Statistical Machine Translation
Part III – Phrase-based SMT / Decoding

Alexander Fraser
Institute for Natural Language Processing
Universität Stuttgart

2012.09.15 Seminar: Statistical MT
NSSNLP, University of Kathmandu

Where we have been

- We defined the overall problem and talked about evaluation
- We have now covered word alignment
  - IBM Model 1, true Expectation Maximization
  - IBM Model 4, approximate Expectation Maximization
  - Symmetrization Heuristics (such as Grow)
    - Applied to two Model 4 alignments
    - Results in final word alignment

Where we are going

- We will define a high performance translation model
- We will show how to solve the search problem for this model

Outline

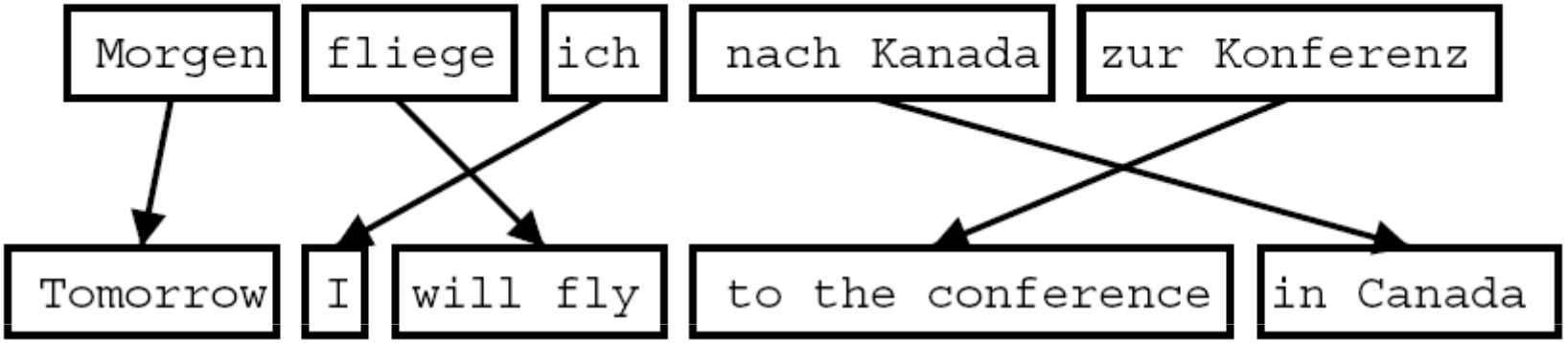
- Phrase-based translation
  - Model
  - Estimating parameters
- Decoding

- We could use IBM Model 4 in the direction p(f|e), together with a language model, p(e), to translate

$$\underset{e}{\text{argmax}}\ P(e \mid f) = \underset{e}{\text{argmax}}\ P(f \mid e)\ P(e)$$

- However, decoding using Model 4 doesn't work well in practice
  - One strong reason is the bad 1-to-N assumption
  - Another problem would be defining the search algorithm
    - If we add additional operations to allow the English words to vary, this will be very expensive
  - Despite these problems, Model 4 decoding was briefly state of the art
- We will now define a better model…

Phrase-based translation

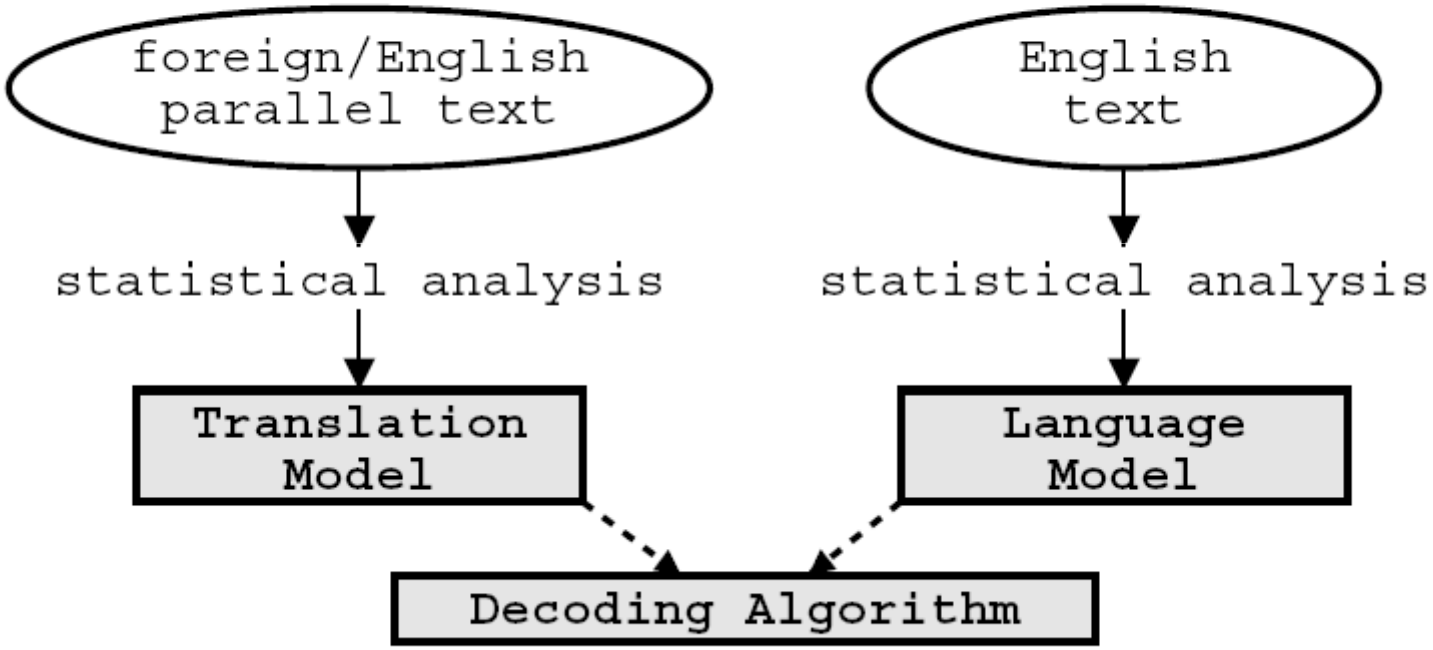


- Foreign input is segmented in phrases
  - any sequence of words, not necessarily linguistically motivated
- Each phrase is translated into English
- Phrases are reordered

Slide from Koehn 2008

Statistical Machine Translation

- Components: Translation model, language model, decoder



Slide from Koehn 2008

Language Model

- Often a trigram language model is used for p(e)
  - P(the man went home) = p(the | START) p(man | START the) p(went | the man) p(home | man went)
- Language models work well for comparing the grammaticality of strings of the same length
  - However, when comparing short strings with long strings they favor short strings
  - For this reason, an important component of the language model is the length bonus
    - This is a constant > 1 multiplied for each English word in the hypothesis
    - It makes longer strings competitive with shorter strings

Phrase-based translation model

- Major components of phrase-based model
  - **phrase translation model** $\phi(\mathbf{f}|\mathbf{e})$
  - **reordering model** d
  - **language model** $p_{\text{LM}}(\mathbf{e})$
- Bayes rule

$$\begin{aligned} \text{argmax}_{\mathbf{e}} p(\mathbf{e}|\mathbf{f}) &= \text{argmax}_{\mathbf{e}} p(\mathbf{f}|\mathbf{e}) p(\mathbf{e}) \\ &= \text{argmax}_{\mathbf{e}} \phi(\mathbf{f}|\mathbf{e}) p_{\text{LM}}(\mathbf{e}) \omega^{\text{length}(\mathbf{e})} \end{aligned}$$

- Sentence $\mathbf{f}$ is decomposed into I phrases $\bar{f}_1^I = \bar{f}_1, ..., \bar{f}_I$
- Decomposition of $\phi(\mathbf{f}|\mathbf{e})$

$$\phi(\bar{f}_1^I|\bar{e}_1^I) = \prod_{i=1}^{I} \phi(\bar{f}_i|\bar{e}_i) d(a_i - b_{i-1})$$

Modified from Koehn 2008

Advantages of phrase-based translation

- *Many-to-many* translation can handle non-compositional phrases
- Use of *local context* in translation
- The more data, the *longer phrases* can be learned

Slide from Koehn 2008

Phrase translation table

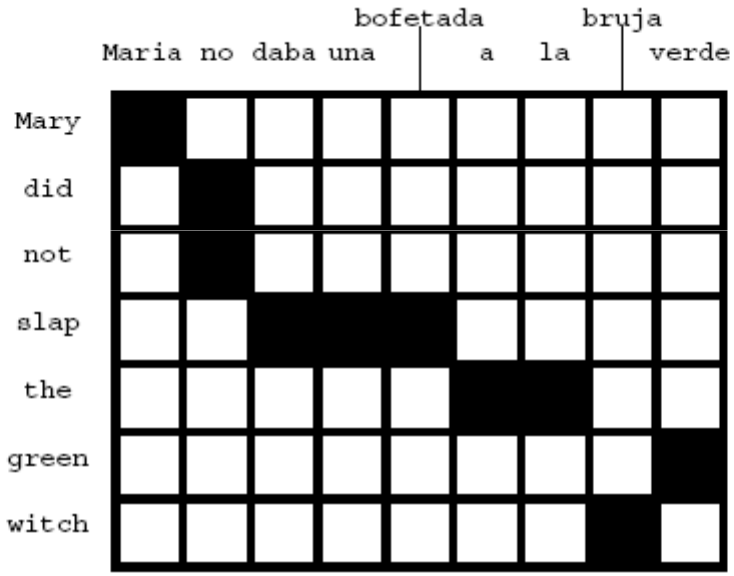
- Phrase translations for *den Vorschlag*

English	$\phi(e\|f)$	English	$\phi(e\|f)$
the proposal	0.6227	the suggestions	0.0114
's proposal	0.1068	the proposed	0.0114
a proposal	0.0341	the motion	0.0091
the idea	0.0250	the idea of	0.0091
this proposal	0.0227	the proposal ,	0.0068
proposal	0.0205	its proposal	0.0068
of the proposal	0.0159	it	0.0068
the proposals	0.0159	...	...

Slide from Koehn 2008

How to learn the phrase translation table?

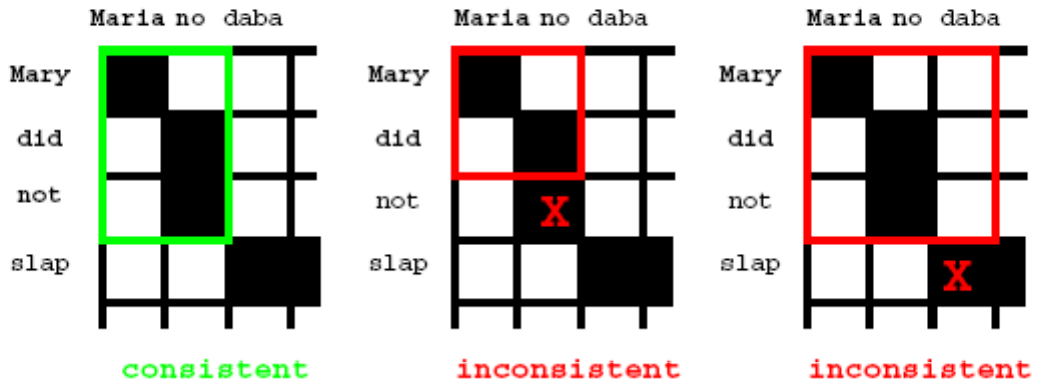
- Start with the *word alignment*:



- Collect all phrase pairs that are **consistent** with the word alignment

Slide from Koehn 2008

Consistent with word alignment



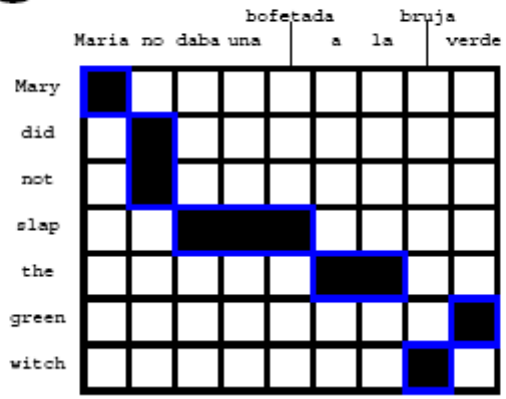
- **Consistent with the word alignment** :=

  phrase alignment has to *contain all alignment points* for all covered words

$$(\overline{e}, \overline{f}) \in BP \Leftrightarrow \quad \forall e_i \in \overline{e} : (e_i, f_j) \in A \rightarrow f_j \in \overline{f}$$
$$\text{AND} \quad \forall f_j \in \overline{f} : (e_i, f_j) \in A \rightarrow e_i \in \overline{e}$$

Slide from Koehn 2008

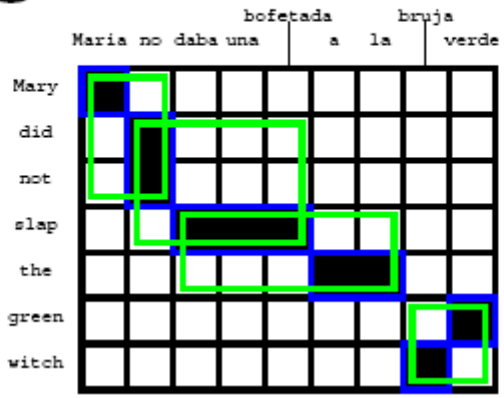
Word alignment induced phrases



(Maria, Mary), (no, did not), (slap, daba una bofetada), (a la, the), (bruja, witch), (verde, green)

Slide from Koehn 2008

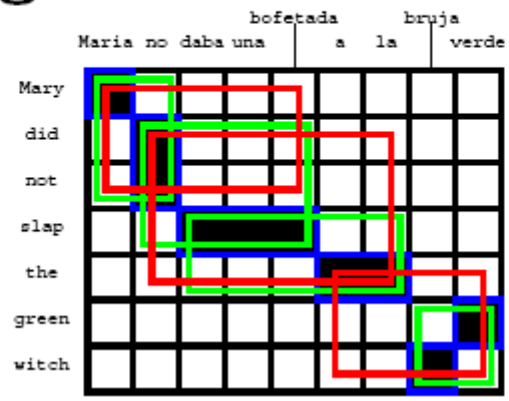
Word alignment induced phrases



(Maria, Mary), (no, did not), (slap, daba una bofetada), (a la, the), (bruja, witch), (verde, green),
(Maria no, Mary did not), (no daba una bofetada, did not slap), (daba una bofetada a la, slap the),
(bruja verde, green witch)

Slide from Koehn 2008

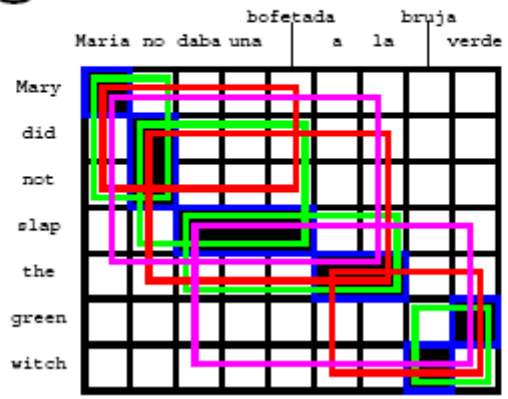
Word alignment induced phrases



(Maria, Mary), (no, did not), (slap, daba una bofetada), (a la, the), (bruja, witch), (verde, green),
(Maria no, Mary did not), (no daba una bofetada, did not slap), (daba una bofetada a la, slap the),
(bruja verde, green witch), (Maria no daba una bofetada, Mary did not slap),
(no daba una bofetada a la, did not slap the), (a la bruja verde, the green witch)

Slide from Koehn 2008

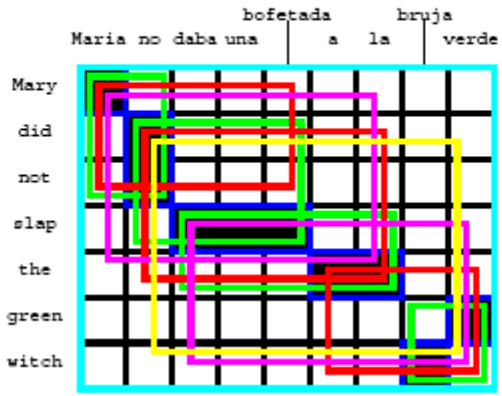
Word alignment induced phrases



(Maria, Mary), (no, did not), (slap, daba una bofetada), (a la, the), (bruja, witch), (verde, green),
(Maria no, Mary did not), (no daba una bofetada, did not slap), (daba una bofetada a la, slap the),
(bruja verde, green witch), (Maria no daba una bofetada, Mary did not slap),
(no daba una bofetada a la, did not slap the), (a la bruja verde, the green witch),
(Maria no daba una bofetada a la, Mary did not slap the),
(daba una bofetada a la bruja verde, slap the green witch)

Slide from Koehn 2008

Word alignment induced phrases (5)



(Maria, Mary), (no, did not), (slap, daba una bofetada), (a la, the), (bruja, witch), (verde, green),
(Maria no, Mary did not), (no daba una bofetada, did not slap), (daba una bofetada a la, slap the),
(bruja verde, green witch), (Maria no daba una bofetada, Mary did not slap),
(no daba una bofetada a la, did not slap the), (a la bruja verde, the green witch),
(Maria no daba una bofetada a la, Mary did not slap the), (daba una bofetada a la bruja verde,
slap the green witch), (no daba una bofetada a la bruja verde, did not slap the green witch),
(Maria no daba una bofetada a la bruja verde, Mary did not slap the green witch)

Slide from Koehn 2008

Probability distribution of phrase pairs

- We need a **probability distribution** $\phi(\overline{f}|\overline{e})$ over the collected phrase pairs

$\Rightarrow$ Possible *choices*

- *relative frequency* of collected phrases: $\phi(\overline{f}|\overline{e}) = \frac{\text{count}(\overline{f},\overline{e})}{\sum_{\overline{f}} \text{count}(\overline{f},\overline{e})}$
- or, conversely $\phi(\overline{e}|\overline{f})$
- use *lexical translation probabilities*

Slide from Koehn 2008

Reordering

- *Monotone* translation
  - do not allow any reordering

  → worse translations

- *Limiting* reordering (to movement over max. number of words) helps

- *Distance-based* reordering cost
  - moving a foreign phrase over n words: cost z^n

- *Lexicalized* reordering model

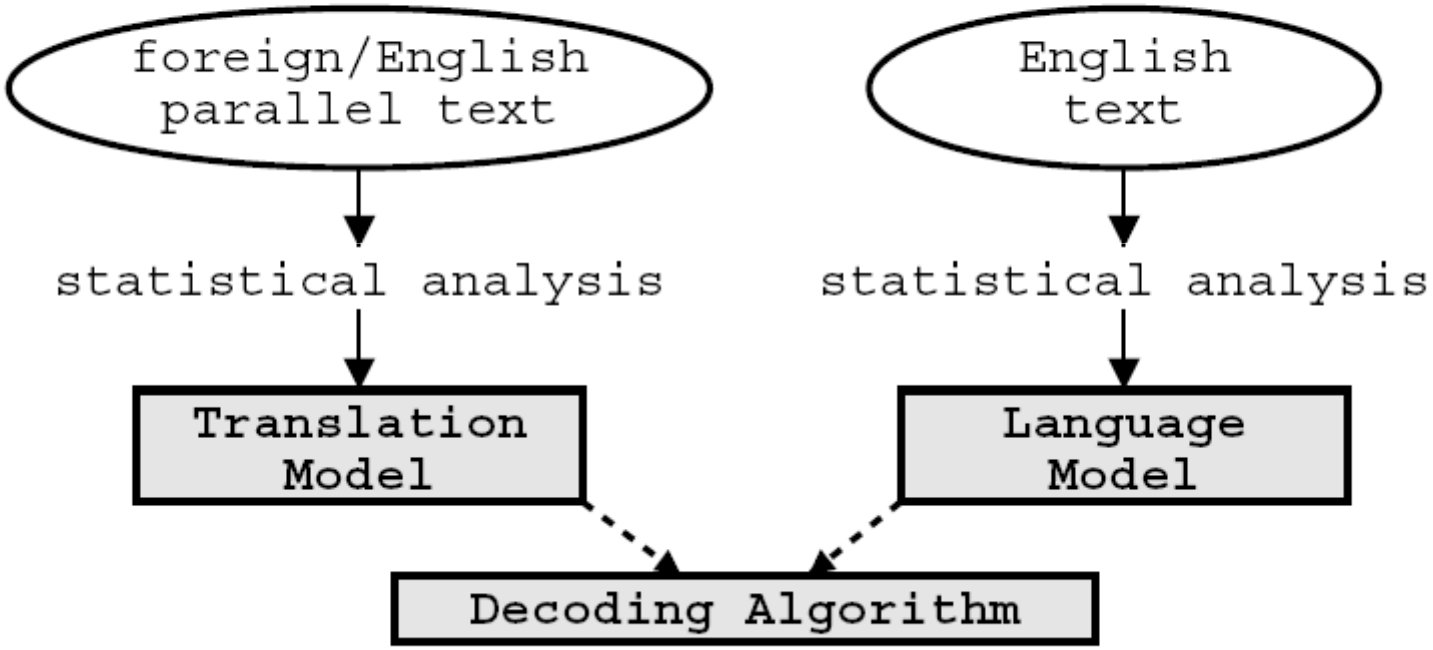
Slide from Koehn 2008

Outline

- *Phrase-based translation model*
- Decoding
  - Basic phrase-based decoding
  - Dealing with complexity
    - Recombination
    - Pruning
    - Future cost estimation

Statistical Machine Translation

- Components: Translation model, language model, decoder



Slide from Koehn 2008

Decoding

- Goal: find the best target translation of a source sentence
- Involves search
  - Find maximum probability path in a dynamically generated search graph
- Generate English string, from left to right, by covering parts of Foreign string
  - Generating English string left to right allows scoring with the n-gram language model
- Here is an example of one path

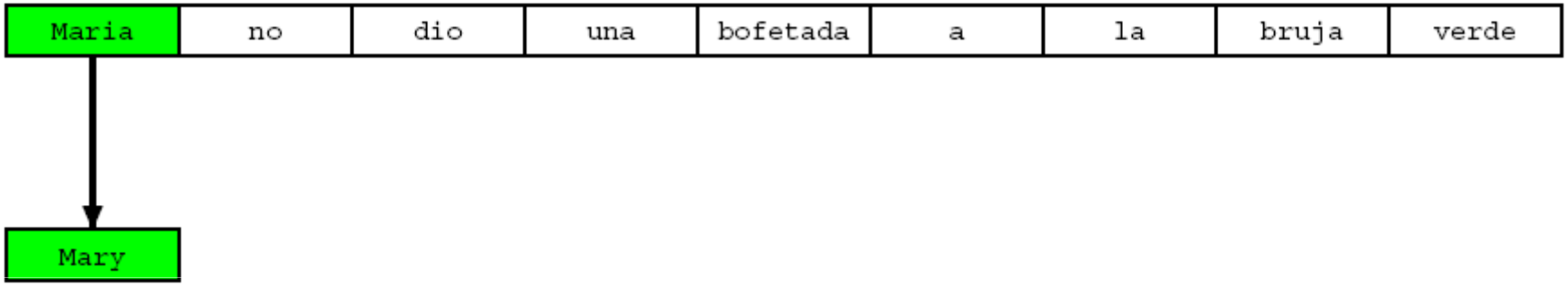
Decoding Process

Maria	no	dio	una	bofetada	a	la	bruja	verde

- Build translation left to right
  - *select foreign* words to be translated

Slide from Koehn 2008

Decoding Process



- Build translation *left to right*
  - select foreign words to be translated
  - *find English* phrase translation
  - *add English* phrase to end of partial translation

Slide from Koehn 2008

Decoding Process

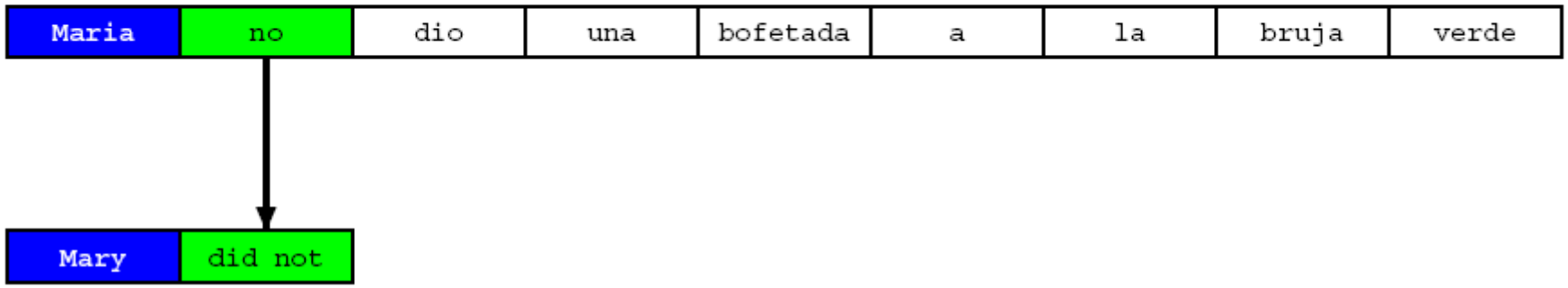
Maria	no	dio	una	bofetada	a	la	bruja	verde

Mary

- Build translation left to right
  - select foreign words to be translated
  - find English phrase translation
  - add English phrase to end of partial translation
  - *mark foreign* words as translated

Slide from Koehn 2008

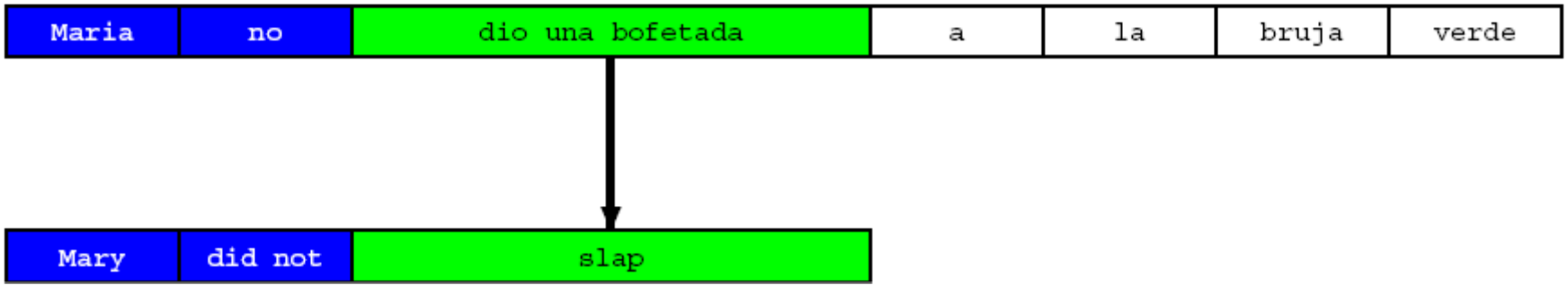
Decoding Process



- *One to many* translation

Slide from Koehn 2008

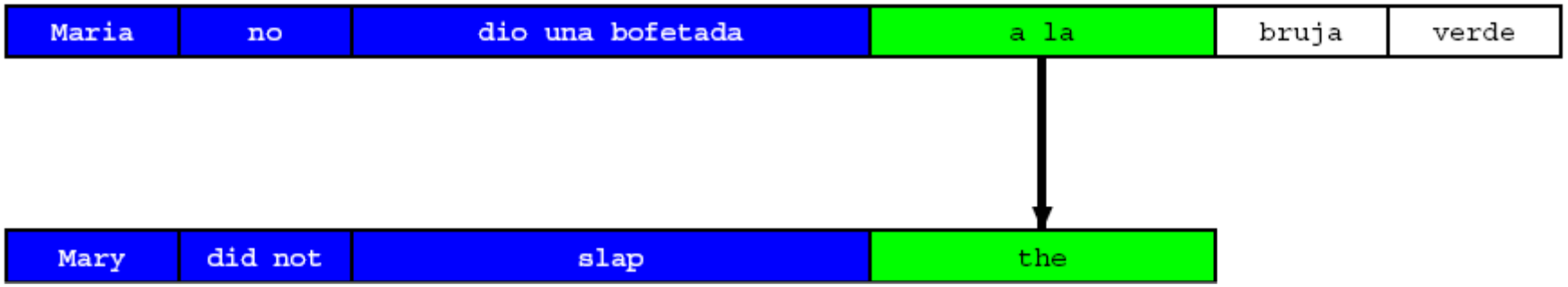
Decoding Process



- Many to one translation

Slide from Koehn 2008

Decoding Process



- *Many to one* translation

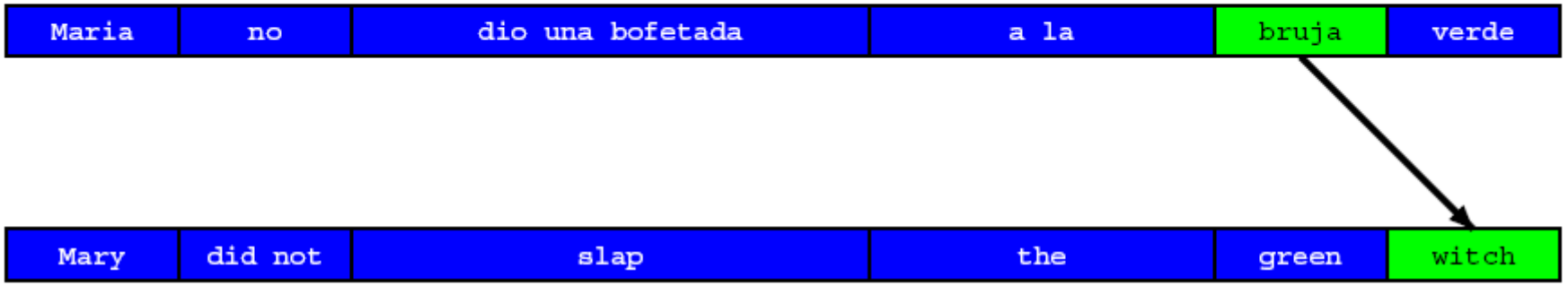
Slide from Koehn 2008

Decoding Process

- *Reordering*

Slide from Koehn 2008

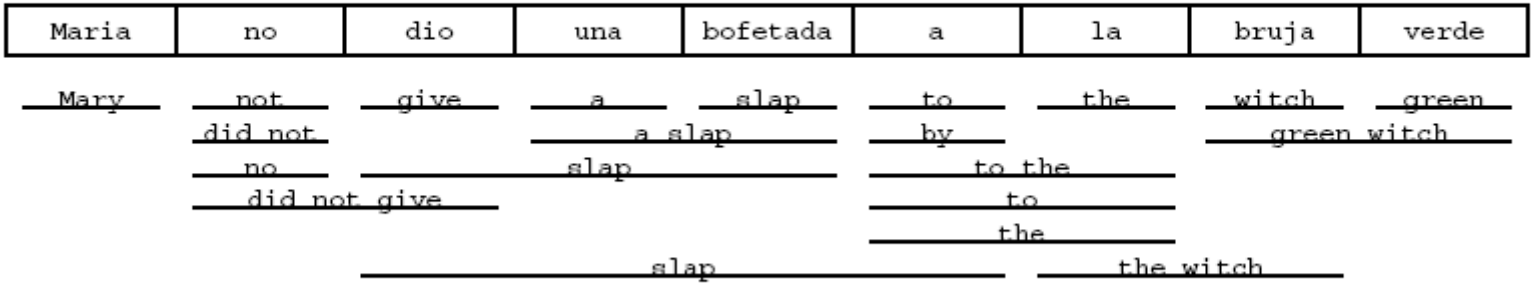
Decoding Process



- Translation *finished*

Slide from Koehn 2008

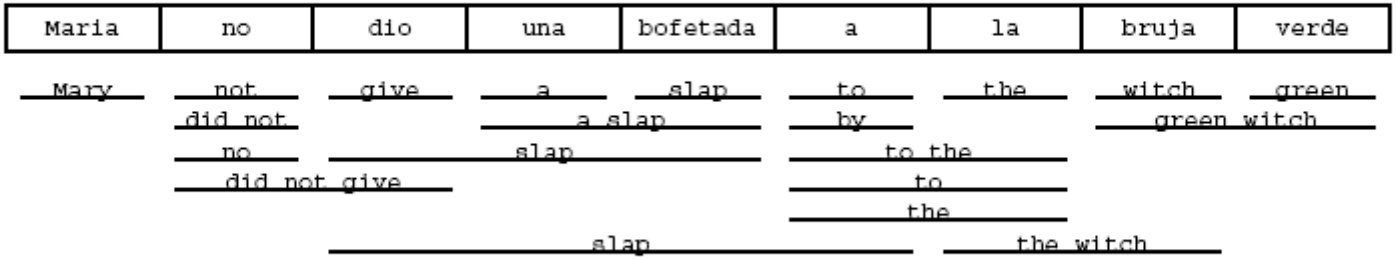
Translation Options

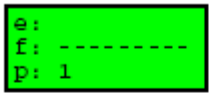


- Look up *possible phrase translations*
  - many different ways to *segment* words into phrases
  - many different ways to *translate* each phrase

Slide from Koehn 2008

Hypothesis Expansion

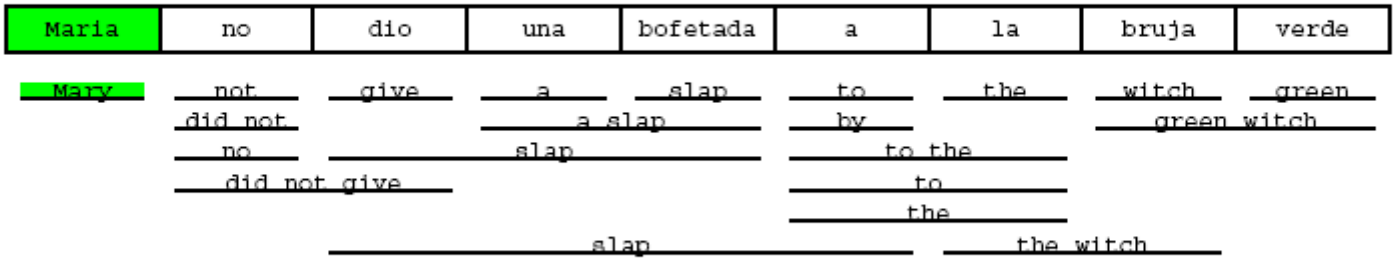


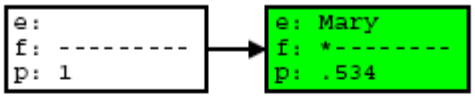


- Start with **empty hypothesis**
  - e: no English words
  - f: no foreign words covered
  - p: probability 1

Slide from Koehn 2008

Hypothesis Expansion

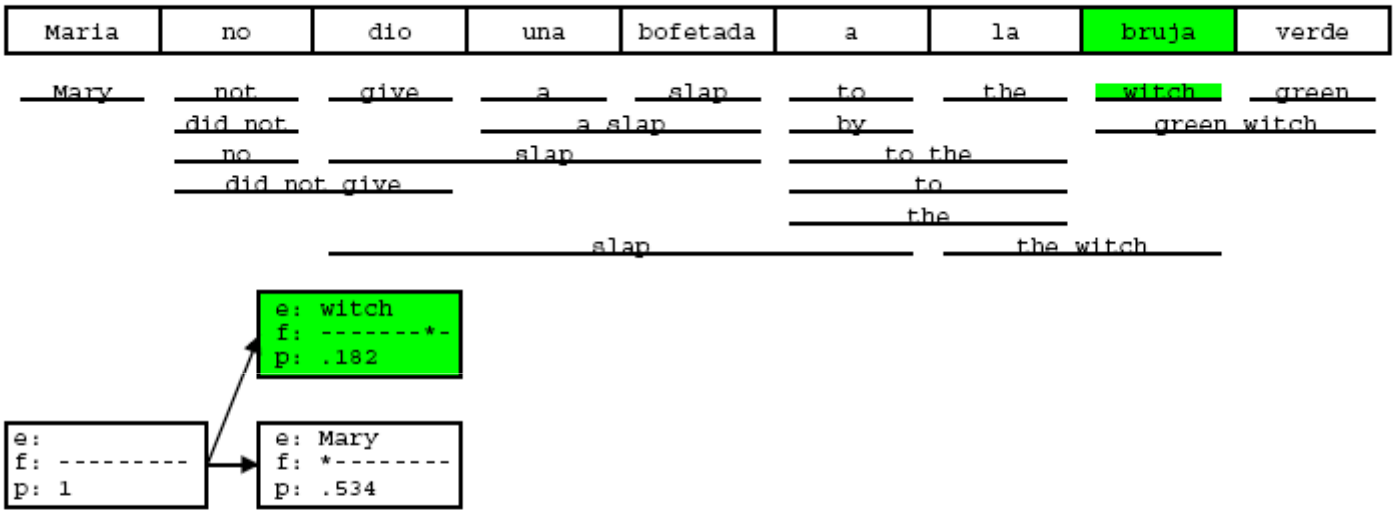




- Pick *translation option*
- Create *hypothesis*
  - e: add English phrase Mary
  - f: first foreign word covered
  - p: probability 0.534

Slide from Koehn 2008

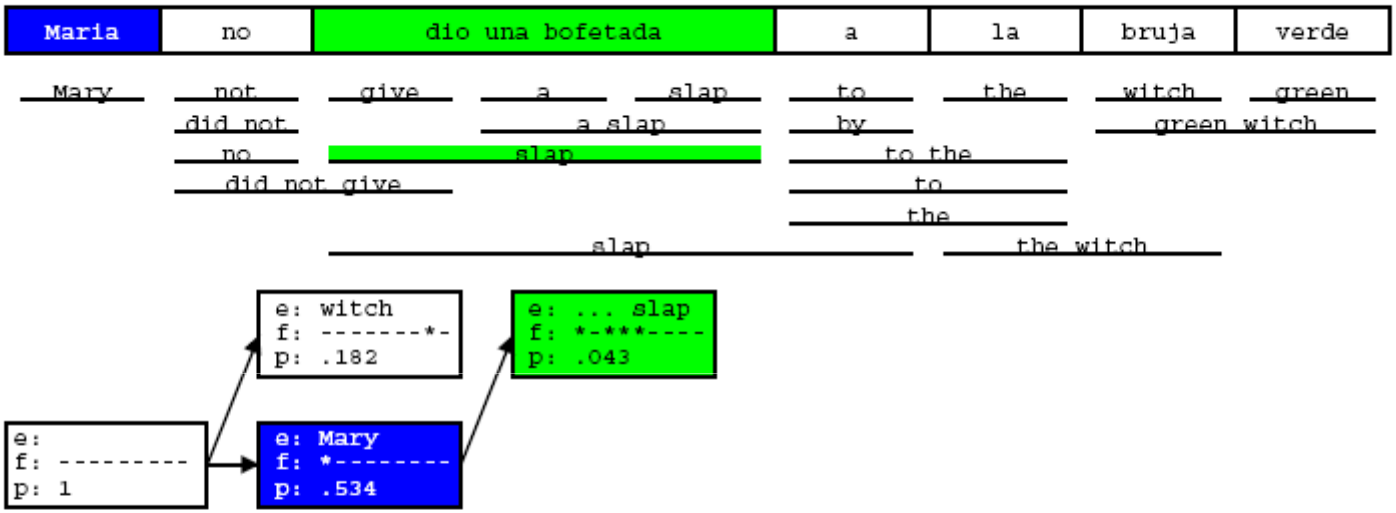
Hypothesis Expansion



- Add another *hypothesis*

Slide from Koehn 2008

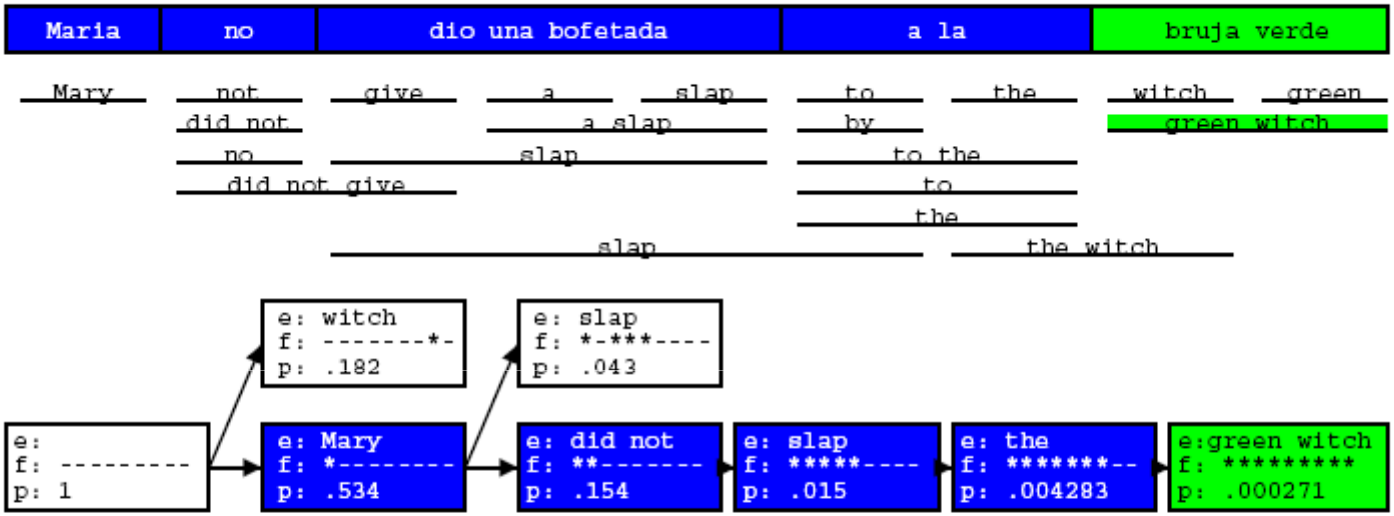
Hypothesis Expansion



- Further *hypothesis expansion*

Slide from Koehn 2008

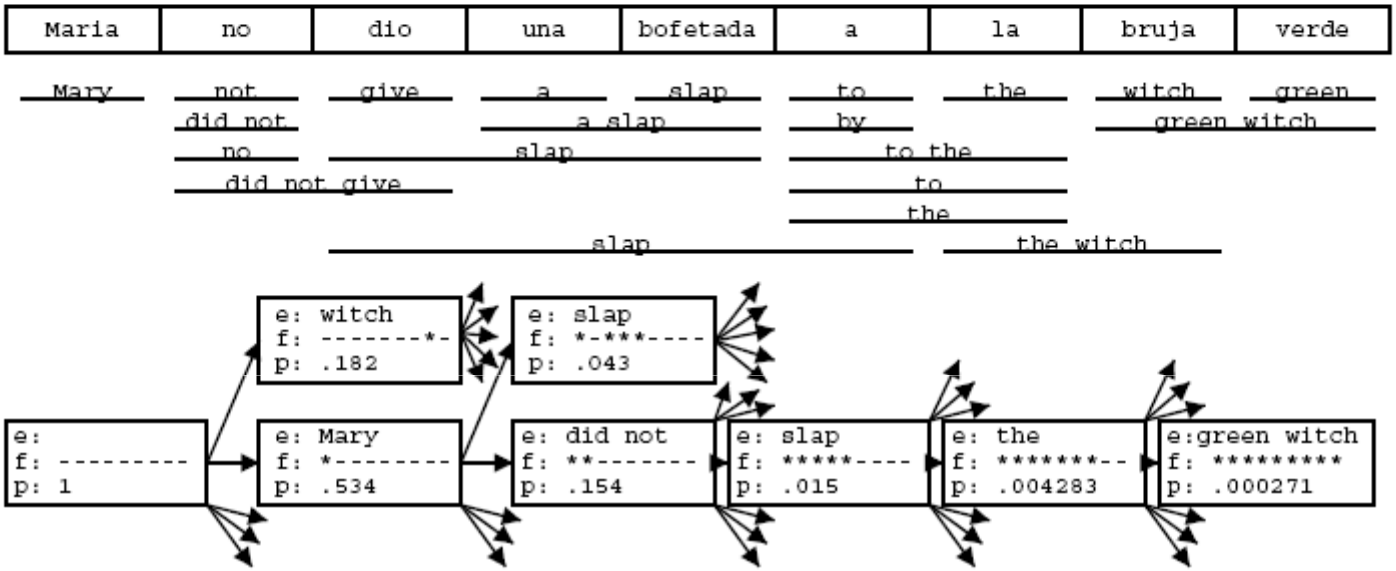
Hypothesis Expansion



- ... until all foreign words *covered*
  - find *best hypothesis* that covers all foreign words
  - *backtrack* to read off translation

Slide from Koehn 2008

Hypothesis Expansion



- Adding more hypothesis

⇒ *Explosion* of search space

Slide from Koehn 2008

Explosion of Search Space

- Number of hypotheses is *exponential* with respect to sentence length

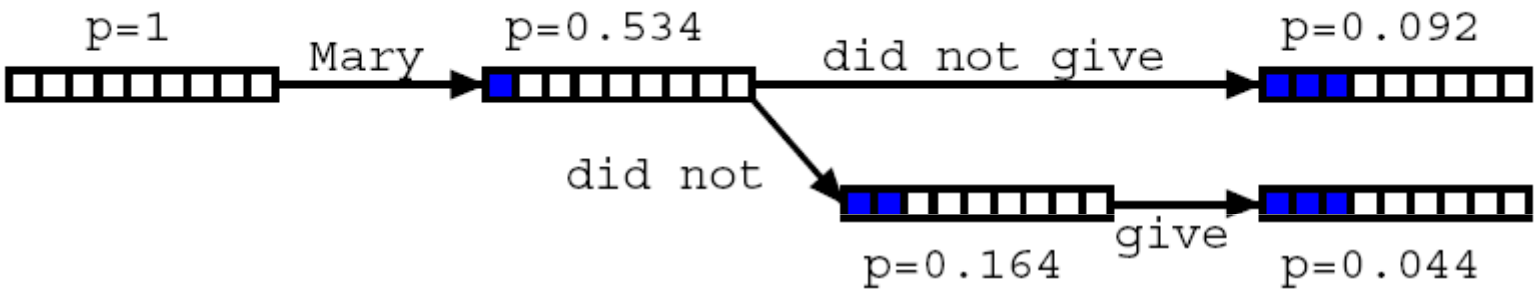
$\Rightarrow$ Decoding is NP-complete [Knight, 1999]

$\Rightarrow$ Need to *reduce search space*

  - risk free: hypothesis **recombination**
  - risky: **histogram/threshold pruning**

Slide from Koehn 2008

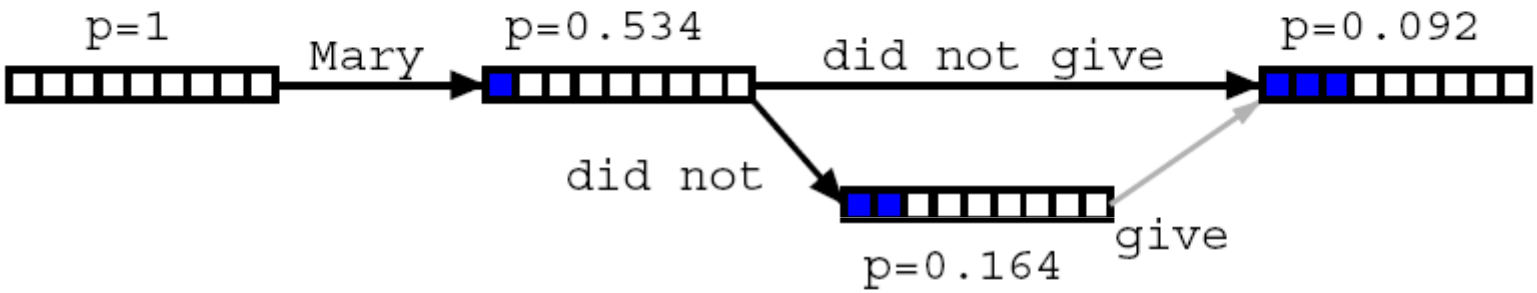
Hypothesis Recombination



- Different paths to the *same* partial translation

Slide from Koehn 2008

Hypothesis Recombination

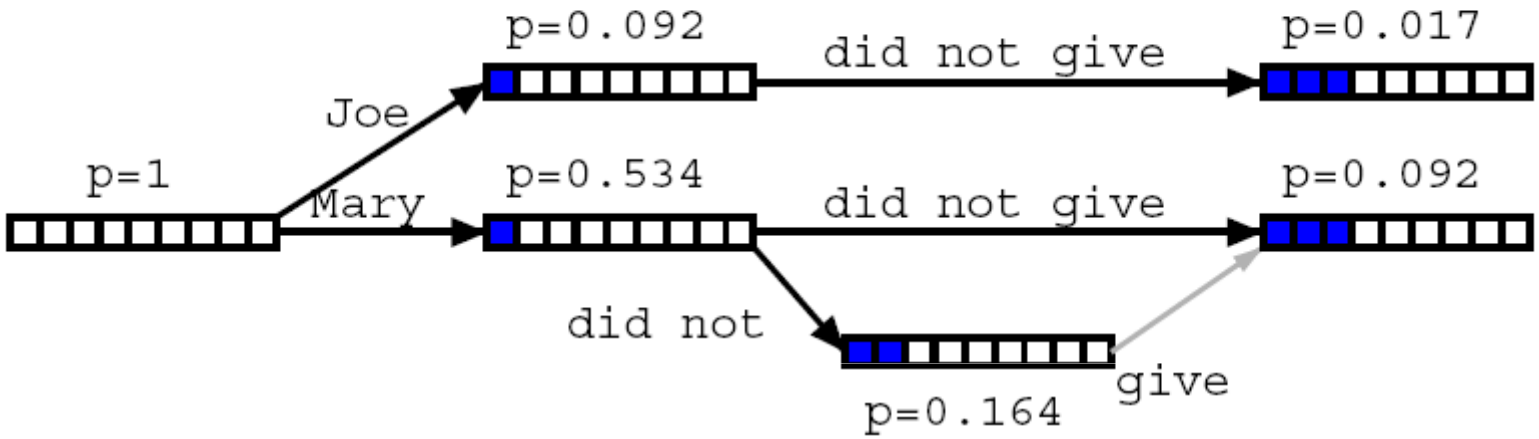


- Different paths to the same partial translation

⇒ *Combine paths*
  - *drop weaker* path
  - keep pointer from weaker path (for lattice generation)

Slide from Koehn 2008

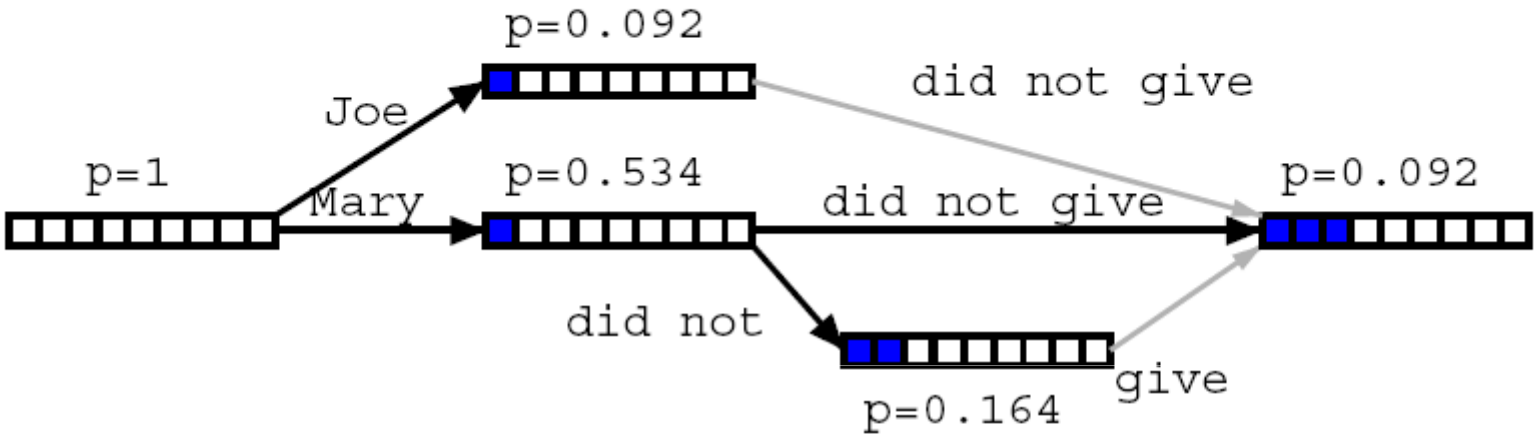
Hypothesis Recombination



- Recombined hypotheses do *not* have to *match completely*
- No matter what is added, weaker path can be dropped, if:
  - *last two English words* match (matters for language model)
  - *foreign word coverage* vectors match (possible future paths are the same)

Modified from Koehn 2008

Hypothesis Recombination



- Recombined hypotheses do not have to match completely
- No matter what is added, weaker path can be dropped, if:
  - last two English words match (matters for language model)
  - foreign word coverage vectors match (possible future paths are the same)

⇒ *Combine paths*

Modified from Koehn 2008

Pruning

- Hypothesis recombination is *not sufficient*

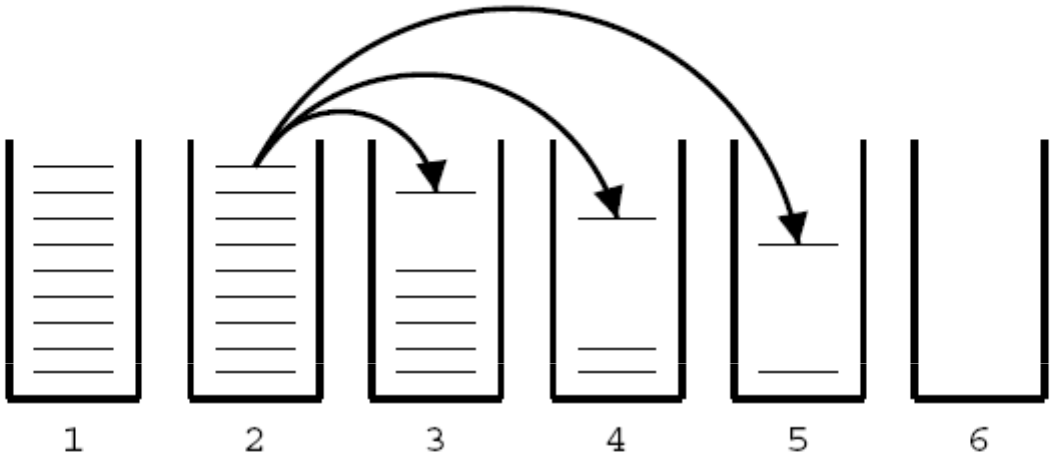
$\Rightarrow$ Heuristically *discard* weak hypotheses early

- Organize Hypothesis in **stacks**, e.g. by
  - *same* foreign words covered
  - *same number* of foreign words covered
  - *same number* of English words produced

- Compare hypotheses in stacks, discard bad ones
  - **histogram pruning**: keep top n hypotheses in each stack (e.g., n=100)
  - **threshold pruning**: keep hypotheses that are at most α times the cost of best hypothesis in stack (e.g., $\alpha = 0.001$)

Slide from Koehn 2008

Hypothesis Stacks

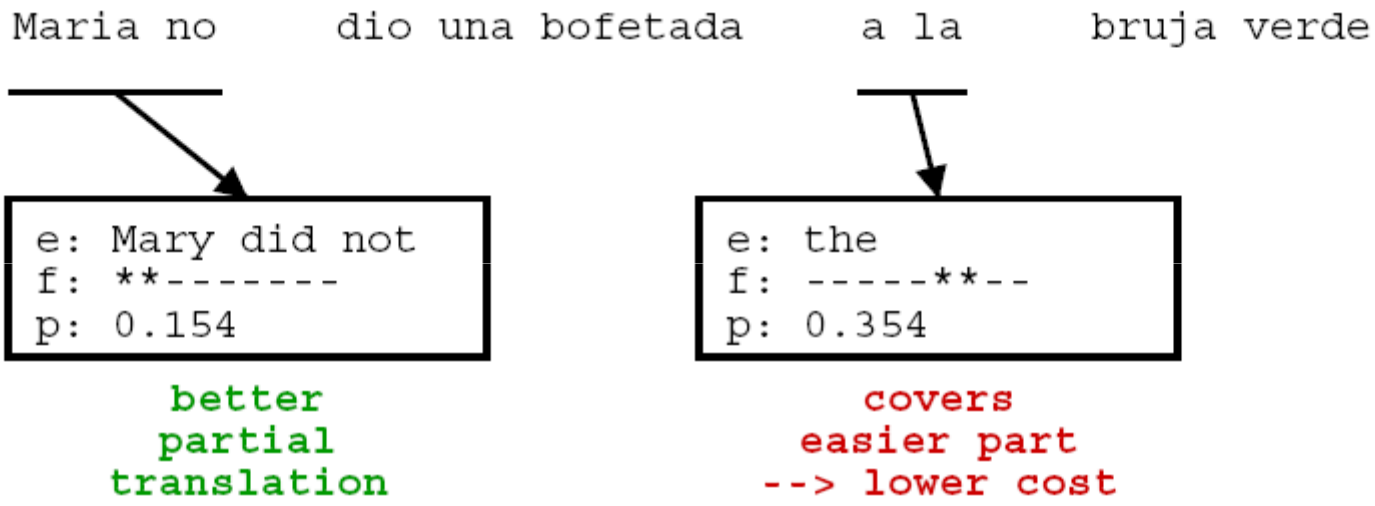


- Organization of hypothesis into stacks
  - here: based on *number of foreign words* translated
  - during translation all hypotheses from one stack are expanded
  - expanded Hypotheses are placed into stacks

Slide from Koehn 2008

Comparing Hypotheses

- Comparing hypotheses with *same number of foreign words* covered

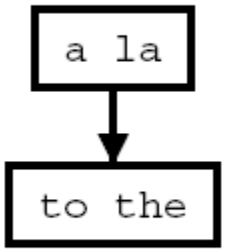


- Hypothesis that covers *easy part* of sentence is preferred

$\Rightarrow$ Need to consider **future cost** of uncovered parts

Slide from Koehn 2008

Future Cost Estimation

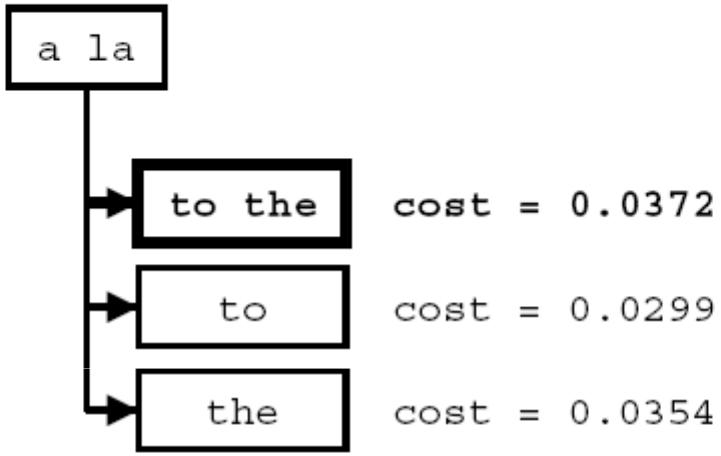


- *Estimate cost* to translate remaining part of input
- Step 1: estimate future cost for each *translation option*
  - look up translation model cost
  - estimate language model cost (no prior context)
  - ignore reordering model cost

  → LM * TM = p(to) * p(the|to) * p(to the|a la)

Slide from Koehn 2008

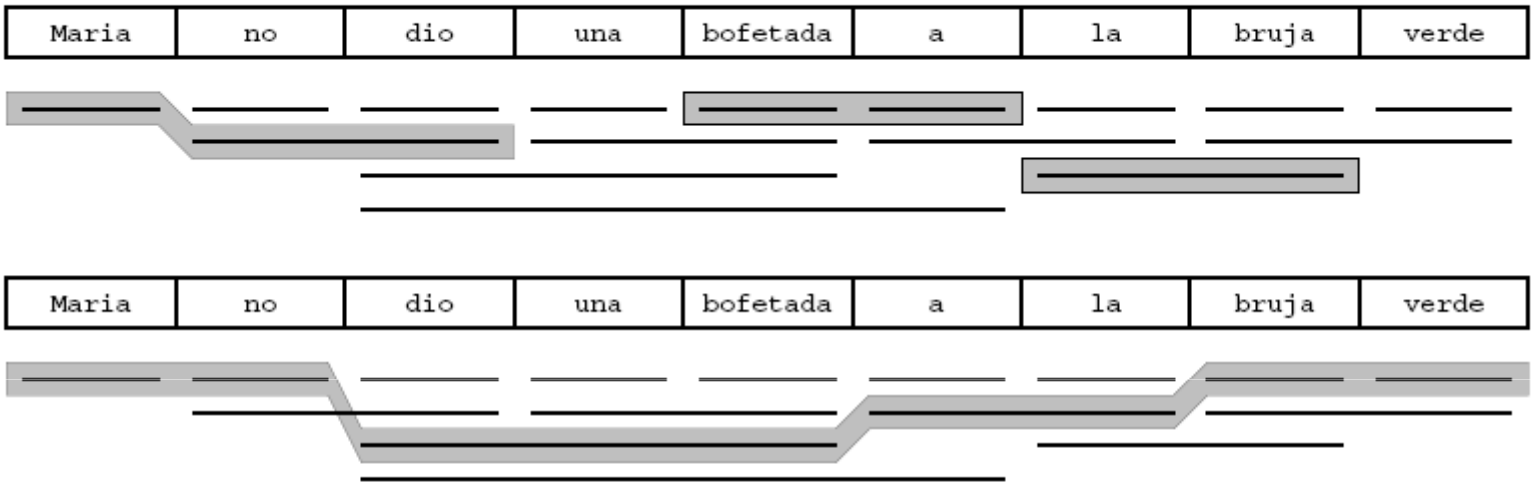
Future Cost Estimation: Step 2



- Step 2: find *cheapest cost* among translation options

Slide from Koehn 2008

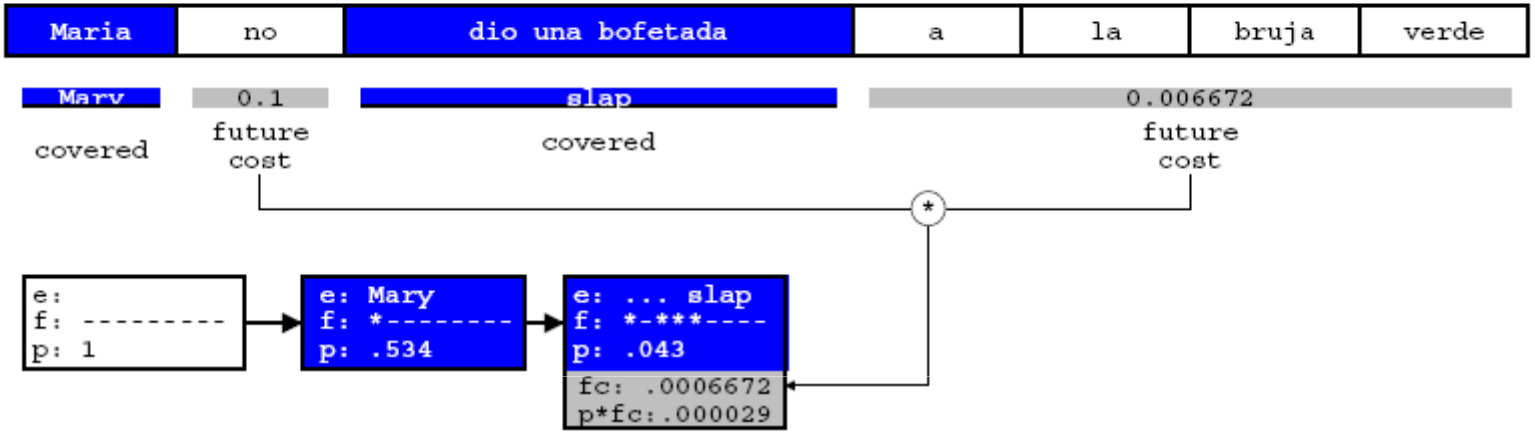
Future Cost Estimation: Step 3



- Step 3: find *cheapest future cost path* for each span
  - can be done *efficiently* by dynamic programming
  - future cost for every span can be *pre-computed*

Slide from Koehn 2008

Future Cost Estimation: Application



- Use future cost estimates when *pruning* hypotheses
- For each *uncovered contiguous span*:
  - look up *future costs* for each maximal contiguous uncovered span
  - *add* to actually accumulated cost for translation option for pruning

Slide from Koehn 2008

A* search

- Pruning might drop hypothesis that lead to the best path (**search error**)
- **A* search**: safe pruning
  - future cost estimates have to be accurate or underestimates
  - **lower bound** for probability is established early by **depth first search**: compute cost for one complete translation
  - if cost-so-far and future cost are worse than *lower bound*, hypothesis can be safely discarded
- Not commonly done, since not aggressive enough

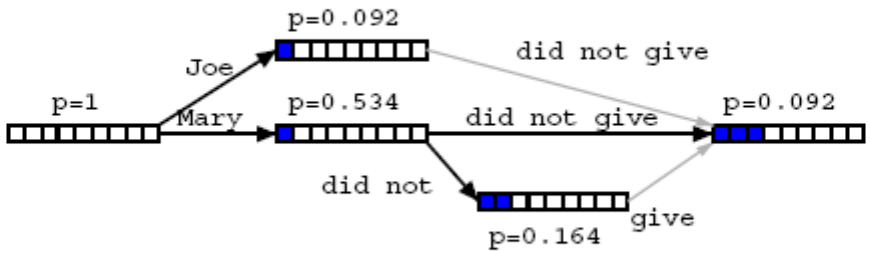
Slide from Koehn 2008

Limits on Reordering

- Reordering may be **limited**
  - **Monotone** Translation: No reordering at all
  - Only phrase movements of at most n words
- Reordering limits *speed* up search (polynomial instead of exponential)
- Current reordering models are weak, so limits *improve* translation quality

Slide from Koehn 2008

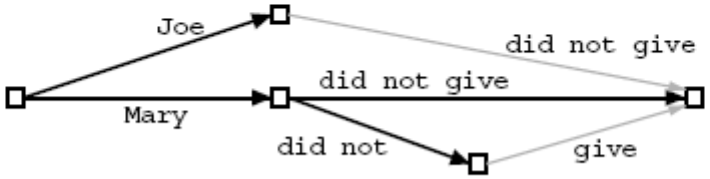
Word Lattice Generation



- **Search graph** can be easily converted into a **word lattice**
  - can be further mined for **n-best lists**

  → enables **reranking** approaches

  → enables **discriminative training**



Slide from Koehn 2008