Decoding

Phrase-Based Decoding

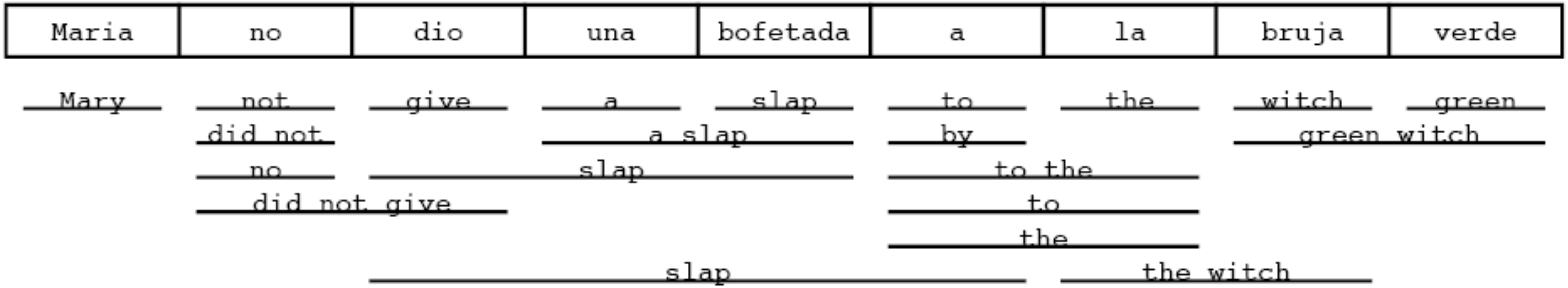
- Inputs:
  - Language model that scores $P(e_i|e_1,\ldots,e_{i-1}) \approx P(e_i|e_{i-n-1},\ldots,e_{i-1})$
  - Phrase table: set of phrase pairs (**e**, **f**) with probabilities P(**f**|**e**)
- What we want to find: **e** produced by a series of phrase-by-phrase translations from an input **f**

Phrase lattices are big!

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				

Slide credit: Dan Klein

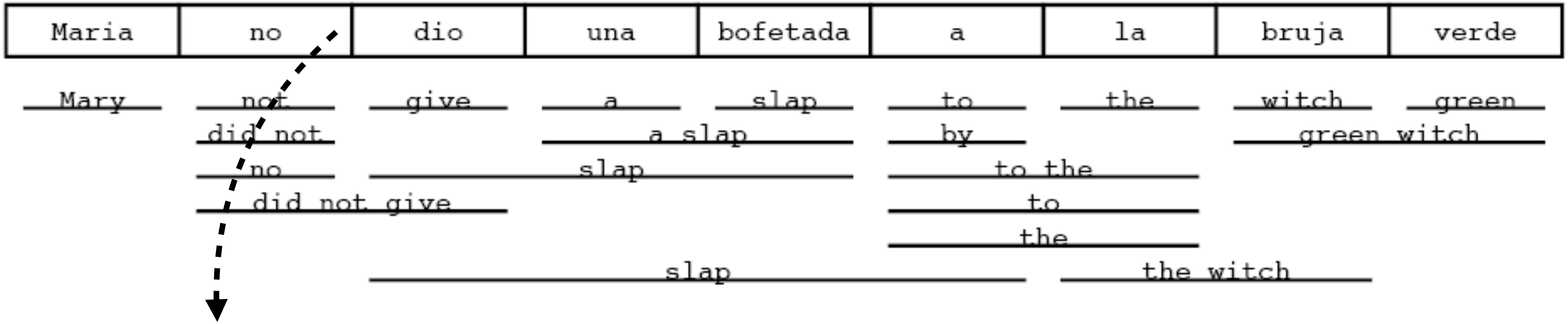
Monotonic Translation

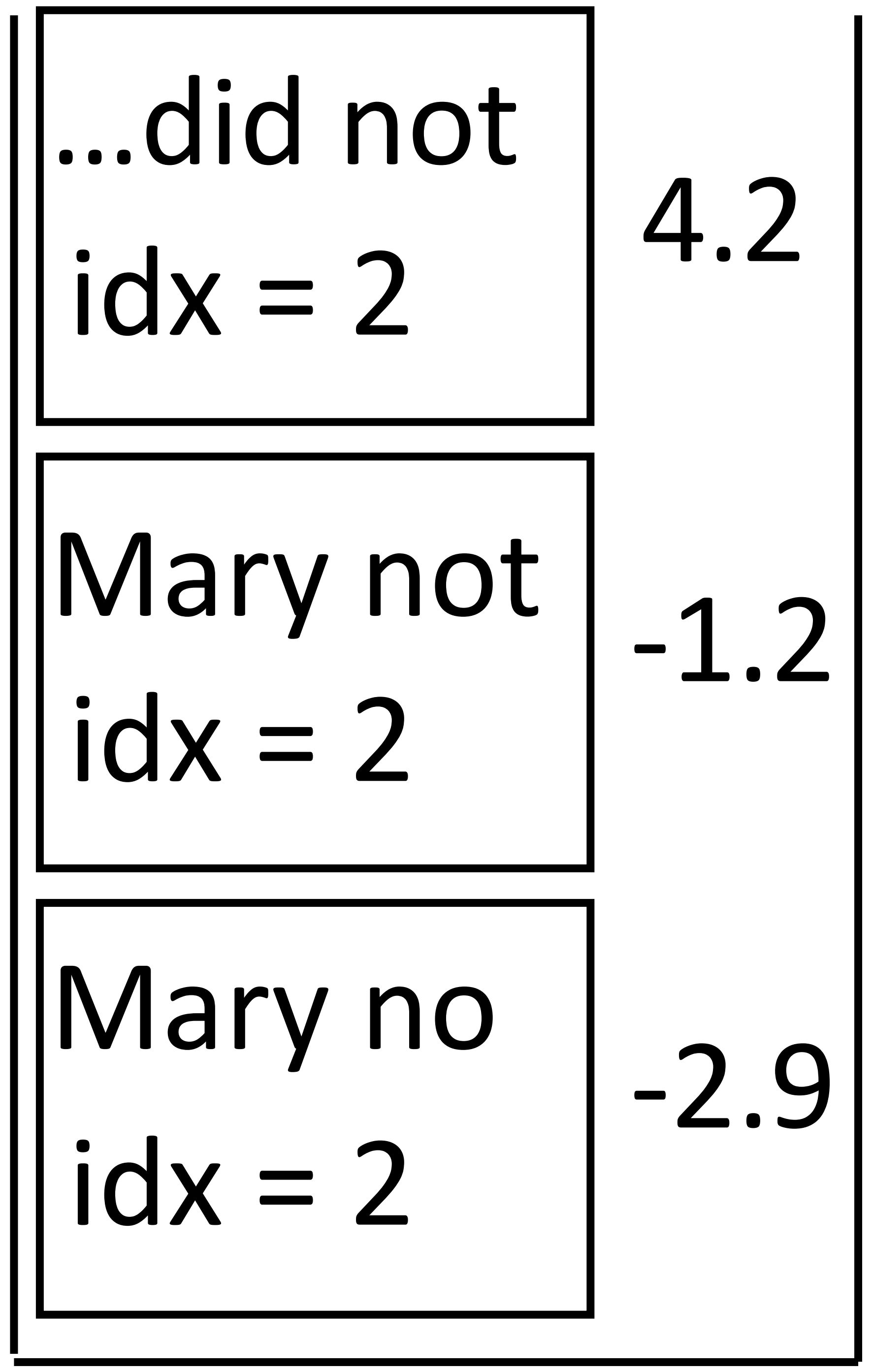


- If we translate with beam search, what state do we need to keep in the beam?
  - What have we translated so far?
  - What words have we produced so far?
  - When using a 3-gram LM, only need to remember the last 2 words!

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

Monotonic Translation



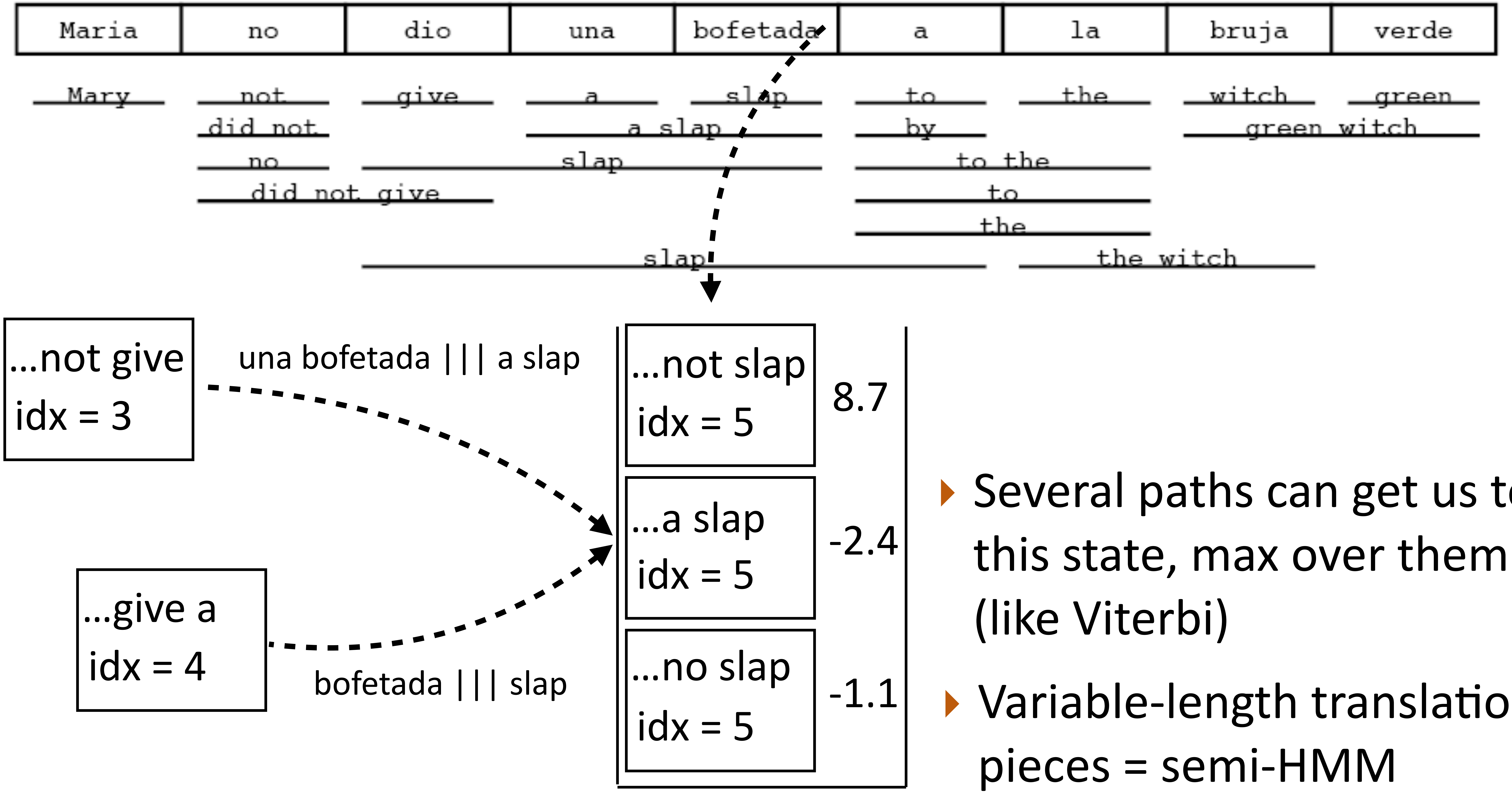


$$\text{score} = \log \, [\underbrace{P(\text{Mary})\, P(\text{not} \mid \text{Mary})}_{\text{LM}}\ \underbrace{P(\text{Mary} \mid \text{Maria})\, P(\text{not} \mid \text{no})}_{\text{TM}}]$$

In reality: score = $\alpha \log P(\text{LM}) + \beta \log P(\text{TM})$

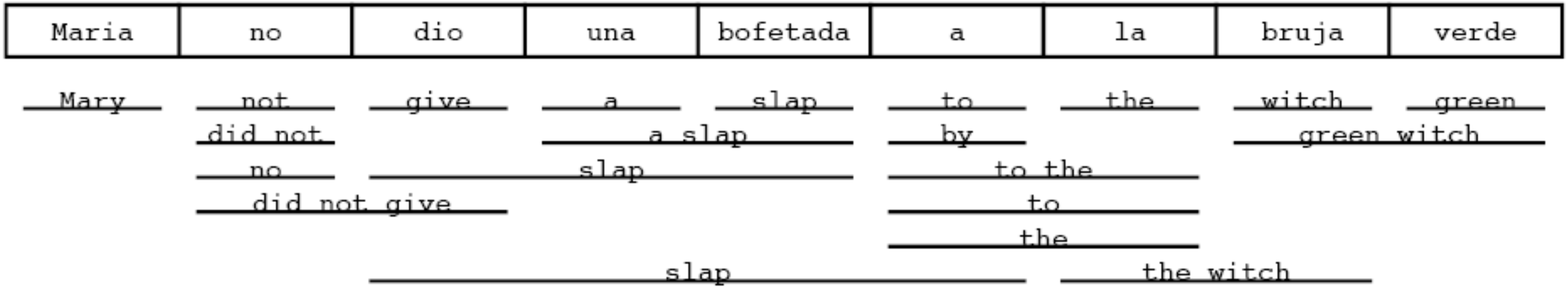
...and TM is broken down into several features

Monotonic Translation

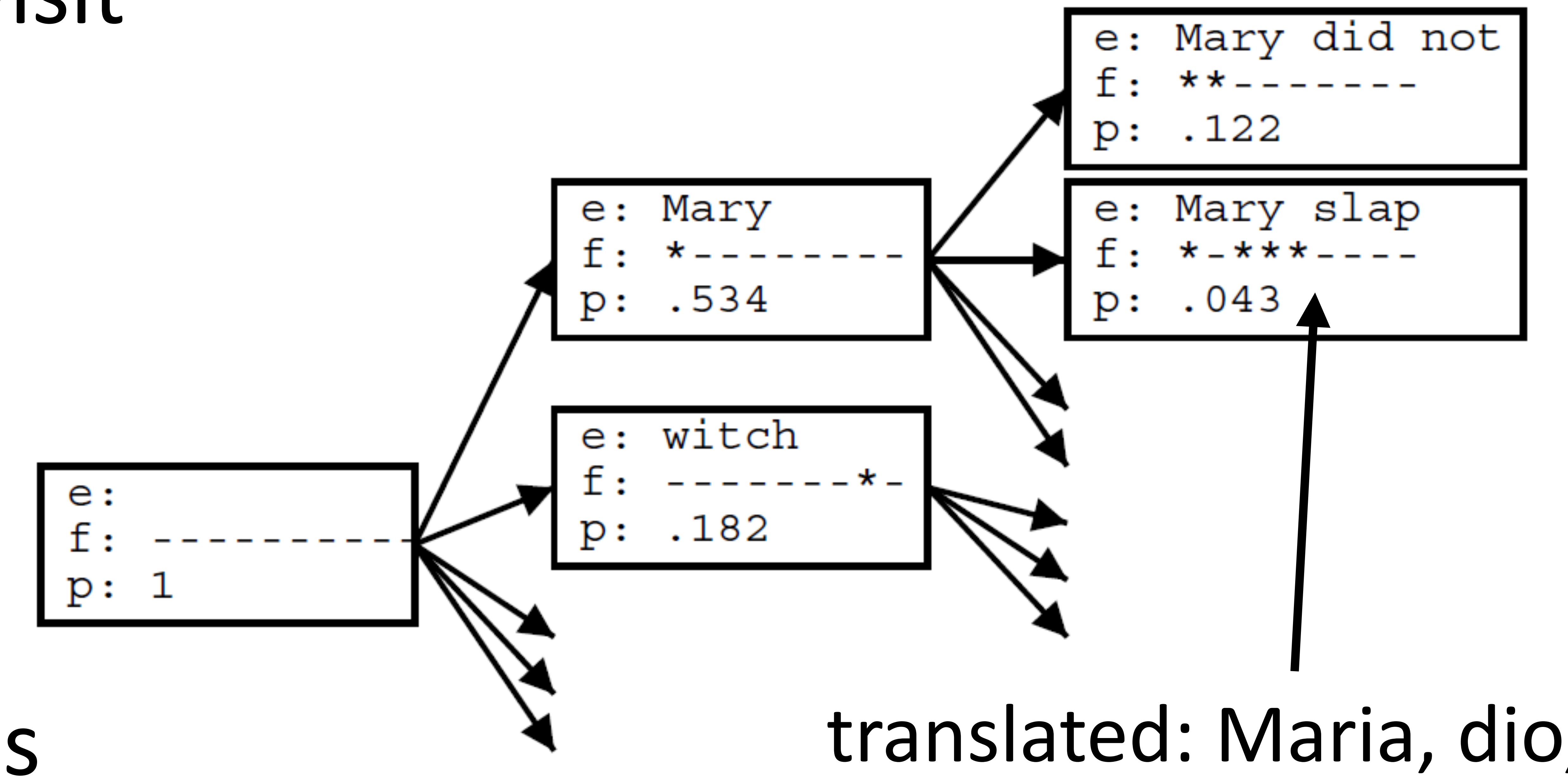


- Several paths can get us to this state, max over them (like Viterbi)
- Variable-length translation pieces = semi-HMM

Non-Monotonic Translation



- Non-monotonic translation: can visit source sentence "out of order"
- State needs to describe which words have been translated and which haven't
- Big enough phrases already capture lots of reorderings, so this isn't as important as you think



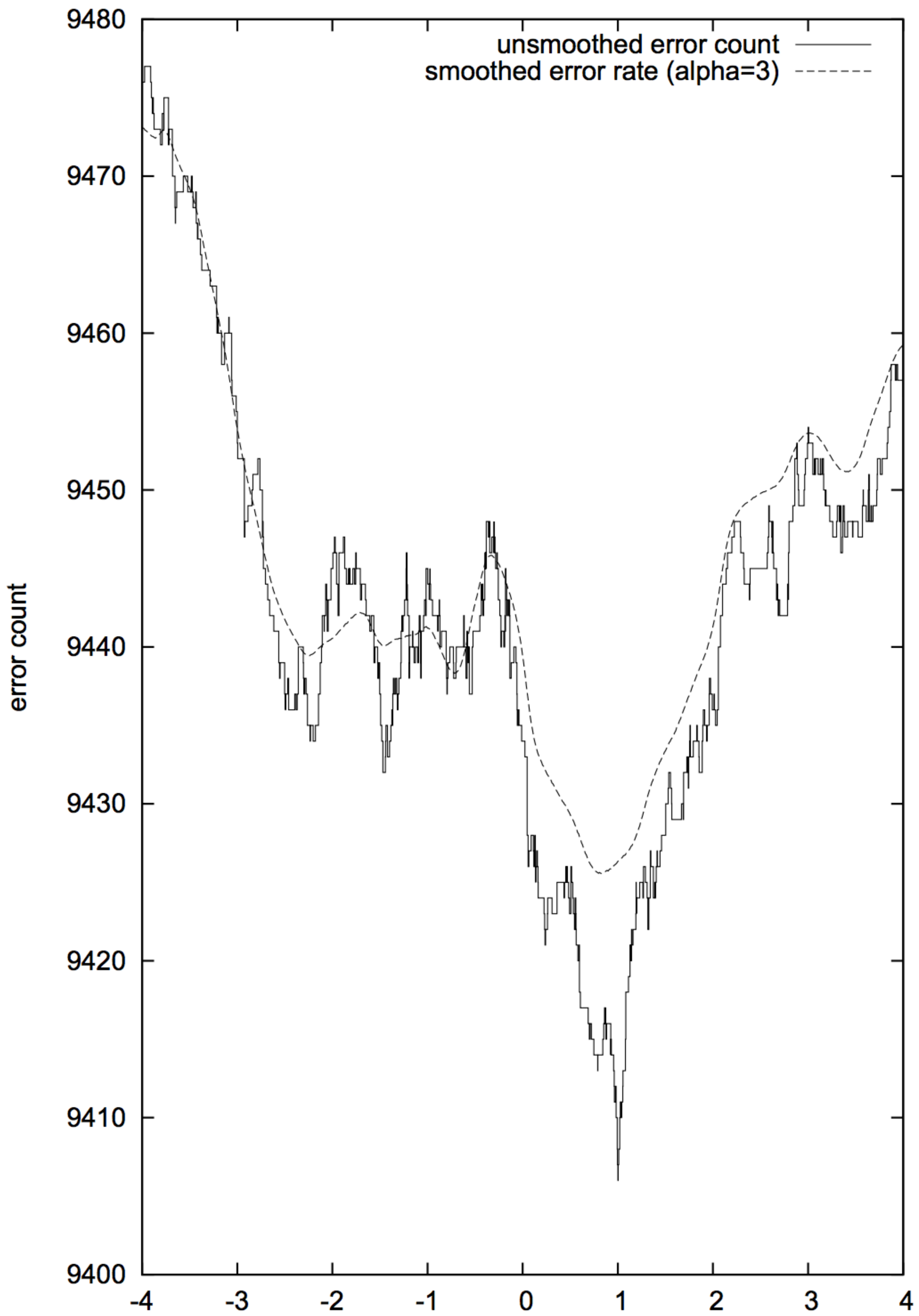
translated: Maria, dio, una, bofetada

Training Decoders

score = α log P(**t**) + β log P(**s**|**t**)

…and P(**s**|**t**) is in fact more complex

- Usually 5-20 feature weights to set, want to optimize for BLEU score which is not differentiable
- MERT (Och 2003): decode to get 1000-best translations for each sentence in a small training set (<1000 sentences), do line search on parameters to directly optimize for BLEU



Moses

- Toolkit for machine translation due to Philipp Koehn + Hieu Hoang
  - Pharaoh (Koehn, 2004) is the decoder from Koehn's thesis

- Moses implements word alignment, language models, and this decoder, plus *a ton* more stuff
  - Highly optimized and heavily engineered, could more or less build SOTA translation systems with this from 2007-2013

Moses

SOURCE	Cela constituerait une solution transitoire qui permettrait de conduire à terme à une charte à valeur contraignante.
HUMAN	That would be an interim solution which would make it possible to work towards a binding charter in the long term .
1x DATA	[this] [constituerait] [assistance] [transitoire] [who] [permettrait] [licences] [to] [terme] [to] [a] [charter] [to] [value] [contraignante] [.]
10x DATA	[it] [would] [a solution] [transitional] [which] [would] [of] [lead] [to] [term] [to a] [charter] [to] [value] [binding] [.]
100x DATA	[this] [would be] [a transitional solution] [which would] [lead to] [a charter] [legally binding] [.]
1000x DATA	[that would be] [a transitional solution] [which would] [eventually lead to] [a binding charter] [.]

slide credit: Dan Klein

Syntactic MT

- Rather than use phrases, use a *synchronous context-free grammar*

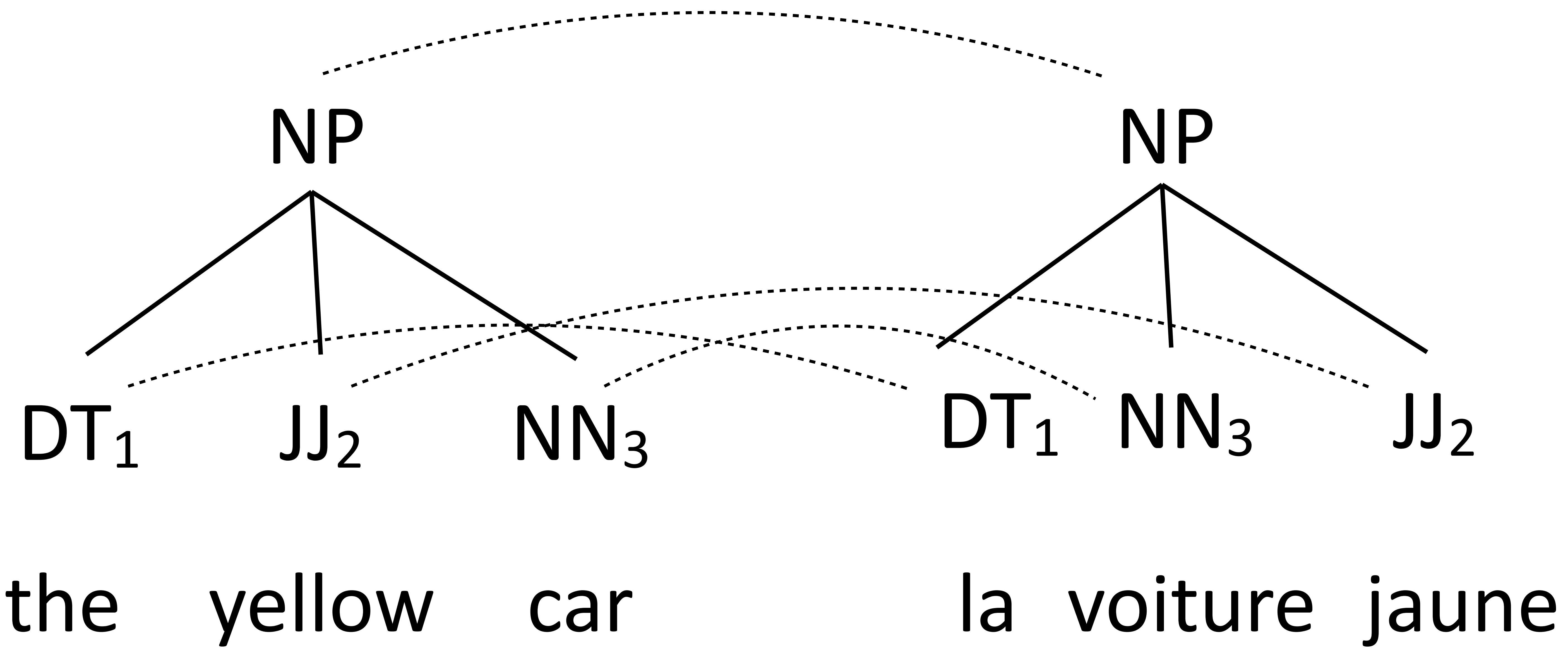
$\mathrm{NP} \rightarrow [\mathrm{DT}_1\ \mathrm{JJ}_2\ \mathrm{NN}_3;\ \mathrm{DT}_1\ \mathrm{NN}_3\ \mathrm{JJ}_2]$

$\mathrm{DT} \rightarrow$ [the, la]

$\mathrm{DT} \rightarrow$ [the, le]

$\mathrm{NN} \rightarrow$ [car, voiture]

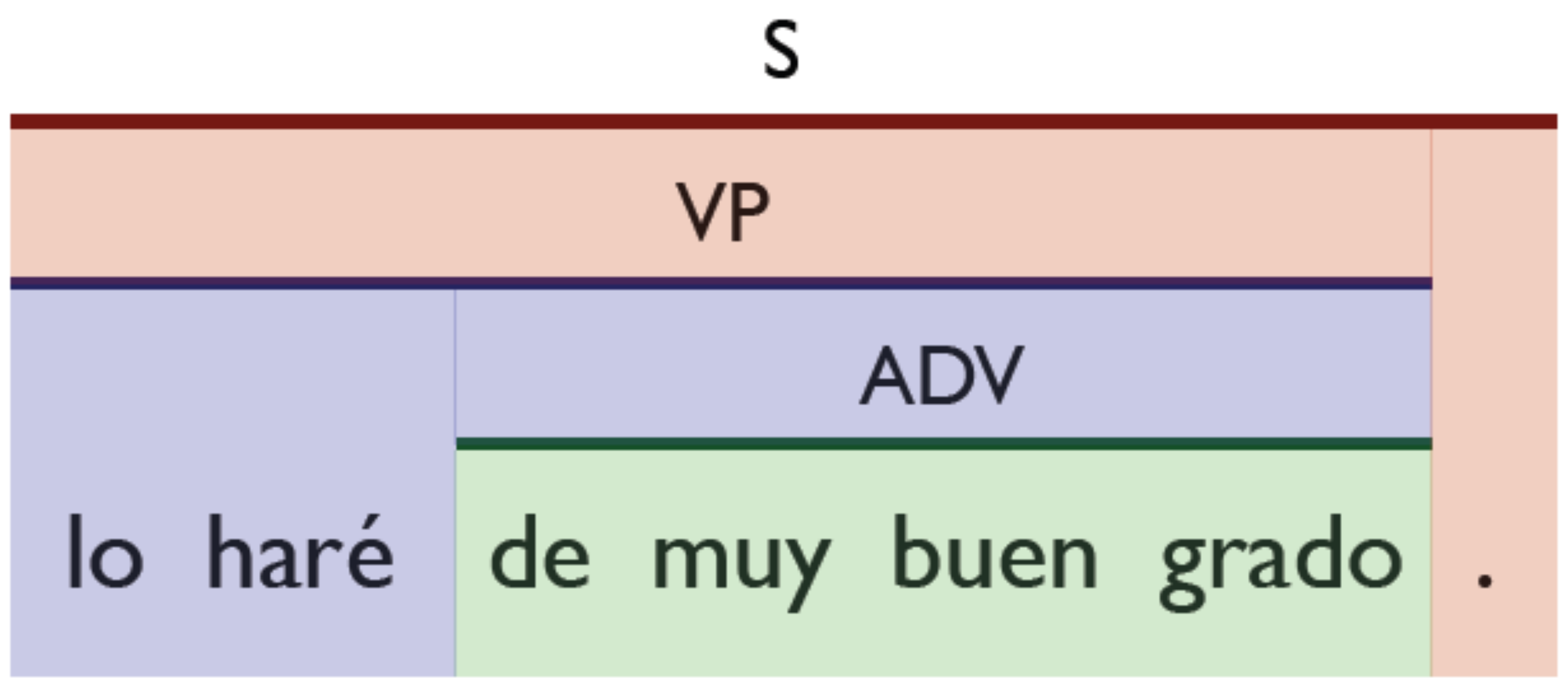
$\mathrm{JJ} \rightarrow$ [yellow, jaune]



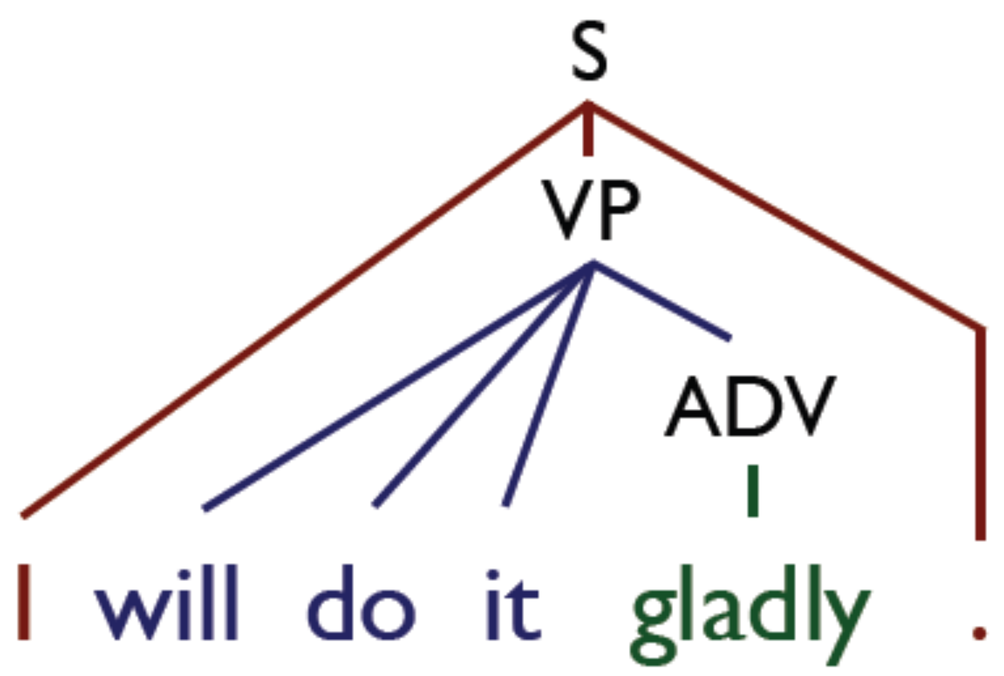
- Translation = parse the input with “half” of the grammar, read off the other half
- Assumes parallel syntax up to reordering

Syntactic MT

Input



Output



- Use lexicalized rules, look like "syntactic phrases"
- Leads to HUGE grammars, parsing is slow

Grammar

S → ⟨ VP . ; I VP . ⟩ **OR** S → ⟨ VP . ; you VP . ⟩

VP → ⟨ lo haré ADV ; will do it ADV ⟩

S → ⟨ lo haré ADV . ; I will do it ADV . ⟩

ADV → ⟨ de muy buen grado ; gladly ⟩

Slide credit: Dan Klein