Universitat Politècnica de València

Master in Artificial Intelligence, Pattern Recognition and Digital Imaging

2022-2023

MACHINE TRANSLATION

2. Statistical Machine Translation

Francisco Casacuberta

fcn@prhlt.upv.es

November 8, 2022

Index

Index

General framework

- Every sentence y in one language is a translation of any sentence x in another language.
- For each possible pair of sentences, y and x, there is a probability $\Pr(\mathrm{y} \mid \mathrm{x})$.
- $\Pr(\mathrm{y} \mid \mathrm{x})$ should be low in the case of:

  $\mathrm{y} =$ *quiero una habitación doble con vistas al mar*
  $\mathrm{x} =$ *are all expenses included in the bill?*

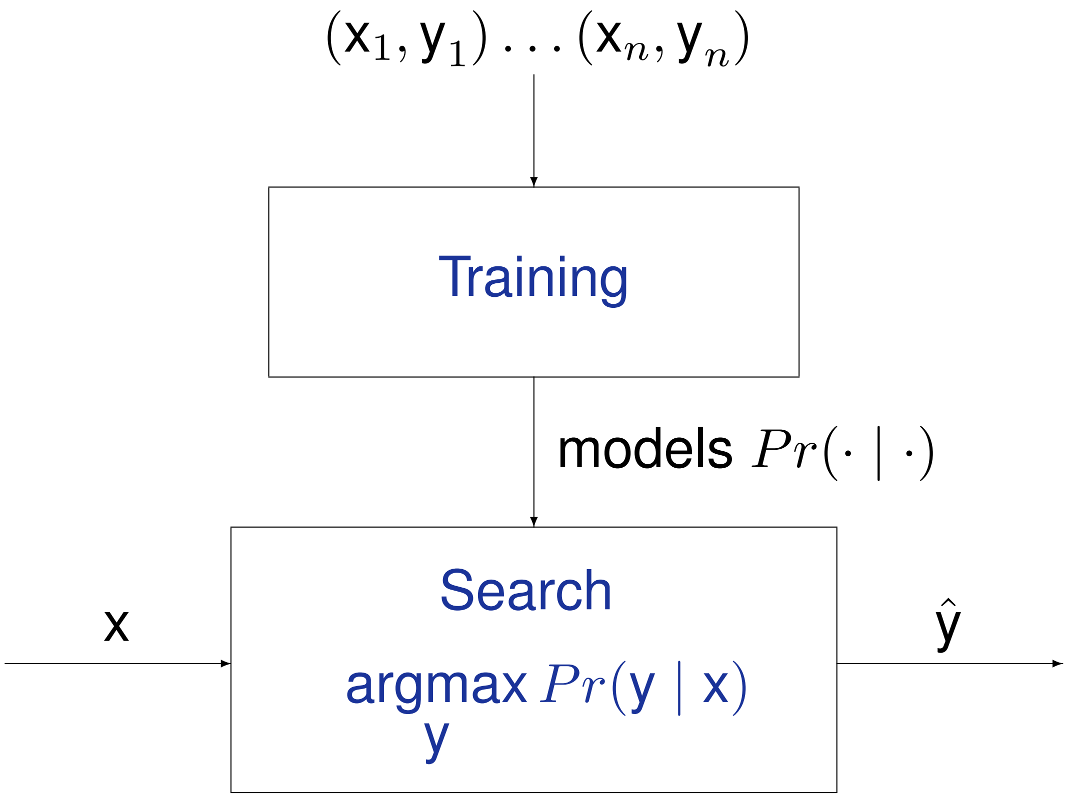
- $\Pr(\mathrm{y} \mid \mathrm{x})$ should be high in the case of:

  $\mathrm{y} =$ *¿ hay alguna habitación tranquila libre ?*
  $\mathrm{x} =$ *is there a quiet room available?*

Training and search

$$x \longrightarrow \boxed{\begin{array}{c}\text{Search} \\ \underset{y}{\operatorname{argmax}}\, Pr(y \mid x)\end{array}} \longrightarrow \hat{y}$$

Training and search



Models

- Language models
- Word-based models
  - Alignment models
  - Stochastic dictionary
- Phrase-based models
  - Phrase tables
  - Lexicalized phrase models
  - Reordering model
  - ...

Index

Language models

In Computational Linguistics (MIARFID)

Word n-grams

$$\Pr(\mathsf{y}) = \prod_{i=1}^{|\mathsf{y}|} \Pr(\mathsf{y}_i \mid \mathsf{y}_1^{i-1}) \approx Pr(\mathsf{y}) = \prod_{i=1}^{|\mathsf{y}|} p_n(\mathsf{y}_i \mid \mathsf{y}_{i-n+1}^{i-1})$$

Regular or context-free grammars

$$\Pr(\mathsf{y}) \approx Pr(\mathsf{y}) = \sum_{d(\mathsf{y})} p_G(d(\mathsf{y})) \approx \max_{d(\mathsf{y})} p_G(d(\mathsf{y}))$$

Neural language models

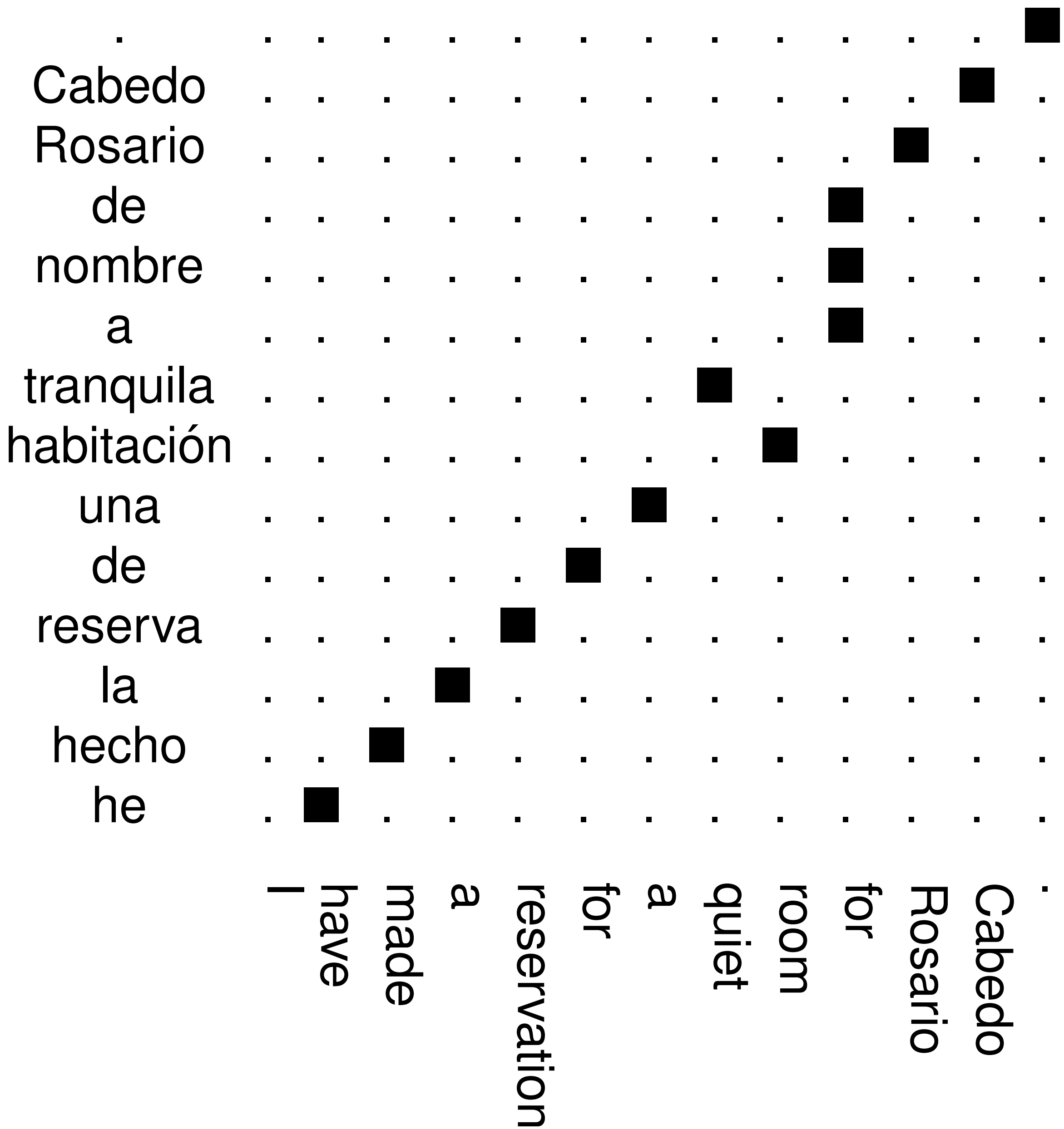
$$\Pr(\mathsf{y}) = \prod_{i=1}^{|\mathsf{y}|} \Pr(\mathsf{y}_i \mid \mathsf{y}_1^{i-1}) \approx Pr(\mathsf{y}) = \prod_{i=1}^{|\mathsf{y}|} p(\mathsf{y}_i \mid \mathsf{y}_1^{i-1})$$

Learning language models

- Probabilistic estimation techniques: SMOOTHING.
- Grammatical inference techniques.
- Widely used statistical toolkits for n-grams:
  - KenLM Language Model Toolkit
    http://kheafield.com/code/kenlm/
  - The IRST Language Modeling Toolkit
    http://sourceforge.net/projects/irstlm/
  - SRILM - The SRI Language Modeling Toolkit
    http://www.speech.sri.com/projects/srilm/
  - The CMU Statistical Language Modeling (SLM) Toolkit
    http://www.speech.cs.cmu.edu/SLM_info.html
- Pre-trained neural language models: BERT (Google) , XLM (Meta), GPT-2 (OpenAI), GPT-3 (OpenAI), BART (Meta), T5 (Google), PaLM (Google), OPT (Meta), BLOOM (BigScience), ...

Index

Example of word alignments

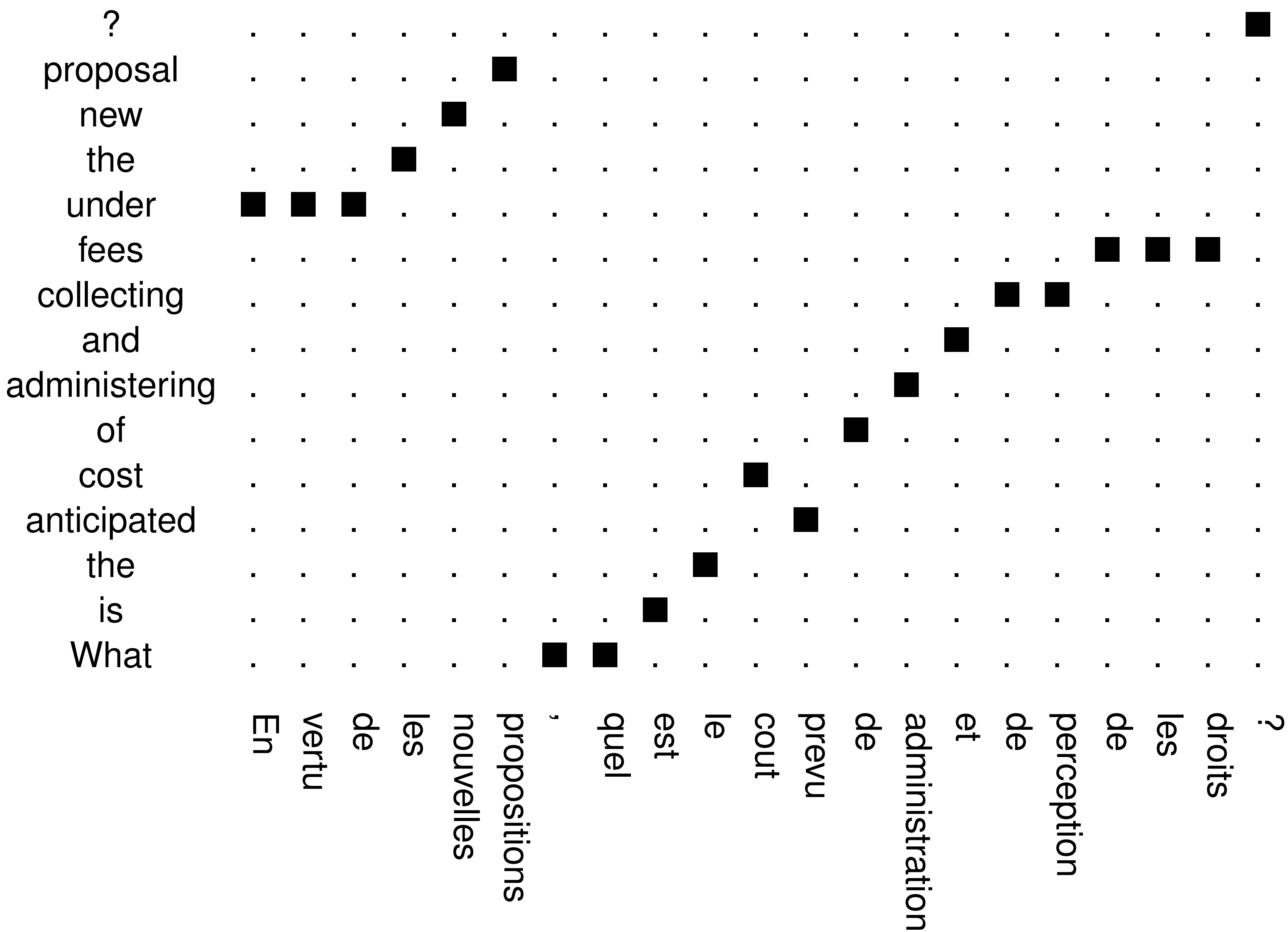


Example of word alignments

	could	you	ask	for	a	taxi	,	please	?
taxi	.	.	.	.	.	■	.	.	.
un	.	.	.	.	■	.	.	.	.
pídame	■	■	■	■	.	.	.	.	.
,	.	.	.	.	.	.	■	.	.
favor	.	.	.	.	.	.	.	■	.
por	.	.	.	.	.	.	.	■	.

Example of word alignments

H. Ney, *Statistical Natural Language Processing*, 2003: Canadian Hansards



Example of word alignments

AMETRA corpus

1996	.	.	■	.	.
de	.	.	■	.	.
marzo	.	.	.	■	.
de	.	.	.	■	.
20	.	.	.	.	■
a	.	.	.	.	■
,	.	■	.	.	.
Lemoa	■	.	.	.	.
En	■	.	.	.	.
	Lemoan	,	1996ko	martxoaren	20an

Example of word alignments

METEO corpus

	sobre	todo	desde	Levante	en	su	mitad	sur
sud	.	.	.	.	.	.	.	■
meitat	.	.	.	.	.	.	■	.
seva	.	.	.	.	.	■	.	.
la	.	.	.	.	.	.	.	.
en	.	.	.	.	■	.	.	.
Llevant	.	.	.	■	.	.	.	.
de	.	.	■	.	.	.	.	.
des	.	.	■	.	.	.	.	.
sobretot	■	■	.	.	.	.	.	.

Alignments

- **Alignments:** (Brown et al. 90) $J = |x|$ y $I = |y|$

$$a \subseteq \{1, ..., J\} \times \{1, ..., I\}$$

  – Number of connections: $I\ J$

  – Number of alignments: $2^{I\ J}$

- Constrain: $a : \{1, ..., I\} \to \{0, ..., J\}$, ($a_i = 0 \Rightarrow \quad i$ in y is not aligned with any position in x).

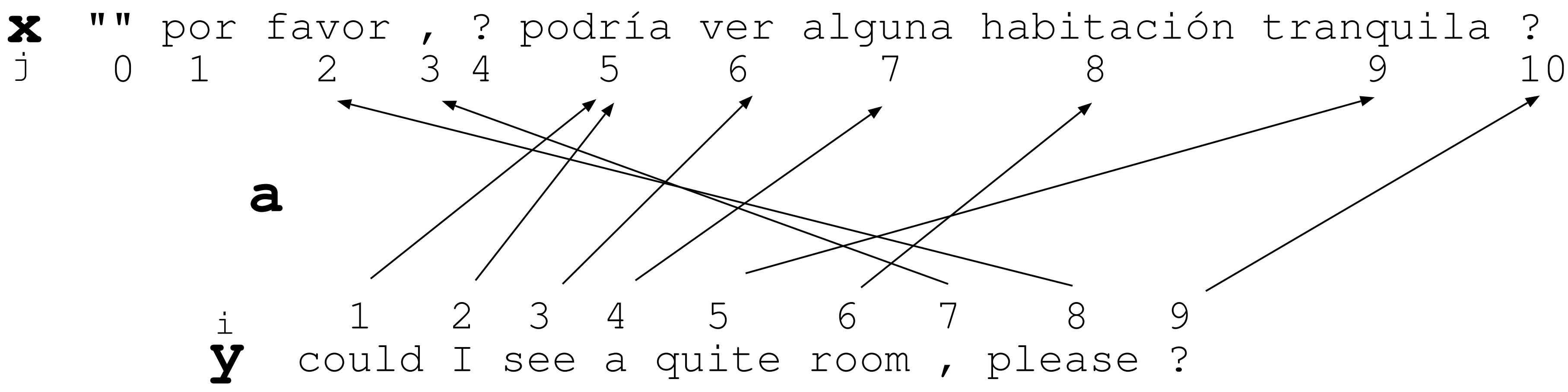
  – Number of alignments: $(J+1)^I$

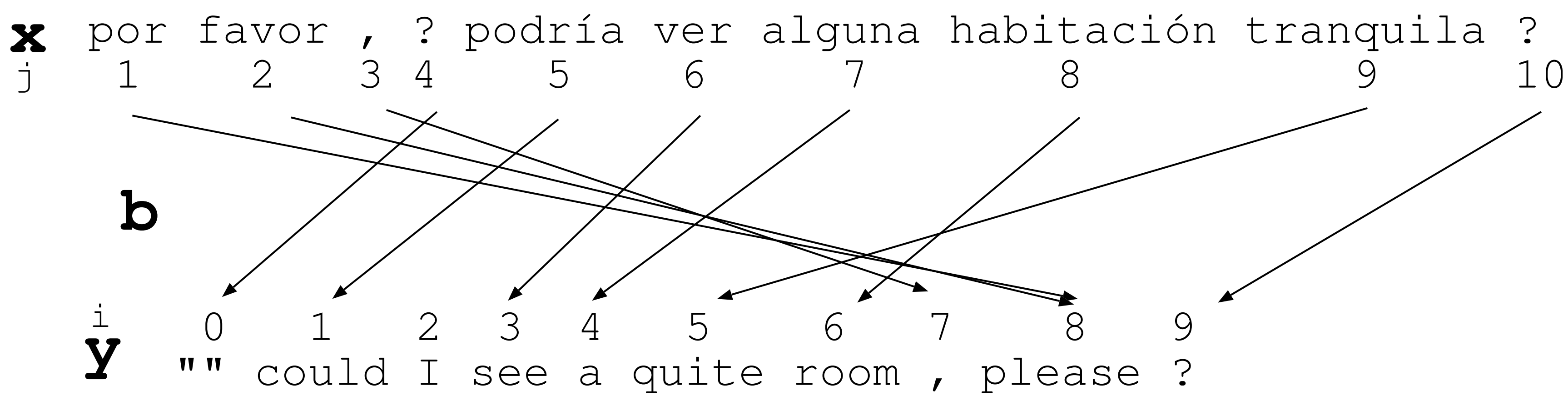
- Inverted alignments: $b : \{1, ..., J\} \to \{0, ..., I\}$,

- Notation:

$$x \equiv x_1, \ldots, x_J \equiv x_1^J$$
$$y \equiv y_1, \ldots, y_I \equiv y_1^I$$
$$a \equiv a_1, \ldots, a_I \equiv a_1^I$$

Alignments


x "" por favor , ? podría ver alguna habitación tranquila ?
j 0 1 2 3 4 5 6 7 8 9 10
a
i 1 2 3 4 5 6 7 8 9
y could I see a quite room , please ?


x por favor , ? podría ver alguna habitación tranquila ?
j 1 2 3 4 5 6 7 8 9 10
b
i 0 1 2 3 4 5 6 7 8 9
y "" could I see a quite room , please ?

Alignments

- Set of possible alignments: $\mathcal{A}(\mathsf{x},\mathsf{y}) = \{\mathsf{a} : \{1,...,I\} \to \{0,...,J\}\}$

- The probability of translation x to y through an alignment a is $\Pr(\mathsf{y},\mathsf{a} \mid \mathsf{x})$

$$\begin{aligned}
\Pr(\mathsf{y} \mid \mathsf{x}) &= \Pr(\mathsf{y}, I \mid \mathsf{x}) \\
&= \Pr(I \mid \mathsf{x})\ \Pr(\mathsf{y} \mid I, \mathsf{x}) \\
&= \Pr(I \mid \mathsf{x}) \sum_{\mathsf{a} \in \mathcal{A}(\mathsf{x},\mathsf{y})} \Pr(\mathsf{y}, \mathsf{a} \mid I, \mathsf{x}) \\
&= \Pr(I \mid \mathsf{x}) \sum_{\mathsf{a} \in \mathcal{A}(\mathsf{x},\mathsf{y})} \Pr(\mathsf{a} \mid I, \mathsf{x})\ \Pr(\mathsf{y} \mid \mathsf{a}, I, \mathsf{x})
\end{aligned}$$

- Length probability: $\Pr(I \mid \mathsf{x}) \approx \mathcal{N}(I \mid J)$

Alignments

$$\Pr(\mathsf{y},\mathsf{a} \mid I,\mathsf{x}) = \Pr(\mathsf{a} \mid I,\mathsf{x})\ \Pr(\mathsf{y} \mid \mathsf{a},I,\mathsf{x})$$

$$\Pr(\mathsf{a} \mid I,\mathsf{x}) = \prod_{i=1}^{I} \Pr(\mathsf{a}_i \mid \mathsf{a}_1^{i-1}, I, \mathsf{x}) \qquad \Pr(\mathsf{y} \mid \mathsf{a},I,\mathsf{x}) = \prod_{i=1}^{I} \Pr(\mathsf{y}_i \mid \mathsf{y}_1^{i-1}, \mathsf{a}, I, \mathsf{x})$$

$$\Pr(\mathsf{y},\mathsf{a} \mid I,\mathsf{x}) = \prod_{i=1}^{I} \Pr(\mathsf{a}_i \mid \mathsf{a}_1^{i-1}, I, \mathsf{x})\ \Pr(\mathsf{y}_i \mid \mathsf{y}_1^{i-1}, \mathsf{a}, I, \mathsf{x})$$

$$\Pr(\mathsf{y} \mid \mathsf{x}) = \Pr(I \mid \mathsf{x}) \sum_{\mathsf{a}\in\mathcal{A}(\mathsf{x},\mathsf{y})} \prod_{i=1}^{I} \Pr(\mathsf{a}_i \mid \mathsf{a}_1^{i-1}, I, \mathsf{x})\ \Pr(\mathsf{y}_i \mid \mathsf{y}_1^{i-1}, \mathsf{a}, I, \mathsf{x})$$

Alignments

$$\Pr(\mathsf{y}, \mathsf{a} \mid I, \mathsf{x}) = \prod_{i=1}^{I} \Pr(\mathsf{a}_i \mid \mathsf{a}_1^{i-1}, I, \mathsf{x}) \;\; \Pr(\mathsf{y}_i \mid \mathsf{y}_1^{i-1}, \mathsf{a}, I, \mathsf{x})$$

- Alignment probability: $\Pr(\mathsf{a}_i \mid \mathsf{a}_1^{i-1}, I, \mathsf{x})$
- Lexicon probability: $\Pr(\mathsf{y}_i \mid \mathsf{y}_1^{i-1}, \mathsf{a}, I, \mathsf{x})$

- Zero-order translation models
  - Model 1
  - Model 2
  - Fertility: Model 3
- First-order translation models
  - Hidden Markov models
  - Model 4
  - Model 5
  - Model 6

Model 1

$$\Pr(\mathsf{y}, \mathsf{a} \mid I, \mathsf{x}) = \prod_{i=1}^{I} \Pr(\mathsf{a}_i \mid \mathsf{a}_1^{i-1}, I, \mathsf{x}) \; \Pr(\mathsf{y}_i \mid \mathsf{y}_1^{i-1}, \mathsf{a}, I, \mathsf{x})$$

- $\Pr(\mathsf{a}_i \mid \mathsf{a}_1^{i-1}, I, \mathsf{x}) \approx \frac{1}{(J+1)}$
- $\Pr(\mathsf{y}_i \mid \mathsf{y}_1^{i-1}, \mathsf{a}, I, \mathsf{x}) \approx l(\mathsf{y}_i \mid \mathsf{x}_{\mathsf{a}_i})$

- $l(\mathsf{y}_i \mid \mathsf{x}_j)$ defines a **statistical lexicon**

$$\Pr(\mathsf{y} \mid \mathsf{x}) = \Pr(I \mid \mathsf{x}) \sum_{\mathsf{a}} \Pr(\mathsf{y}, \mathsf{a} \mid I, \mathsf{x}) \approx P_{M1}(\mathsf{y} \mid \mathsf{x}) = \frac{\mathcal{N}(I \mid J)}{(J+1)^I} \prod_{i=1}^{I} \sum_{j=0}^{J} l(\mathsf{y}_i \mid \mathsf{x}_j)$$

Model 1

$$
\begin{aligned}
\Pr(\mathsf{y} \mid \mathsf{x}) &= \Pr(I \mid \mathsf{x}) \sum_{\mathsf{a}} \Pr(\mathsf{y}, \mathsf{a} \mid I, \mathsf{x}) \\
&\approx \mathcal{N}(I \mid J) \sum_{\mathsf{a}} \prod_{i=1}^{I} \left[\frac{1}{(J+1)}\, l(\mathsf{y}_i \mid \mathsf{x}_{\mathsf{a}_i}) \right] \\
&= \frac{\mathcal{N}(I \mid J)}{(J+1)^I} \sum_{\mathsf{a}_1=0}^{J} \cdots \sum_{\mathsf{a}_I=0}^{J} \prod_{i=1}^{I} l(\mathsf{y}_i \mid \mathsf{x}_{\mathsf{a}_i}) \\
&= \frac{\mathcal{N}(I \mid J)}{(J+1)^I} \prod_{i=1}^{I} \sum_{\mathsf{a}_i=0}^{J} l(\mathsf{y}_i \mid \mathsf{x}_{\mathsf{a}_i}) \\
&= \frac{\mathcal{N}(I \mid J)}{(J+1)^I} \prod_{i=1}^{I} \sum_{j=0}^{J} l(\mathsf{y}_i \mid \mathsf{x}_j) = P_{M1}(\mathsf{y} \mid \mathsf{x})
\end{aligned}
$$

Model 1: parameter estimation

- Training sample: $A = \{(\mathsf{x}^{(1)}, \mathsf{y}^{(1)}), (\mathsf{x}^{(2)}, \mathsf{y}^{(2)}), \ldots, (\mathsf{x}^{(N)}, \mathsf{y}^{(N)})\}$
- Goal: maximize the likelihood (or log-likelihood)

$$\operatorname*{argmax}_{l} \mathcal{L}_A(l) = \operatorname*{argmax}_{l} \log \prod_{n=1}^{N} P_{M1}(\mathsf{y}^{(n)} \mid \mathsf{x}^{(n)}) = \operatorname*{argmax}_{l} \sum_{n=1}^{N} \log P_{M1}(\mathsf{y}^{(n)} \mid \mathsf{x}^{(n)})$$

- Procedure: Expectation-maximization or growth transformations:

Initialize l and counts $c(y,x)$ for all source and target words y and x

Iterate

For all training sample $(\mathsf{x}^{(n)}, \mathsf{y}^{(n)}) \in A$

For all source word $x \in \mathsf{x}^{(n)}$ and target word $y \in \mathsf{y}^{(n)}$

Counting step: $c(y,x) = c(y,x) + \dfrac{l(y \mid x)\ \#(x, \mathsf{x}^{(n)})\ \#(y, \mathsf{y}^{(n)})}{\sum_{j=0}^{J^{(n)}} l(y \mid \mathsf{x}_j^{(n)})}$

For all source word $x \in L_X$ and target word $y \in L_Y$

Normalization step: $l(y \mid x) = \dfrac{c(y,x)}{\sum_{y'} c(y', x)}$

until convergence

Parameter estimation in Model 1

- PROPERTY: the increase of the likelihood of the training set in each iteration:

$$\prod_{n=1}^{N} P_{M1(k)}(\mathsf{y}^{(n)} \mid \mathsf{x}^{(n)}) \leq \prod_{n=1}^{N} P_{M1(k+1)}(\mathsf{y}^{(n)} \mid \mathsf{x}^{(n)})$$

- PROPERTY: eventually an absolute maximum is achieved!
- COMPUTATIONAL COST : if $I_M = \max_n I^{(n)}$ y $J_M = \max_n J^{(n)}$
  - time: $O(N \times (I_M + J_M))$
  - space: $O(|L_X| \times |L_Y|)$
- Public software for training Model 1: `https://github.com/moses-smt/mgiza`

Model 2

$$\Pr(\mathsf{y}, \mathsf{a} \mid I, \mathsf{x}) = \prod_{i=1}^{I} \Pr(\mathsf{a}_i \mid \mathsf{a}_1^{i-1}, I, \mathsf{x}) \; \Pr(\mathsf{y}_i \mid \mathsf{y}_1^{i-1}, \mathsf{a}, I, \mathsf{x})$$

- $\Pr(\mathsf{a}_i \mid \mathsf{a}_1^{i-1}, I, \mathsf{x}) \approx a(\mathsf{a}_i \mid i, I, J)$
- $\Pr(\mathsf{y}_i \mid \mathsf{y}_1^{i-1}, \mathsf{a}, I, \mathsf{x}) \approx l(\mathsf{y}_i \mid \mathsf{x}_{\mathsf{a}_i})$

- $l(\mathsf{y}_i \mid \mathsf{x}_j)$ defines a **statistical lexicon**
- $a(j \mid i, I, J)$ defines **statistical alignments**

$$\Pr(\mathsf{y} \mid \mathsf{x}) \approx P_{M2}(\mathsf{y} \mid \mathsf{x}) = \mathcal{N}(I \mid J) \prod_{i=1}^{I} \sum_{j=0}^{J} a(j \mid i, I, J) \; l(\mathsf{y}_i \mid \mathsf{x}_j) \qquad \text{(Exercise)}$$

Parameter estimation (l and a):

Public software for training Model 2: `https://github.com/moses-smt/mgiza`

Optimal alignment with Model 2

Search for the "best" alignment from $\mathcal{A}(\mathsf{x},\mathsf{y})$

$$\begin{aligned}
\Pr(\mathsf{y} \mid \mathsf{x}) &= \Pr(I \mid \mathsf{x}) \sum_{\mathsf{a}\in\mathcal{A}(\mathsf{y},\mathsf{x})} \Pr(\mathsf{y},\mathsf{a} \mid I,\mathsf{x}) \\
&\approx \Pr(I \mid \mathsf{x}) \max_{\mathsf{a}\in\mathcal{A}(\mathsf{y},\mathsf{x})} \Pr(\mathsf{y},\mathsf{a} \mid I,\mathsf{x}) = \widehat{\Pr}(\mathsf{y} \mid \mathsf{x})
\end{aligned}$$

Using Model 2,

$$\begin{aligned}
\widehat{\Pr}(\mathsf{y} \mid \mathsf{x}) &= \Pr(I \mid \mathsf{x}) \max_{\mathsf{a}} \Pr(\mathsf{y},\mathsf{a} \mid I,\mathsf{x}) \\
&\approx \mathcal{N}(I \mid J) \max_{\mathsf{a}} \prod_{i=1}^{I} \left[a(\mathsf{a}_i \mid i, I, J)\; l(\mathsf{y}_i \mid \mathsf{x}_{\mathsf{a}_i})\right] \\
&= \mathcal{N}(I \mid J) \prod_{i=1}^{I} \max_{0\leq j\leq J} \left[a(j \mid i, I, J)\; l(\mathsf{y}_i \mid \mathsf{x}_j)\right] = \widehat{P}_{M2}(\mathsf{y} \mid \mathsf{x})
\end{aligned}$$

Optimal alignment with Model 2

Algorithm Viterbi $(\mathsf{x}, \mathsf{y}, l, a)$
Input: A pair x, y and the parameters l and a of Model 2
Output: An optimal alignment A between x and y.

For $i = 1$ **until** I

$$A[i] := \operatorname*{argmax}_{0 \leq j \leq J} \left[a(j \mid i, I, J) \; l(\mathsf{y}_i \mid \mathsf{x}_j) \right]$$

End-for
Return: A

- The computational cost of this algorithm is $O(J \times I)$.
- Public software for training Models 1 and 2 and for computing the optimal alignments: https://github.com/moses-smt/mgiza
- Optimal alignments can be used in an alternative training procedure: Viterbi estimation (exercise)

Fast_align: An alternative to alignment with Model 2 [Dyer NAACL 2013]

$a(j \mid i, I, J)$ is not a table[1]

$$h(i, j, I, J) = -\left|\frac{i}{I} - \frac{j}{J}\right|$$

$$a(j \mid i, I, J) = \begin{cases} p_0 & j = 0 \\ (1 - p_0)\,\frac{\exp(\lambda\ h(i,j,I,J))}{Z_\lambda(i,I,J)} & 0 < j \le J \\ 0 & \text{otherwise} \end{cases}$$

$$Z_\lambda(i, I, J) = \sum_{j=1}^{J} \exp(\lambda\ h(i, j, I, J))$$

$Z_\lambda(i, I, J)$ can be computed in $O(1)$[1]

[1] http://github.com/clab/fast_align

Examples of alignments

EuTrans-I corpus: Spanish-English

- Vocabulary: 680 Spanish words, and 513 English words.
- Training: 10,000 pairs (97,000/99,000 words).

An example

1	2	3	4	5	6	7	8	9	10
por	favor	,	¿	podría	ver	alguna	habitación	tranquila	?

- Model 1, Iteration 5
  could (5) I (6) see (6) a (7) quiet (9) room (8) , (3) please (2) ? (4)
- Model 2, Iteration 2
  could (5) I (6) see (6) a (7) quiet (9) room (8) , (3) please (3) ? (10)

Homogeneous HMM alignment

$$\Pr(\mathsf{y}, \mathsf{a} \mid I, \mathsf{x}) = \prod_{i=1}^{I} \Pr(\mathsf{a}_i \mid \mathsf{a}_1^{i-1}, \mathsf{y}_1^{i-1}, I, \mathsf{x}) \; \Pr(\mathsf{y}_i \mid \mathsf{a}_1^{i}, \mathsf{y}_1^{i-1}, I, \mathsf{x})$$

- $\Pr(\mathsf{a}_i \mid \mathsf{a}_1^{i-1}, \mathsf{y}_1^{i-1}, I, \mathsf{x}) \approx h(\mathsf{a}_i \mid \mathsf{a}_{i-1}, I, J)$
- $\Pr(\mathsf{y}_i \mid \mathsf{a}_1^{i}, \mathsf{y}_1^{i-1}, I, \mathsf{x}) \approx l(\mathsf{y}_i \mid \mathsf{x}_{\mathsf{a}_i})$

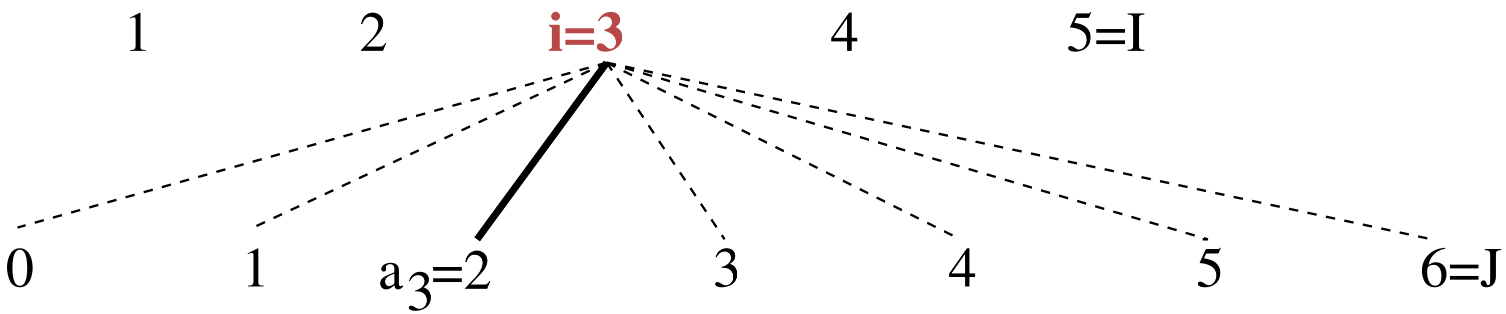
$h(\mathsf{a}_i \mid \mathsf{a}_{i-1}, I, J)$ defines statistical alignment with first-order dependencies

$$h(j \mid j', I, J) = \frac{q(j - j')}{\sum_{j''=1}^{J} q(j'' - j')}$$

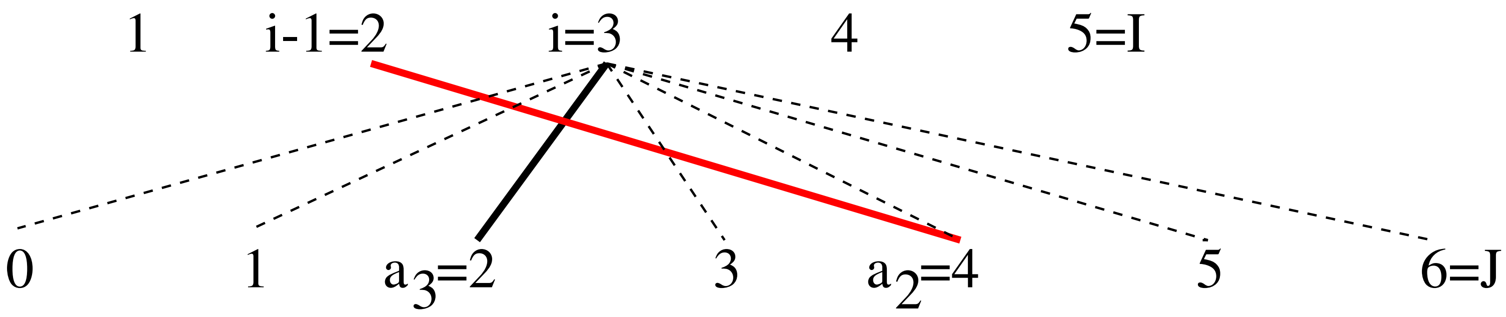
A set of non-negative parameters $q(j - j')$

Homogeneous HMM alignment

A comparison between alignments in M2 and HMM



$\mathfrak{a}(a_3=2 \mid \mathbf{i=3}, I=5, J=6)$



$h(a_3=2 \mid \mathbf{a_2=4}, I=5, J=6)$

Homogeneous HMM alignment

$$P_{HMM}(\mathsf{y} \mid \mathsf{x}) = \mathcal{N}(I \mid J) \sum_{\mathsf{a}} \prod_{i=1}^{I} h(\mathsf{a}_i \mid \mathsf{a}_{i-1}, I, J) \; l(\mathsf{y}_i \mid \mathsf{x}_{\mathsf{a}_i})$$

- Forward computation of $P_{HMM}(\mathsf{y} \mid \mathsf{x})$: $P_{HMM}(\mathsf{y} \mid \mathsf{x}) = \mathcal{N}(I \mid J) \; Q(I, J)$ with

$$Q(i, j) = l(\mathsf{y}_i \mid \mathsf{x}_j) \sum_{j'} h(j \mid j', I, J) \; Q(i-1, j')$$

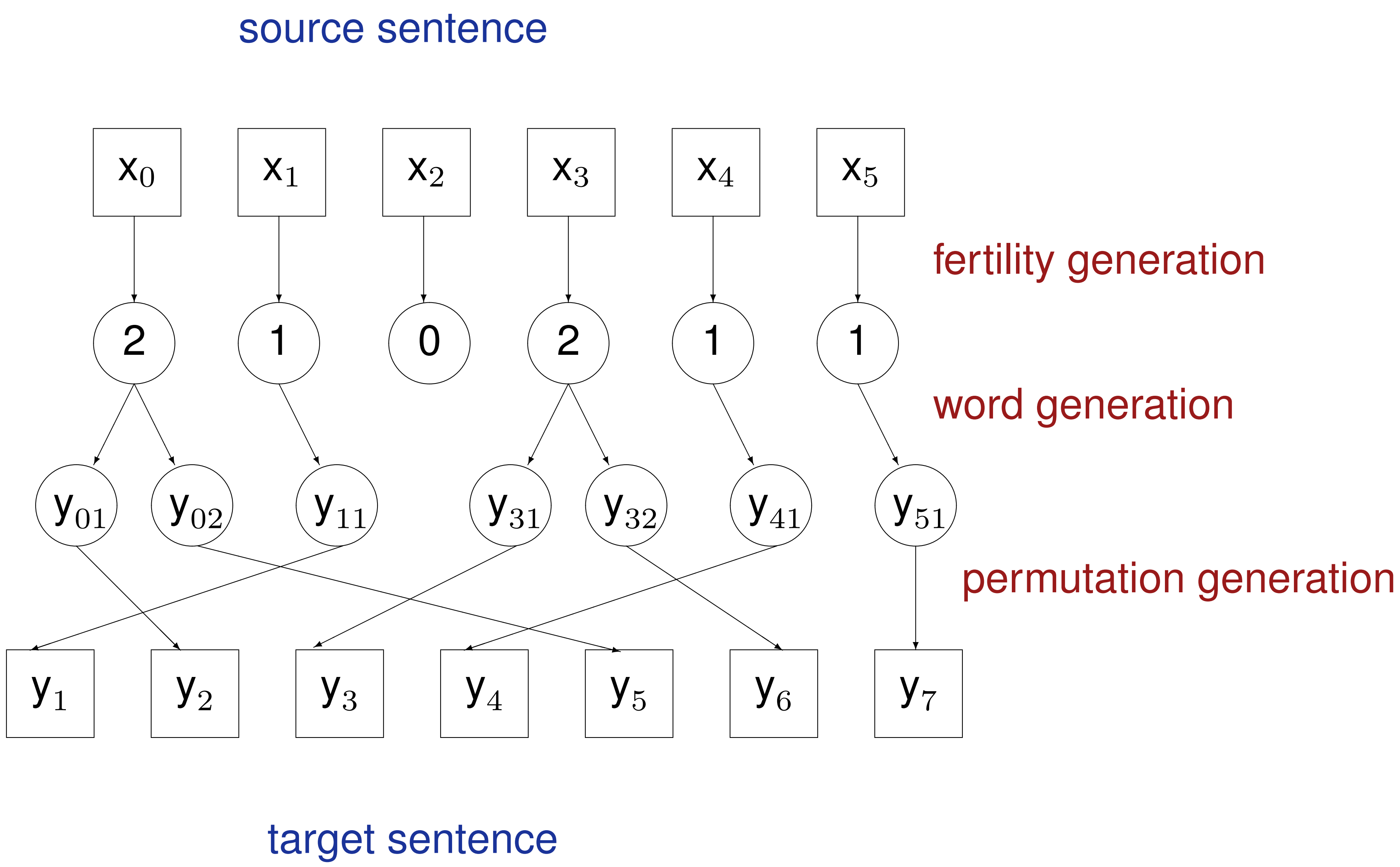
- Using a maximum approach:

$$\widehat{P}_{HMM}(\mathsf{y} \mid \mathsf{x}) = \mathcal{N}(I \mid J) \max_{\mathsf{a}} \prod_{i=1}^{I} h(\mathsf{a}_i \mid \mathsf{a}_{i-1}, I, J) \; l(\mathsf{y}_i \mid \mathsf{x}_{\mathsf{a}_i}) = \mathcal{N}(I \mid J) \; \widehat{Q}(I, J)$$

$$\widehat{Q}(i, j) = l(\mathsf{y}_i \mid \mathsf{x}_j) \max_{j'} \left(h(j \mid j', I, J) \; \widehat{Q}(i-1, j') \right)$$

- Training with the maximum approach
  - Position alignment by computing $\widehat{Q}(i, j)$
  - Parameter estimation (relative frequencies)

Fertility


source sentence
x_0 x_1 x_2 x_3 x_4 x_5
fertility generation
2 1 0 2 1 1
word generation
y_{01} y_{02} y_{11} y_{31} y_{32} y_{41} y_{51}
permutation generation
y_1 y_2 y_3 y_4 y_5 y_6 y_7
target sentence

Fertility

Fertility ϕ of $\mathsf{x}_j \in L_X$: number of the target words connected to a source word x_j.

1. Choose how many target words are connected to a source word x_j: *fertility* of x_j: $\phi_j = \phi(\mathsf{x}_j)$
2. Choose a set of the target words, a *tablet* τ_j, that is connected to j-th target word $\tau_{j,k} \in L_X$ for $1 \leq k \leq \phi(\mathsf{x}_j)$
3. Choose the *position* $\pi_{j,k}$ in the target source sentence of the k-th word $\tau_{j,k}$ that is connected to the j-th source word, $1 \leq \pi_{j,k} \leq I$

Model 3

$$\Pr(\mathsf{y} \mid \mathsf{x}) = \sum_{\mathsf{a}} \Pr(\mathsf{y}, \mathsf{a} \mid \mathsf{x}) = \sum_{\mathsf{a}} \sum_{(\tau,\pi)\in\mathcal{F}(\phi,\mathsf{y},\mathsf{a})} \Pr(\phi, \tau, \pi \mid \mathsf{x})$$

The probability for a tablet τ and a permutation π is:

$$\Pr(\phi, \tau, \pi \mid \mathsf{x}) = \Pr(\phi \mid \mathsf{x}) \;\; \Pr(\tau \mid \phi, \mathsf{x}) \;\; \Pr(\pi \mid \tau, \phi, \mathsf{x})$$

- $\Pr(\phi_j \mid \phi_1^{j-1}, \mathsf{x}) \approx f(\phi_j \mid \mathsf{x}_j)$ *fertility probability*
- $\Pr(\tau_{jk} = y \mid \tau_{j,1}^{k-1}, \tau_0^{j-1}, \phi_0^J, \mathsf{x}) \approx l(y \mid \mathsf{x}_j)$ *lexicon probability*
- $\Pr(\pi_{jk} = i \mid \pi_{j,1}^{k-1}, \pi_1^{j-1}, \tau_0^J, \phi_0^J, \mathsf{x}) \approx d(i \mid j, I, J)$ *distortion probability*

$$P_{M3}(\mathsf{y} \mid \mathsf{x}) = \sum_{\mathsf{a}} \sum_{(\tau,\pi)\in\mathcal{F}(\phi,\mathsf{y},\mathsf{a})} P_{M3}(\phi, \tau, \pi \mid \mathsf{x}) =$$

$$\sum_{a_1=0}^{J} \cdots \sum_{a_I=0}^{J} \binom{I-\phi_0}{\phi_0} p_0^{I-2\phi_0} \, p_1^{\phi_0} \prod_{j=1}^{J} \phi_j! \; f(\phi_j \mid \mathsf{x}_j) \; \prod_{i=1}^{I} l(\mathsf{y}_i \mid \mathsf{x}_{\mathsf{a}_i}) \; d(i \mid \mathsf{a}_i, I, J)$$

Examples of alignments

Corpus EuTrans-I: Spanish-English

1	2	3	4	5	6	7	8	9	10
por	favor	,	¿	podría	ver	alguna	habitación	tranquila	?

- Model 1, Iteration 5
  could (5) I (6) see (6) a (7) quiet (9) room (8) , (3) please (2) ? (4)

- Model 2, Iteration 2
  could (5) I (6) see (6) a (7) quiet (9) room (8) , (3) please (3) ? (10)

- Model 3, Iteration 2
  could (5) I (5) see (6) a (7) quiet (9) room (8) , (3) please (2) ? (10)

Model 4 & 5

- $\Pr(\phi_j \mid \phi_1^{j-1}, \mathsf{x}) \approx f(\phi_j \mid \mathsf{x}_j)$ *fertility probability*
- $\Pr(\pi_{jk} = y \mid \tau_{j,1}^{k-1}, \tau_0^{j-1}, \phi_0^J, \mathsf{x}) \approx l(y \mid \mathsf{x}_j)$ *lexicon probability*
- $\Pr(\pi_{jk} = i \mid \pi_{j,1}^{k-1}, \pi_1^{j-1}, \tau_0^J, \phi_0^J, \mathsf{x}) \approx$ two first-order models *distortion probability*
  - *distortion probability for the first position in a tablet*
  - *distortion probability for the rest of positions in a tablet*

The training process

- Every model has a specific set of free parameters. For example for IBM Model 4: $\theta = \left\{ \{l(y \mid x)\}, \{p_{=1}(\Delta_i)\}, \{p_{>1}(\Delta_i)\}, \{p(\phi \mid x)\}, p_1 \right\}$

- To train the model parameters θ: A maximum likelihood criterium, using a parallel training corpus consisting of S sentence pairs $\{(\mathsf{x}^{(n)}, \mathsf{y}^{(n)}) : n = 1, \ldots, N\}$:

$$\hat{\theta} = \underset{\theta}{\text{argmax}} \prod_{n=1}^{N} \sum_{\mathsf{a}} p_\theta(\mathsf{y}^{(n)}, \mathsf{a} \mid \mathsf{x}^{(n)})$$

- The training is carried out using the Expectation-Maximization (EM) algorithm.

- The estimated counts are approximate by:
  - Computing the (approximate) most probable alignment (Model 2 or HMM)
  - Apply modifications: moves and swaps
  - Sum the estimated counts for all alignments whose probability is larger than the probability of the probable alignment times a given constant.

- Initialization: random, model 1 (5 iterations), model 2 or HMM (0-2 iterations), model 3 (3 iterations), model 4 (3 iterations).

Brown et al. *The mathematics of statistical machine translation: parameter estimation.* Comput. Ling., 19(2):263–310, 1993.

Index

Categorization

- Too many parameters to be estimated

- Many words play the same role: names, dates, etc.

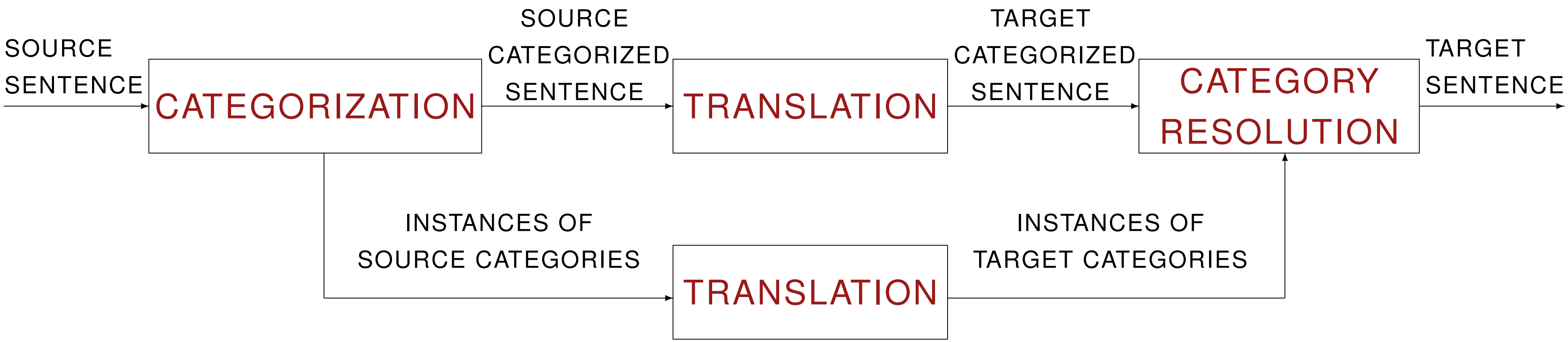
- Substitution of words by categories:
  - The vocabulary size decreases.
  - Easy word addition to the vocabulary.

- Examples:
  - `mi nombre es $NAME.masc $SURNAME . # my name is $NAME.masc $SURNAME .`
  - `nos vamos a ir el  $DATE a $HOUR . # we are leaving on $DATE at $HOUR .`

- Given a bilingual corpus:
  - Automatic extraction of bilingual categories.
  - Manual extraction of bilingual categories.

An approach

(I.Garcia-Varea, F.Casacuberta. *An iterative, DP-based search algorithm for statistical machine translation.* ICSLP. 1998.)



1. CATEGORIZATION: Translating the source sentence into an source categorized sentence and obtaining the source instances of each category.
2. CATEGORIZED TRANSLATION: Translating the source categorized sentence into a target categorized sentence.
3. TRANSLATION OF EACH CATEGORY: Translating the source instances of each category detected.
4. CATEGORY RESOLUTION: Substitution of each target category by the corresponding instance translation.

An example

(I.Garcia-Varea, F.Casacuberta. *An iterative, DP-based search algorithm for statistical machine translation.* ICSLP. 1998.)

me voy a ir el dia veintiseis de abril a las doce en punto de la mañana

Statistical Categorization

Viterbi Alignment

me voy a ir el dia \$DATE a \$HOUR de la mañana

\$DATE = veintiseis de abril
\$HOUR = las doce en punto

Statistical Translation

I am leaving on \$DATE at \$HOUR in the morning

\$DATE = April the twenty-sixth
\$HOUR = twelve o'clock

Category Resolution

I am leaving on April the twenty-sixth at twelve o'clock in the morning

Index

Phrase-based models

- Modelling the correspondences between word segments (phrases)
- Log-linear models: combining different models.
- Moses: a widely used free software, statistical machine translation engine that can be used to train statistical models.

Exemple of word alignments

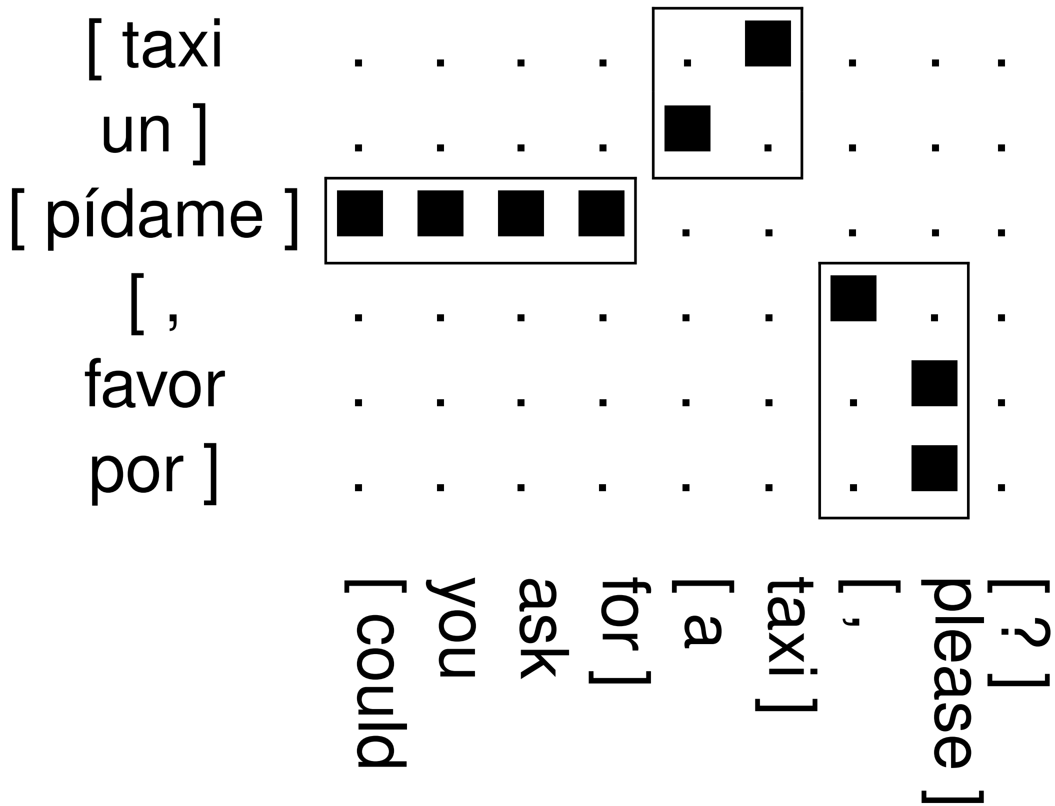
	could	you	ask	for	a	taxi	,	please	?
taxi	·	·	·	·	·	■	·	·	·
un	·	·	·	·	■	·	·	·	·
pídame	■	■	■	■	·	·	·	·	·
,	·	·	·	·	·	·	■	·	·
favor	·	·	·	·	·	·	·	■	·
por	·	·	·	·	·	·	·	■	·

Segment alignment

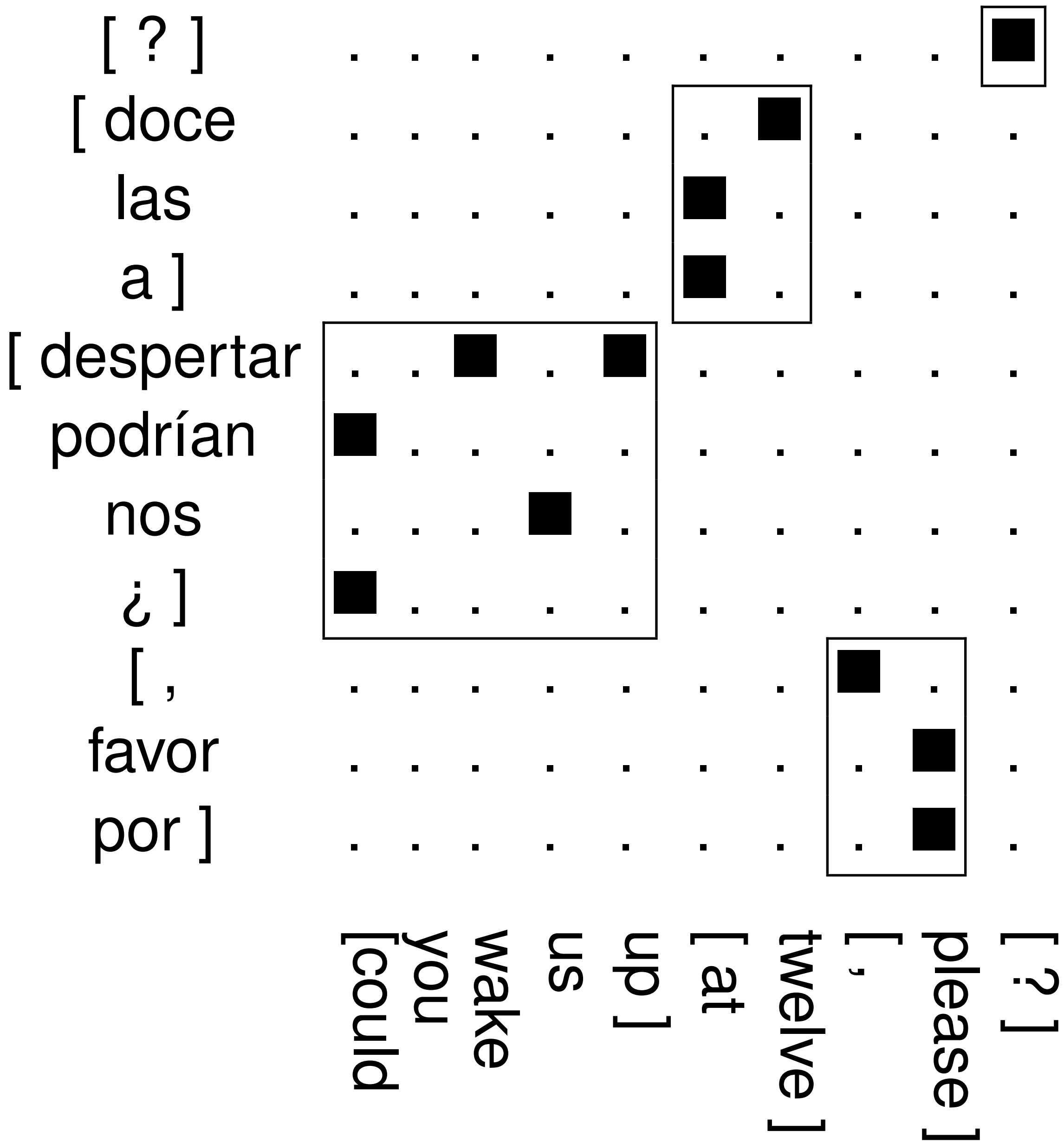
SINGLE-WORD ALIGNMENTS: only model the correspondence between words.

Alternative:

SEGMENT ALIGNMENTS: modelling the correspondences between word segments.



Segment alignment

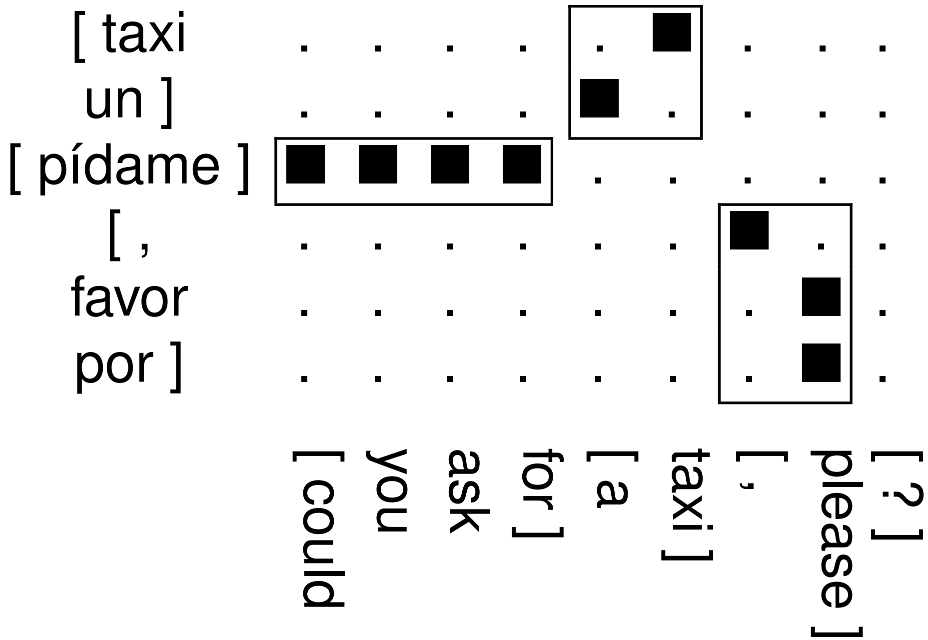


Beyond word-based models

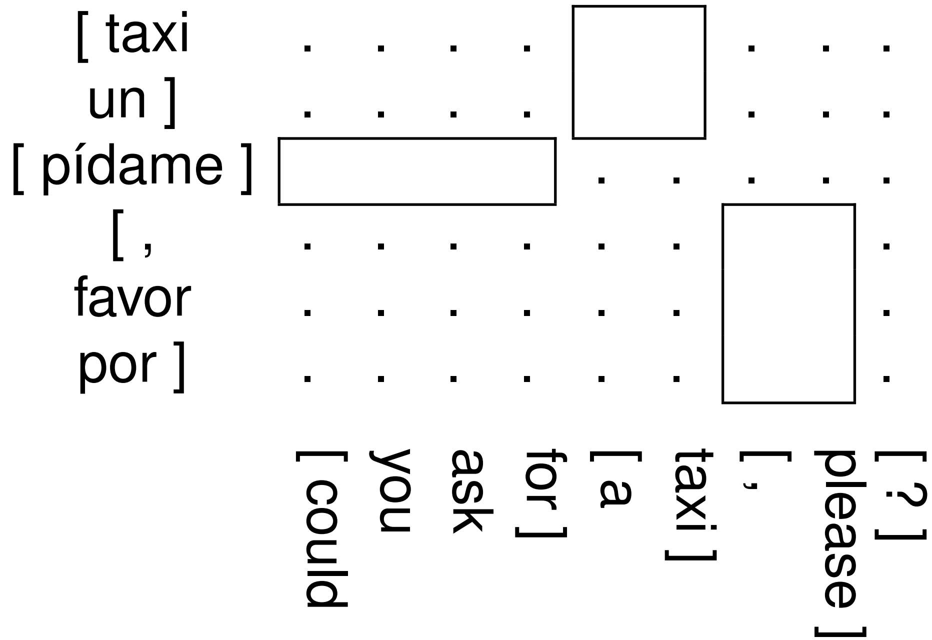
- The basic assumption in the word-based models: Each source word is generated by only one target word.
- This assumption does not correspond to the nature of natural language. In some cases, it is necessary to know the context.
- Solutions:
  - *Context-dependent dictionaries*. The basic unit is the word.
  - *Word sequences*:
    - *Alignment templates*: A sequence of source (classes of) words is aligned with a sequence of target (classes of) words. Inside the templates there are word-to-word correspondences. The basic unit is the word. (Och & Ney, CL, 2006)
    - *Phrase-based models*: [1] A sequence of source words is aligned with a sequence of target words. The basic unit is the phrase. (Koehn, 2010)
    - *Hierarchical phrase-based models*: Phrases that contain subphrases. The model is formally a synchronous context-free grammar. (Koehn, 2010)

[1] By “phrase” we will mean a possible word sequence.

Word sequences



Alignment templates



Bilingual phrases

Phrase-based models

- The *bilingual phrases* are pairs of word sequences.
- Bilingual phrases are related with a bilingual segmentation.
- The statistical dictionaries of single word pairs are substituted by statistical dictionaries of bilingual phrases.
- Problems:
  - The generalisation capability, since only word sequences that appear in a segmentation of the training corpus are accepted.
  - The selection of adequate bilingual phrases.

An example

x: could you ask for a taxi , please ?

	x	could you ask for	a taxi	, please ?
	j	1 2 3 4	5 6	7 8 9=J
Segmentation	μ	μ_1	μ_2	μ_3
Translation	y	[pídame]	[un taxi .]	[por favor ,]
Permutation	α	$\alpha_1 = 2$	$\alpha_2 = 3$	$\alpha_3 = 1$
		por favor ,	pídame	un taxi .
	i	1 2 3	4	5 6 7=I
Segmentation	γ	γ_1	γ_2	γ_3

y: por favor , pídame un taxi .

General framework

- Given a source sentence x and a target sentence y.
- Assumption: Let K be the number of segments in x and in y,
- Process:
  - Segmentation of the source sentence

$$\mu : \{1, \ldots, K\} \rightarrow \{1, \ldots, J\} : \mu_k \geq \mu_{k-1} \;\; 1 < k \leq K \;\; \& \;\; \mu_K = J$$

  - Source phrases: $\bar{\mathsf{x}}_k = \mathsf{x}_{\mu_{k-1}+1}, \ldots, \mathsf{x}_{\mu_k} \equiv \mathsf{x}_{\mu_{k-1}+1}^{\mu_k}$ for $1 \leq k \leq K$
  - Target phrases: $\bar{\mathsf{y}}_k$ translation of $\bar{\mathsf{x}}_k$
  - Segment alignment (Permutation): $\alpha : \{1, \ldots, K\} \rightarrow \{1, \ldots, K\} : \alpha_k = \alpha_{k'}$ iff $k = k'$
  - Target sentence: $\bar{\mathsf{y}}_{\alpha_1}, \ldots, \bar{\mathsf{y}}_{\alpha_K}$

General framework

- The most used models are inspired in HMM word-models:
  - Statistical dictionaries: $l(\mathsf{y}_i \mid \mathsf{x}_j)$
  - Statistical alignment with first-order dependencies: $h(\mathsf{a}_i \mid \mathsf{a}_{i-1}, J)$

  For a given source sentence x and a given target length I:

$$P_{HMM}(\mathsf{y} \mid \mathsf{x}) = \sum_{\mathsf{a}} \prod_{i=1}^{I} h(\mathsf{a}_i \mid \mathsf{a}_{i-1}, J)\; l(\mathsf{y}_i \mid \mathsf{x}_{\mathsf{a}_i})$$

- For phrase-based models
  - Phrase tables: $p(\bar{\mathsf{y}} \mid \bar{\mathsf{x}})$
  - First-order alignments between phrases or segments: $q(\alpha_k \mid \alpha_{k-1}, K)$

  For a given source sentence x and a given target length I:

$$P_{PB}(\mathsf{y} \mid \mathsf{x}) = \sum_{K} \sum_{\mu_1^K} \sum_{\alpha_1^K} \sum_{\gamma_1^K} \prod_{k=1}^{K} q(\alpha_k \mid \alpha_{k-1}, K)\; p(\mathsf{y}_{\gamma_{\alpha_k - 1}+1}^{\gamma_{\alpha_k}} \mid \mathsf{x}_{\mu_{k-1}+1}^{\mu_k})$$

Monotone vs. no monotone alignments

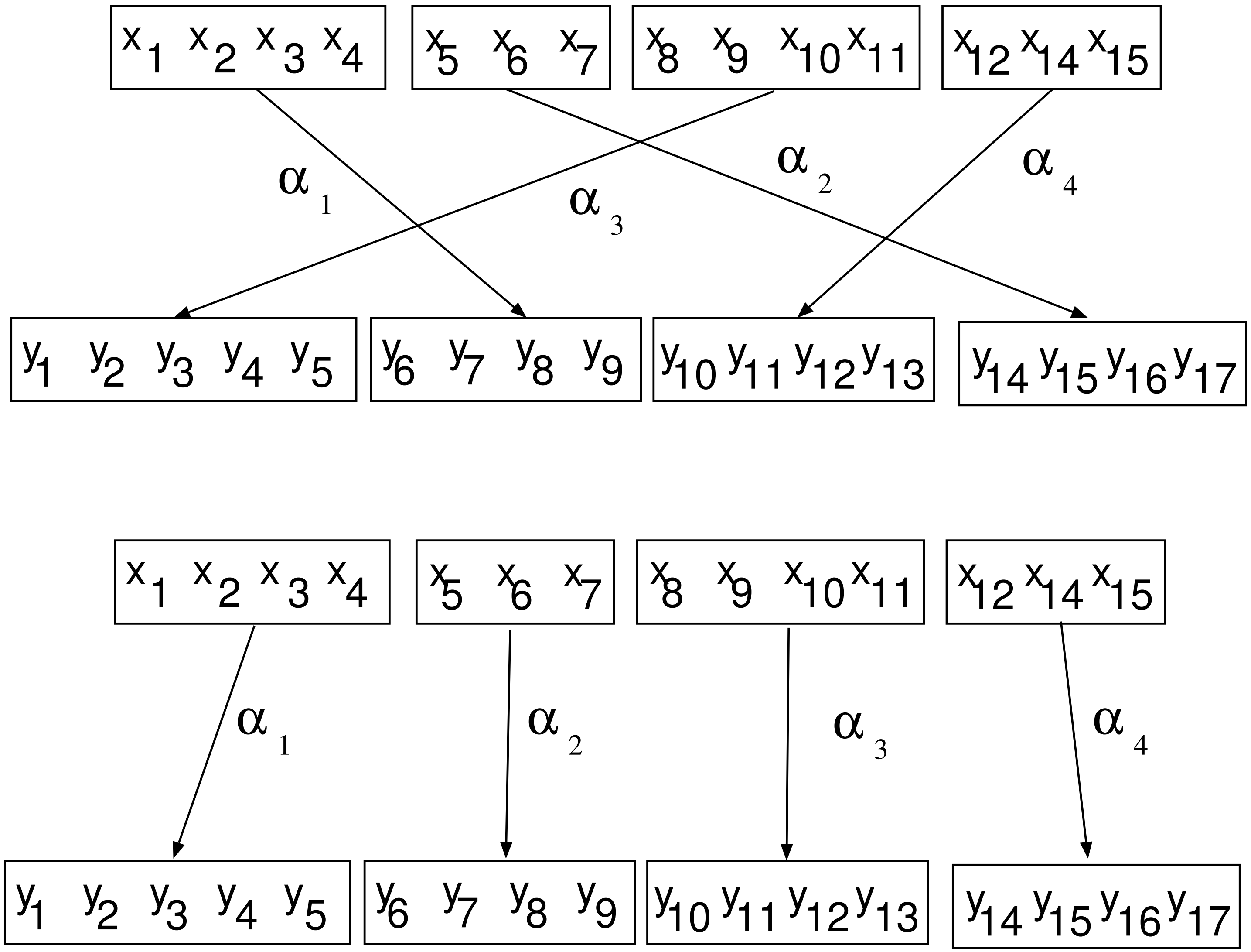
NO MONOTONE ALIGNMENT

$$\Pr(\mathsf{y} \mid \mathsf{x}) \approx P(\mathsf{y} \mid \mathsf{x}) = \mathcal{N}(I \mid J) \sum_{K} \sum_{\mu_1^K} \sum_{\alpha_1^K} \sum_{\gamma_1^K} \prod_{k=1}^{K} q(\alpha_k \mid \alpha_{k-1})\, p(\mathsf{y}_{\gamma_{\alpha_k-1}+1}^{\gamma_{\alpha_k}} \mid \mathsf{x}_{\mu_{k-1}+1}^{\mu_k})$$

MONOTONE ALIGNMENT $\Rightarrow \quad \alpha_k = k$

$$\Pr(\mathsf{y} \mid \mathsf{x}) \approx P(\mathsf{y} \mid \mathsf{x}) = \mathcal{N}(I \mid J) \sum_{K} \sum_{\mu_1^K} \sum_{\gamma_1^K} \prod_{k=1}^{K} p(\mathsf{y}_{\gamma_{k-1}+1}^{\gamma_k} \mid \mathsf{x}_{\mu_{k-1}+1}^{\mu_k})$$

Monotone vs. no monotone alignments



Log-linear models

Search for a target sentence with maximum *posterior* probability:

$$\underset{\mathsf{y}}{\text{argmax}} \Pr(\mathsf{y} \mid \mathsf{x}) = \underset{\mathsf{y}}{\text{argmax}} \frac{\exp\left(\sum_{m=1}^{M} \lambda_m h_m(\mathsf{x}, \mathsf{y})\right)}{\sum_{\mathsf{y}'} \exp\left(\sum_{m=1}^{M} \lambda_m h_m(\mathsf{x}, \mathsf{y}')\right)} = \underset{\mathsf{y}}{\text{argmax}} \sum_{m=1}^{M} \lambda_m h_m(\mathsf{x}, \mathsf{y})$$

- Target language model: $h_1(\mathsf{x}, \mathsf{y}) = \log Pr(\mathsf{y})$ (from a n-gram model)
- Phrase-based model: $h_2(\mathsf{x}, \mathsf{y}) = \log Pr_{PB}(\mathsf{y} \mid \mathsf{x})$, (from the phrase table $p(\tilde{\mathsf{y}} \mid \tilde{\mathsf{x}})$)
- Phrase-based inverse model: $h_3(\mathsf{x}, \mathsf{y}) = \log Pr_{PB}(\mathsf{x} \mid \mathsf{y})$, (from the inverse phrase table $p(\tilde{\mathsf{x}} \mid \tilde{\mathsf{y}})$)
- Reordering model: $h_4(\mathsf{x}, \mathsf{y}) = q(\alpha_k \mid \alpha_{k-1}) = \frac{\gamma^{|\alpha_k - \alpha_{k-1}|}}{\mathcal{Q}_{\mathcal{N}}}$ ($\mathcal{Q}_{\mathcal{N}}$ is a normalization factor)
- More features

Log-linear models: More features

- Lexicalized model: $h_4(\mathsf{x}, \mathsf{y}) = \log Pr_{LEX}(\mathsf{x} \mid \mathsf{y})$ computed from $p_{lex}(\tilde{\mathsf{y}} \mid \tilde{\mathsf{x}}, \tilde{\mathsf{a}})$, where $\tilde{\mathsf{a}}$ is a word alignement from $\tilde{\mathsf{x}}$ to $\tilde{\mathsf{y}}$.

$$p_{lex}(\tilde{\mathsf{y}} \mid \tilde{\mathsf{x}}, \tilde{\mathsf{a}}) = \prod_{i=1}^{|\tilde{\mathsf{y}}|} \frac{1}{|\{j \mid (i,j) \in \tilde{\mathsf{a}}\}|} \sum_{(i,j) \in \tilde{\mathsf{a}}} l(\mathsf{y}_i \mid \mathsf{x}_j)$$

- Inverse lexicalized model: $h_5(\mathsf{x}, \mathsf{y}) = \log Pr_{LEX}(\mathsf{y} \mid \mathsf{x})$ computed from $p_{lex}(\tilde{\mathsf{x}} \mid \tilde{\mathsf{y}}, \tilde{\mathsf{b}})$, where $\tilde{\mathsf{b}}$ is a word alignement from $\tilde{\mathsf{y}}$ to $\tilde{\mathsf{x}}$.

$$p_{lex}(\tilde{\mathsf{x}} \mid \tilde{\mathsf{y}}, \tilde{\mathsf{b}}) = \prod_{j=1}^{|\tilde{\mathsf{x}}|} \frac{1}{|\{i \mid (i,j) \in \tilde{\mathsf{b}}\}|} \sum_{(i,j) \in \tilde{\mathsf{b}}} l(\mathsf{x}_j \mid \mathsf{y}_i)$$

- Target word penalty: $\log I$
- Phrase penalty: $\log K$
- More ...

Index

Learning phrase-based models

- Training with a sentence-aligned corpus.
  - Using the EM algorithm
    (Marcu & Wong, EMNLP, 2002, Andrés-Ferrer et al., AAI, 2008)

- Training with a word-aligned corpus.
  - Symmetrization alignments and counting
    (Koehn, Statistical Machine Translation, 2010)

- Complementary techniques.
  - Phrase-table pruning (Sanchis et al., EAMT, 2011)
  - Incremental learning (Ortiz et al, NAACL, 2010)

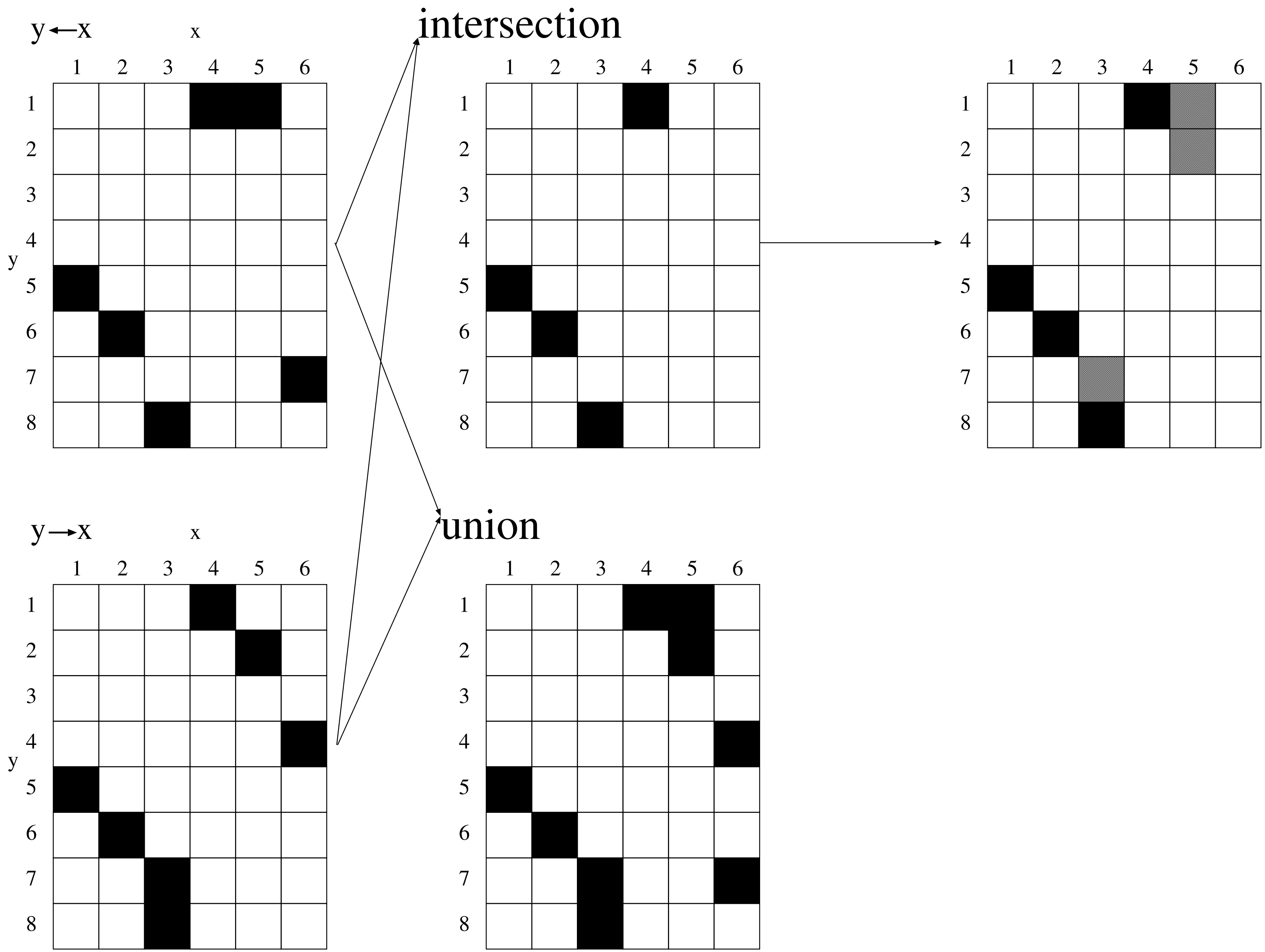
Symmetrized aligments

[M. Federico 2008]

Given a sentence-aligned corpus $\mathcal{T} = \{(\mathsf{x}_1, \mathsf{y}_1), \ldots, (\mathsf{x}_N, \mathsf{y}_N)\}$:

- For a pair $(\mathsf{x}_n, \mathsf{y}_n)$, of length J_n and I_n, alignments are computed using mgiza:
  - Alignment from target to source: a : $\{1, ..., I_n\} \rightarrow \{0, ..., J_n\}$
  - Alignment from source to target: b : $\{1, ..., J_n\} \rightarrow \{0, ..., I_n\}$
- The symmetrized aligments can be obtained:
  - Union: u $= \{(i, j) \mid 1 \leq i \leq I_n, 1 \leq j \leq J_n, \mathsf{a}_i = j \text{ OR } \mathsf{b}_j = i\}$
  - Intersection i $= \{(i, j) \mid 1 \leq i \leq I_n, 1 \leq j \leq J_n, \mathsf{a}_i = j \text{ AND } \mathsf{b}_j = i\}$
  - Grow-diagonal: intersection plus selected links from a and b
- A set of bilingual word sequences from a word simmetrized aligned corpus
- The parameters of the phrase-model are estimated.

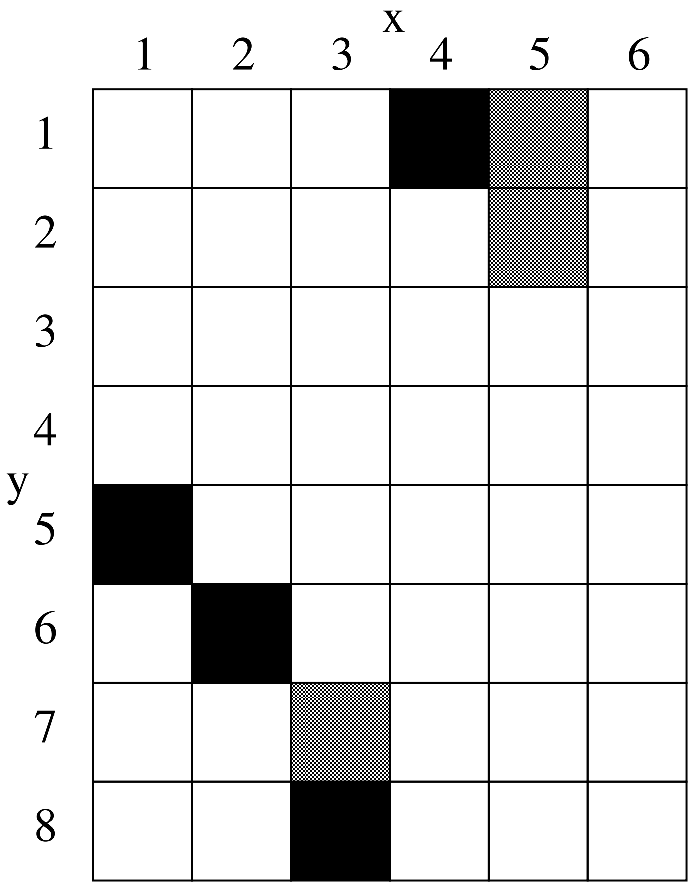
Symmetrized aligments



Extracting bilingual phrases

[M. Federico 2008]

$(\widetilde{x}, \widetilde{y})$ is a phrase-pair $(x_{j_1}^{j_2}, y_{i_1}^{i_2})$ from a pair (x, y) by a symmetrized aligment if the set of target positions linked to source positions in $[j_1, \ldots, j_2]$ by the alignment is included in $[i_1, \ldots, i_2]$ and viceversa



$(x_4, y_1),\ (x_4x_5, y_1),\ (x_4x_5, y_1y_2),$

$(x_5, y_1),\ (x_5, y_1y_2),\ (x_5, y_2)$

$(x_1, y_5),\ (x_1x_2, y_5y_6),\ \ldots$

$(x_2x_3, y_6y_7y_8),\ \ldots$

Estimating the phrase table and the distortion model

- Phrase probabilities by relative frequencies: for each pair of segments $(\widetilde{x}, \widetilde{y})$:

$$p(\widetilde{y} \mid \widetilde{x}) \;=\; \frac{N(\widetilde{y}, \widetilde{x})}{N(\widetilde{x})} \qquad\qquad p(\widetilde{x} \mid \widetilde{y}) \;=\; \frac{N(\widetilde{y}, \widetilde{x})}{N(\widetilde{y})}$$

  where $N(\widetilde{y})$ denotes the number of times that phrase $\widetilde{y}$ has appeared, and $N(\widetilde{x}, \widetilde{y})$ is the number of times that the bilingual phrase $(\widetilde{x}, \widetilde{y})$ has appeared.

- Distortion model $p(\alpha_k \mid \alpha_{k-1})$:

$$p(\alpha_k \mid \alpha_{k-1}) = p_0^{|\gamma_{\alpha_k} - \gamma_{\alpha_{k-1}}|}$$

  where p_0 is a parameter to be ajusted using a validation set.

Learning log-linear models

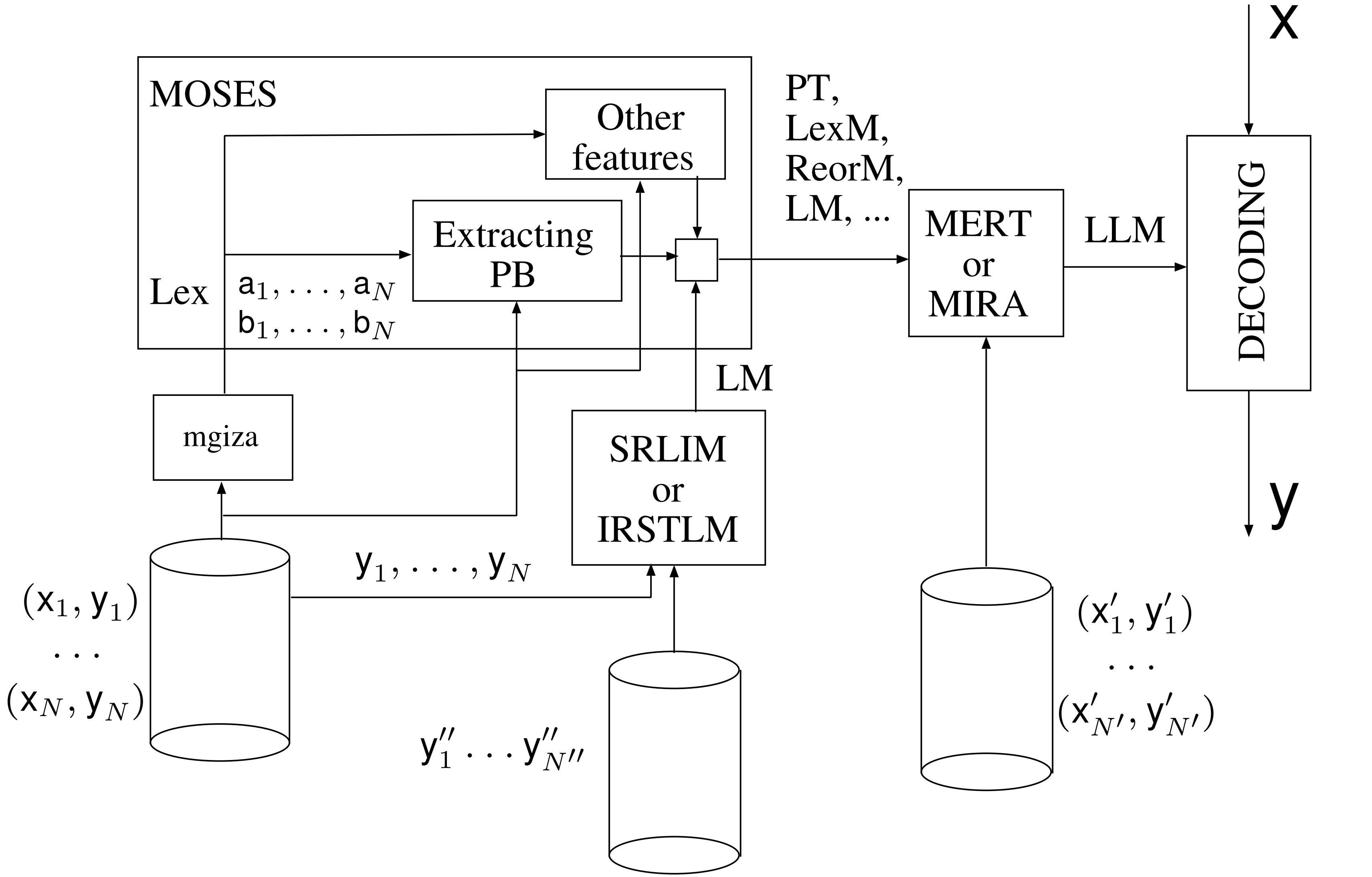
- Learning the features $h_m(\mathrm{x}, \mathrm{y})$ for $1 \leq m \leq M$. Target language model, phrase-based models, phrase-based inverse models, reordering model, lexicalized model, inverse lexicalized model, target word penalty, phrase penalty, ...

- Estimate the weights λ_i of the log-linear model using a developpment set and the Minimum Error Rate Training (MERT) or Margin Infused Relaxed Algorithm (MIRA) procedures

Learning features of the log-linear models

- Given a sentence-aligned corpus $\mathcal{T} = \{(\mathrm{x}_1, \mathrm{y}_1), \ldots, (\mathrm{x}_N, \mathrm{y}_N)\}$
- Training target language models ($Pr(\mathrm{y})$) using the target sentences of the bilingual training corpus $\{\mathrm{y}_1, \ldots, \mathrm{y}_N\}$ (maybe with the addition of other target data)
- Use mgiza with the bilingual corpus $\mathcal{T}$ to build:
  - a symetrized word-aligned corpus from word-alignement in both directions;
  - a stochastic dictionary
- Build the set of bilingual phrases and estimate $p(\widetilde{x} \mid \widetilde{y})$ and $p(\widetilde{y} \mid \widetilde{x})$ (inverse phrase translation probability and direct phrase translation probability)
- Build the set of lexical bilingual weighting lex $p_{lex}(\widetilde{x} \mid \widetilde{y})$ and $p_{lex}(\widetilde{y} \mid \widetilde{x})$ (inverse lexical weighting and direct lexical weighting)
- Build the rest of the models: phrase penalty, distance-based reordering model, word penalty and others

Learning log-linear models



Index

The search problem in statistical machine translation

$$\hat{y} = \underset{y}{\text{argmax}}\, \Pr(y \mid x) = \underset{y}{\text{argmax}} \sum_{k=1}^{K} \lambda_k h_k(x, y)$$

- $h_1(x, y) = \log Pr(y)$, a language model
- $h_2(x, y) = \log Pr(y \mid x)$, a translation model model
- $h_3(x, y) = \log Pr(x \mid y)$, an inverse translation model model
- $\ldots$

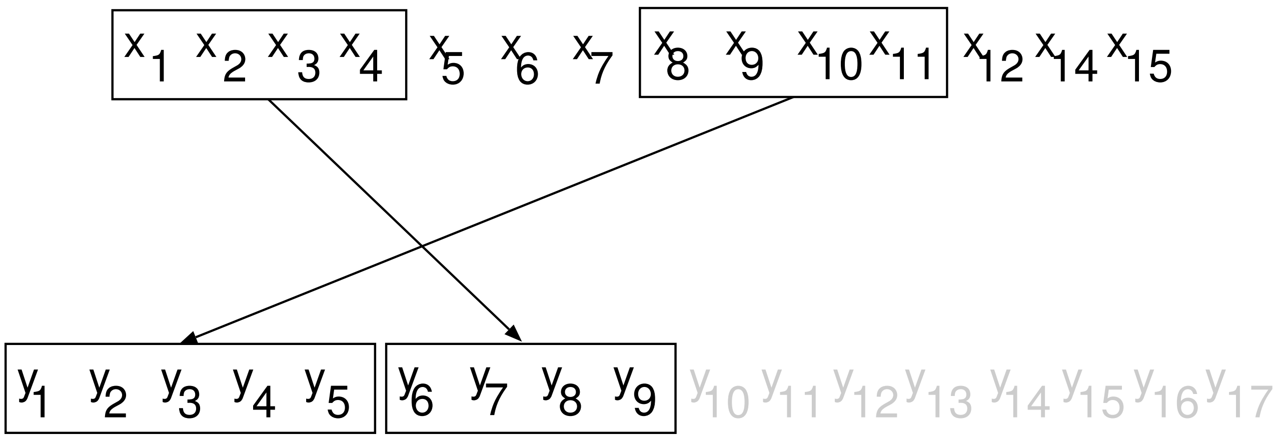
- Search is a **NP**-Hard problem. (Knight, 1999) (Udupa and Maji, 2006)
- Algorithmic solutions: (+ heuristics for efficient suboptimal solutions)
  - *Stack-decoding* (*$A^\star$ or Branch & Bound*) (Ortiz, 2003) (Koehn, 2010)
  - *Dynamic Programming* (Ney 2003) (Tillmann & Ney 2003)

Basic stack-decoding strategy

- Origin of the *stack decoding* or $A^{\star}$: ASR (Jelinek, 1976)
- Optimal solution to the search problem.
- Applied to translation with word-based models: Candide systems [Berger et al. 96], [Wang and Waibel 98], [Ueffing et al. 01] [Och and Ney 03]
- Incremental development of partial hyphotesis.
- The hypotheses are stored in a stack (a type of ‘prioritary queue’)
- Selection and expansion of the top of the stack(s)
- Prunning the search space:
  - Beam-search or threshold prunnig
  - Histogram prunning

Basic multiple stack decoding (I)

- A hypothesis in a stack:
  - A prefix of the target sentence (y_1^i)
  - A coverage subset of source positions ($\mathcal{C}$)
  - A score (S).



- y_1^9
- $\mathcal{C} = \{1, 2, 3, 4, 8, 9, 10, 11\}$
- S = a function of $p_{LM}(y_1^9)$, $p(x_1^4 \mid y_6^9)$, $p(x_8^{11} \mid y_1^9)$, $p(y_6^9 \mid x_1^4)$, $p(y_1^9 \mid x_8^{11})$, $p_{lex}(x_1^4 \mid y_6^9)$, $p_{lex}(x_8^{11} \mid y_1^9)$, $p_{lex}(y_6^9 \mid x_1^4)$, $p_{lex}(y_1^9 \mid x_8^{11})$, ...

- There is one stack for each possible subset of source positions which words has already been translated.

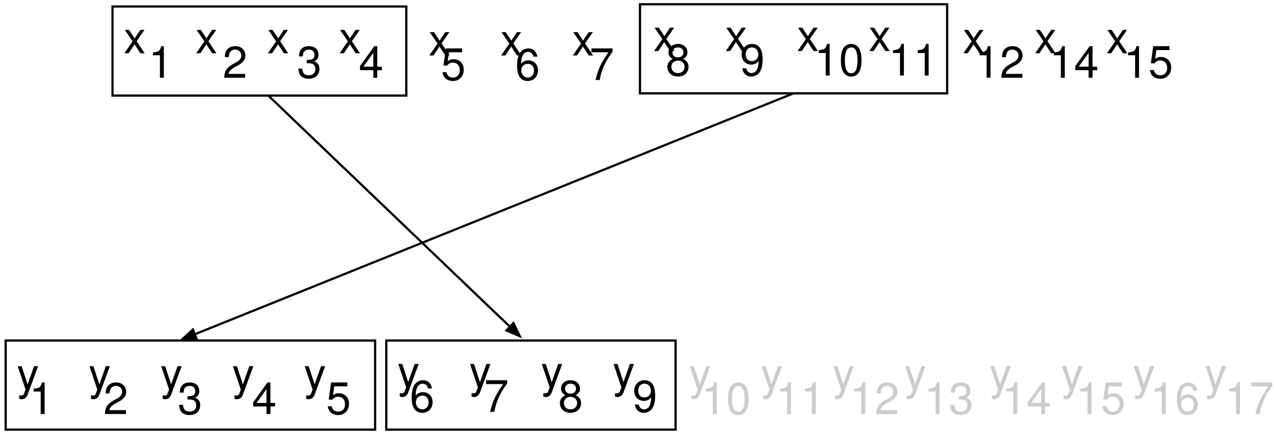
- The possible number of stacks can be very high ($\leq 2^J$)

- In practice, there is one stack for each size of subset of source positions which words has already been translated.

Basic multiple stack decoding (II)

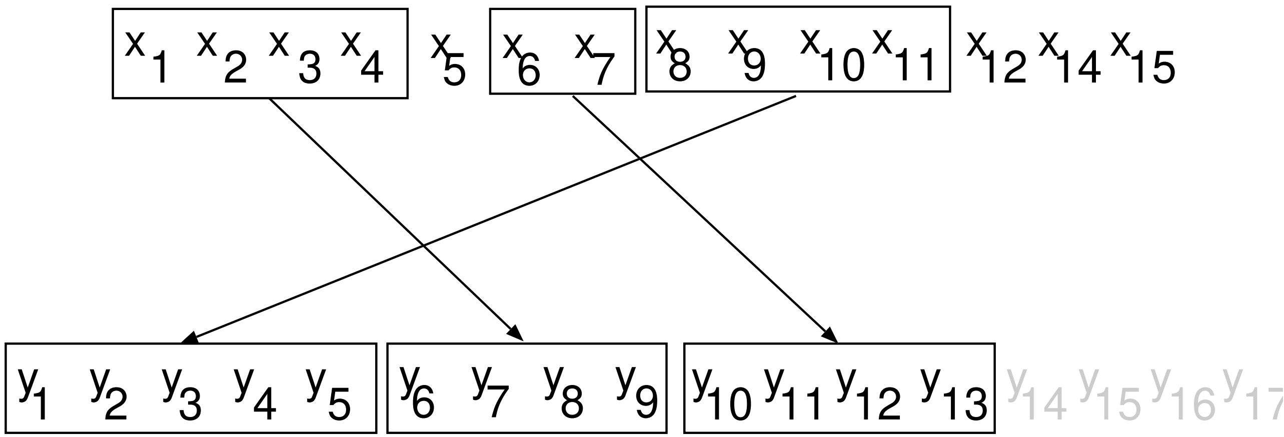
- In each iteration, the best hypothesis from each available stack is selected to generate new extended hypothesis.
- Selection of phrases that match a subset of free source positions (from the complementary set of $\mathcal{C}$ (assuming some constraints))
- The new target prefix is the concatenation of the target prefix of the selected hypothesis and the target words of the selected phrase.
- The new score is computed using the new n-gram and the new source positions.
- A new $\mathcal{C}$ is produced.
- The new hypothesis is stored in the corresponding stack.

Basic multiple stack decoding (III)



- y_1^9
- $\mathcal{C} = \{1, 2, 3, 4, 8, 9, 10, 11\}$
- $S =$ a function of $p_{LM}(\mathrm{y}_1^9)$, $p(\mathrm{x}_1^4 \mid \mathrm{y}_6^9)$, $p(\mathrm{x}_8^1 \mid \mathrm{y}_1^9)$, $p(\mathrm{y}_6^9 \mid \mathrm{x}_1^4)$, $p(\mathrm{y}_1^9 \mid \mathrm{x}_8^1)$, $p_{lex}(\mathrm{x}_1^4 \mid \mathrm{y}_6^9)$, $p_{lex}(\mathrm{x}_8^1 \mid \mathrm{y}_1^9)$, $p_{lex}(\mathrm{y}_6^9 \mid \mathrm{x}_1^4)$, $p_{lex}(\mathrm{y}_1^9 \mid \mathrm{x}_8^1)$, ...

If there is bililngual phrase $(\mathrm{x}_6\mathrm{x}_7, \mathrm{y}_{10}\mathrm{y}_{11}\mathrm{y}_{12}\mathrm{y}_{13})$ in the phrase table:



- y_1^{13}
- $\mathcal{C} = \{1, 2, 3, 4, 6, 7, 8, 9, 10, 11\}$
- $S =$ a function of $p_{LM}(\mathrm{y}_1^{13})$, $p(\mathrm{x}_1^4 \mid \mathrm{y}_6^9)$, $p(\mathrm{x}_8^1 \mid \mathrm{y}_1^9)$, $p(\mathrm{y}_6^9 \mid \mathrm{x}_1^4)$, $p(\mathrm{y}_1^9 \mid \mathrm{x}_8^1)$, $p(\mathrm{y}_{10}^{13} \mid \mathrm{x}_6^7)$, $p(\mathrm{x}_6^7 \mid \mathrm{y}_{10}^{13})$, $p_{lex}(\mathrm{x}_1^4 \mid \mathrm{y}_6^9)$, $p_{lex}(\mathrm{x}_8^1 \mid \mathrm{y}_1^9)$, $p_{lex}(\mathrm{y}_6^9 \mid \mathrm{x}_1^4)$, $p_{lex}(\mathrm{y}_1^9 \mid \mathrm{x}_8^1)$, $p_{lex}(\mathrm{y}_{10}^{13} \mid \mathrm{x}_6^7)$, $p_{lex}(\mathrm{x}_6^7 \mid \mathrm{y}_{10}^{13})$, ...

Assessment

- Word error rate (WER): The minimum number of substitution, insertion and deletion operations needed to convert the word string hypothesized by the translation system into a given single reference word string.
- Multi reference WER (mWER): Similar to WER, but for each source test sentence there are more than one target sentences as references.
- BiLingual Evaluation Understudy (BLEU): it is based on the n-grams of the hypothesized translation that occur in the reference translations (geometric mean). The BLEU metric ranges from 0.0 (worst score) to 1.0 (best score).
- Translation error rate (TER): Similar to WER, but swaps are allowed without penalty. `http://www.cs.umd.edu/~snover/tercom/`
- NIST: similar to BLEU but arithmetic mean is used
- METEOR: Metric for evaluation of translation with explicit ordering and no exact matching.
- BEER: A trained system of a linear combination of features.

EuroMatrix (2006-2009)

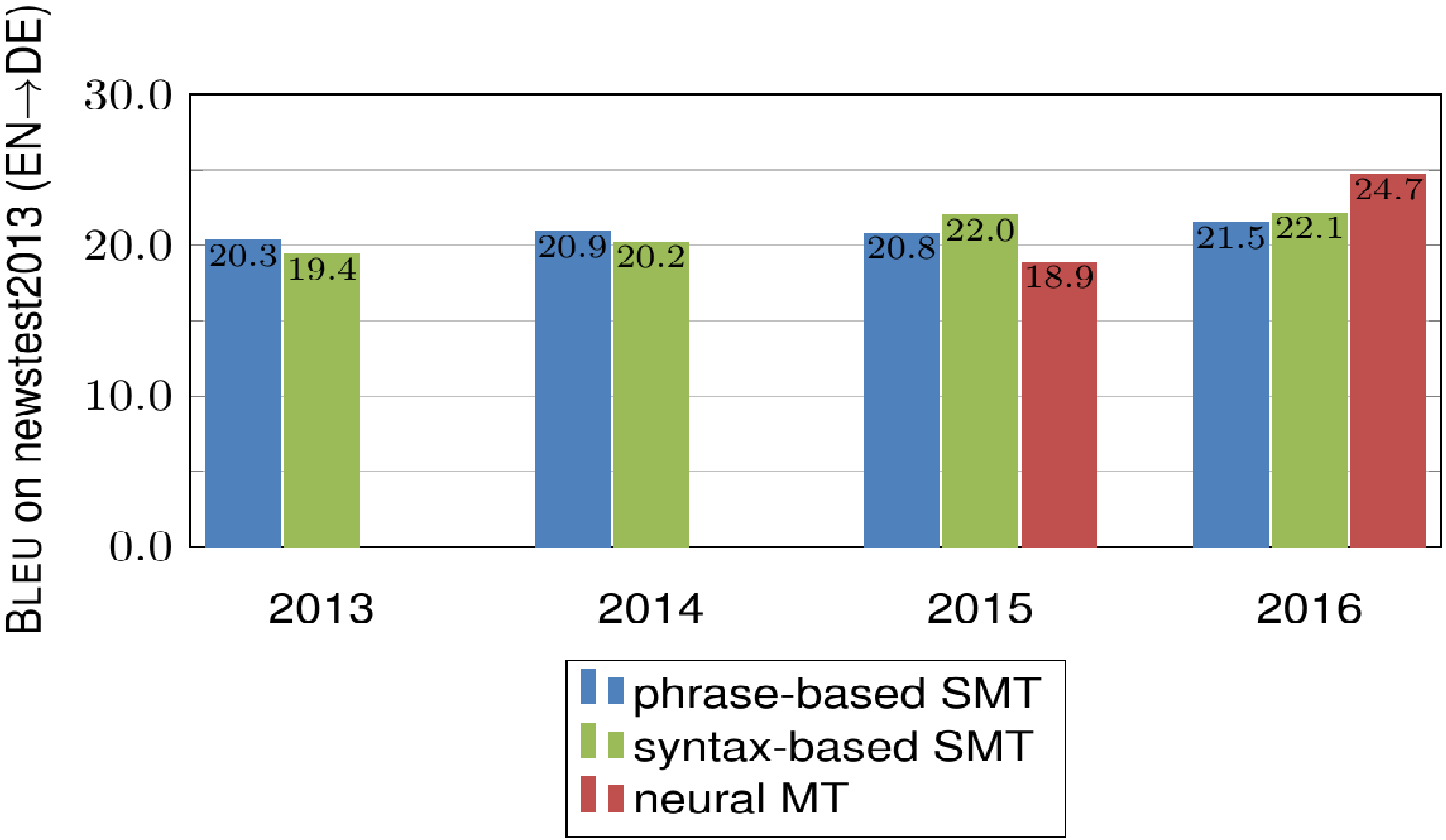
http://www.statmt.org/matrix/

EURO MATRIX

output language

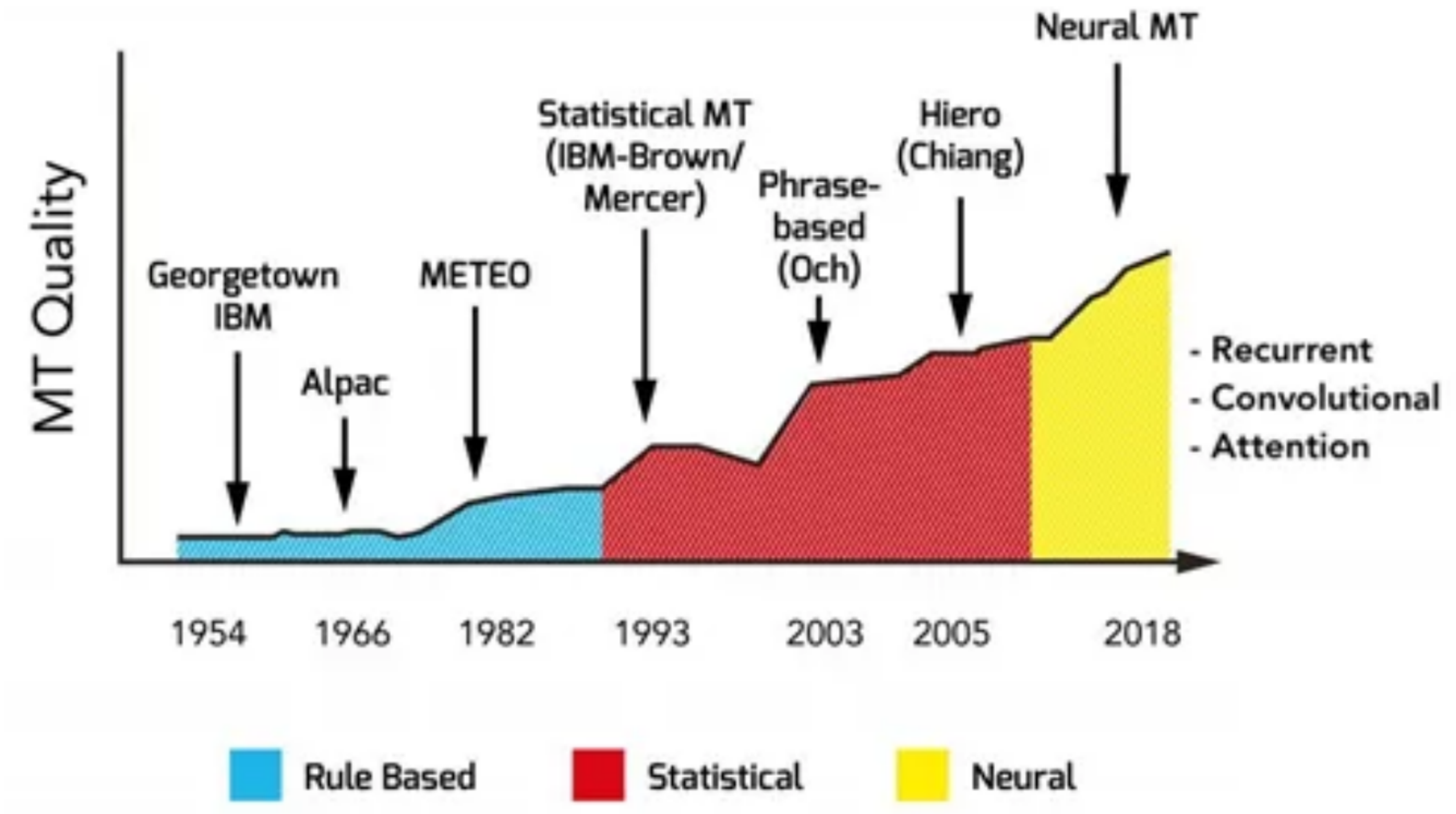
input language											
	Danish	BLEU 21.47	BLEU 18.49	BLEU 21.12	BLEU 28.57	BLEU 14.24	BLEU 28.79	BLEU 22.22	BLEU 24.32	BLEU 26.49	BLEU 28.33
	BLEU 20.51	**Dutch**	BLEU 18.39	BLEU 17.49	BLEU 23.01	BLEU 10.34	BLEU 24.67	BLEU 20.07	BLEU 20.71	BLEU 22.95	BLEU 19.03
	BLEU 22.35	BLEU 23.40	**German**	BLEU 20.75	BLEU 25.36	BLEU 11.88	BLEU 27.75	BLEU 21.36	BLEU 23.28	BLEU 25.49	BLEU 20.51
	BLEU 22.79	BLEU 20.02	BLEU 17.42	**Greek**	BLEU 27.28	BLEU 11.44	BLEU 32.15	BLEU 26.84	BLEU 27.67	BLEU 31.26	BLEU 21.23
	BLEU 25.24	BLEU 21.02	BLEU 17.64	BLEU 23.23	**English**	BLEU 13.00	BLEU 31.16	BLEU 25.39	BLEU 27.10	BLEU 30.16	BLEU 24.83
	BLEU 20.02	BLEU 17.09	BLEU 14.57	BLEU 18.20	BLEU 21.86	**Finnish**	BLEU 22.49	BLEU 18.39	BLEU 19.14	BLEU 21.16	BLEU 18.85
	BLEU 23.73	BLEU 21.13	BLEU 18.54	BLEU 26.13	BLEU 30.00	BLEU 12.63	**French**	BLEU 32.48	BLEU 35.37	BLEU 38.47	BLEU 22.68
	BLEU 21.47	BLEU 20.07	BLEU 16.92	BLEU 24.83	BLEU 27.89	BLEU 11.08	BLEU 36.09	**Italian**	BLEU 31.20	BLEU 34.04	BLEU 20.26
	BLEU 23.27	BLEU 20.23	BLEU 18.27	BLEU 26.46	BLEU 30.11	BLEU 11.99	BLEU 39.04	BLEU 32.07	**Portuguese**	BLEU 37.95	BLEU 21.96
	BLEU 24.10	BLEU 21.42	BLEU 18.29	BLEU 28.38	BLEU 30.51	BLEU 12.57	BLEU 40.27	BLEU 32.31	BLEU 35.92	**Spanish**	BLEU 23.90
	BLEU 30.35	BLEU 21.94	BLEU 18.97	BLEU 22.86	BLEU 30.20	BLEU 15.37	BLEU 29.77	BLEU 23.94	BLEU 25.95	BLEU 28.66	**Swedish**

Edinburgh's WMT results over the years[1]



[1] Sennrich et al. Advances in Neural Machine Translation. AMTA. 2016.

MT quality over the years[1]



[1] https://syncedreview.com/2020/05/20/neural-network-ai-is-the-future-of-the-translation-industry/

Index

Bibliography

1. P. F. Brown et al. *A statistical approach to machine translation*. Computational Linguistics, 16(2): 79-85, 1990.
2. P. F. Brown et al. *The mathematics of statistical machine translation: parameter estimation.* Computational Linguistics, 19(2): 263-310, 1993.
3. S. Barrachina and J. Vilar. *Bilingual clustering using monolingual algorithms*. TMI. 1999.
4. F. Och. *An Efficient method for determining bilingual word classes*. EACL. 1999.
5. F. J. Och, H. Ney: *A Systematic Comparison of Various Statistical Alignment Models*. Computational Linguistics, 29(1): 19-51, 2003.
6. I. García-Varea, F. Casacuberta. *Maximum Entropy Modeling: A Suitable Framework to Learn Context-Dependent Lexicon Models for Statistical Machine Translation*. Machine Learning 60(1-3): 135-158. 2005.
7. Federico. *S*tatistical Machine Translation. Galileo Galilei PhD School. University of Pisa, 2008 `http://medialab.di.unipi.it/web/SMT/SMT-0508-part-6-pp.pdf`
8. Koehn and Knight. *Feature-rich statistical translation of noun phrases*. ACL. 2003.
9. López: *Statistical Machine Translation*. ACM Computing Surveys 40(3): 1-49, 2008.
10. Ortiz-Martínez, García-Varea and Casacuberta. *Online Learning for Interactive Statistical Machine Translation*. NAACL, 2010.
11. Chiang: *Hierarchical Phrase-Based Translation*, Computational Linguistics, 33(2):201-228, June 2007
12. P. Koehn: *Statistical Machine Translation*, Cambridge University Press. 2010