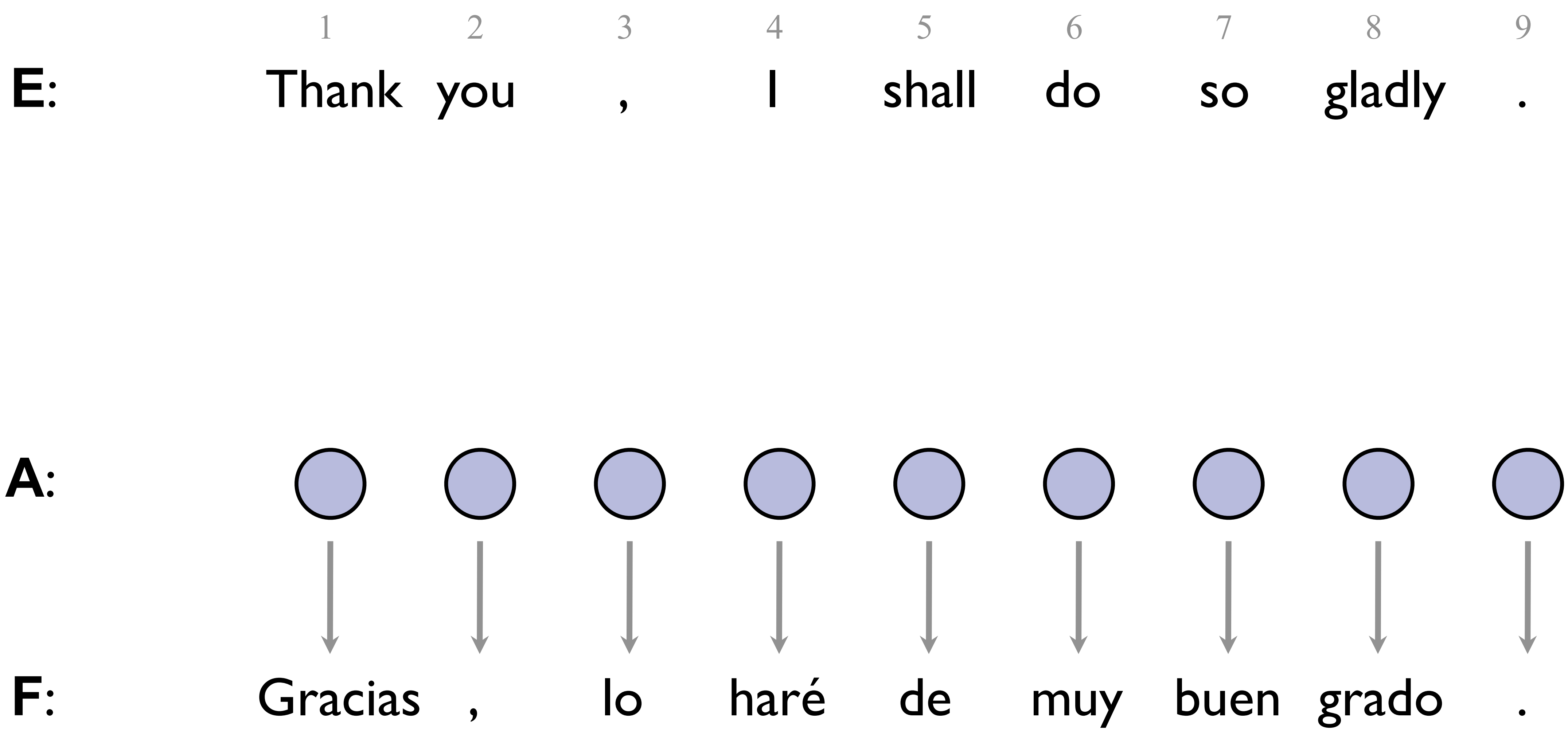
E: Thank you , I shall do so gladly .

F: Gracias , lo haré de muy buen grado .

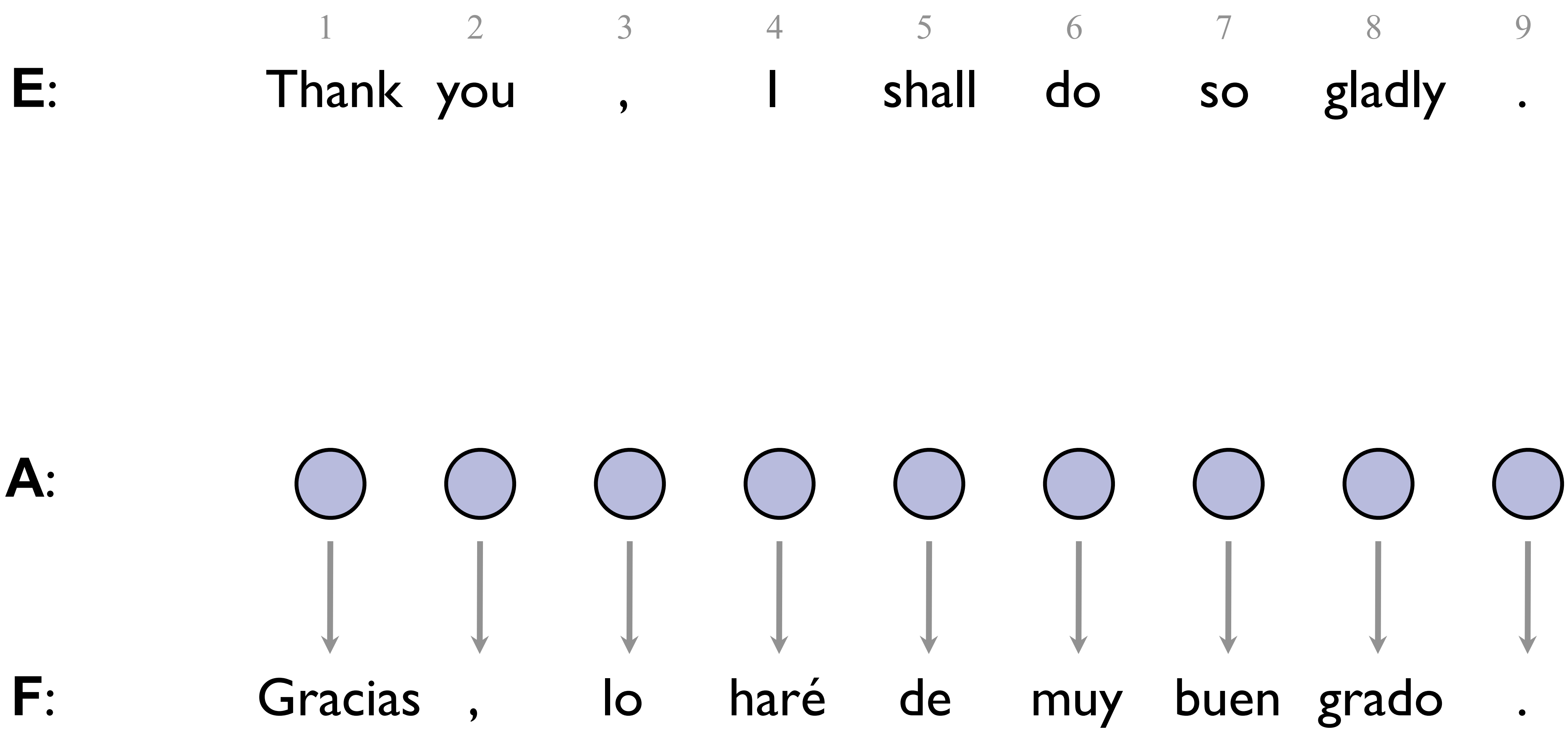
IBM Models 1/2



IBM Models 1/2



IBM Models 1/2

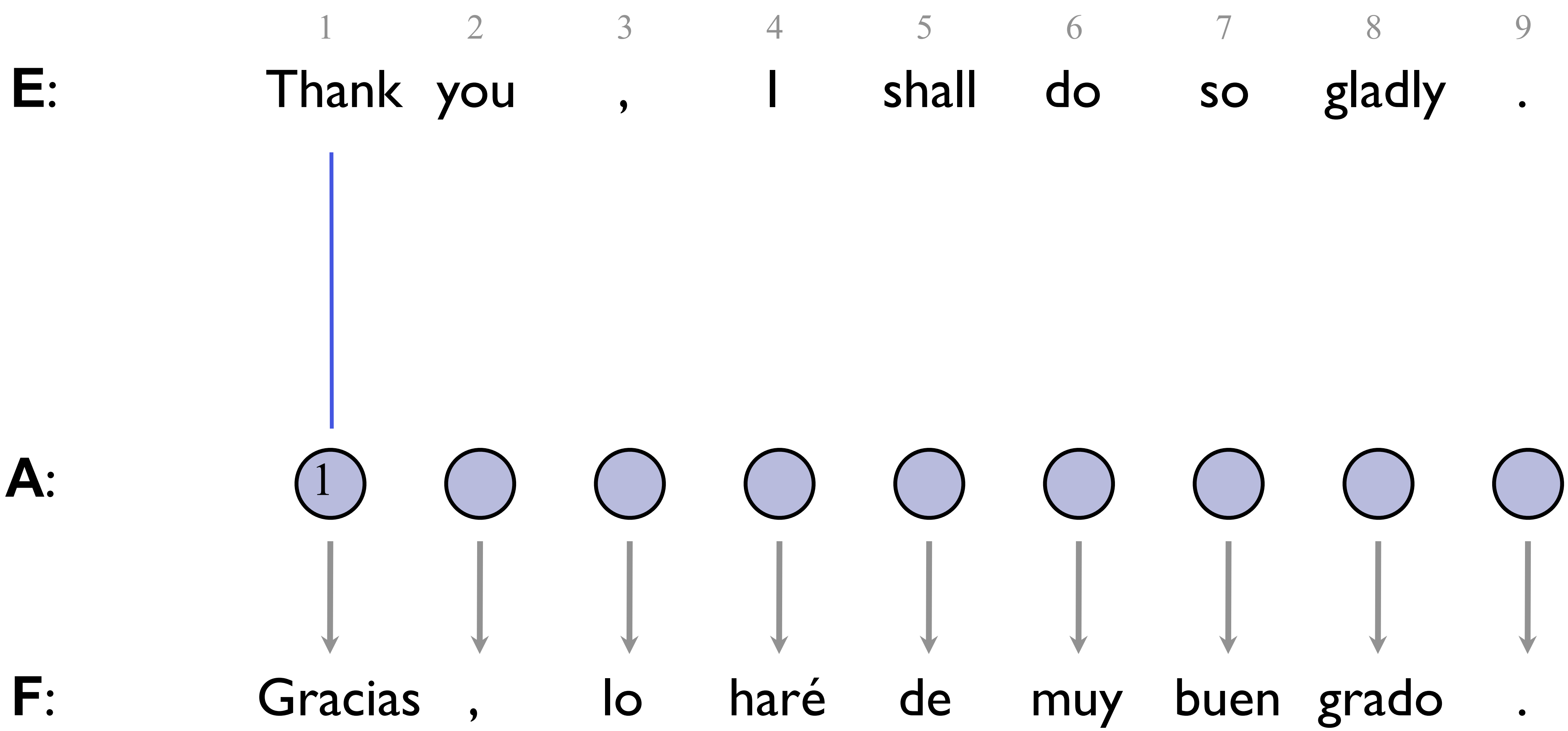


Model Parameters

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

Transitions: $P(A_2 = 3 \mid I, J)$

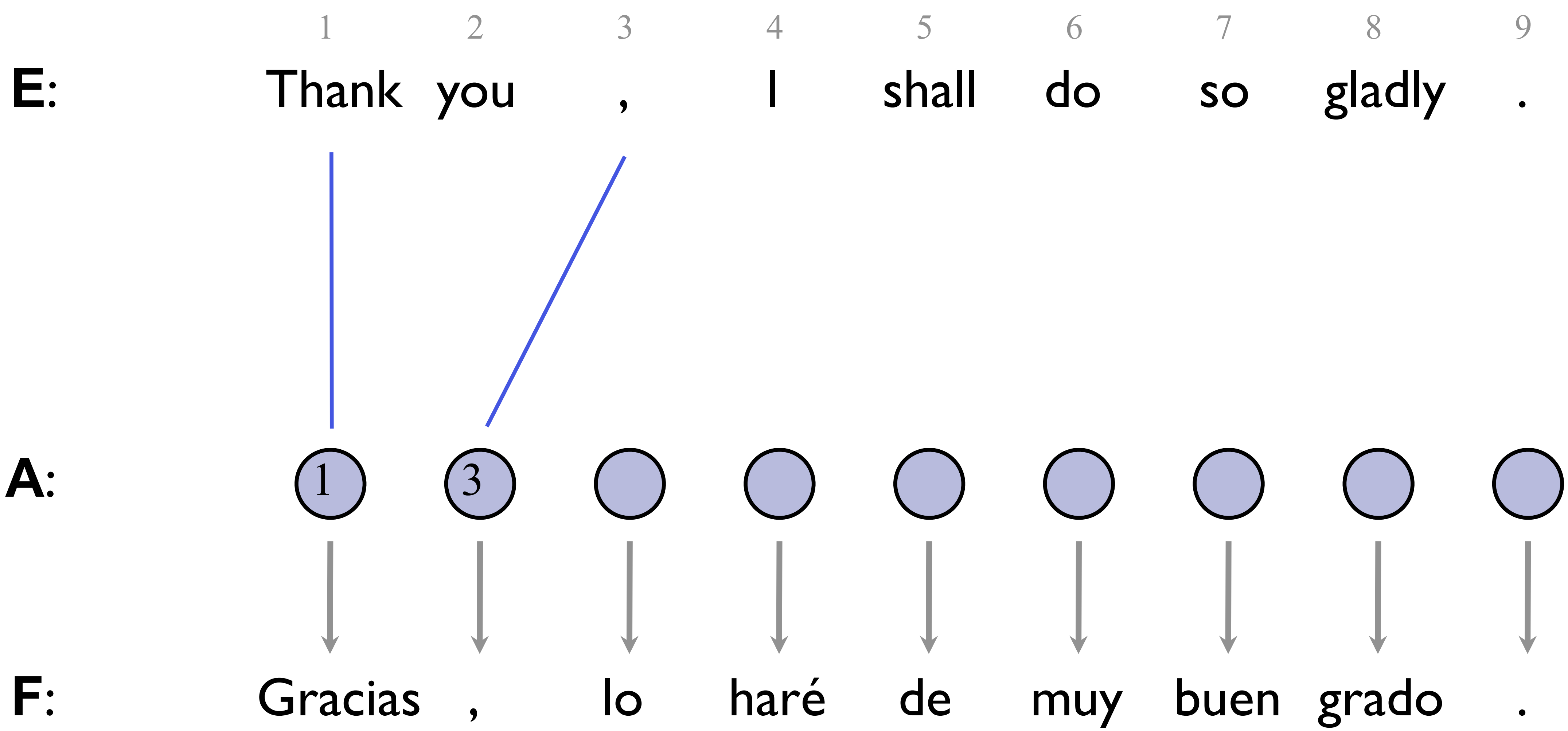
IBM Models 1/2



Model Parameters

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

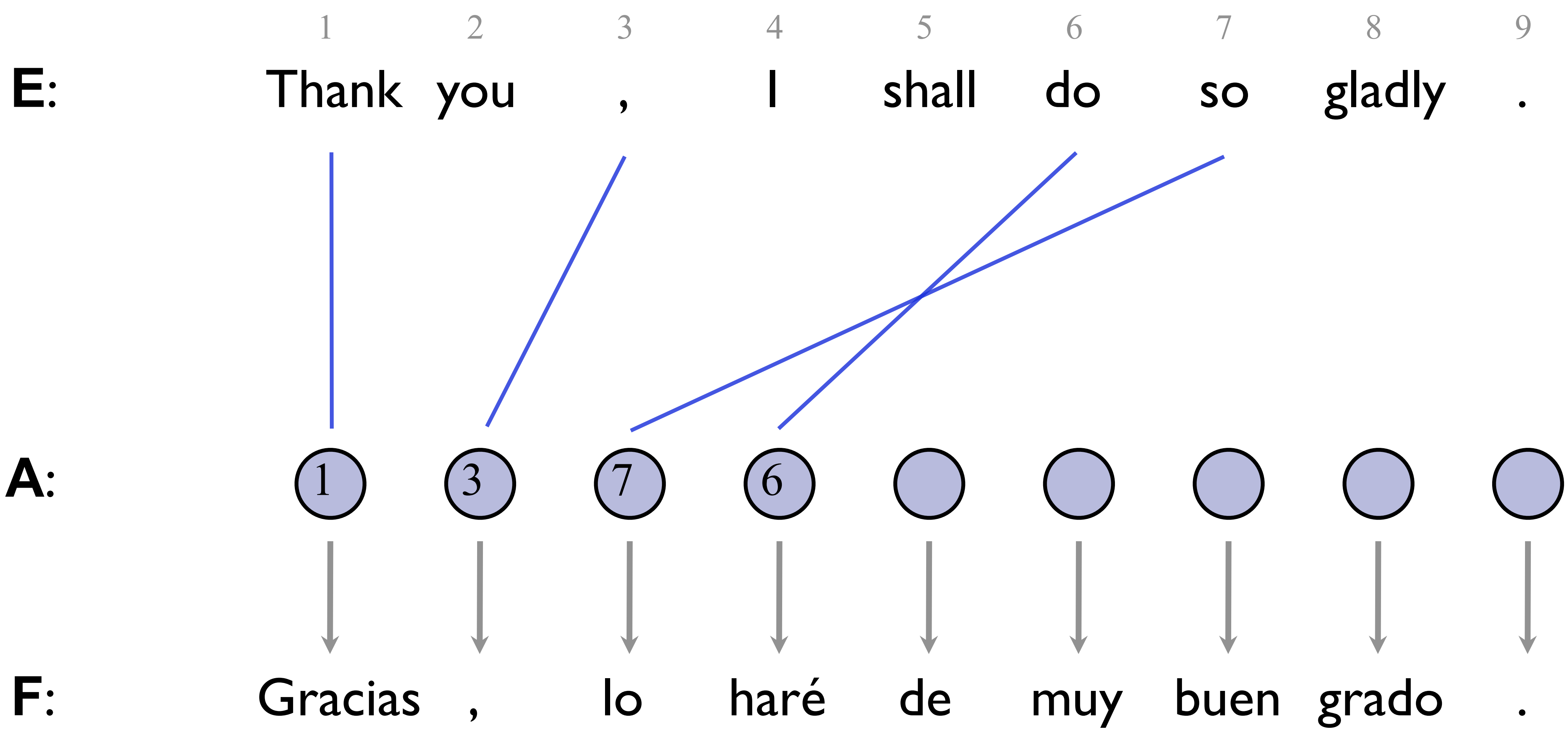
IBM Models 1/2



Model Parameters

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

IBM Models 1/2

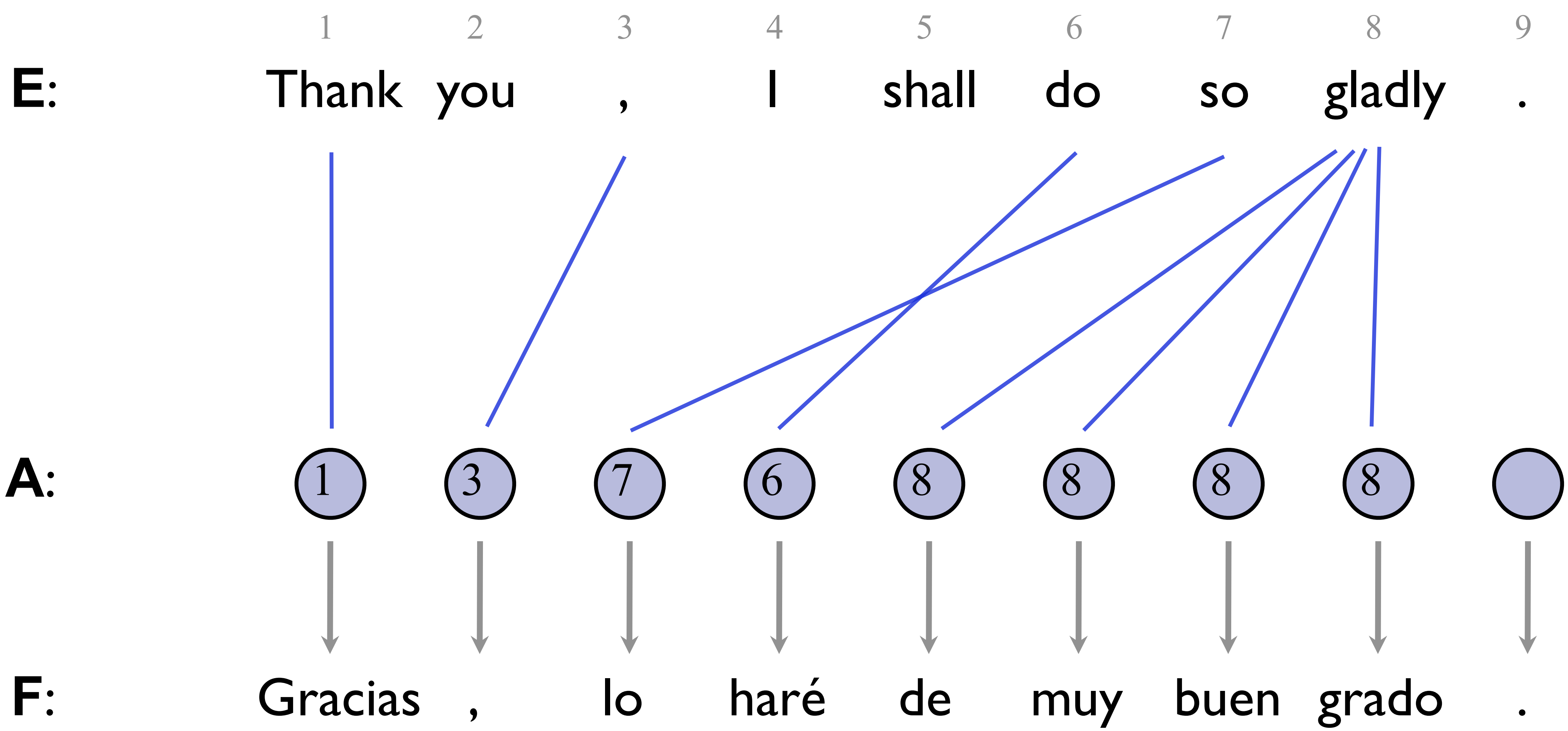


Model Parameters

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

Transitions: $P(A_2 = 3 \mid I, J)$

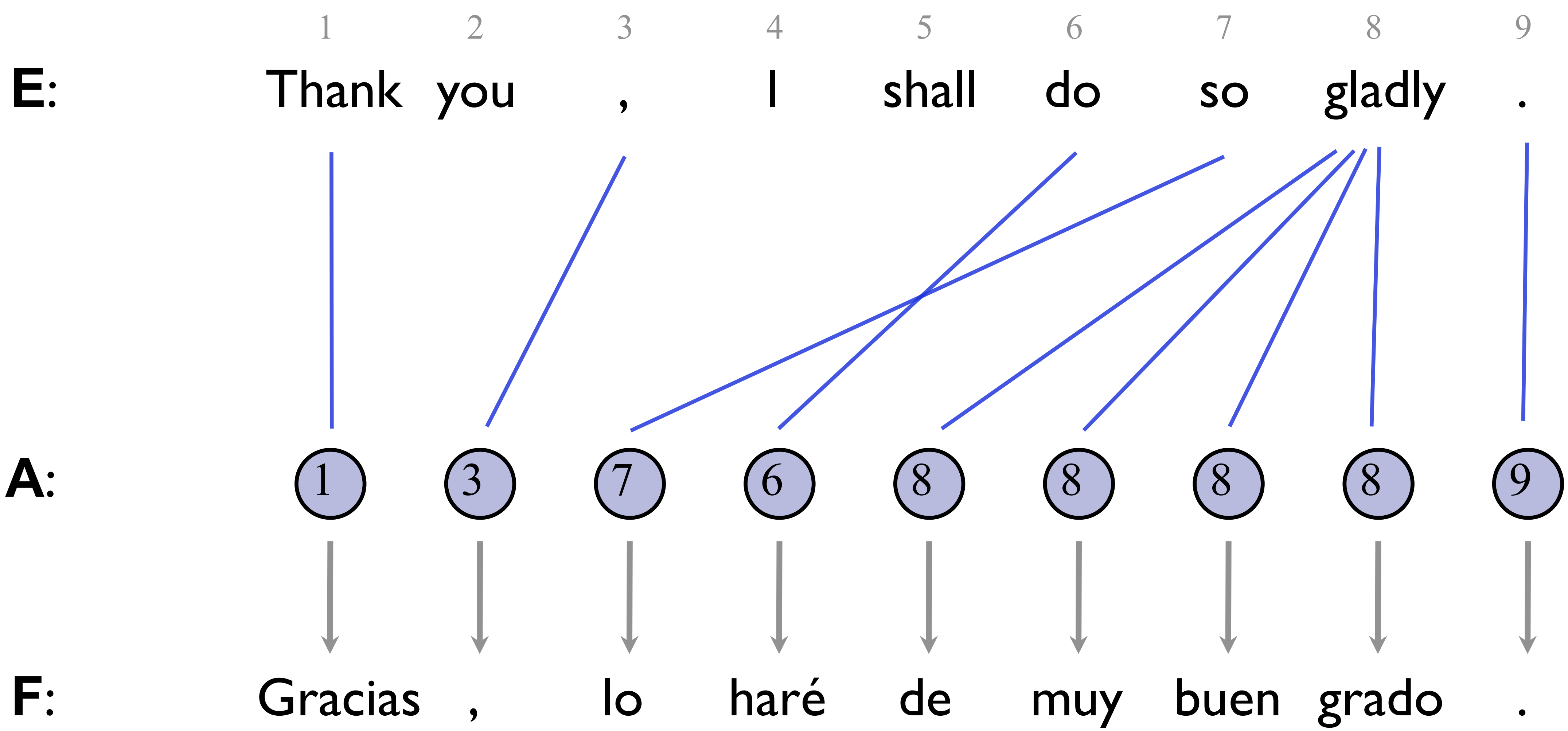
IBM Models 1/2



Model Parameters

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

IBM Models 1/2



Model Parameters

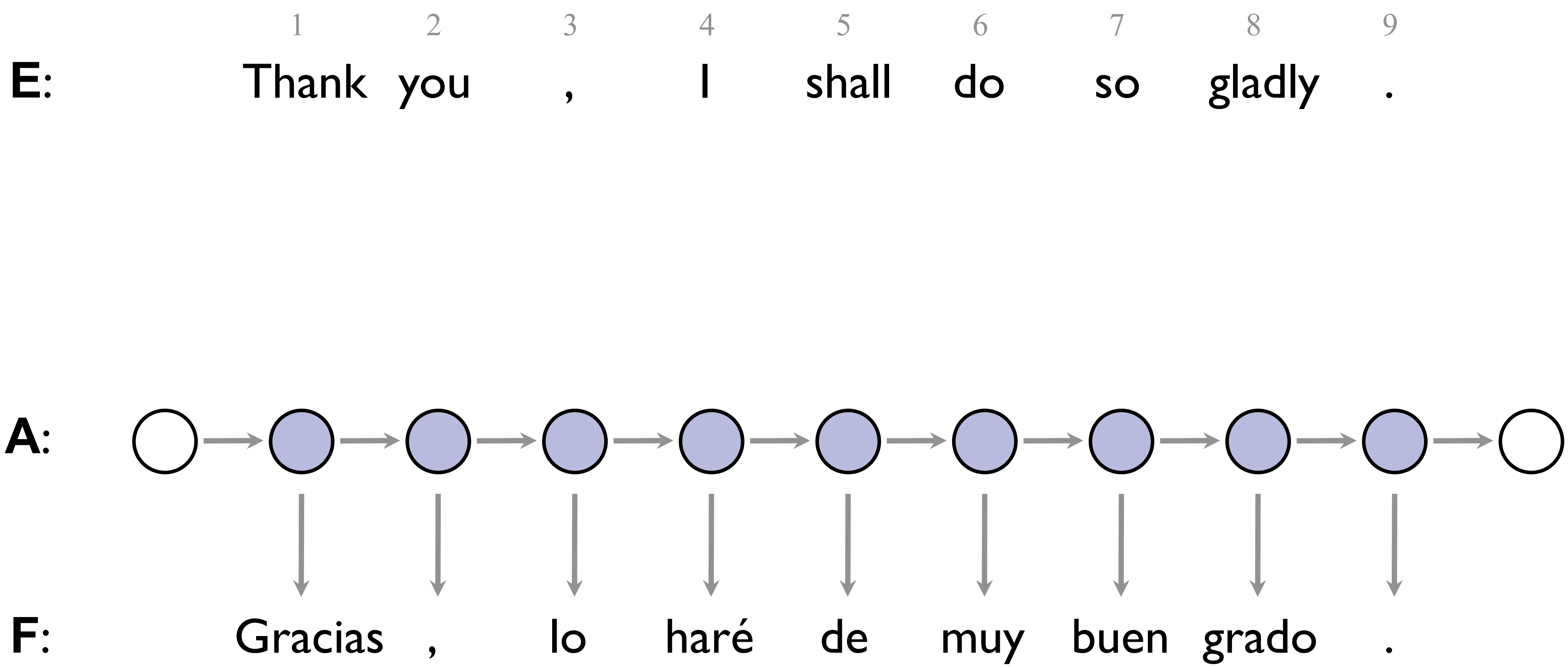
Emissions: $P(F_1 = \text{Gracias} \mid E_{A_1} = \text{Thank})$ *Transitions*: $P(A_2 = 3 \mid I, J)$

The HMM Model

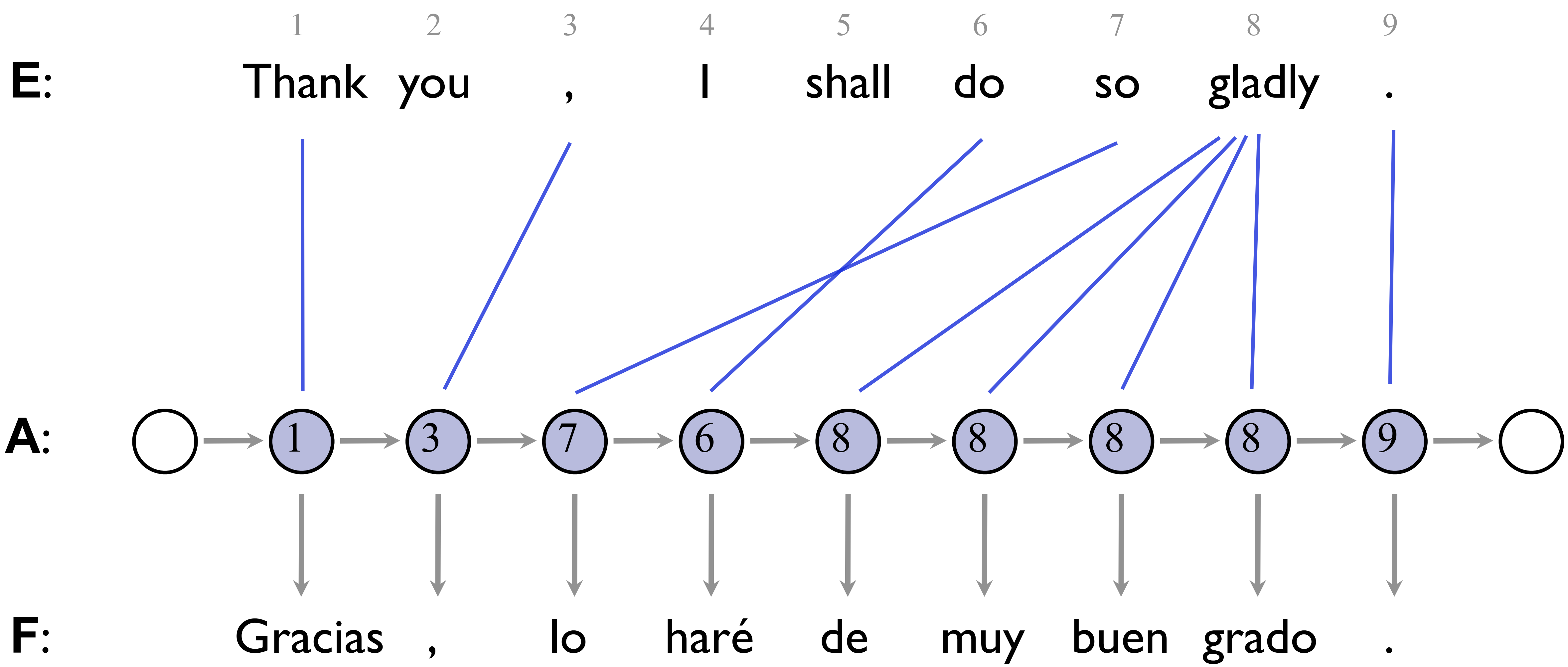
	1	2	3	4	5	6	7	8	9
E:	Thank	you	,	I	shall	do	so	gladly	.

F: Gracias , lo haré de muy buen grado .

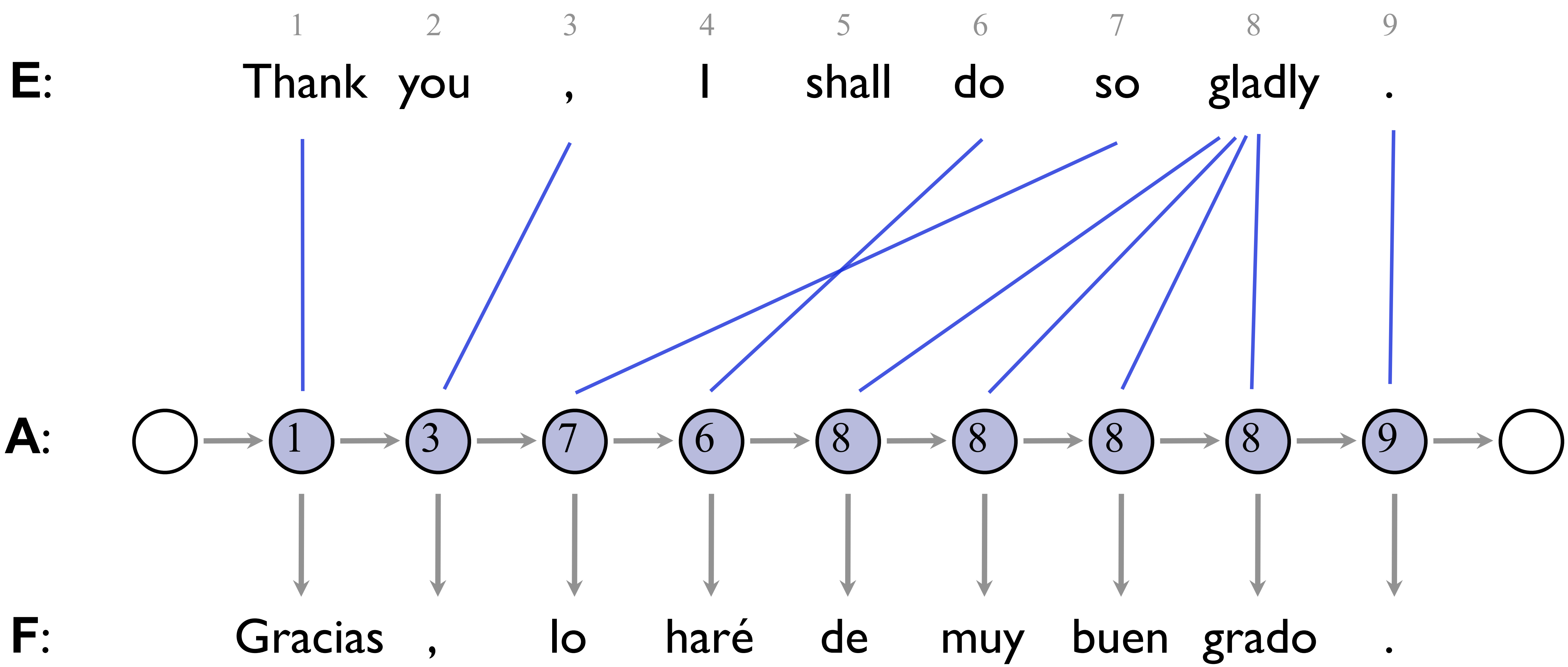
The HMM Model



The HMM Model



The HMM Model



Model Parameters

Emissions: $P(F_1 = \text{Gracias} \mid E_{A_1} = \text{Thank})$ *Transitions*: $P(A_2 = 3 \mid A_1 = 1)$

The HMM Model

- Model 2 preferred global monotonicity
- We want local monotonicity (small jumps)
- HMM model (Vogel et al 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

The HMM Model

- Model 2 preferred global monotonicity
- We want local monotonicity (small jumps)
- HMM model (Vogel et al 96)

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

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

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

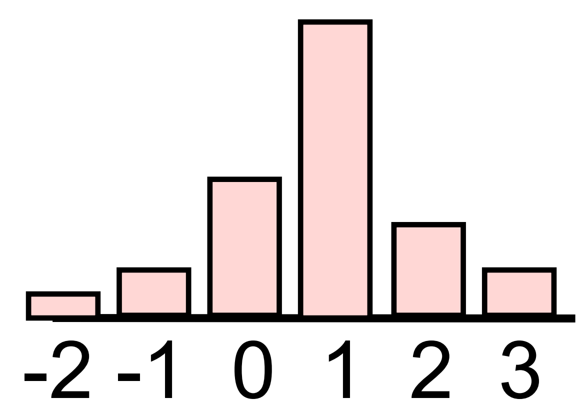
The HMM Model

- Model 2 preferred global monotonicity
- We want local monotonicity (small jumps)
- HMM model (Vogel et al 96)

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

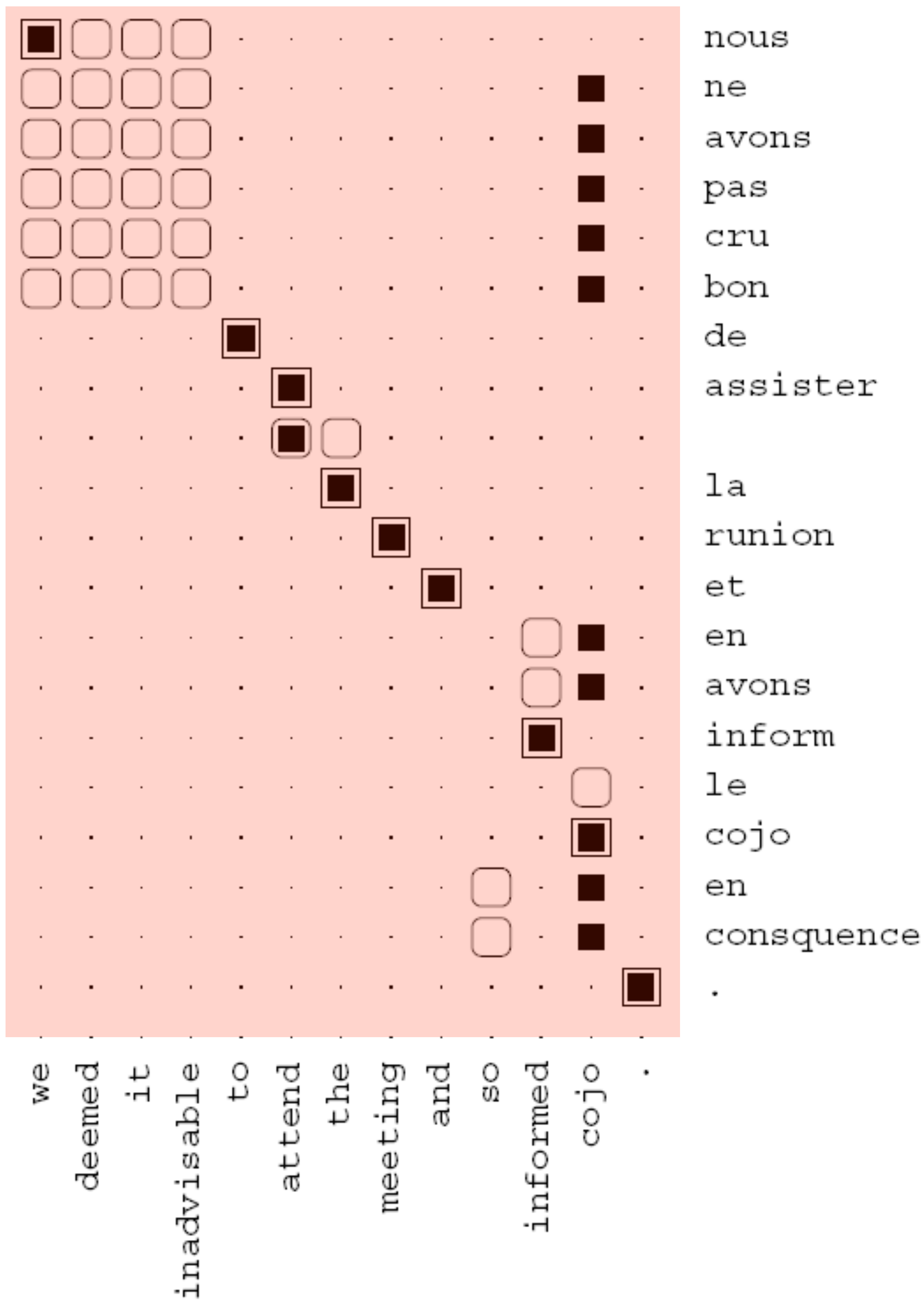
$$P(f, a|e) = \prod_j P(a_j|a_{j-1}) P(f_j|e_i)$$

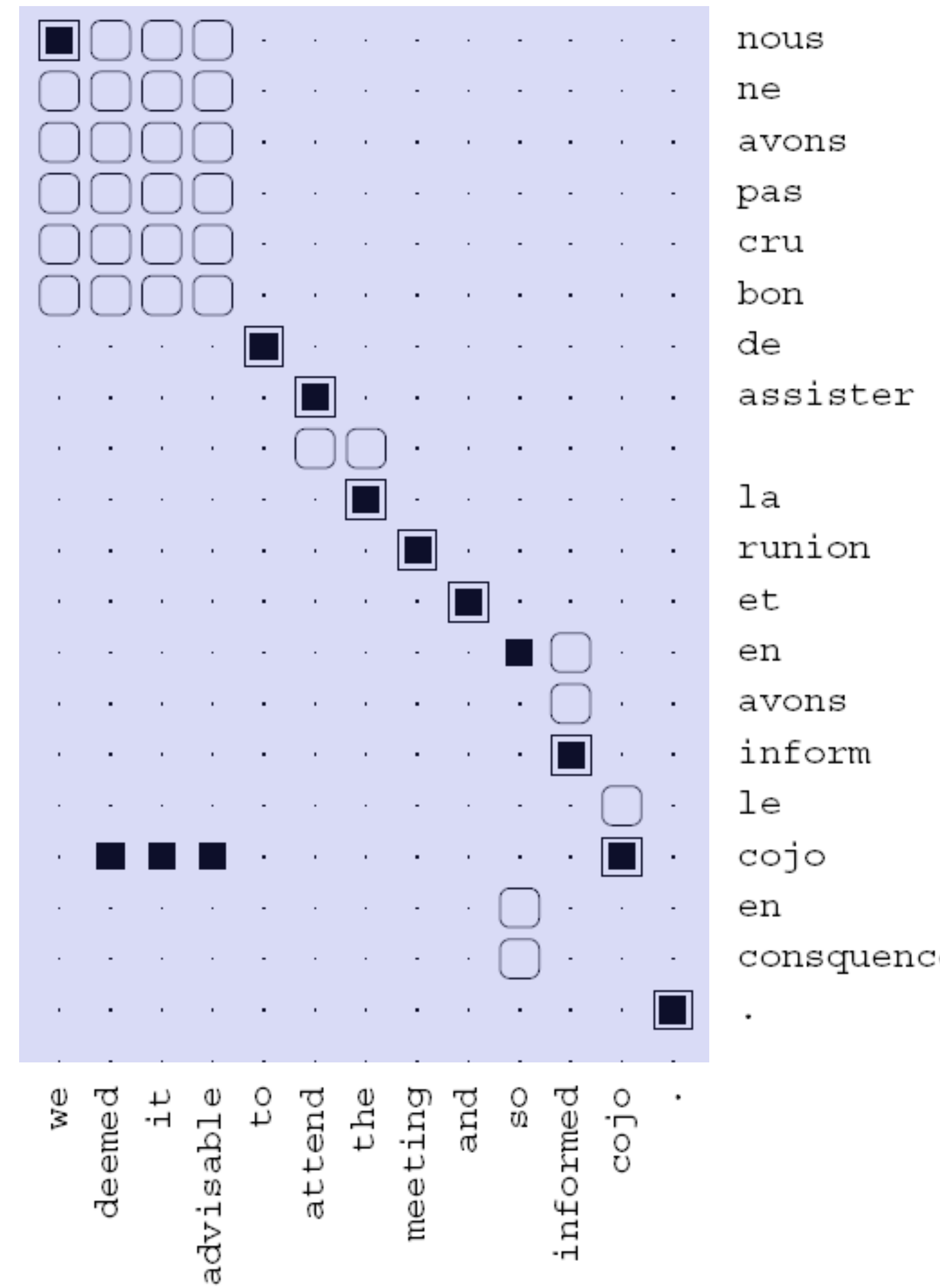
$$P(a_j - a_{j-1})$$



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

HMM Examples



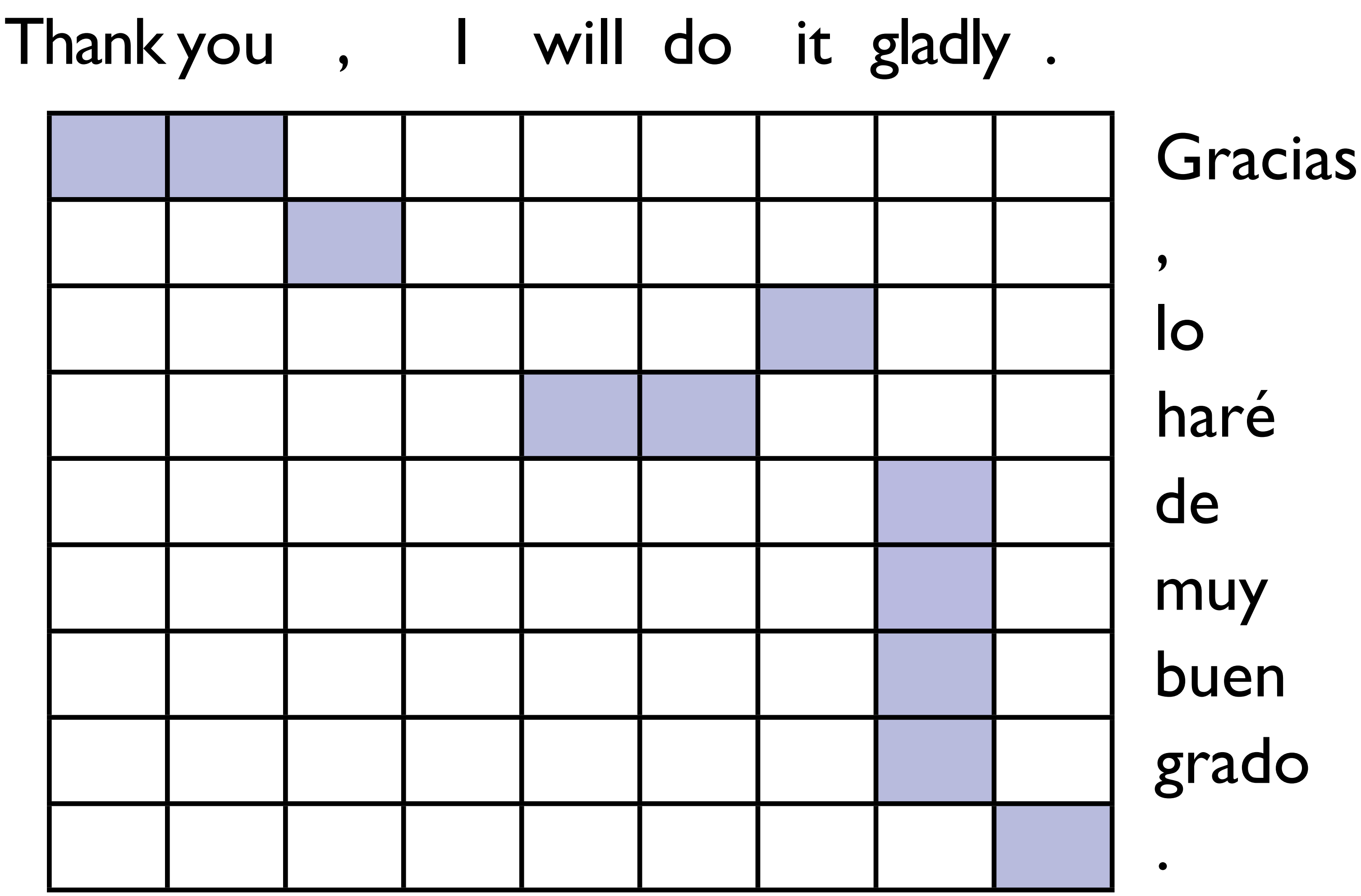


AER for HMMs

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

Estimating Rule Parameters from Words

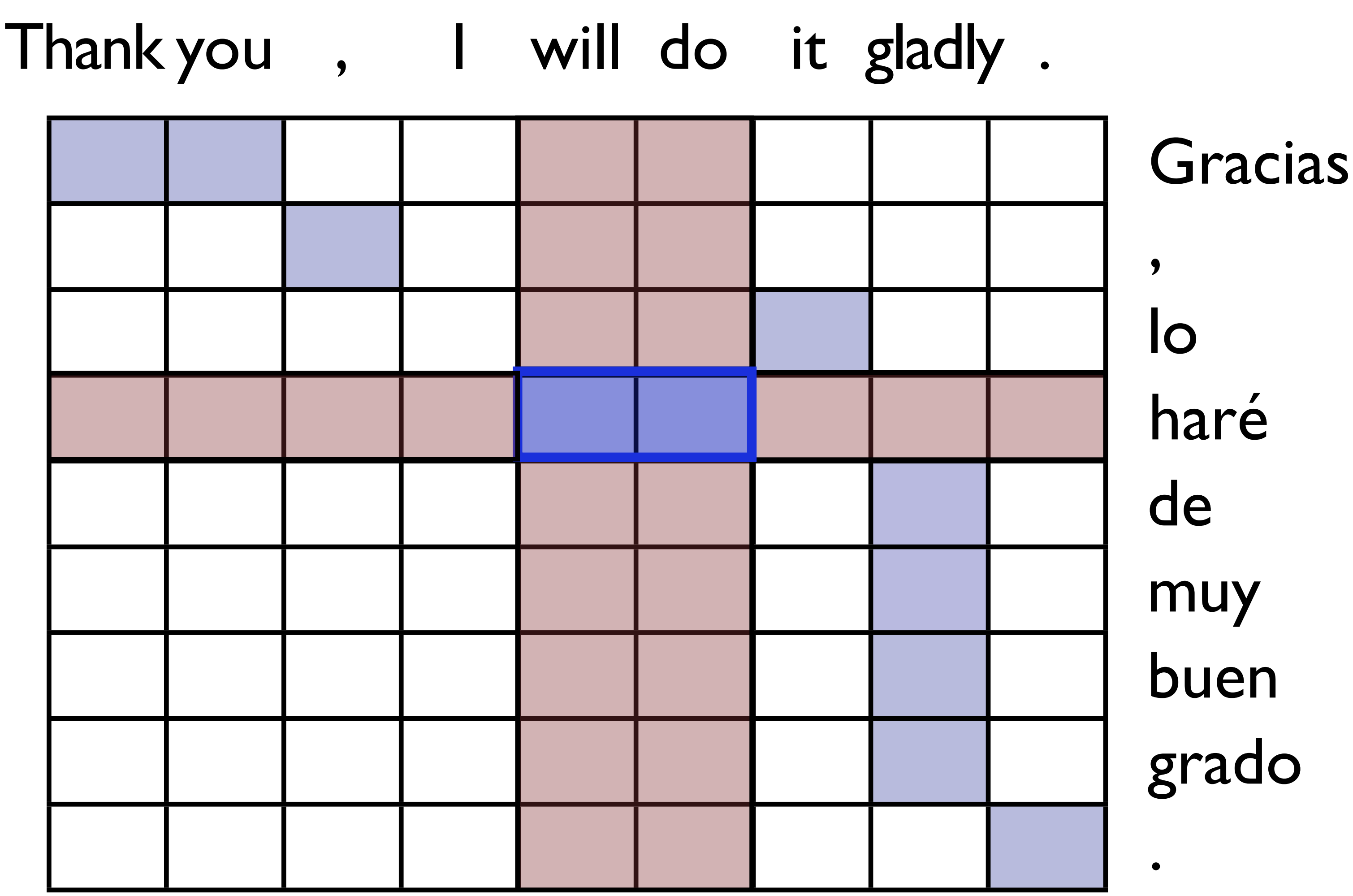
Word Aligned Sentence Pair



Grammar Rules

Estimating Rule Parameters from Words

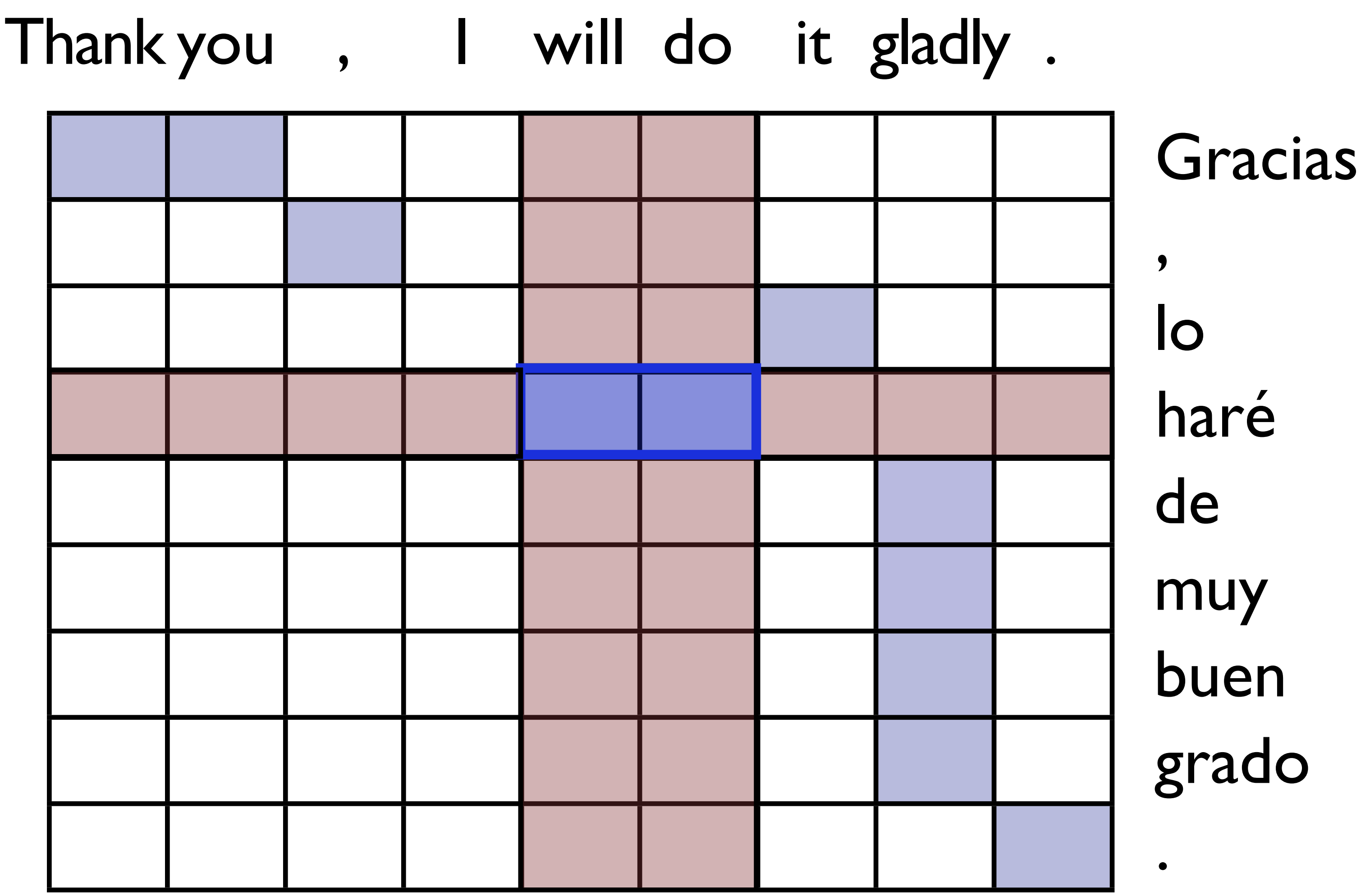
Word Aligned Sentence Pair



Grammar Rules

Estimating Rule Parameters from Words

Word Aligned Sentence Pair

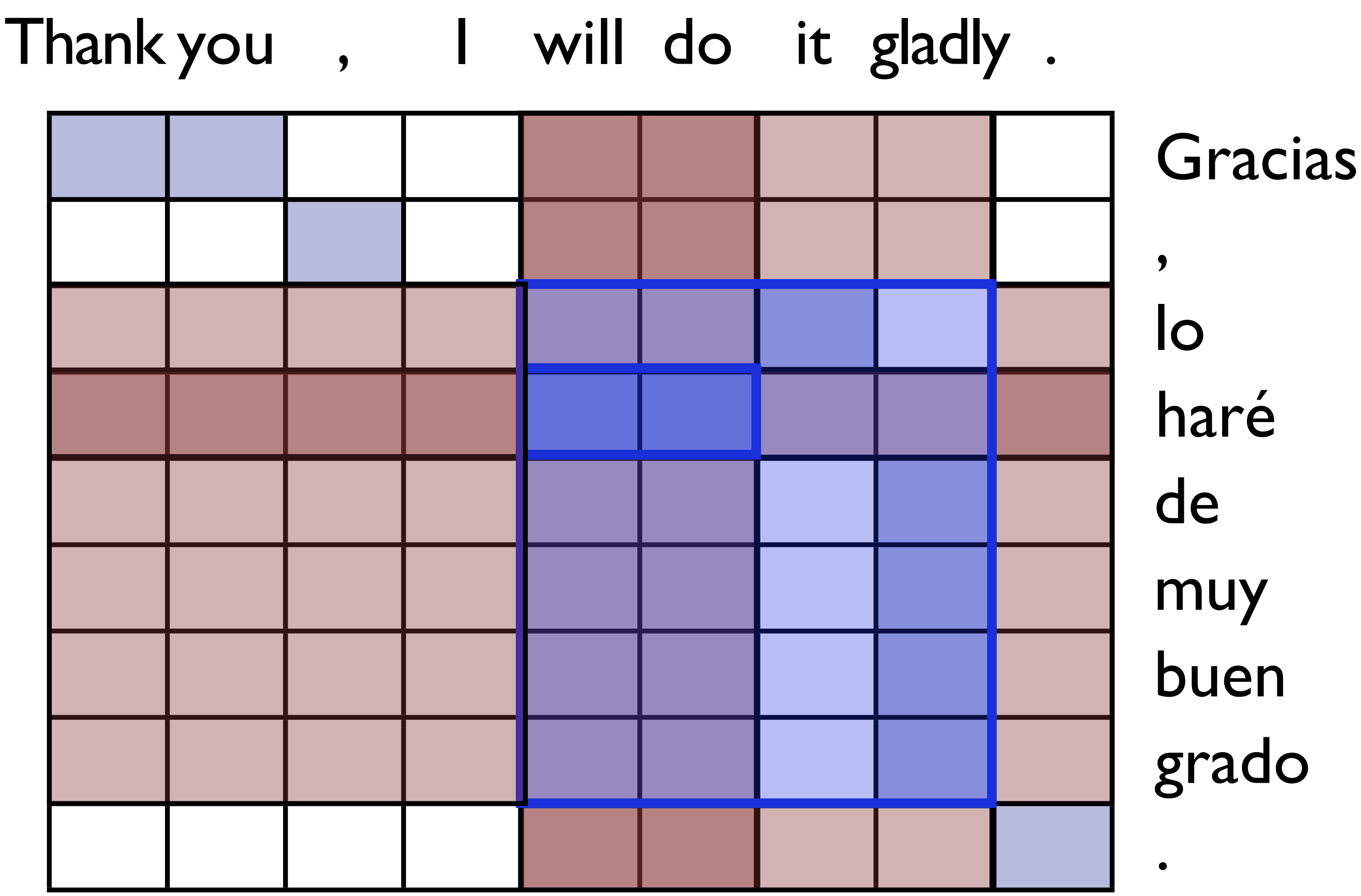


Grammar Rules

⟨haré ; will do⟩

Estimating Rule Parameters from Words

Word Aligned Sentence Pair

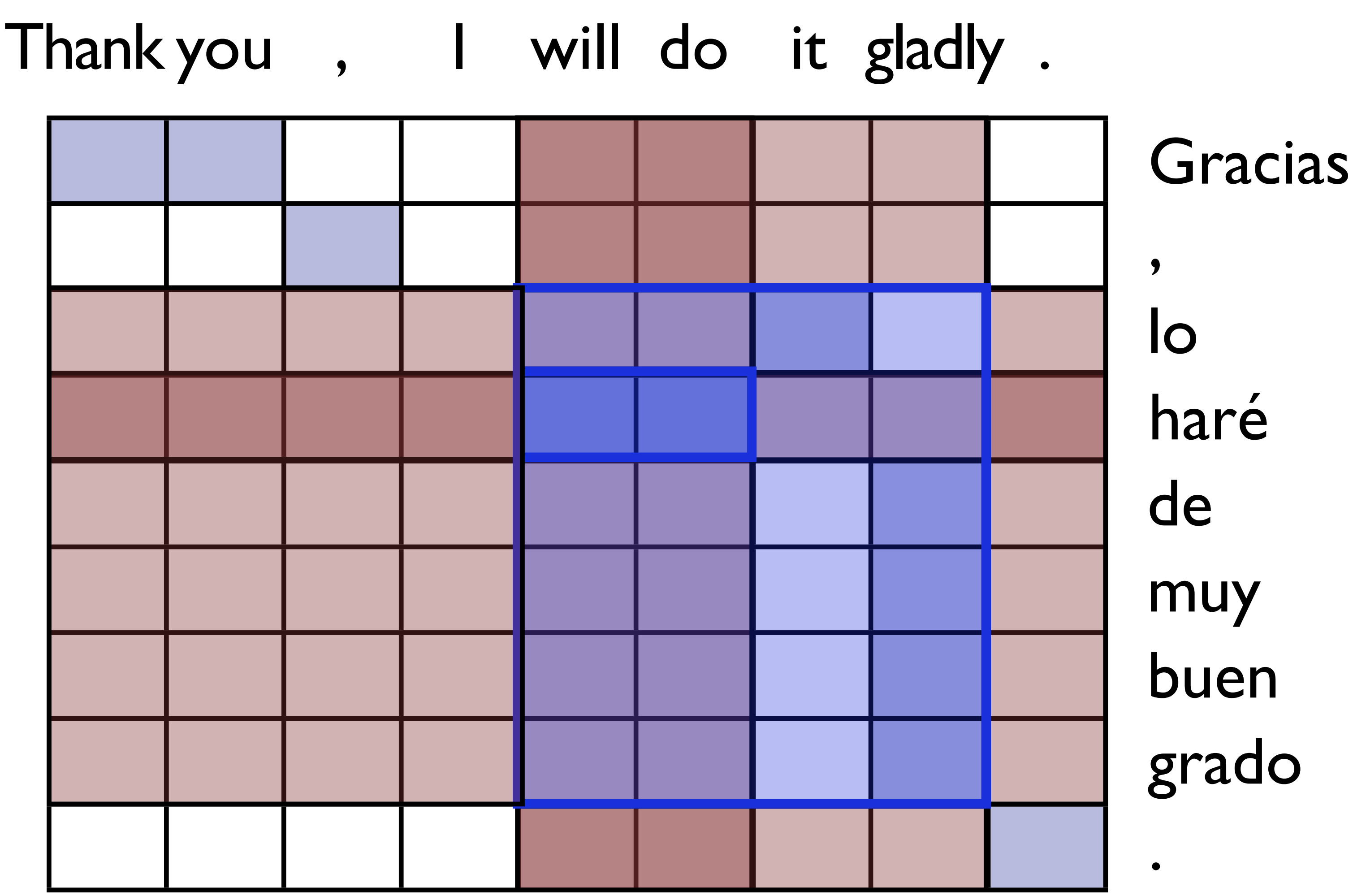


Grammar Rules

⟨haré ; will do⟩

Estimating Rule Parameters from Words

Word Aligned Sentence Pair



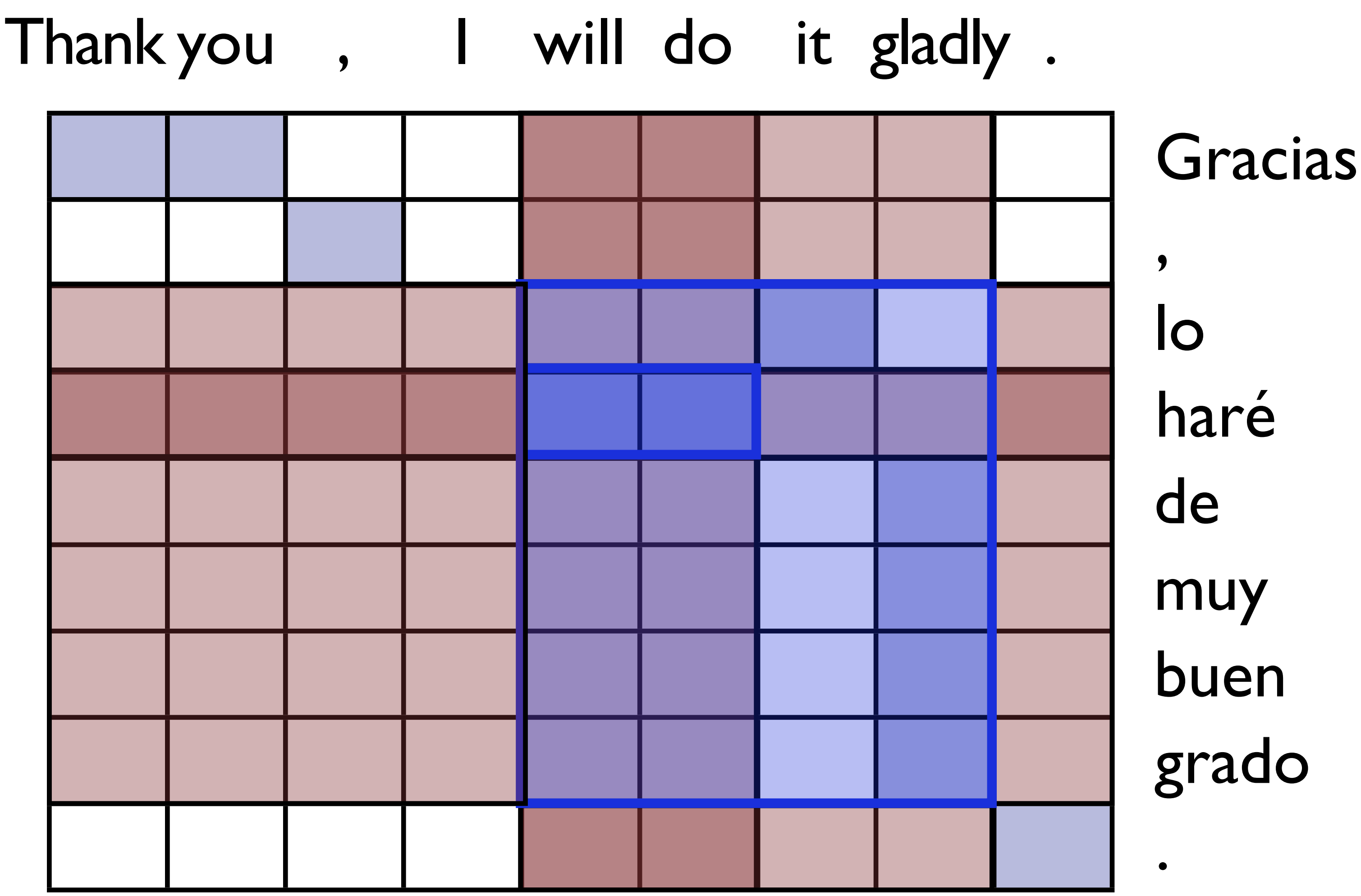
Grammar Rules

⟨haré ;
will do⟩

⟨lo X de ... grado ;
X it gladly⟩

Estimating Rule Parameters from Words

Word Aligned Sentence Pair



Grammar Rules

⟨haré ; will do⟩

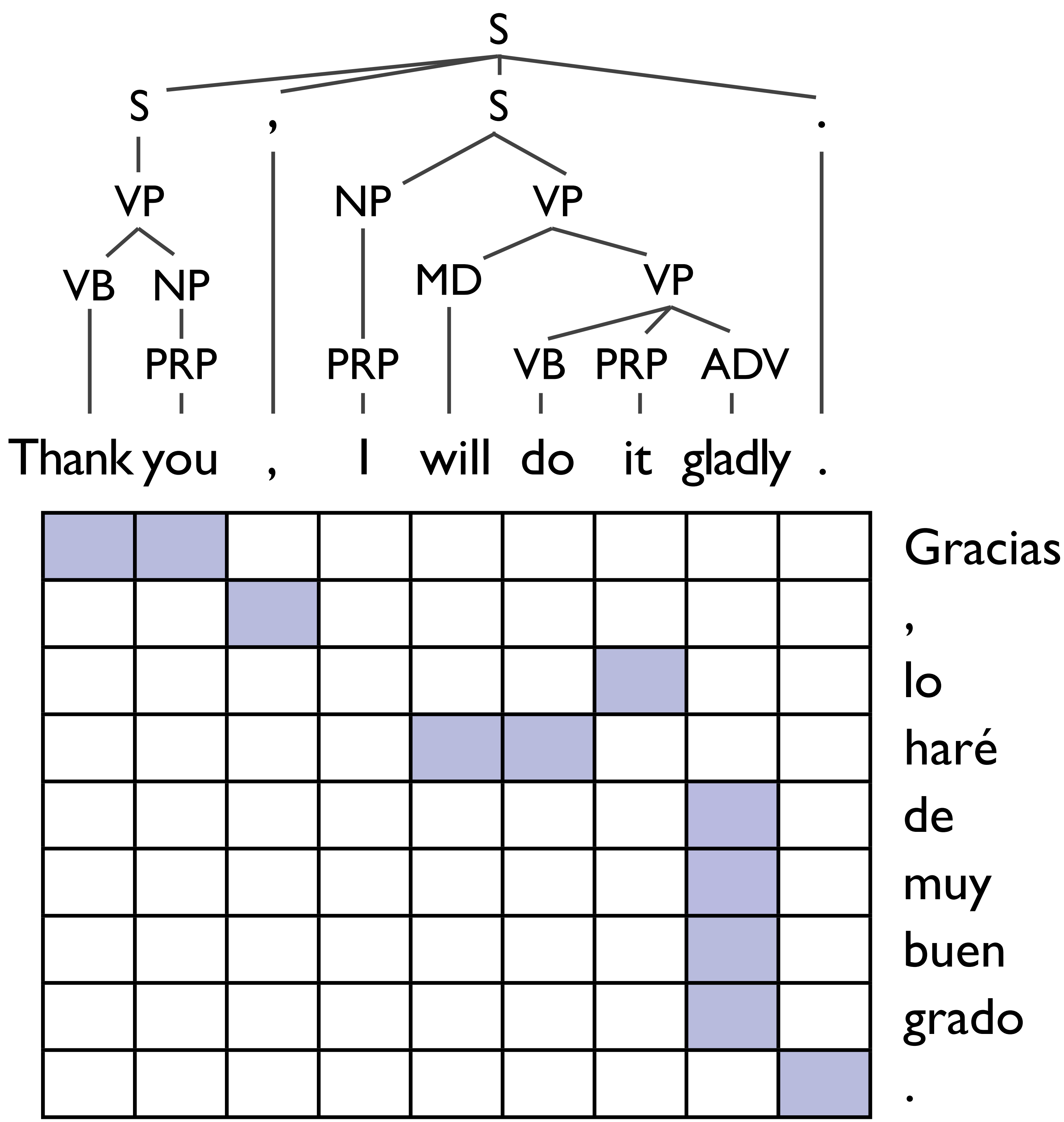
⟨lo X de ... grado ; X it gladly⟩

Model Parameters

Relative frequency counts

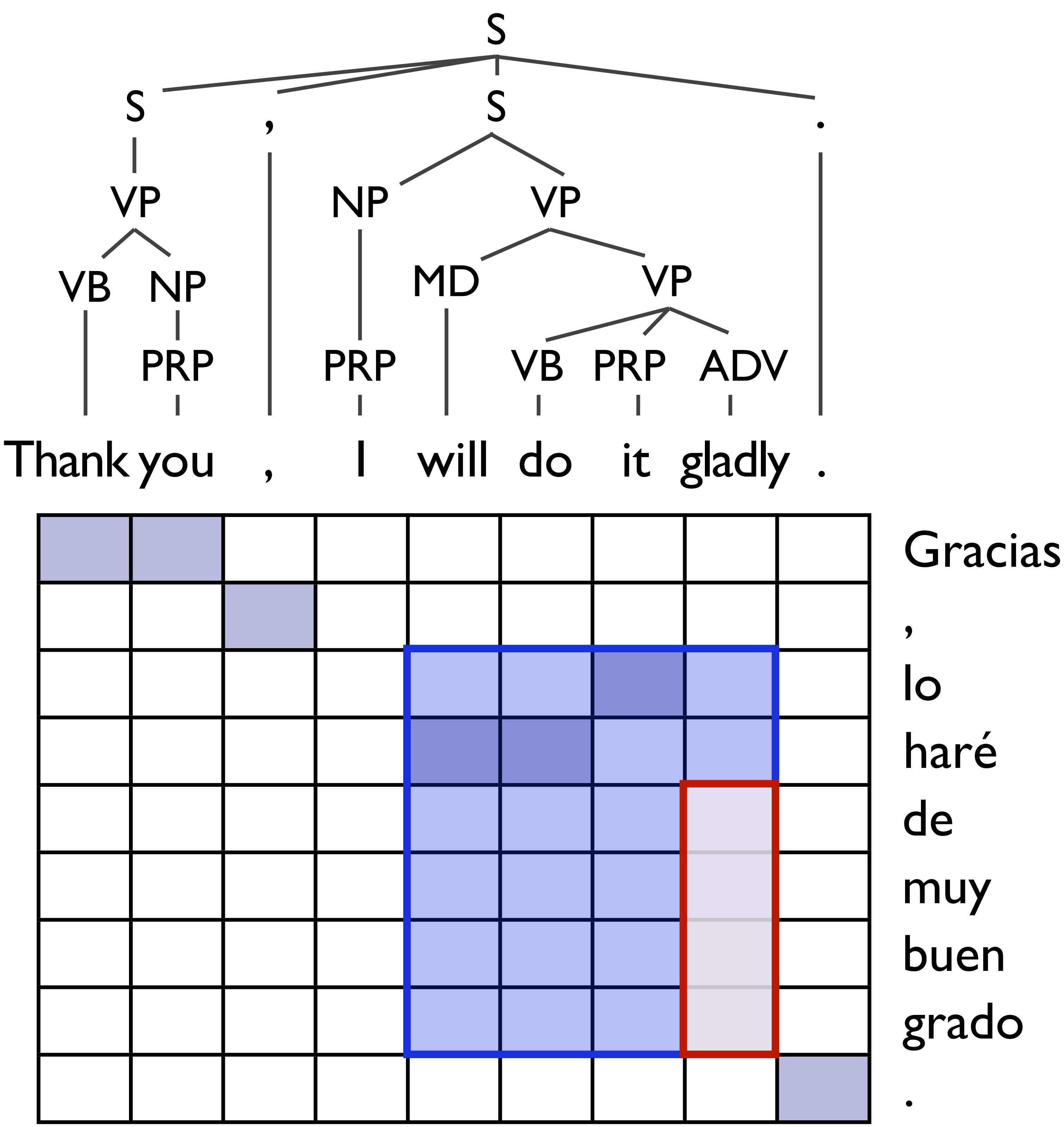
$$P(f|e) = \frac{c(\text{ lo X de muy buen grado ; X it gladly })}{c(\text{ * ; X it gladly })}$$

Learning Grammars for Translation



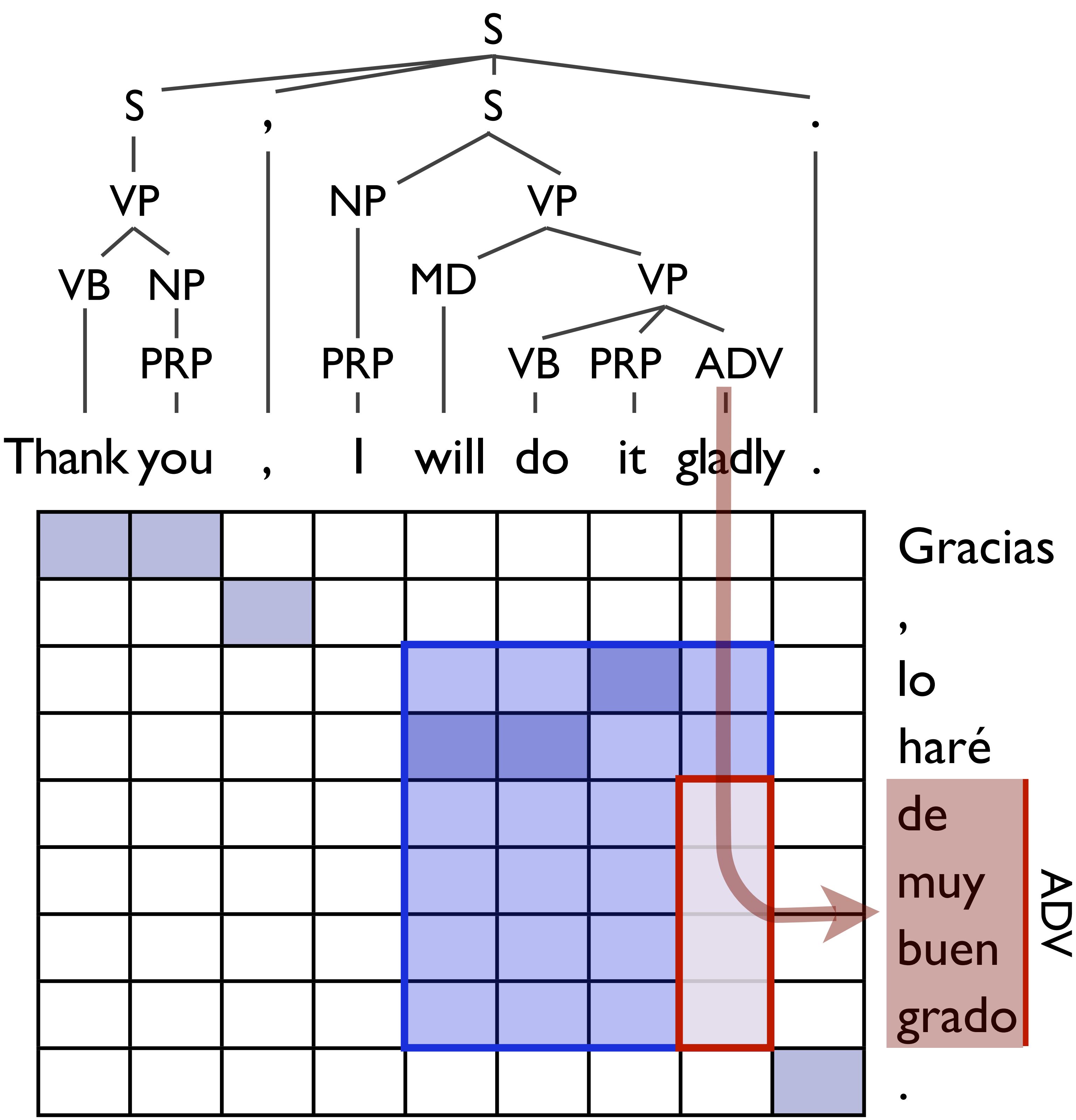
Grammar Rules

Learning Grammars for Translation



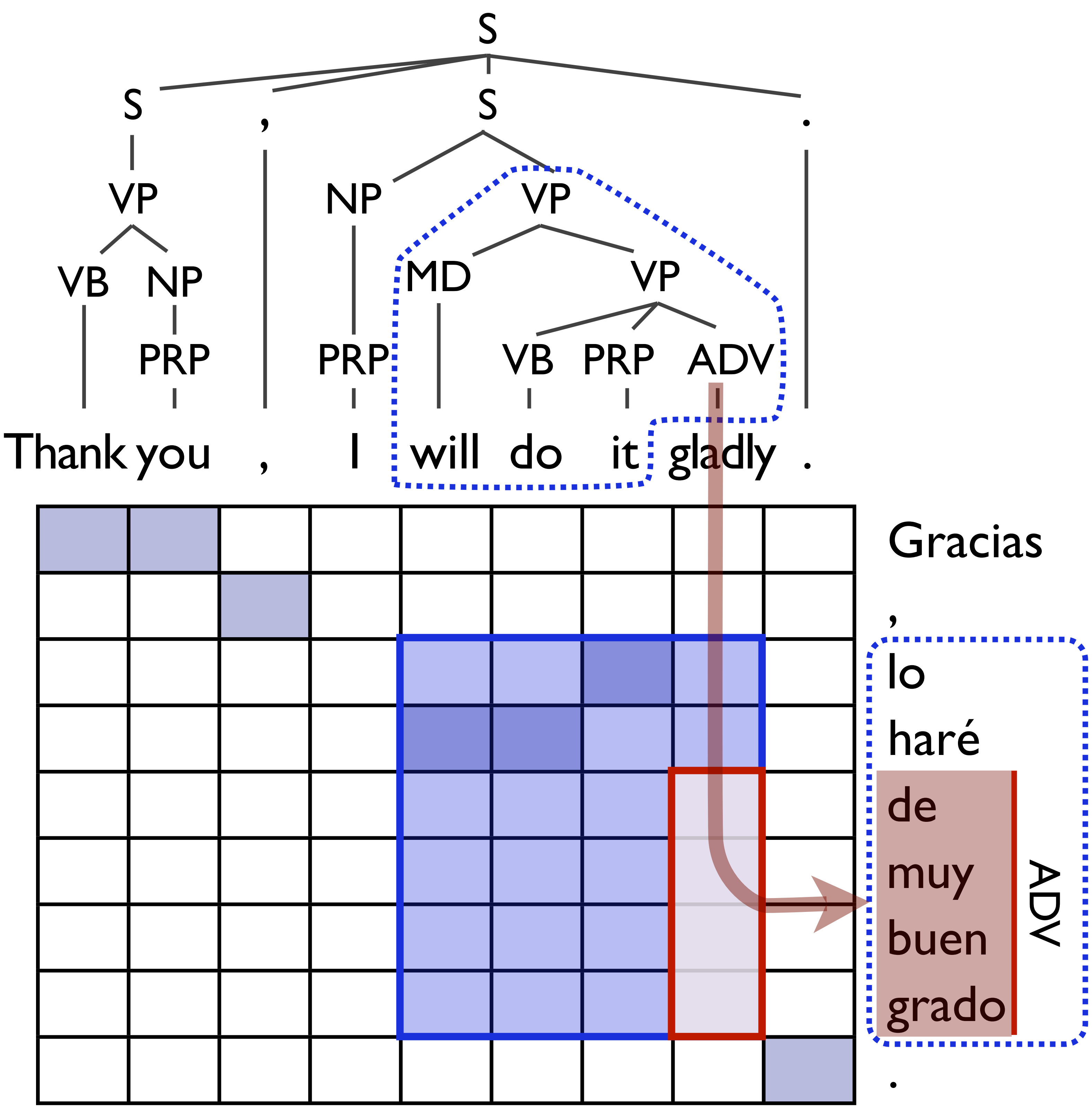
Grammar Rules

Learning Grammars for Translation



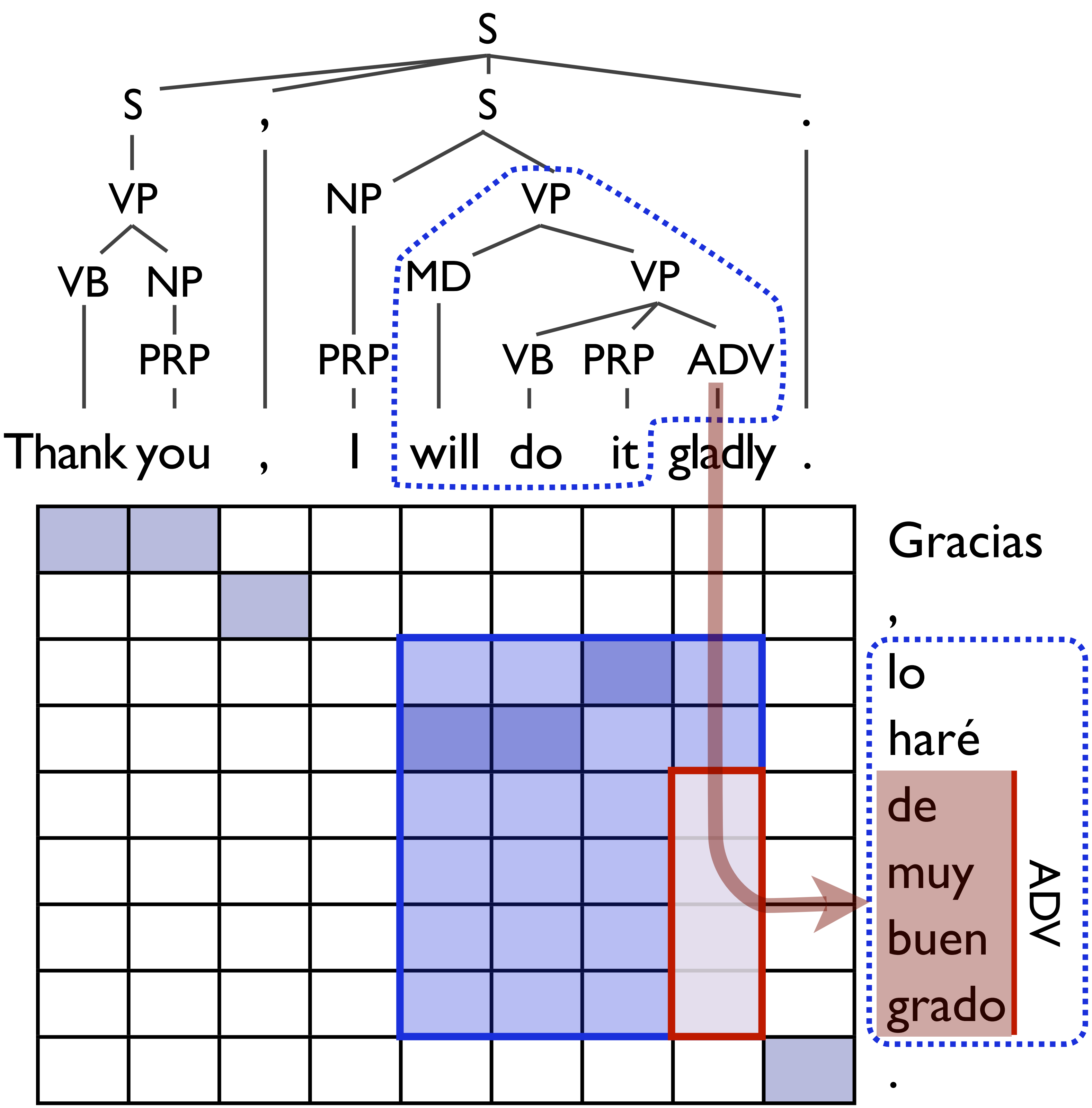
Grammar Rules

Learning Grammars for Translation



Grammar Rules

Learning Grammars for Translation



Grammar Rules

VP →

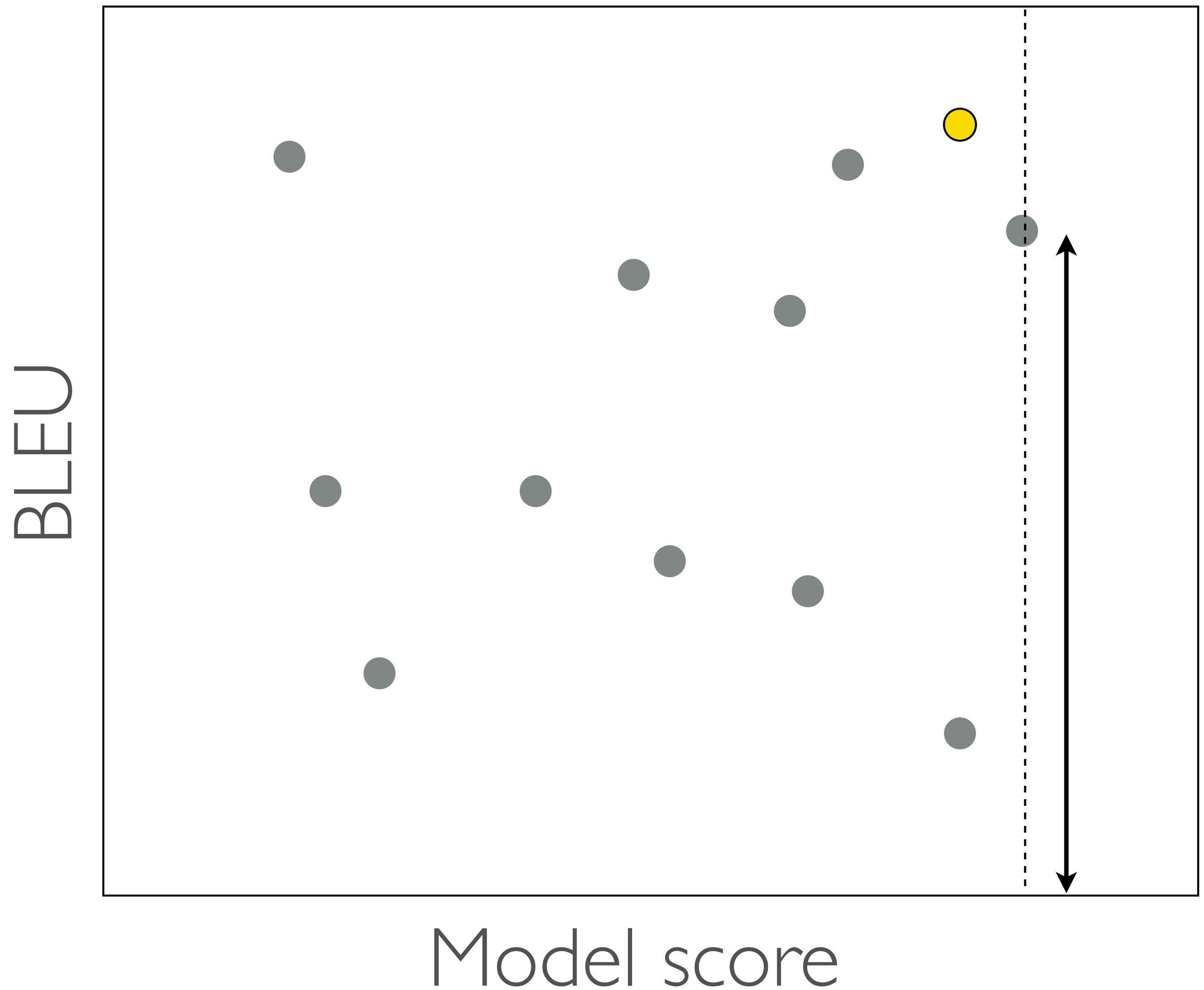
⟨lo haré ADV ;

will do it ADV⟩

Estimating the Log-Linear Model

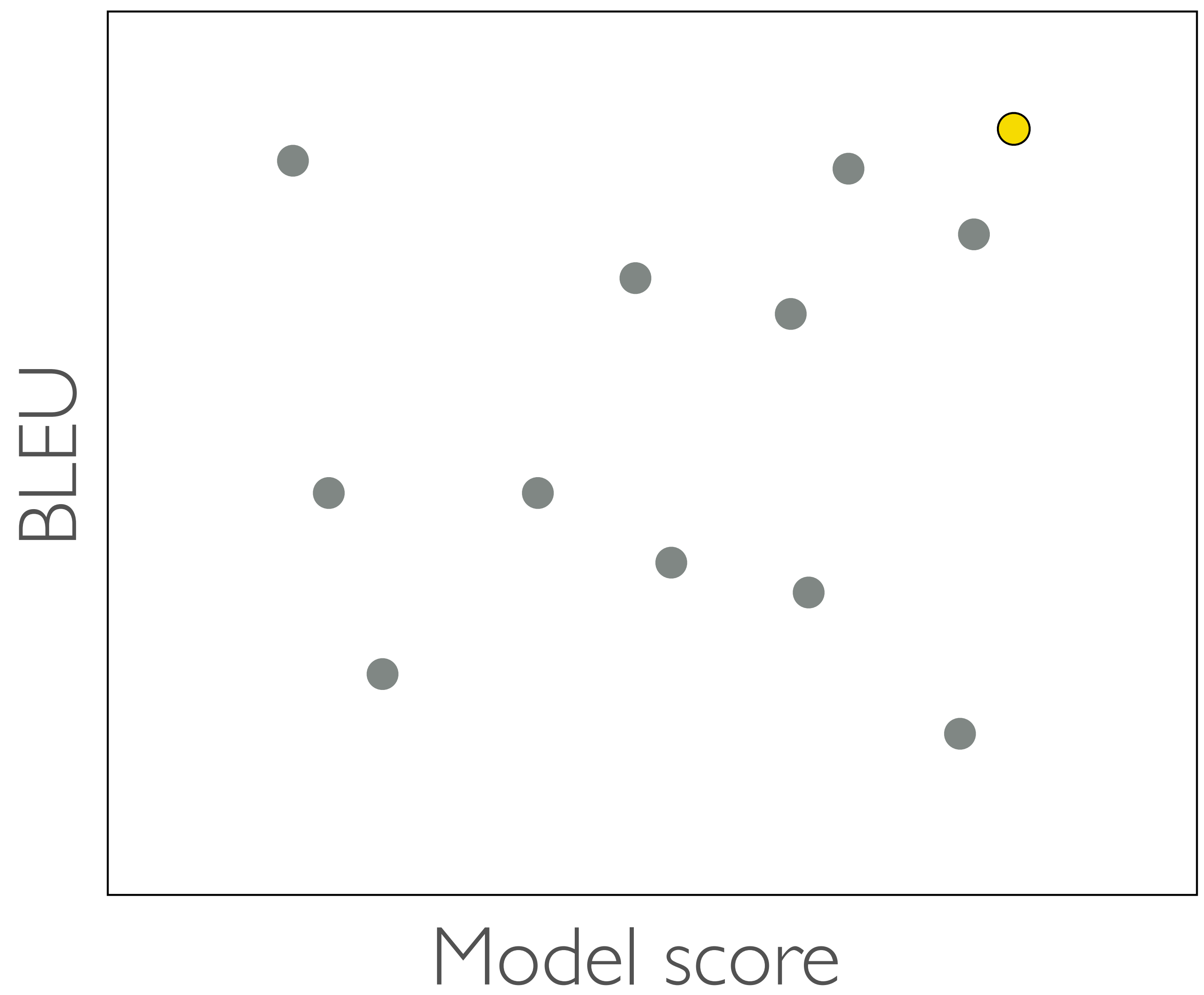
- We have all the features needed to translate:
  $P(f|e), P(e|f), P(e), \ldots$
- Now we need weights for these features
- Typically called the *tuning* stage in an MT pipeline
- Discriminative training for structured problems:
  - Need an output scoring function (BLEU)
  - Choose an optimization method

Minimum Error Rate Training (Och '02)



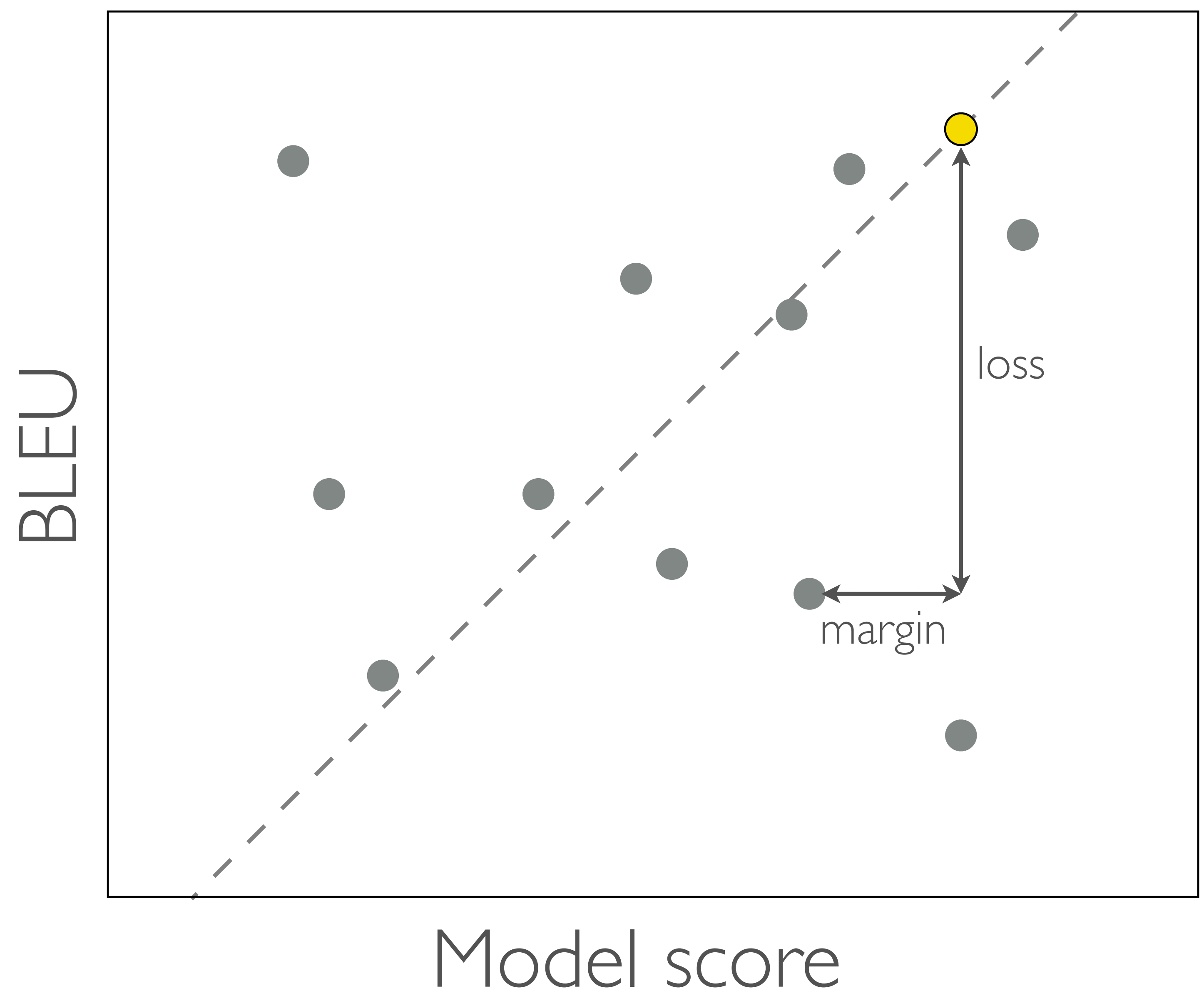
Maximize the BLEU score of the highest scoring translation

Minimum Error Rate Training (Och '02)



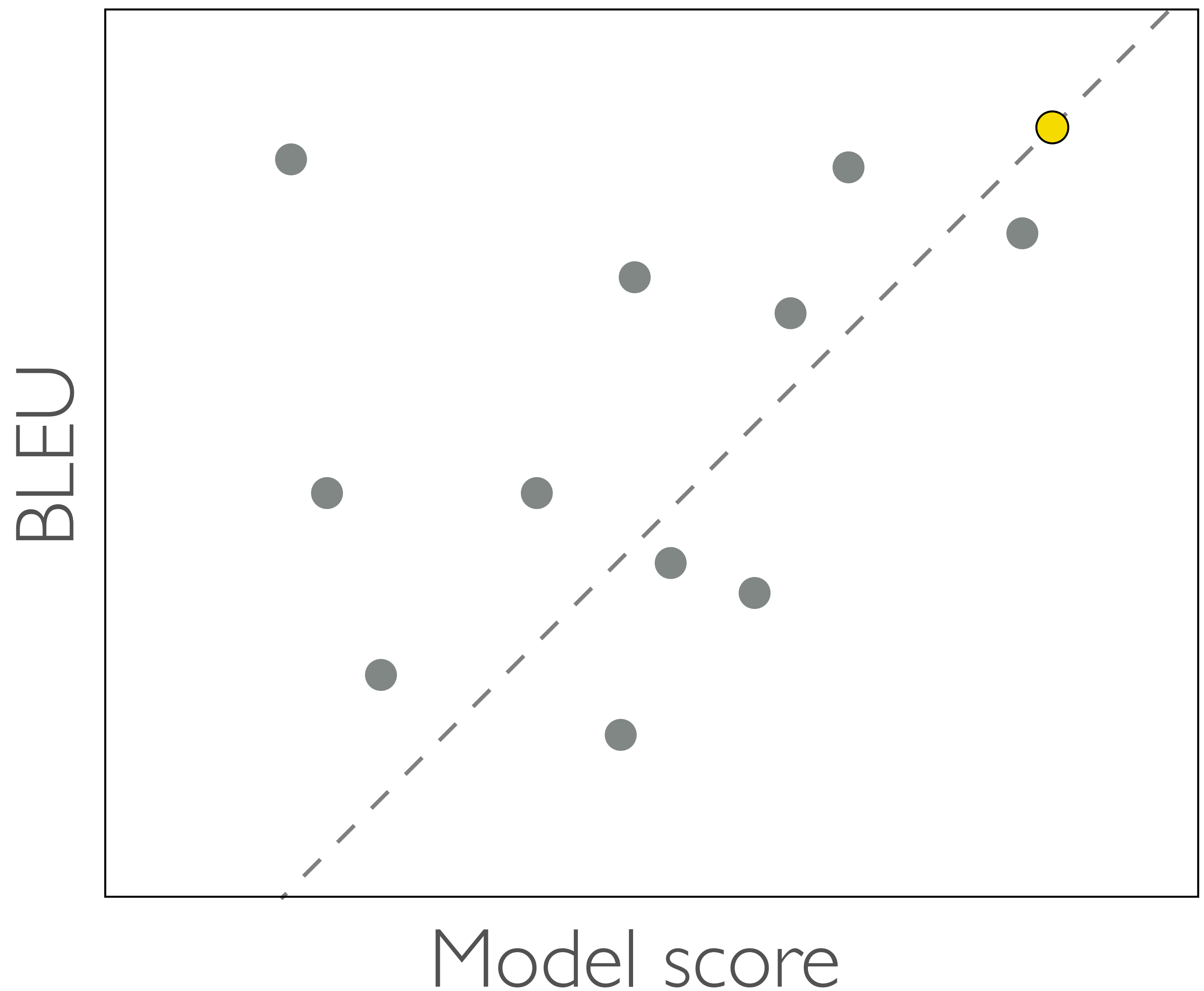
Maximize the BLEU score of the highest scoring translation

Max Margin Training (Chiang et al '08)



Make the margin larger than the loss

Max Margin Training (Chiang et al '08)



Make the margin larger than the loss

Current Research on MT Estimation

- Add linguistic knowledge to the pipeline (syntax, disambiguation models, etc.)
- Use synchronous grammars and phrase models for alignment (instead of words)
- Condition on more context when translating a word or phrase
- Add lots of features to the log-linear model