Introduction to Machine Translation

Joan Andreu Sánchez

Departamento Sistemas Informáticos y Computación
Instituto Tecnológico de Informática
Universidad Politécnica Valencia

PASCAL 2 Ghana Bootcamp 2011

URL: http://www.dsic.upv.es/~jandreu
e-mail: jandreu@dsic.upv.es

Index

Slides in Sections 1, 2 and 3 have been prepared from slides supplied by F. Casacuberta.

Index

MT objectives: Erroneus conceptions

- MT is a waste of time because a machine never will translate Shakespeare
- In general, the quality of translation you can get from an MT system is very low
- MT threatens the jobs of translators
- There is an MT system that translates what you say into Japanese and translates the other speaker's replies in English

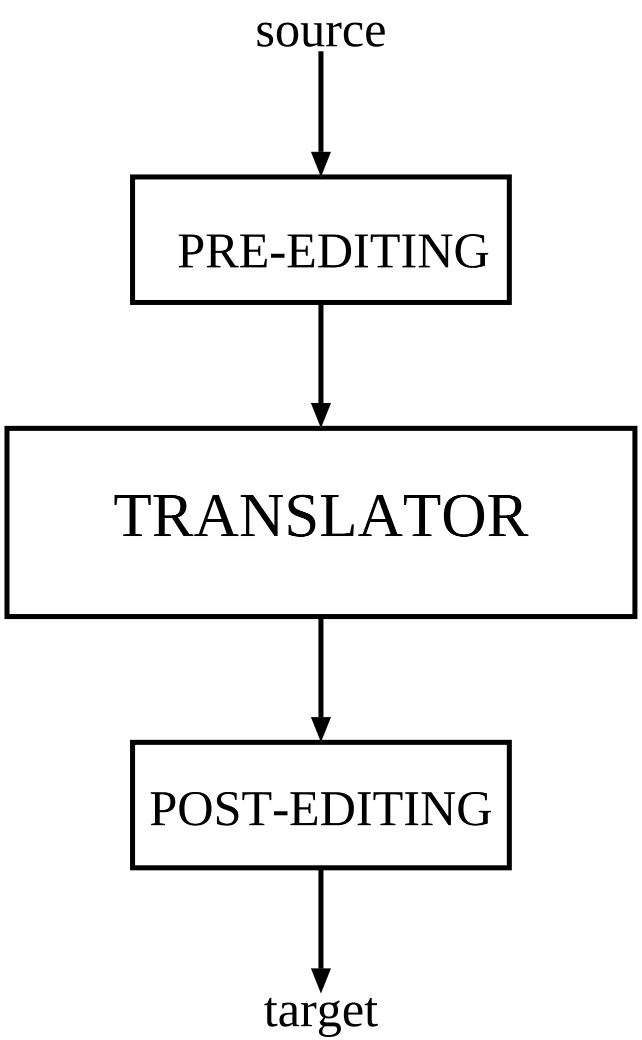
MT objectives: Facts

- There are many situations that a MT systems produce reliable, if less than perfect, translations at high speed
- In some circumstances, MT systems can produce good quality outputs
- MT does not threaten transaltors' jobs: High demand of translations and too repetitive translation jobs
- Speech-to-speech MT is still a research topic
- There are many open research problems in MT
- Building a traditional MT system is a time consuming job
- A user will typically have to invest a considerable amount of effort in customizing an MT system

MT objectives: need of pre/post-editing

- While the number of errors and bad constructions is high, “post-editing” can make the result useful.
- Many problems could have been avoided by making the source text “simpler”.
- Simplification of the translation problem by using adequate rules to produce “controled” (i.e., simple and regular) source text.

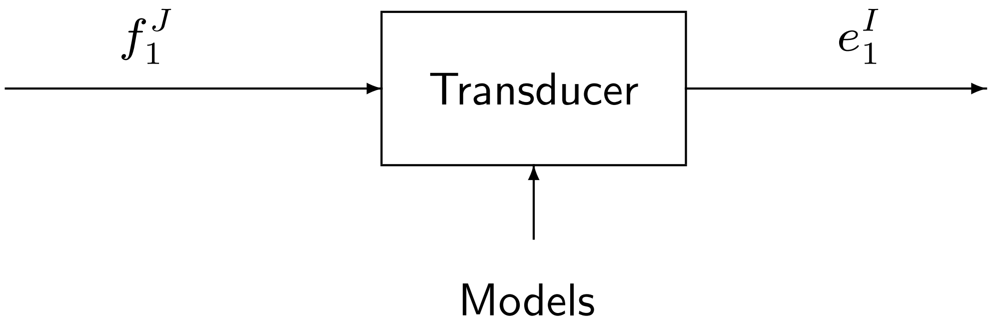
General scheme for MT



Technologies

- (Linguistic) knowledge-based methods
- (Memorized) example-based methods
  - Translation memories
- Statistical models
  - Alignment models
  - Syntax-based models
  - Finite-State models
- Hybrid models

Statistical MT



- Inverse approach (noisy channel)

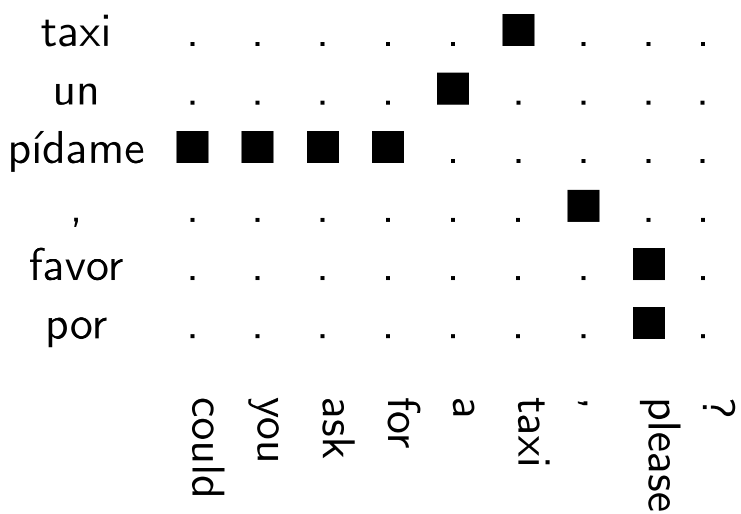
$$\widehat{e_1^I} = \arg\max_{e_1^I} \Pr(e_1^I | f_1^J) = \arg\max_{e_1^I} \Pr(f_1^J | e_1^I) \Pr(e_1^I)$$

- Direct approach

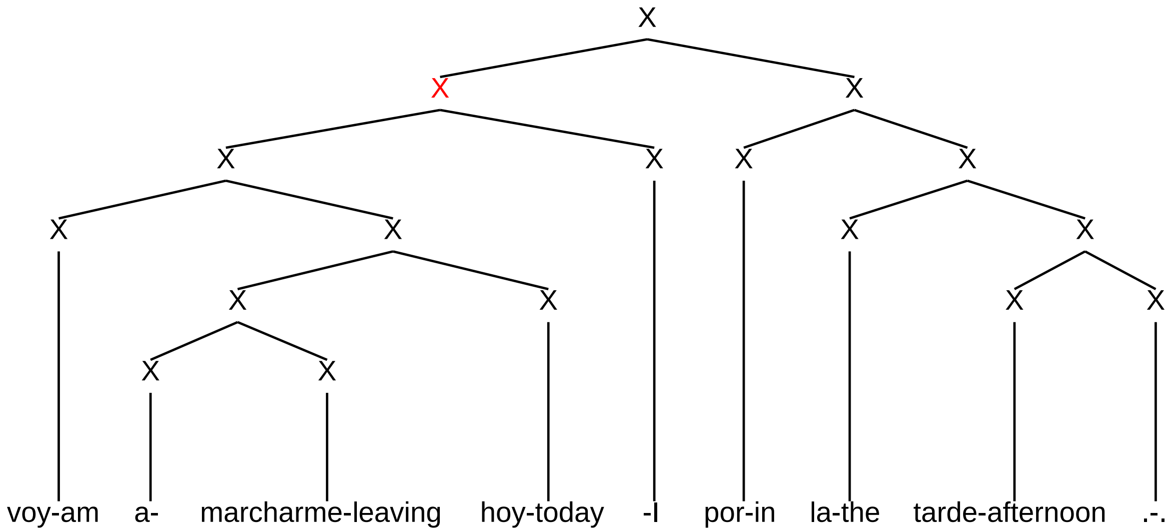
$$\widehat{e_1^I} = \arg\max_{e_1^I} \Pr(e_1^I | f_1^J)$$

Statistical approaches to MT

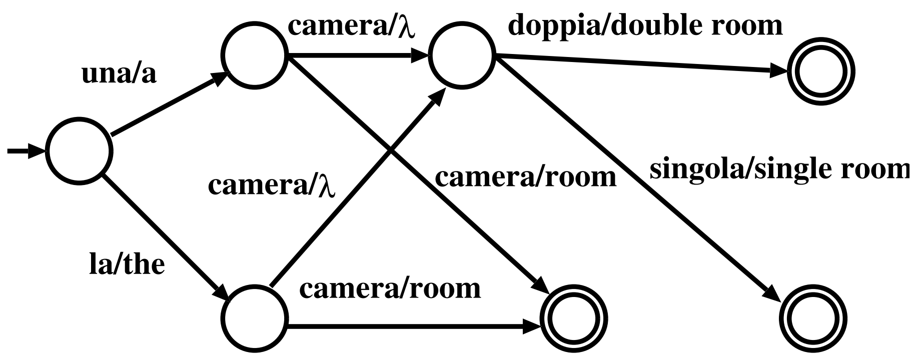
➔ Word-alignment approaches



➔ Syntactic approaches



➔ Finite-state approaches



Resources

- Dictionaries
- Grammars
- Corpora
- Paragraph-aligned and Labeled Corpora

1.4 Assesment

- ➢ Test sentences with reference translation
- ➢ Automatic assessment
  - Editing Distances:
    Translation Word Error Rate (TWER)
    Translation Error Rate (TER)
  - Multireference TWER
  - N-Gram based: *BLUE* and *NIST* score
  - . . .

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 $\mathrm{Pr}(\mathrm{y} \mid \mathrm{x})$
- The probability of pairs of sentences as
  quiero una habitación doble con vistas al mar # are all expenses included in the bill ?
  should be low
- The probability of pairs of sentences as
  ¿ hay alguna habitación tranquila libre ? # is there a quiet room available ?
  should be high

General framework

Given a source sentence x, search for the sentence ŷ

$$\boxed{\hat{y} = \arg\max_{y} \Pr(y \mid x)}$$

Approaches

- A direct approach: *maximum entropy models*
- An inverse approach: *channel models*

An inverse approach

Given a source sentence x, search for the sentence $\hat{y}$

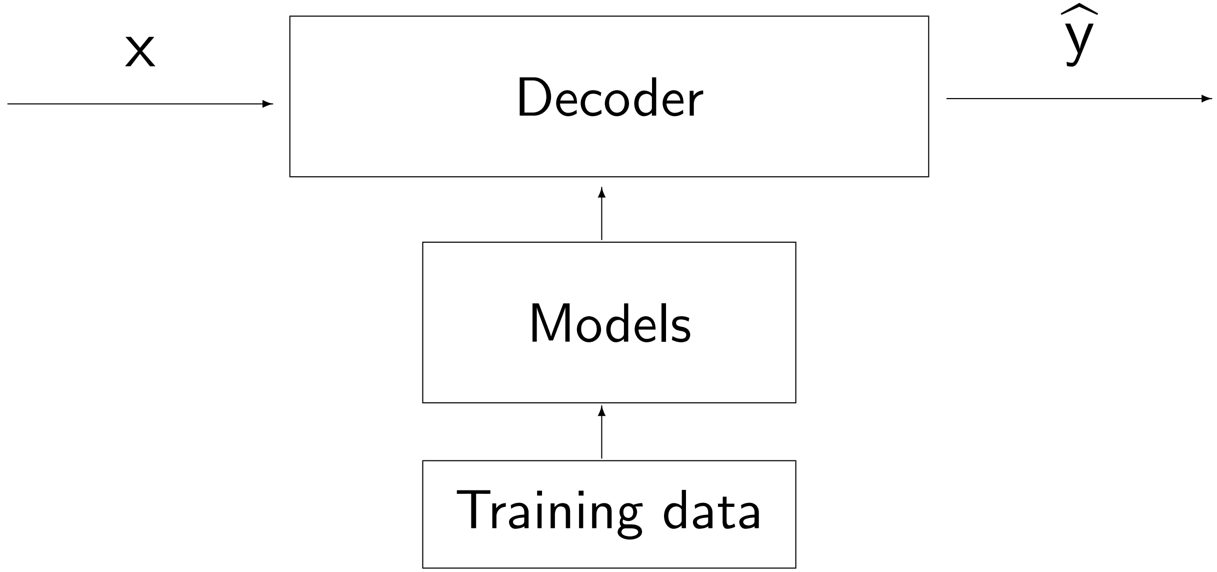
$$\boxed{\hat{y} = \arg\max_{y} \Pr(y \mid x) = \arg\max_{y} \Pr(x \mid y) \cdot \Pr(y)}$$

A channel model

$$\boxed{\Pr(y)} \xrightarrow{\ y\ } \boxed{\Pr(x \mid y)} \xrightarrow{\ x\ }$$

A target-language model + alignment and lexicon models

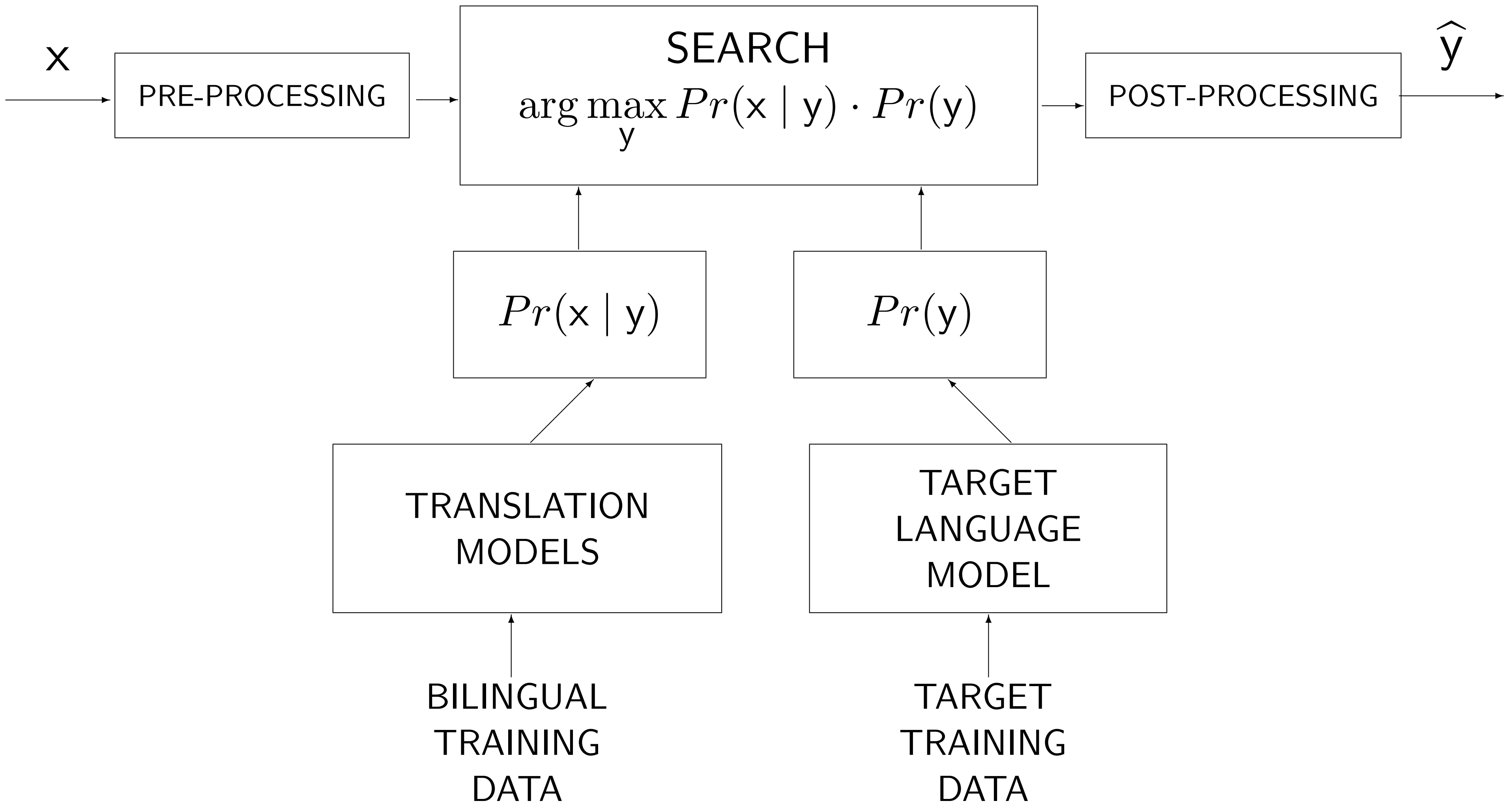
Translation search



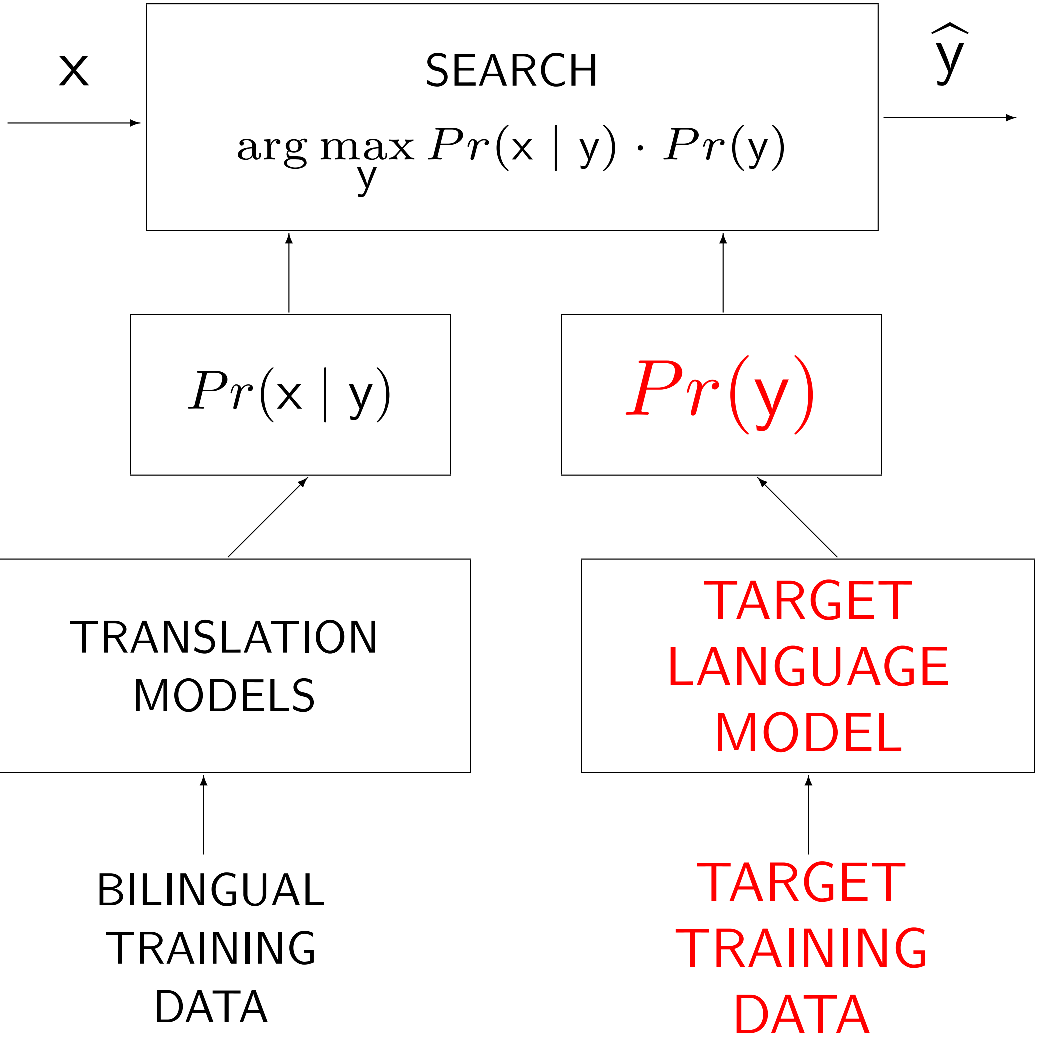
Inverse approach:

- A target-language model: $\mathrm{Pr}(y) \approx Pr(y)$
- Translation models (alignment and lexicon models): $\mathrm{Pr}(x \mid y) \approx Pr(x \mid y)$
- Search procedure: $\widehat{y} = \arg\max_{y} Pr(x \mid y) \cdot Pr(y)$

An inverse approach



An inverse approach: The target language model



Language models

Word n-grams

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

n-grams of categories

$$\Pr(\mathsf{y}) \approx Pr(\mathsf{y}) = \prod_{i=1}^{|\mathsf{y}|} p_n(C_i|C_{i-N+1} \ldots C_{i-1}) \cdot p(\mathsf{y}_i|C_i)$$

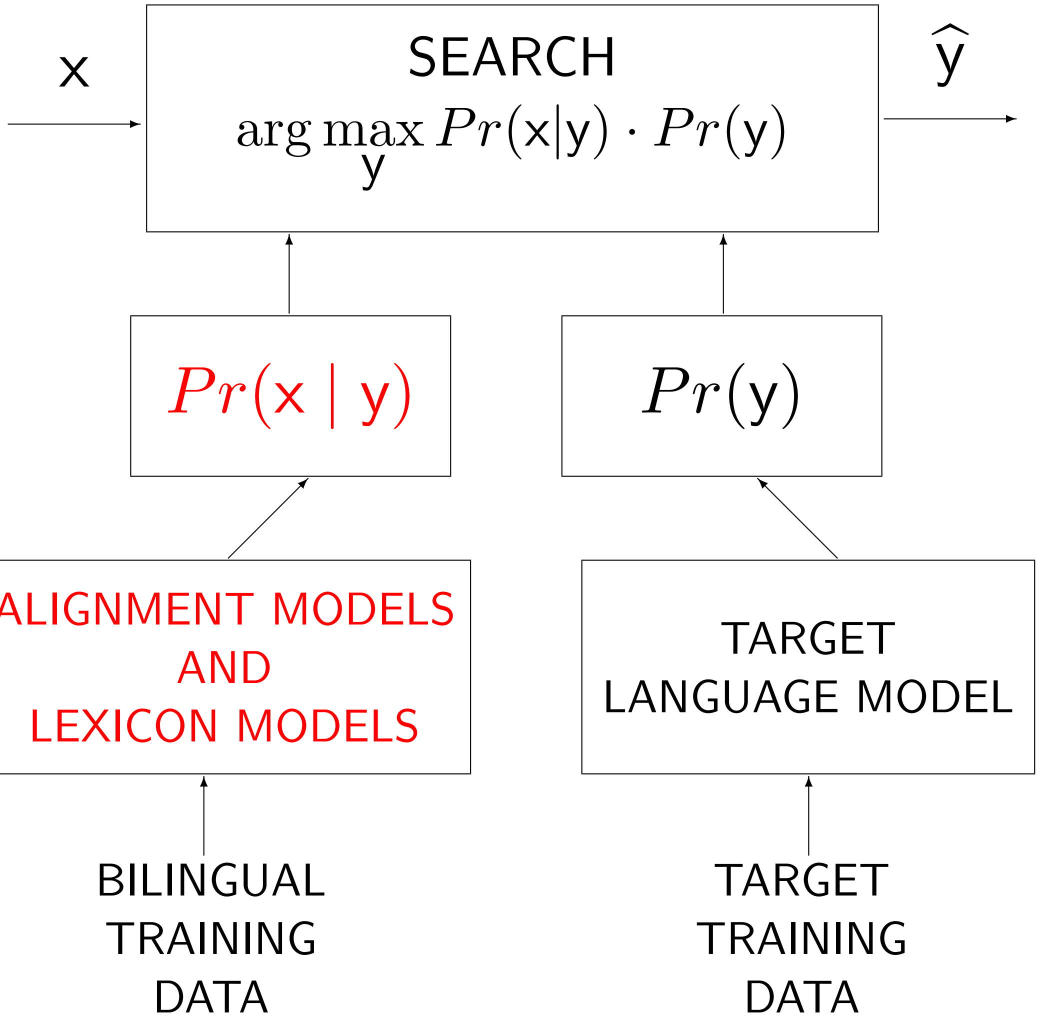
Regular or context-free grammars

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

Learning language models

- Probabilistic estimation techniques.
  - Maximum likelihood
  - Maximum entropy.
- Smoothing.
- Extensions: cache, triggers, categories, etc.
- Widely used toolkits for n-grams:
  - 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

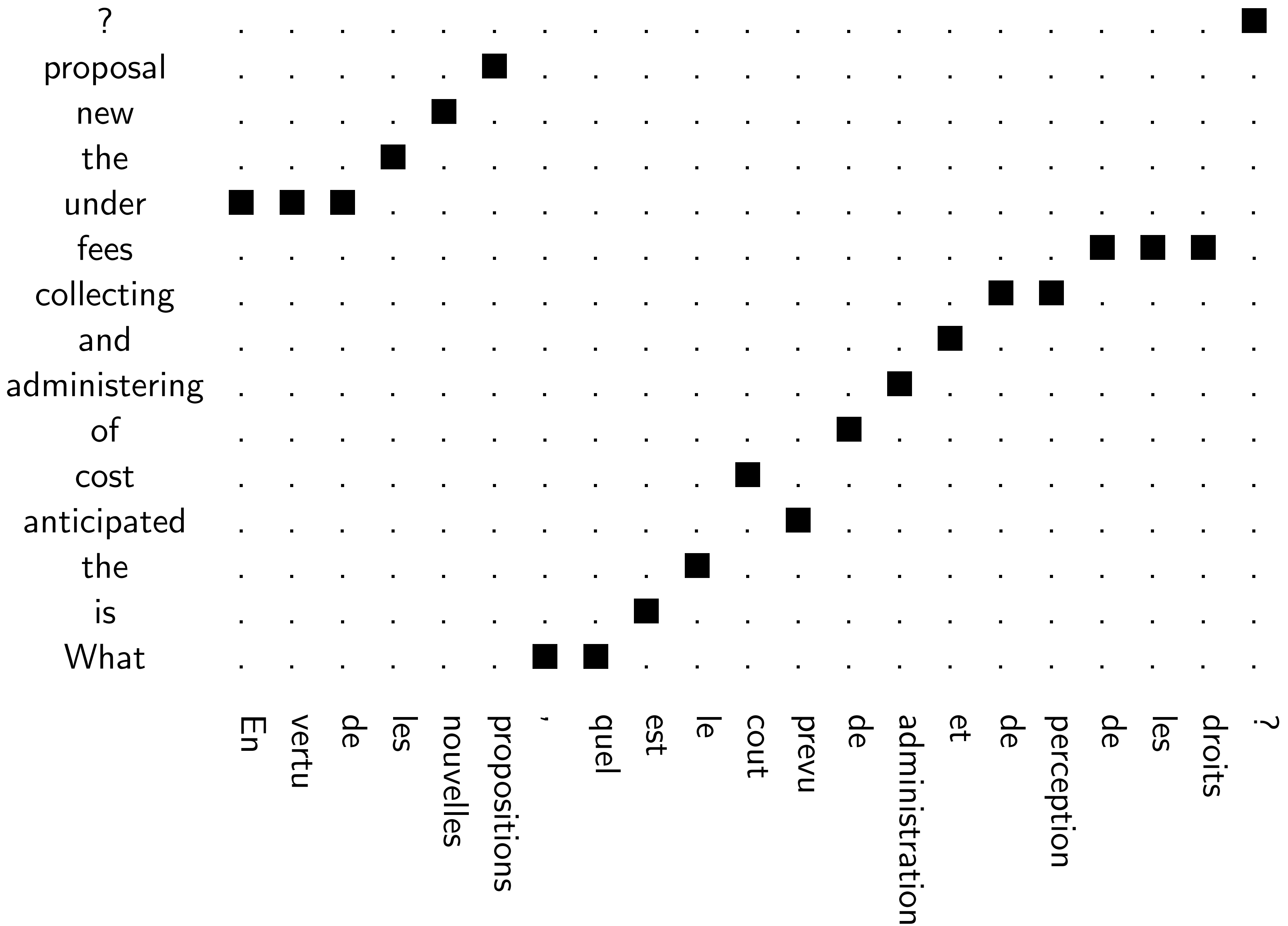
An inverse approach



Example of word alignments

	I	have	made	a	reservation	for	a	quiet	room	for	Rosario	Cabedo	.
.	·	·	·	·	·	·	·	·	·	·	·	·	■
Cabedo	·	·	·	·	·	·	·	·	·	·	·	■	·
Rosario	·	·	·	·	·	·	·	·	·	·	■	·	·
de	·	·	·	·	·	·	·	·	·	■	·	·	·
nombre	·	·	·	·	·	·	·	·	·	■	·	·	·
a	·	·	·	·	·	·	·	·	·	■	·	·	·
tranquila	·	·	·	·	·	·	·	■	·	·	·	·	·
habitación	·	·	·	·	·	·	·	·	■	·	·	·	·
una	·	·	·	·	·	·	■	·	·	·	·	·	·
de	·	·	·	·	·	■	·	·	·	·	·	·	·
reserva	·	·	·	·	■	·	·	·	·	·	·	·	·
la	·	·	·	■	·	·	·	·	·	·	·	·	·
hecho	·	·	■	·	·	·	·	·	·	·	·	·	·
he	·	■	·	·	·	·	·	·	·	·	·	·	·

Example of word alignments [Ney 03a]



2.2 Alignments

➢ **Alignments** [Brown 90]: $J = |\mathsf{x}|$ y $I = |\mathsf{y}|$

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

- Number of connections: $I \cdot J$
- Number of alignments: $2^{I \cdot J}$

➢ Constrain: $\mathsf{a} : \{1, ..., J\} \rightarrow \{0, ..., I\}$, ($\mathsf{a}_j = 0 \Rightarrow j$ in x is not aligned with any position in y).

- Number of alignments: $(I + 1)^J$

➢ Set of possible alignments: $\mathcal{A}(\mathsf{x}, \mathsf{y})$

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

$$\Pr(\mathsf{x} \mid \mathsf{y}) = \sum_{\mathsf{a} \in \mathcal{A}(\mathsf{y},\mathsf{x})} \Pr(\mathsf{x}, \mathsf{a} \mid \mathsf{y})$$

2.2 Alignments

$$\begin{aligned}\Pr(\mathsf{x},\mathsf{a} \mid \mathsf{y}) &= \Pr(J \mid \mathsf{y}) \cdot \Pr(\mathsf{x},\mathsf{a} \mid J,\mathsf{y}) \\ &= \Pr(J \mid \mathsf{y}) \cdot \Pr(\mathsf{a} \mid J,\mathsf{y}) \cdot \Pr(\mathsf{x} \mid \mathsf{a}, J,\mathsf{y})\end{aligned}$$

- Length probability: $\Pr(J \mid \mathsf{y})$
- **Alignment probability**: $\Pr(\mathsf{a} \mid J,\mathsf{y})$
- **Lexicon probability**: $\Pr(\mathsf{x} \mid \mathsf{a}, J,\mathsf{y})$

$$\Pr(\mathsf{a} \mid J,\mathsf{y}) = \prod_{j=1}^{J} \Pr(\mathsf{a}_j \mid \mathsf{a}_1^{j-1}, J,\mathsf{y}) \qquad \Pr(\mathsf{x} \mid \mathsf{a}, J,\mathsf{y}) = \prod_{j=1}^{J} \Pr(\mathsf{x}_j \mid \mathsf{x}_1^{j-1}, \mathsf{a}, J,\mathsf{y})$$

$$\Pr(\mathsf{x},\mathsf{a} \mid \mathsf{y}) = \Pr(J \mid \mathsf{y}) \cdot \prod_{j=1}^{J} \Pr(\mathsf{a}_j \mid \mathsf{a}_1^{j-1}, \mathsf{x}_1^{j-1}, J,\mathsf{y}) \cdot \Pr(\mathsf{x}_j \mid \mathsf{a}_1^{j}, \mathsf{x}_1^{j-1}, J,\mathsf{y})$$

Zero-order models

- Model 1
- Model 2
- The Viterbi approximation
- The search problem

Model 1

$$\Pr(\mathsf{x}, \mathsf{a} \mid \mathsf{y}) = \Pr(J \mid \mathsf{y}) \cdot \prod_{j=1}^{J} \Pr(\mathsf{a}_j \mid \mathsf{a}_1^{j-1}, \mathsf{x}_1^{j-1}, J, \mathsf{y}) \cdot \Pr(\mathsf{x}_j \mid \mathsf{a}_1^{j}, \mathsf{x}_1^{j-1}, J, \mathsf{y})$$

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

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

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

Model 1

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

Generative process: Given a target sentence y of length I,

1. Choose the length of the source sentence J according to $n(J|I)$
2. For each $1 \leq j \leq J$, choose a position a_j in the target sentence according to an uniform distribution.
3. For each $1 \leq j \leq J$ choose a source word x_j according to $l(\mathsf{x}_j \mid \mathsf{y}_{\mathsf{a}_j})$

Model 1: An example

Given y: *a* *double* *room* $(I = 3)$

Choose J ($n(J \mid 3)$): ($J = 5$)	1	2	3	4	5
Choose a_j (uniform)	1 a	3 room	2 double	2 double	2 double
Choose x_j ($l(x_j \mid y_i)$)	Una	habitación	con	dos	camas

Model 2

$$\Pr(\mathsf{x}, \mathsf{a} \mid \mathsf{y}) = \Pr(J \mid \mathsf{y}) \cdot \prod_{j=1}^{J} \Pr(\mathsf{a}_j \mid \mathsf{a}_1^{j-1}, \mathsf{x}_1^{j-1}, J, \mathsf{y}) \cdot \Pr(\mathsf{x}_j \mid \mathsf{a}_1^{j}, \mathsf{x}_1^{j-1}, J, \mathsf{y})$$

- $\Pr(J \mid \mathsf{y}) \approx n(J|I)$
- $\Pr(\mathsf{a}_j \mid \mathsf{a}_1^{j-1}, \mathsf{x}_1^{j-1}, J, \mathsf{y}) \approx a(\mathsf{a}_j \mid j, J, I)$
- $\Pr(\mathsf{x}_j \mid \mathsf{a}_1^{j}, \mathsf{x}_1^{j-1}, J, \mathsf{y}) \approx l(\mathsf{x}_j \mid \mathsf{y}_{\mathsf{a}_j})$

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

$a(i \mid j, J, I)$ defines **statistical alignments**

$$\Pr(\mathsf{x} \mid \mathsf{y}) \approx P_{M2}(\mathsf{x} \mid \mathsf{y}) = n(J|I) \cdot \prod_{j=1}^{J} \sum_{i=0}^{I} a(i \mid j, J, I) \cdot l(\mathsf{x}_j \mid \mathsf{y}_i)$$

Model 2

- $\Pr(J \mid \mathsf{y}) \approx n(J|I)$
- $\Pr(\mathsf{a}_j \mid \mathsf{a}_1^{j-1}, \mathsf{x}_1^{j-1}, J, \mathsf{y}) \approx a(\mathsf{a}_j \mid j, J, I)$
- $\Pr(\mathsf{x}_j \mid \mathsf{a}_1^{j}, \mathsf{x}_1^{j-1}, J, \mathsf{y}) \approx l(\mathsf{x}_j \mid \mathsf{y}_{\mathsf{a}_j})$

Generative process: Given a target sentence y of length I,

1. Choose the length of the source sentence J according to $n(J|I)$.
2. For each $1 \leq j \leq J$, choose a position a_j in the target sentence according to $a(\mathsf{a}_j \mid j, J, I)$.
3. For each $1 \leq j \leq J$ choose a source word x_j according to $l(\mathsf{x}_j \mid \mathsf{y}_{\mathsf{a}_j})$.

Model 2: An example

Given y: *a* *double* *room* $(I = 3)$

Choose J $(n(J \mid 3))$: $(J = 5)$	1	2	3	4	5
Choose a_j $(a(\mathrm{a}_j \mid, j, I, J))$	1 a	3 room	2 double	2 double	2 double
Choose x_j $(l(\mathrm{x}_j \mid \mathrm{y}_i))$	Una	habitación	con	dos	camas

The translation process: searching

$$\arg\max_{\mathsf{y}} Pr(\mathsf{x} \mid \mathsf{y}) \cdot Pr(\mathsf{y})$$

A computational difficult problem [Knight 99]

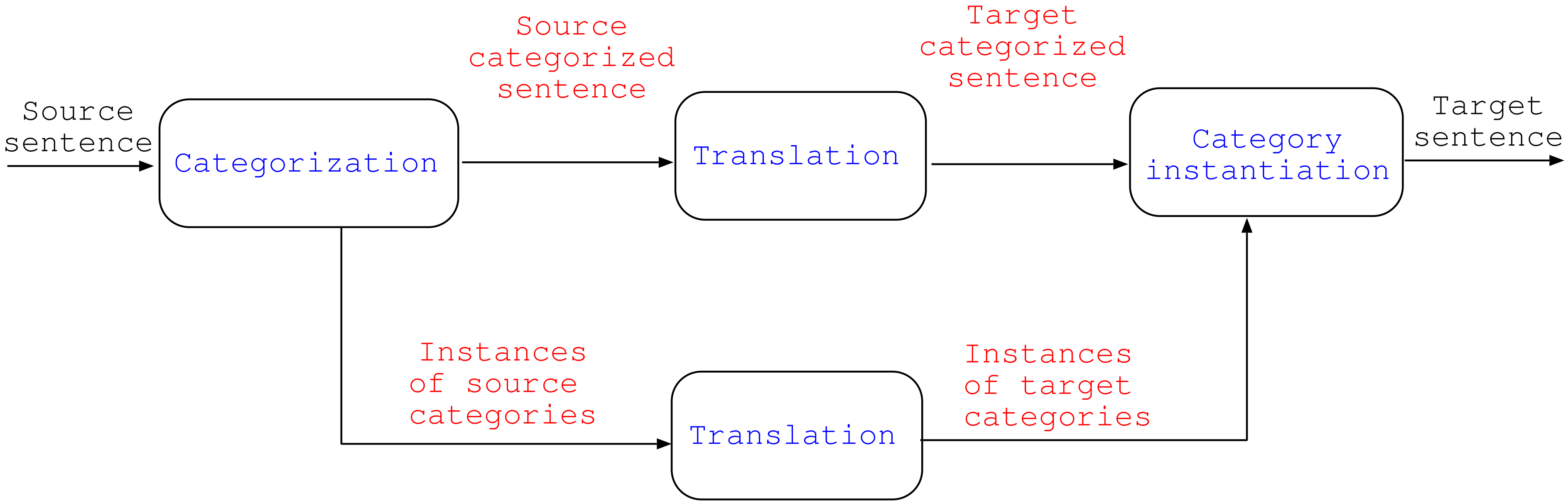
Algorithmic solutions:

- Dynamic Programming like [Ney 00a]
- Stack-Decoding: A* or Branch & Bound [Brown 90]

2.4 Categorization in MT

- 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



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

An example

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

Automatic categorization

- *Extended word categories* [Barrachina 99]

  1. Align a bilingual corpus
  2. Build extended words using the alignments
  3. Apply a clustering algorithm to the corpus of extended word sentences

- *Statistical bilingual categories* [Och 99]

  1. Align a bilingual corpus
  2. Apply a clustering algorithm to the target corpus.
  3. Apply a clustering algorithm to the source corpus taking into account the categories of target words aligned to the source words.

Index

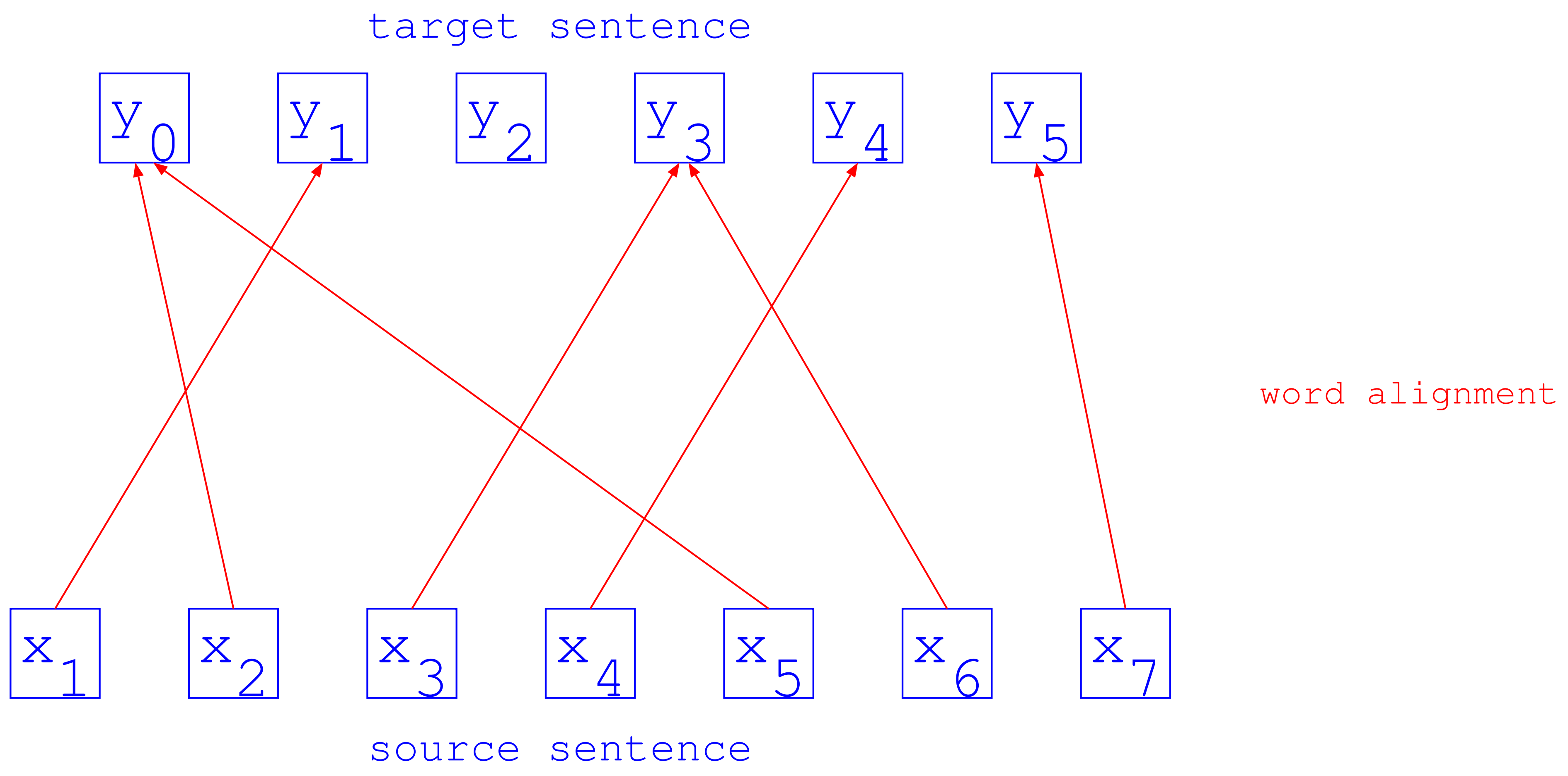
Alignments

$$\Pr(\mathsf{x} \mid \mathsf{y}) = \sum_{\mathsf{a} \in \mathcal{A}(\mathsf{y},\mathsf{x})} \Pr(\mathsf{x}, \mathsf{a} \mid \mathsf{y}) = \Pr(J \mid \mathsf{y}) \cdot \sum_{\mathsf{a} \in \mathcal{A}(\mathsf{y},\mathsf{x})} \Pr(\mathsf{x}, \mathsf{a} \mid J, \mathsf{y})$$

Alignment probabilities and lexicon probabilities

- Model 1
- Model 2

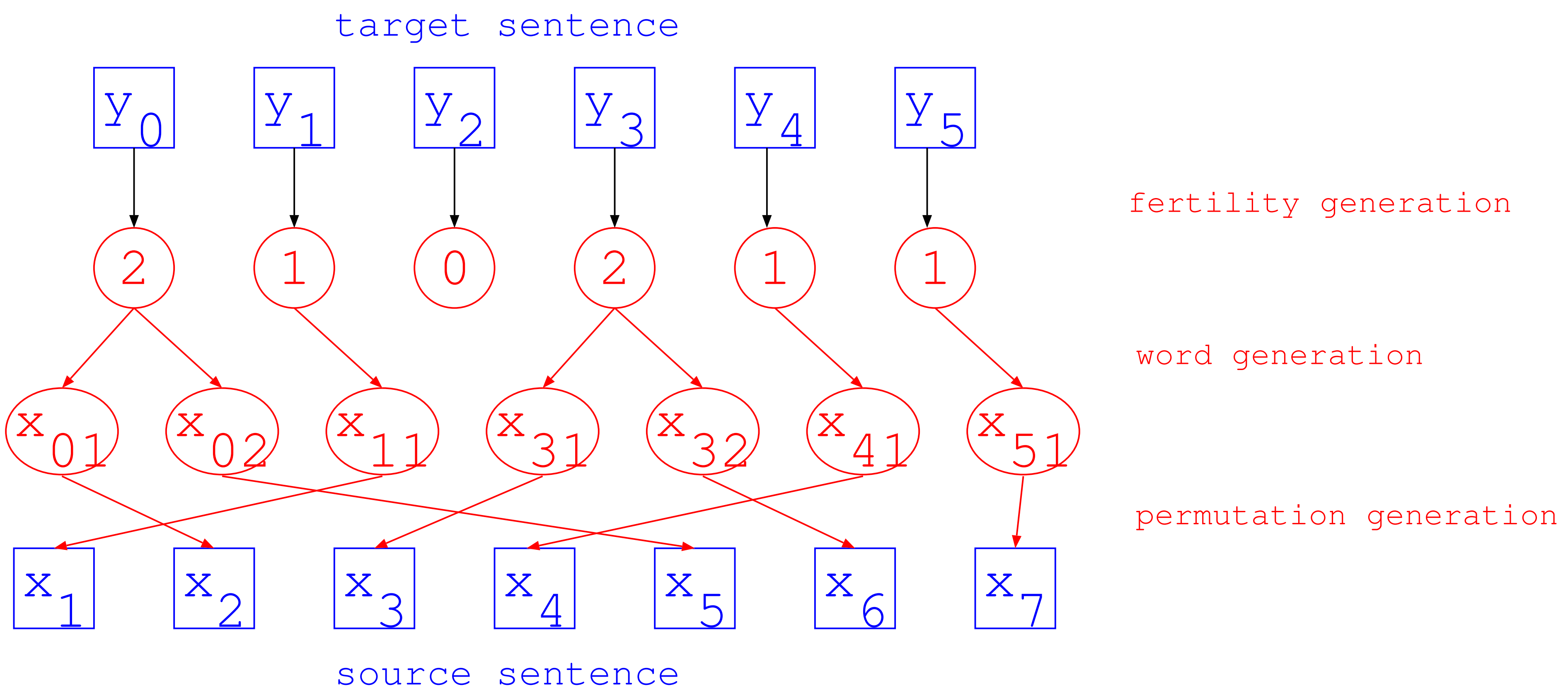
Models 1, 2 or HMM



Models 3, 4 and 5

- Model 3: Lexicon, fertility and distortion models
- Model 4 is a refined version of distortion distribution in Model 3
- Model 5 is a consistent version of distortion distribution in Model 4

Fertility


target sentence
y 0
y 1
y 2
y 3
y 4
y 5
fertility generation
2
1
0
2
1
1
word generation
x 01
x 02
x 11
x 31
x 32
x 41
x 51
permutation generation
x 1
x 2
x 3
x 4
x 5
x 6
x 7
source sentence

3.1 Fertility-based models

Fertility ϕ of $\mathrm{y}_i \in \Delta$: number of the source words connected to an target word y_i

1. Choose how many source words are connected to a target word y_i: *fertility* of y_i

2. Choose a set of the source words, a *tablet* τ_i, that is connected to i-th target word

3. Choose the *position* $\pi_{i,k}$ in the source sentence of the k-th word $\tau_{i,k}$ that is connected to the i-th target word

Model 3

Given a target sentence y of length I:

1. For each $1 \leq i \leq I$ choose a length ϕ_i
2. Choose a length ϕ_0
3. $J = \sum_{i=0}^{I} \phi_i$.
4. For each $1 \leq i \leq I$ and $1 \leq k \leq \phi_i$, choose a source word
5. For each $1 \leq i \leq I$ and $1 \leq k \leq \phi_i$, choose a position
6. If any position has been choosen then **error** (*inconsistent model*).
7. For each $1 \leq k \leq \phi_0$ choose a positionfrom the vacant positions according to a uniform distribution.

Example

Given y: *a* *double* *room* $(I = 3)$

i	1		2		3
Choose $\phi(\mathrm{y}_i) = \phi$ using $f(\phi\|\mathrm{y}_i)$	1		3		1
Choose $\tau_{i,k} = x$ using $l(x\|\mathrm{y}_i)$	{una}	{con,	camas,	dos}	{habitación}
Choose $\pi_{i,k} = j$ using $d(j\|i, I, J)$	1	3	5	4	2

j	1	2	3	4	5
x	una	habitación	con	dos	camas

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)

Conventional IBM Models Training

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

$$\hat{\theta} = \arg\max_{\theta} \prod_{n=1}^{N} \sum_{\mathsf{a}} p_{\theta}(\mathsf{x}^{(n)}, \mathsf{a} | \mathsf{y}^{(n)}) \quad .$$

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

3.2 The search problem

$$\widehat{\mathsf{y}} = \arg\max_{\mathsf{y}} Pr(\mathsf{x} \mid \mathsf{y}) \cdot Pr(\mathsf{y})$$

➢ Search is a **NP**-Hard problem. [Knight 99]

➢ Algorithmic solutions: (+ heuristics for efficient suboptimal solutions)

- *Dynamic Programming* [Tillmann 03]
- *Stack-decoding*, *A* or Branch & Bound* (Ortiz , 2003)

Some stack-decoding proposals

- Candide systems from IBM [Berger et al. 96]: Multiple stacks, model 3.
- Multiple stack-decoding [Wang and Waibel 98]: Model 2.
- Algorithm $A^{\star}$ [Ueffing et al. 01]: model 4.
- Basic stack-decoding strategy:
  - Origin of the *stack decoding* or $A^{\star}$: ASR
  - Optimal solution to the search problem (Jelinek, 1976)
  - Incremental development of pratical hyphotesis
  - The hypothesis are stored in a prioritary queue (a type of 'stack')
  - Selection and expansion of the top of the stack(s).

A taxonomy of the stack-decoding algorithms

➢ Basic stack-decoding algorithm:

- All the hypothesis are stored in a one stack
- A hypothesis is selected in each iteration: the hypothesis with higher score in the stack

➢ Problem: hypothesis with a high number of aligned words are discarded.

➢ Possible solutions:

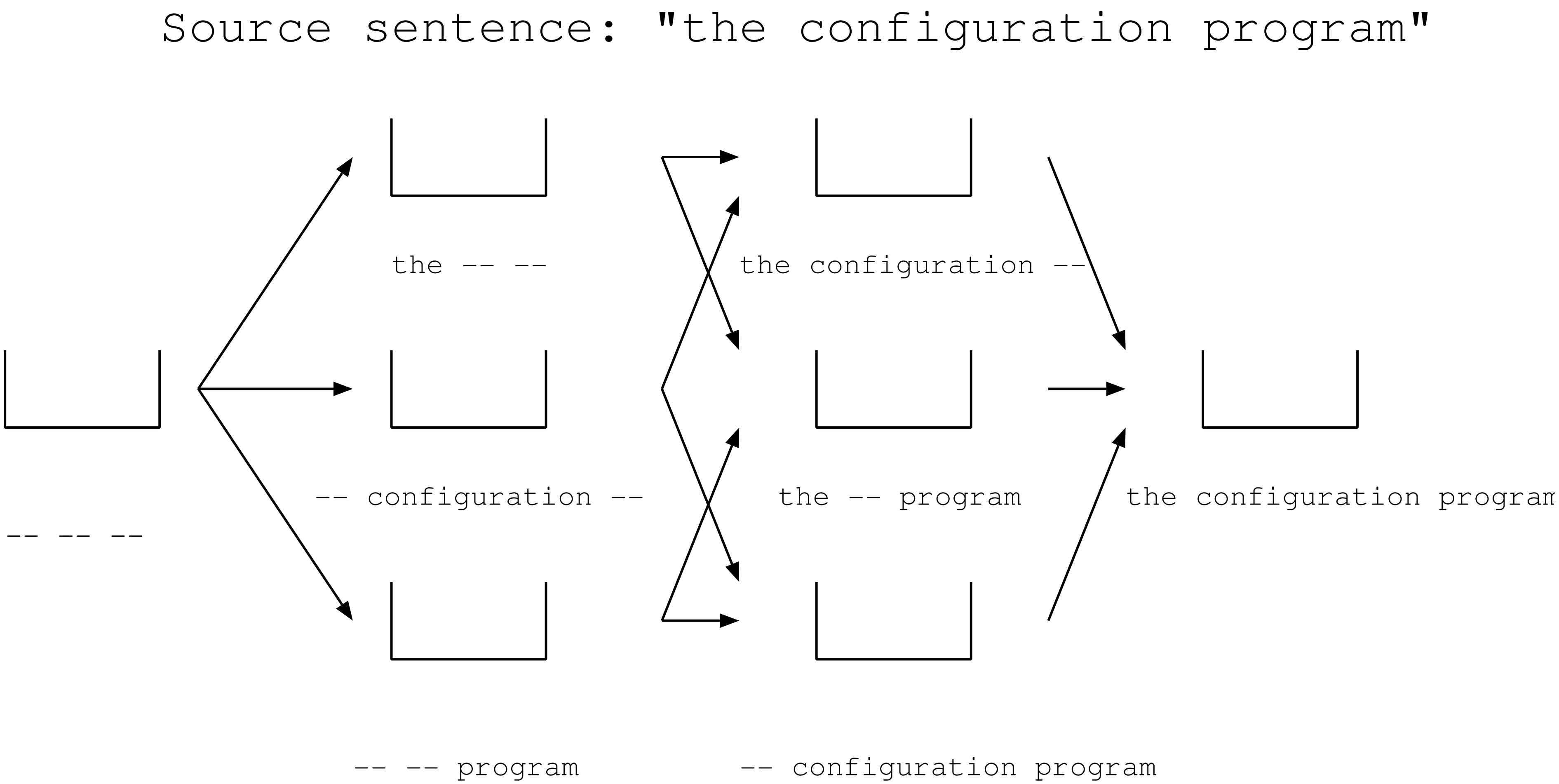
- Use of heuristics: an estimation of the contribution to the set of the optimal score.
- Multiple stacks.

➢ Taxonomy:

- Single stack algorithms $A^\star$
- Multiple stack algorithms

Basic multiple stack decoding *StackDecoding*

- 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).
- 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.
- In each iteration, the best hypothesis from each available stack is selected to generate new extended hypothesis.
- The new hypothesis is stored in the corresponding stack.


Source sentence: "the configuration program"
the -- --
the configuration --
-- -- --
-- configuration --
the -- program
the configuration program
-- -- program
-- configuration program

Is the linguistic knowledge needed for statistical machine translation?

➢ YES?

- There are many linguistic knowledge available.
- The bilingual training data can be better exploited.

➢ NOT?

- Many linguistic knowledge is hard to formalize.
- The generation of new linguistic knowledge requires great human effort.

Linguistic knowledge that has been used in statistical machine translation

- Morpho-syntactic knowledge: lexicon, Part-of-Speech, etc...
  (Nießen and Ney, 2004)

  Hybrid linguistic-statistical approaches have been used with success (i.e. *hidden markov models*)

- Others: Cognates (Kondrak, Marcu and Knight, 2003), named entities (Huang, Vogel and Waibel, 2003), ...

- Syntactic information: next topic!

Morpho-syntactic knowledge in statistical machine translation

Nießen and Ney, 2004. *Statistical machine translation with scarce resources using morpho-syntactic information*. Computational Linguistics.

- Present statistical machine translation systems often treat different inflected forms of the same lemma as if they were independent of one another.

- The bilingual data can be better exploited by explicitely taking into account the interdependencies of related inflected forms.

Morpho-syntactic knowledge in statistical machine translation

yo **como** pan

- Morphological and syntactic tags (POS, tense, person, ...)
- The *base form*

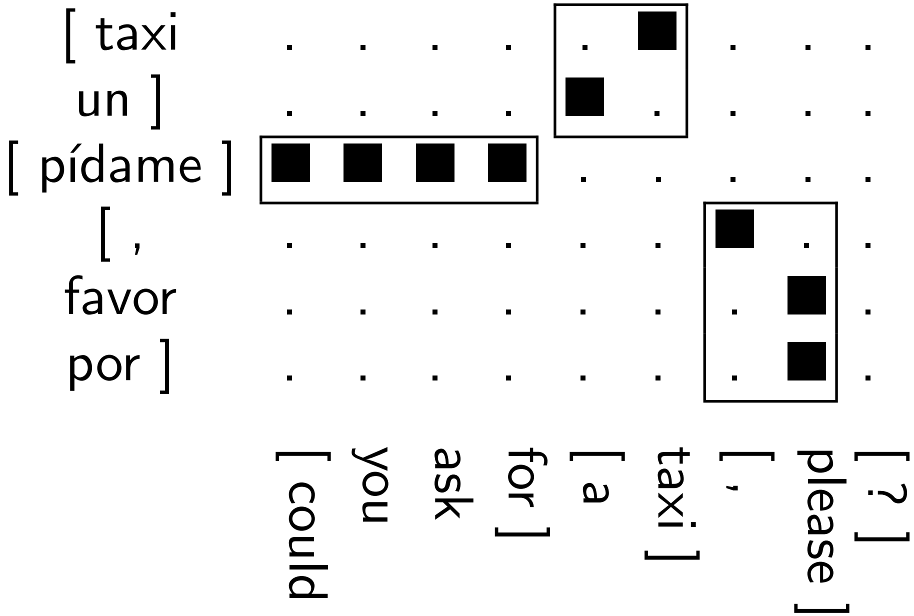
comer verb indicative present singular 1

Index

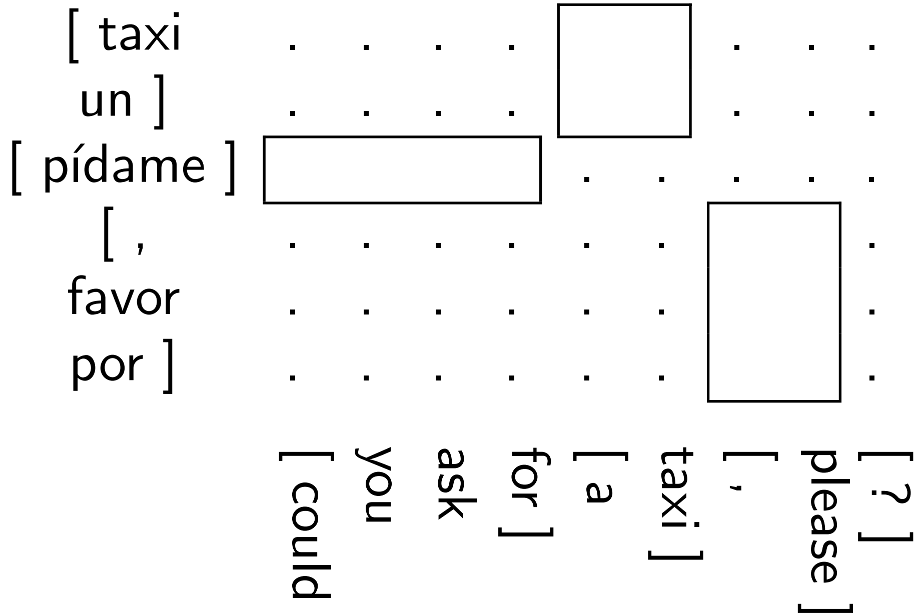
4.1 Beyond word models

- The basic assumption in the current 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.
    - *Phrase-based models*: A sequence of source words is aligned with a sequence of target words. The basic unit is the phrase.

Word sequences



Alignment templates



Bilingual phrases

Word sequences

The statistical dictionaries of single word pairs are substituted by statistical dictionaries of *bilingual phrases*.

Bilingual phrases are related with a bilingual segmentation.

- Problem: The generalisation capability, since only sequences of segments that have been seeing in the training corpus are accepted.

- Problem: The selection of adequate bilingual phrases.

An example

y: could you ask for a taxi , please ?

	y	could you ask for	a taxi	, please ?
	i	1 2 3 4	5 6	7 8 9=I
Segmentation	i	i_1	i_2	i_3
Translation	x	[pídame]	[un taxi .]	[por favor ,]
Permutation	α	$\alpha_1 = 2$	$\alpha_2 = 3$	$\alpha_3 = 1$
		por favor ,	pídame	un taxi .
	j	1 2 3	4	5 6 7
Segmentation	γ	γ_1	γ_2	γ_3

x: por favor , pídame un taxi .

Log-linear models

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

$$\hat{\mathsf{y}} = \arg\max_{\mathsf{y}} \Pr(\mathsf{y} \mid \mathsf{x})$$

$$\hat{\mathsf{y}} = \arg\max_{\mathsf{y}} \frac{\exp\left(\sum_{k=1}^{K} \lambda_k h_k(\mathsf{x},\mathsf{y})\right)}{\sum_{\mathsf{y}'} \exp\left(\sum_{k=1}^{K} \lambda_k h_k(\mathsf{x},\mathsf{y}')\right)} = \arg\max_{\mathsf{y}} \sum_{k=1}^{K} \lambda_k h_k(\mathsf{x},\mathsf{y})$$

- $h_1(\mathsf{x},\mathsf{y}) = \log Pr(\mathsf{y})$, a language model
- $h_2(\mathsf{x},\mathsf{y}) = \log Pr_{PB}(\mathsf{y} \mid \mathsf{x})$, phrase-based models
- $h_3(\mathsf{x},\mathsf{y}) = \log Pr_{PB}(\mathsf{x} \mid \mathsf{y})$, phrase-based inverse model
- $h_4(\mathsf{x},\mathsf{y}) = \log Pr_{M1}(\mathsf{x} \mid \mathsf{y})$, statistical dictionaries
- $h_5(\mathsf{x},\mathsf{y}) = \log Pr_{M1}(\mathsf{y} \mid \mathsf{x})$, statistical inverse dictionaries
- ...

Learning phrase-based models

Given a sentence-aligned corpus $\mathcal{T}$:

- A word-aligned corpus is generated using the GIZA++ toolkit with $\mathcal{T}$

  http://www-i6.informatik.rwth-aachen.de/Colleagues/och/software/GIZA++.html
- A set of bilingual word sequences from the word aligned corpus is extracted.
- The parameters of the phrase-model are estimated.

Estimating the parameters

Estimating the parameters

By relative frequencies, for each pair of segments (x, y):

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

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

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

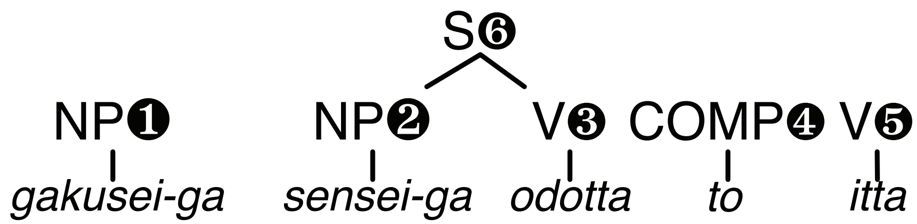
where p_0 is a parameter to be ajusted using a validation set.

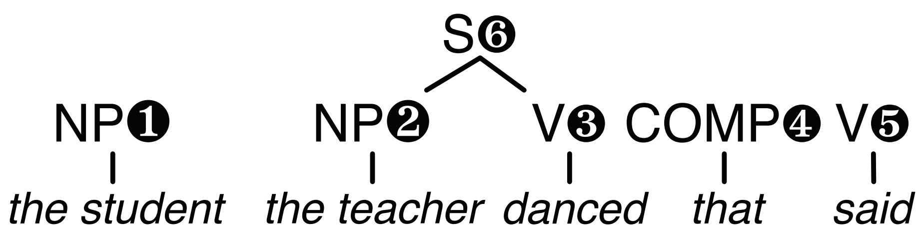
Index

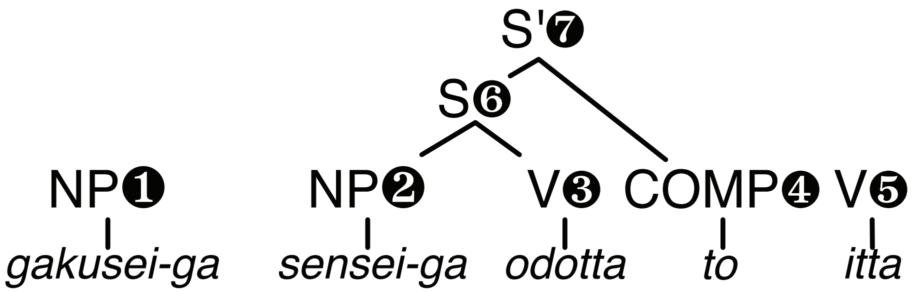
Example SCFG*

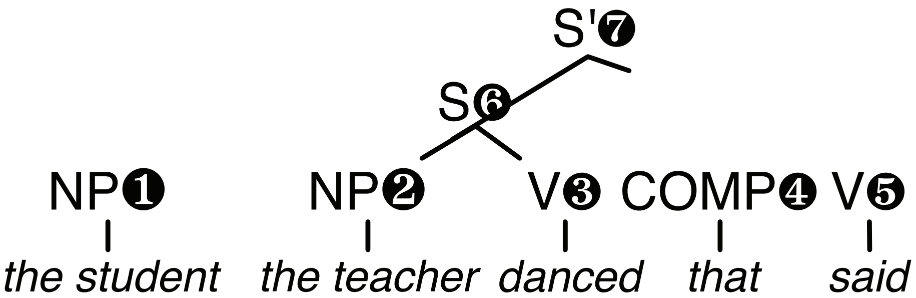
	Japanese	English
S →	NP① VP②	NP① VP②
S'→	S① COMP②	COMP② S①
VP →	NP① V②	V② NP①
NP →	*gakusei-ga*	*student*
NP →	*sensei-ga*	*teacher*
V →	*odotta*	*danced*
V →	*itta*	*said*
COMP →	*to*	*that*

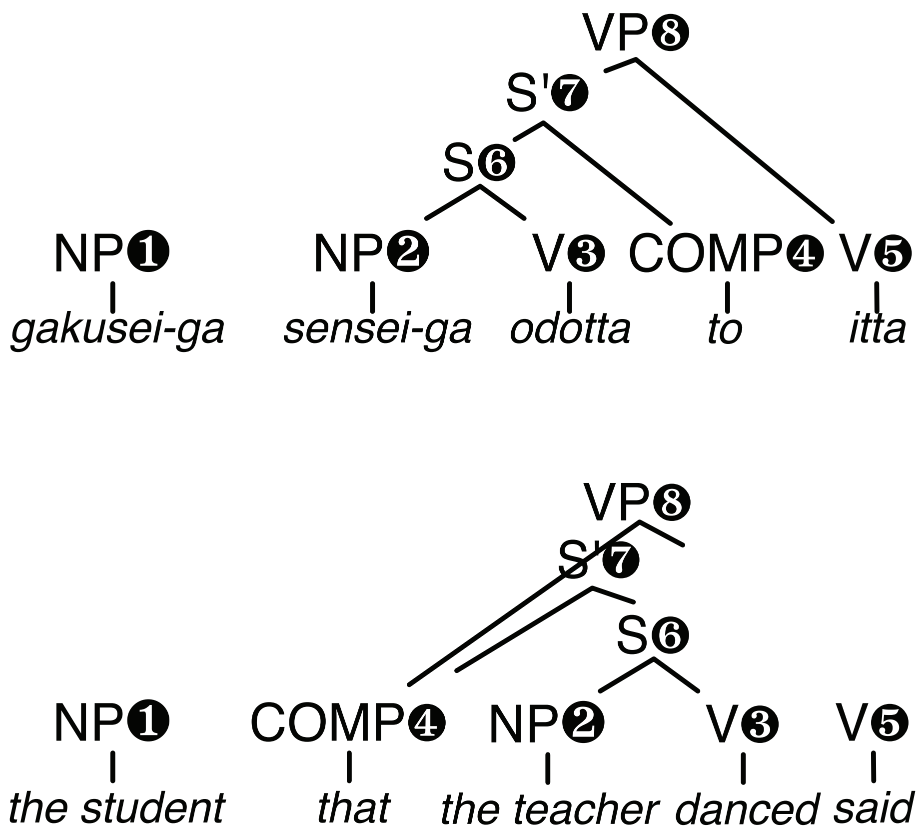
* Slide source: http://www.mt-archive.info/MTMarathon-2009-Li-ppt.pdf

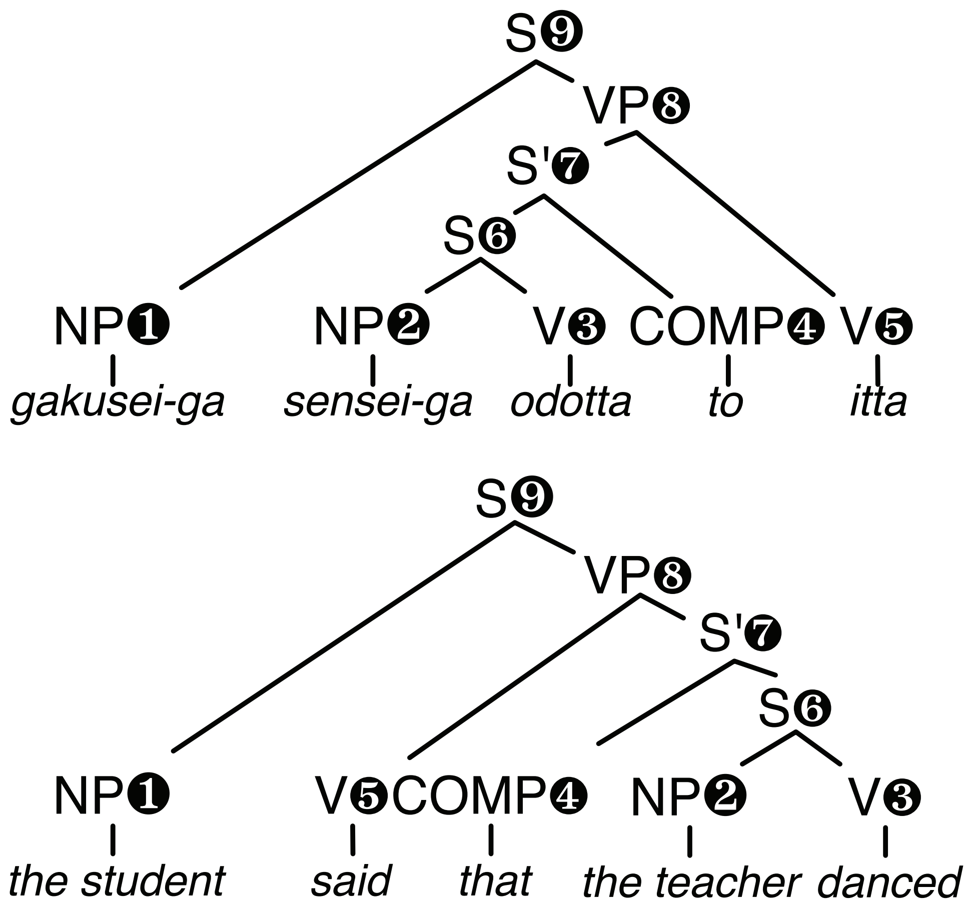

S❻
NP❶
NP❷
V❸
COMP❹
V❺
gakusei-ga
sensei-ga
odotta
to
itta


S❻
NP❶
NP❷
V❸
COMP❹
V❺
the student
the teacher
danced
that
said


S'❼
S❻
NP❶
NP❷
V❸
COMP❹
V❺
gakusei-ga
sensei-ga
odotta
to
itta


S'❼
S❻
NP❶
NP❷
V❸
COMP❹
V❺
the student
the teacher
danced
that
said

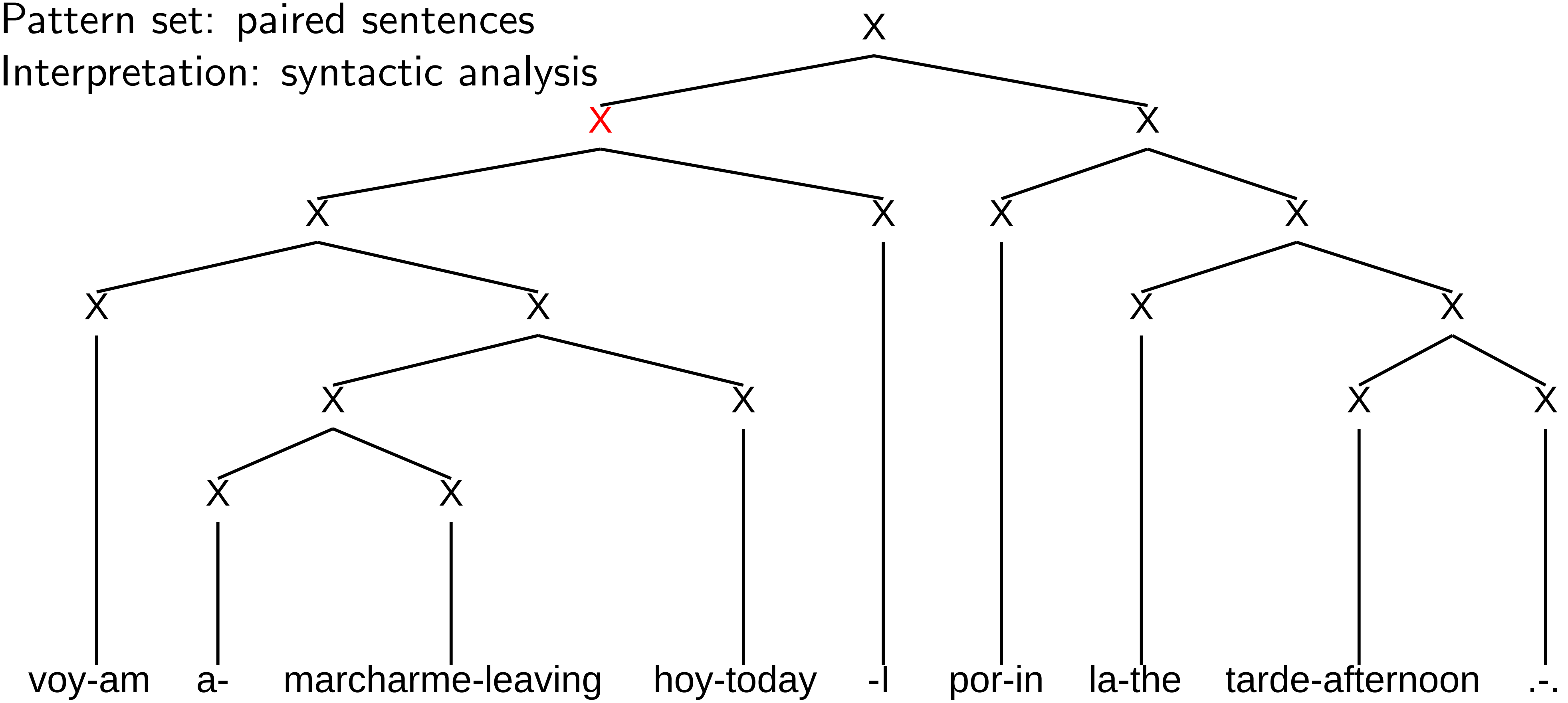

VP❽
S'❼
S❻
NP❶
NP❷
V❸
COMP❹
V❺
gakusei-ga
sensei-ga
odotta
to
itta
VP❽
S'❼
S❻
NP❶
COMP❹
NP❷
V❸
V❺
the student
that
the teacher
danced
said


S❾
VP❽
S'❼
S❻
NP❶
NP❷
V❸
COMP❹
V❺
gakusei-ga
sensei-ga
odotta
to
itta
S❾
VP❽
S'❼
S❻
NP❶
V❺
COMP❹
NP❷
V❸
the student
said
that
the teacher
danced

Stochastic inversion transduction grammars [Wu 97, Maryanski 79]

- Primitives: two alphabets (words, punctuation symbols, . . .)
- Object representation: two paired written sentences

  “voy a marcharme hoy por la tarde” $\Longleftrightarrow$ “I am leaving today in the afternoon”
- Pattern set: paired sentences
- Interpretation: syntactic analysis



- **ITG:** $G = (N, W_1, W_2, R, S)$

  R is a finite set of straight orientation rules $A \rightarrow [a_1 a_2 \ldots a_r]$ and inverted orientation rules $A \rightarrow \langle a_1 a_2 \ldots a_r \rangle$, $a_i \in N \cup X$ and $X = (W_1 \cup \{\epsilon\}) \times (W_2 \cup \{\epsilon\})$

  Theorem. For any ITG G, there exists an equivalent ITG G' in which every production takes one of the following forms:

$$\begin{array}{lll} S \rightarrow \epsilon/\epsilon & A \rightarrow x/\epsilon & A \rightarrow [BC] \\ S \rightarrow x/y & A \rightarrow \epsilon/y & A \rightarrow \langle BC \rangle \end{array}$$

- **SITG:** $G_s = (G, p)$ where:

  - G is an ITG

  - p is a function that attaches a probability to each rule:
    $p : R \rightarrow]0, 1]$

$$\sum_{1 \leq j \leq n_i} p(A_i \rightarrow \alpha_j) = 1, \qquad \forall A_i \in N$$

Stochastic derivation for SITG

Given a sequence of stochastic events:

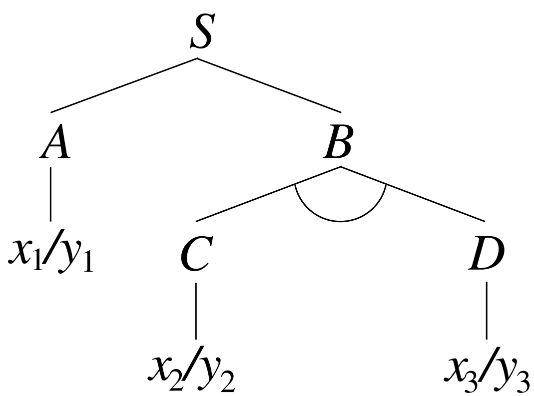
$$(S,S) = (\alpha_0,\beta_0) \overset{r_1}{\Rightarrow} (\alpha_1,\beta_1) \overset{r_2}{\Rightarrow} (\alpha_2,\beta_2) \cdots (\alpha_{m-1},\beta_{m-1}) \overset{r_m}{\Rightarrow} (\alpha_m,\beta_m) = (x,y)$$

Probability of (x,y) being generated by $G_s = (G,p)$ from the rule sequence $d_x = (r_1,\ldots,r_m)$, is:

$$P_{G_s}((x,y),d_x) = \prod_{j=1\cdots m} p(r_j)$$

Example

$$\begin{array}{lcl} S & \rightarrow & [AB] \\ A & \rightarrow & x_1/y_1 \\ B & \rightarrow & \langle CD \rangle \\ C & \rightarrow & x_2/y_2 \\ D & \rightarrow & x_3/y_3 \end{array}$$



$(S,S) \Rightarrow (AB,AB) \Rightarrow (x_1B,y_1B) \Rightarrow (x_1CD,y_1DC) \Rightarrow (x_1x_2D,y_1Dy_2) \Rightarrow (x_1x_2x_3,y_1y_3y_2)$

Probability of a string pair

$$\Pr_{G_s}(x, y) = \sum_{d_x \in D_x} \Pr_{G_s}((x, y), d_x)$$

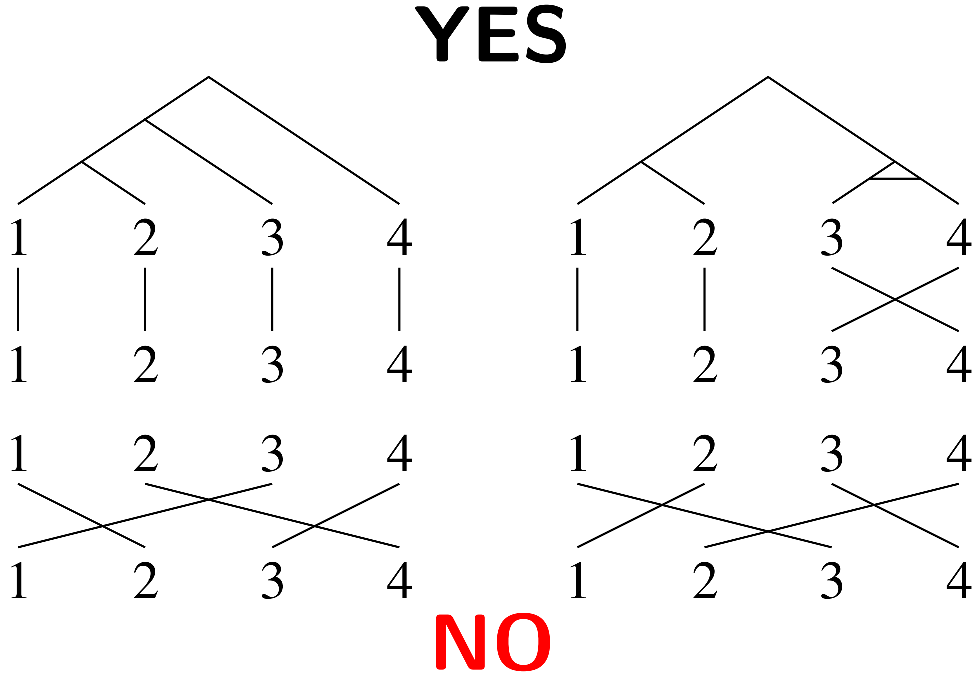
Probability of the best derivation

$$\widehat{\Pr}_{G_s}(x, y) = \max_{d_x \in D_x} \Pr_{G_s}((x, y), d_x)$$

Language generated by a SITG

$$L(G_s) = \{(x, y) \mid \Pr_{G_s}(x, y) > 0\}$$

Expressiveness of ITGs



r	ITG	all matchings	ratio
1	1	1	1.000
2	2	2	1.000
3	6	6	1.000
4	22	24	0.917
5	90	120	0.750

r	ITG	all matchings	ratio
6	394	720	0.547
7	1,806	5,040	0.358
8	8,558	40,320	0.212
9	41,586	362,880	0.115
10	206,098	3,628,800	0.057

5.2 ITG for MT

- Parsing:
  - Inside algorithm
  - Viterbi algorithm
- Learning:
  - Structure learning
  - Probabilistic estimation: Inside-outside estimation
    Viterbi-based estimation
- Translation:
  - Adapted Cooke-Kasami-Younger parser algorithm

Viterbi algorithm [Wu 97, Gascó 10b]

- Given $(x,y) \in (W_1^*, W_2^*)$ and $A \in N$

$$\delta_{i,j,k,l}(A) = \widehat{\Pr}(A \stackrel{*}{\Rightarrow} x_{i+1} \cdots x_j / y_{k+1} \cdots y_l)$$

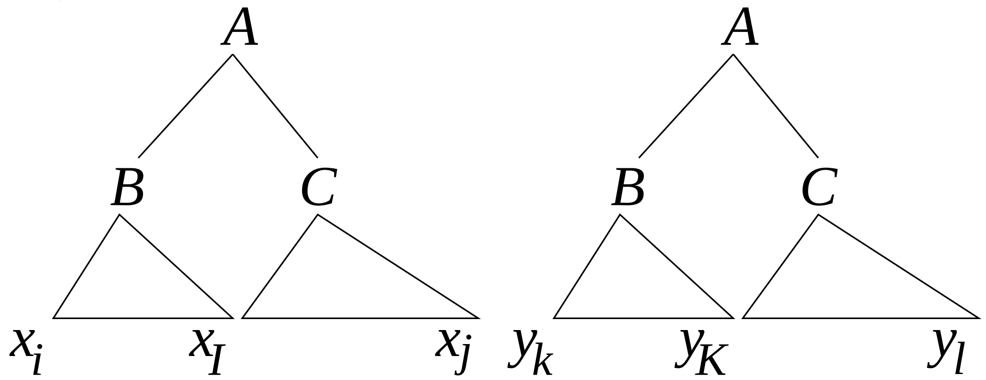
- Initialization

$$\begin{aligned} \delta_{i-1,i,k-1,k}(A) &= p(A \rightarrow x_i/y_k) && 1 \leq i \leq |x|, 1 \leq k \leq |y| \\ \delta_{i-1,i,k,k}(A) &= p(A \rightarrow x_i/\epsilon) && 1 \leq i \leq |x|, 0 \leq k \leq |y| \\ \delta_{i,i,k-1,k}(A) &= p(A \rightarrow \epsilon/y_k) && 0 \leq i \leq |x|, 1 \leq k \leq |y| \end{aligned}$$

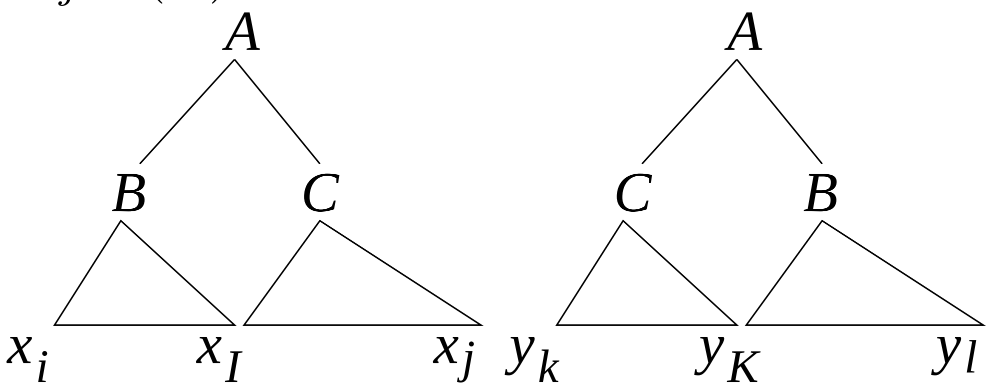
- Recursion. For all $A \in N$, and i, j, k, l such that $0 \leq i < j \leq |x|$, $0 \leq k < l \leq |y|$ and $j - i + l - k \geq 2$:

$$\delta_{ijkl}(A) = \max(\delta^{[]}_{ijkl}(A), \delta^{\langle\rangle}_{ijkl}(A))$$

$$\delta^{[]}_{ijkl}(A) = \max_{\substack{B,C \in N \\ i \leq I \leq j, k \leq K \leq l \\ ((j-I)+(l-K)) \times ((l-i)+(K-k) \neq 0}} p(A \rightarrow [BC])\delta_{iIkK}(B)\delta_{IjKl}(C)$$



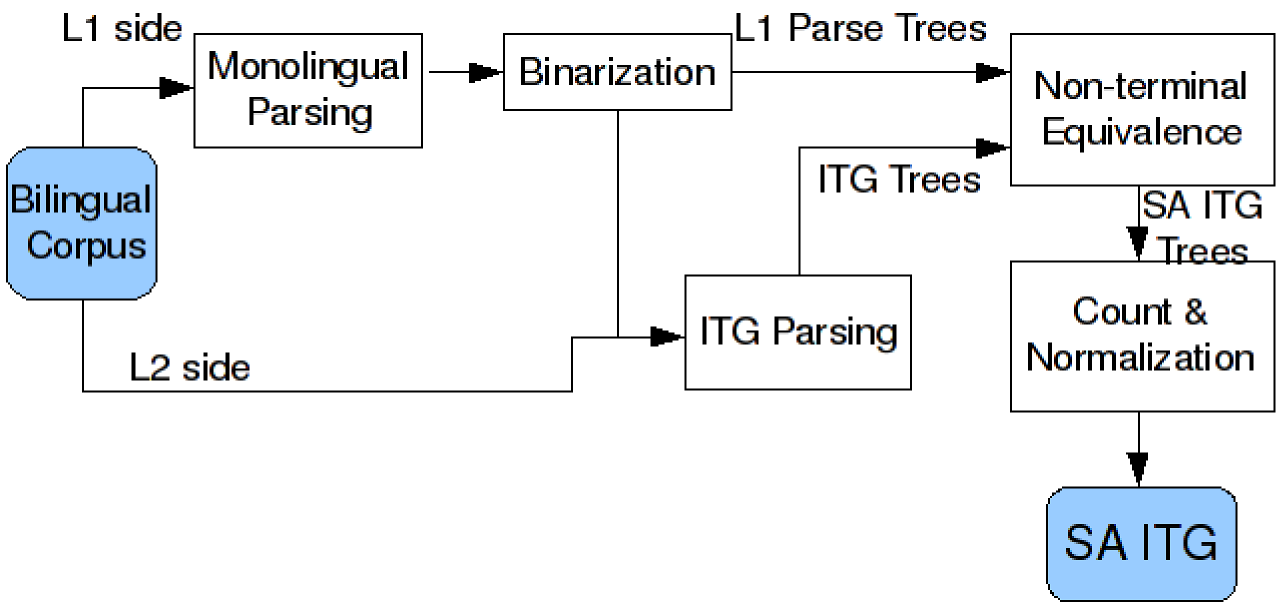
$$\delta^{\langle\rangle}_{ijkl}(A) = \max_{\substack{B,C \in N \\ i \leq I \leq j, k \leq K \leq l \\ ((j-I)+(K-k)) \times ((I-i)+(I-K)) \neq 0}} p(A \rightarrow \langle BC \rangle)\delta_{iIKl}(B)\delta_{IjkK}(C)$$



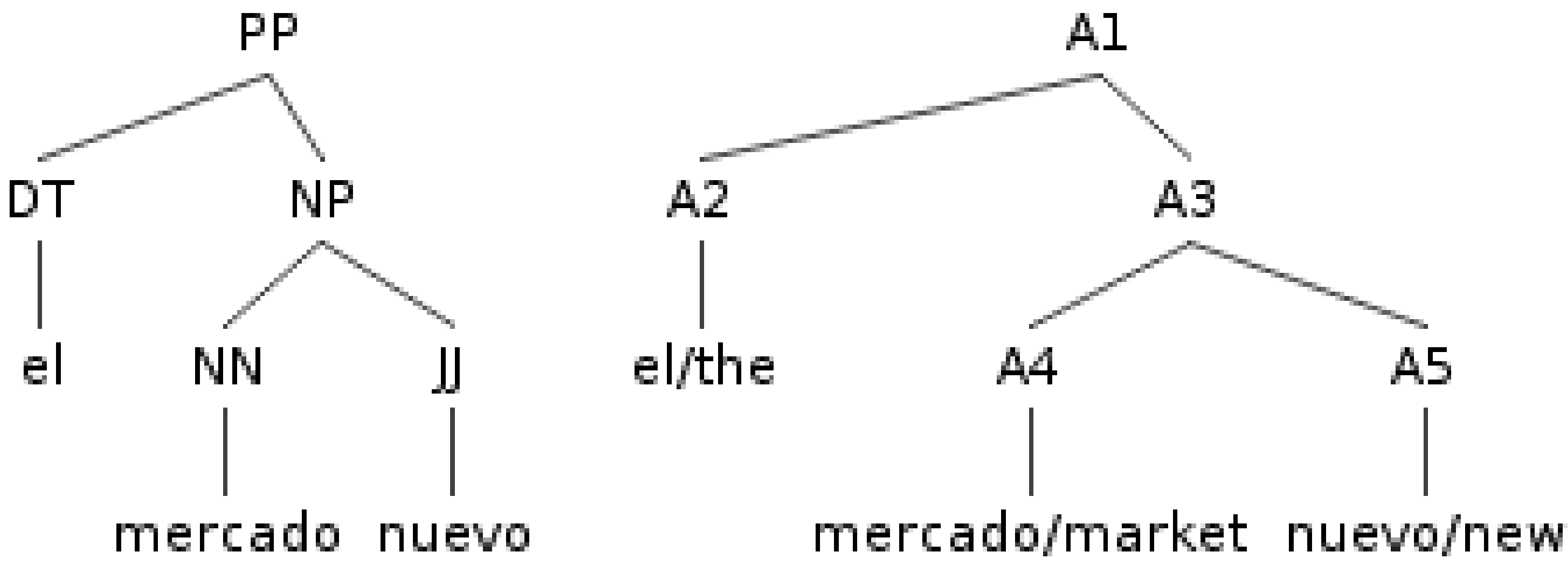
5.2 ITG for MT

[Gascó 10a]

1. Create an initial SITG
2. Estimate the probabilities
3. Attach linguistic information to the non-terminal symbols



5.2 ITG for MT


PP
DT
NP
el
NN
JJ
mercado
nuevo
A1
A2
A3
el/the
A4
A5
mercado/market
nuevo/new

5.2 ITG for MT

- IWSLT 2008 (Chinese-English BTEC)
- Standard tools: GIZA++, ZMERT
- Stanford parser for Chinese
- Baseline: Moses, 5-gram

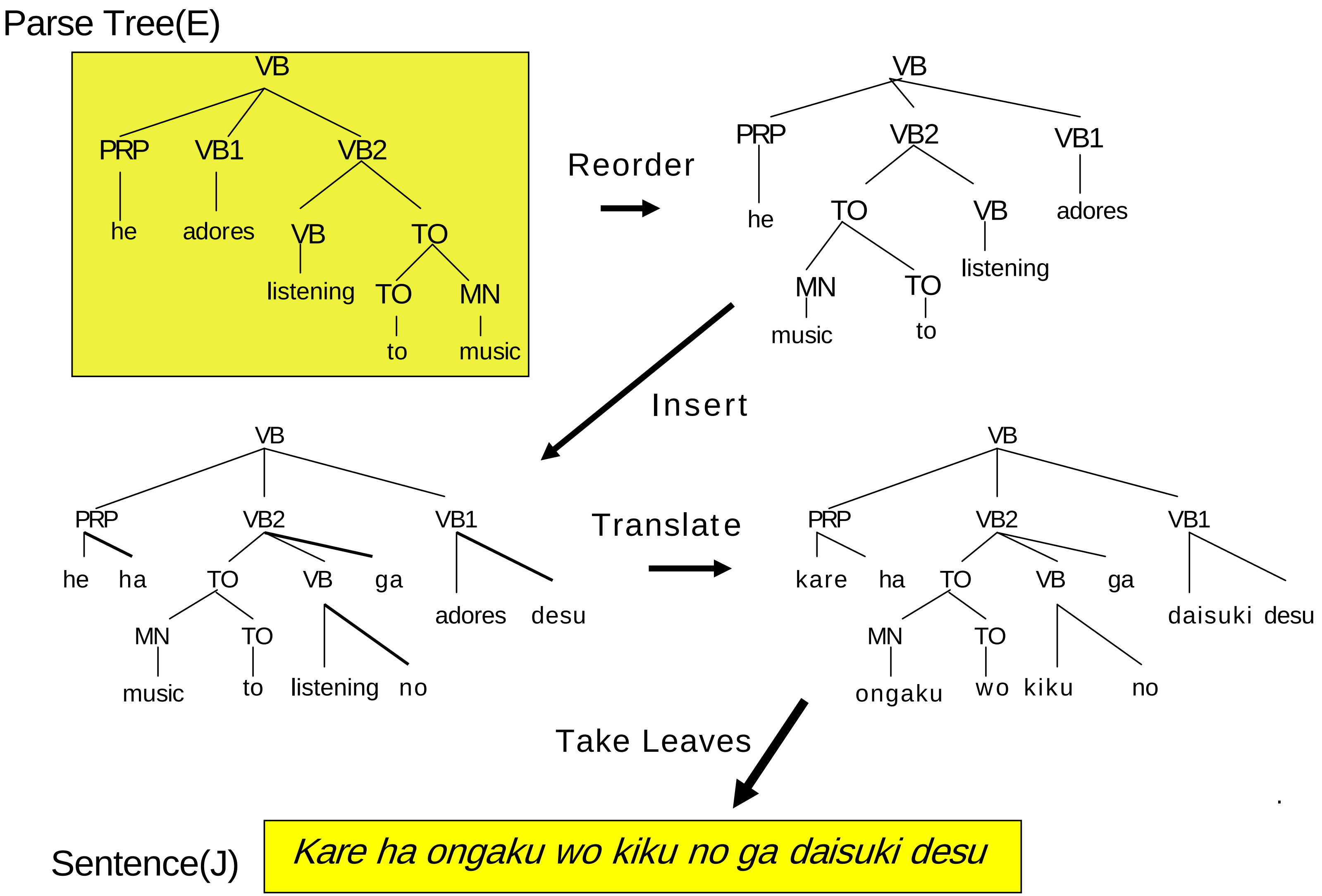
Corpus Set	Statistic	Chinese	English
Training	Sentences	42,655	
	Words	330,163	380,431
	Voc. Size	8,773	8,387
DevSet	Sentences	489	
	Words	3,169	3,861
	OOV Words	111	115
Test	Sentences	507	
	Words	3,357	-
	OOV Words	97	-

System	%BLEU
Baseline PBT	41.1
Initial ITG	41.2
Re-estimated ITG	41.8
Source SAITG	42.9
Target SAITG	43.0

Main ideas [Yamada 01]

- The input sentence is preprocessed by a syntactic parser
- The channel performs operations on each node of the parse tree:
  - reordering child nodes
  - inserting extra words at each node
  - translating leaf words
- The output of the the model is a string.

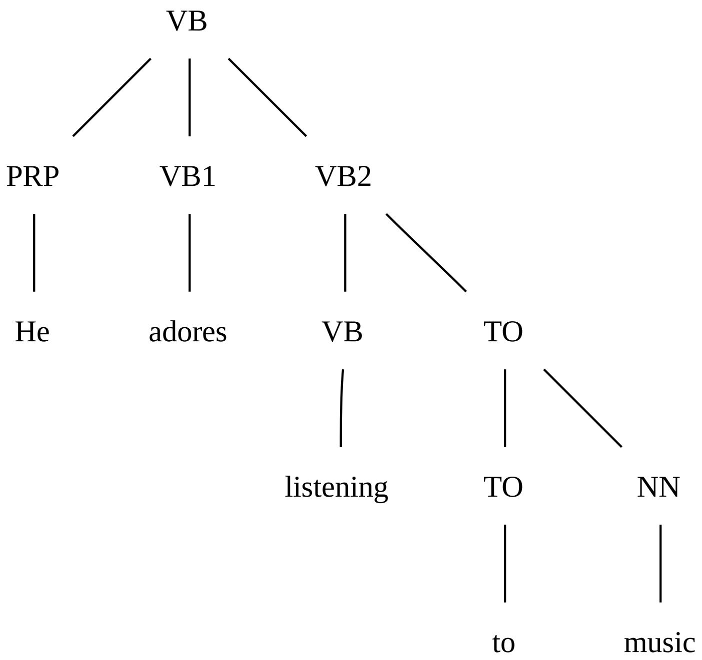
An example*

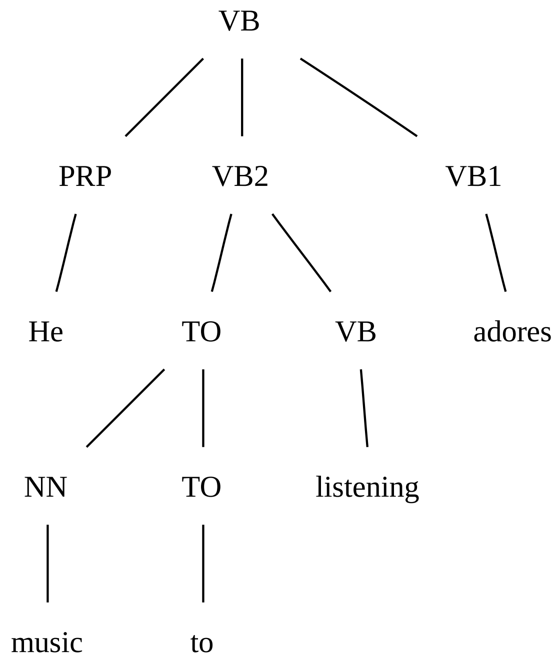


*Source: http://www.isi.edu/natural-language/people/cs562-8-22-06.pdf

⇒ The reordering is decided according to the *r-table*

original order	reordering	P(reorder)
PRP VB1 VB2	PRP VB1 VB2	0.074
	PRP VB2 VB1	0.723
	VB1 PRP VB2	0.061
	...	...
VB TO	VB TO	0.252
	TO VB	0.749
TO NN	TO NN	0.107
	NN TO	0.893
	...	...



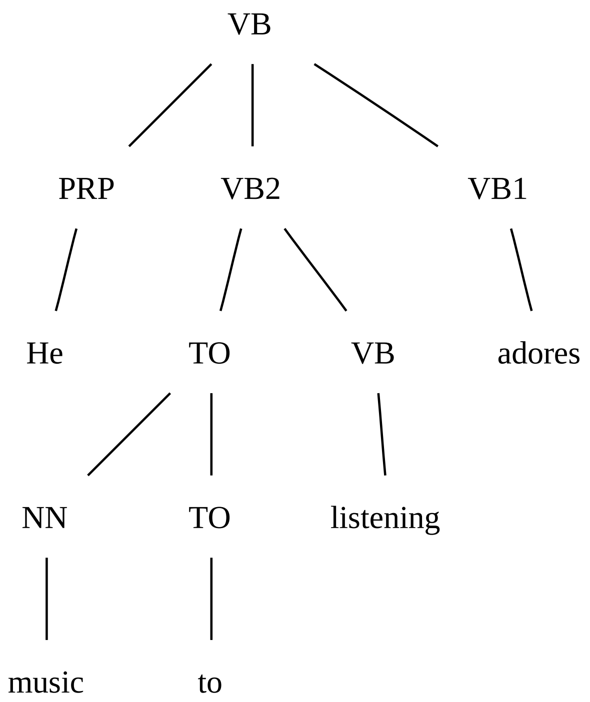


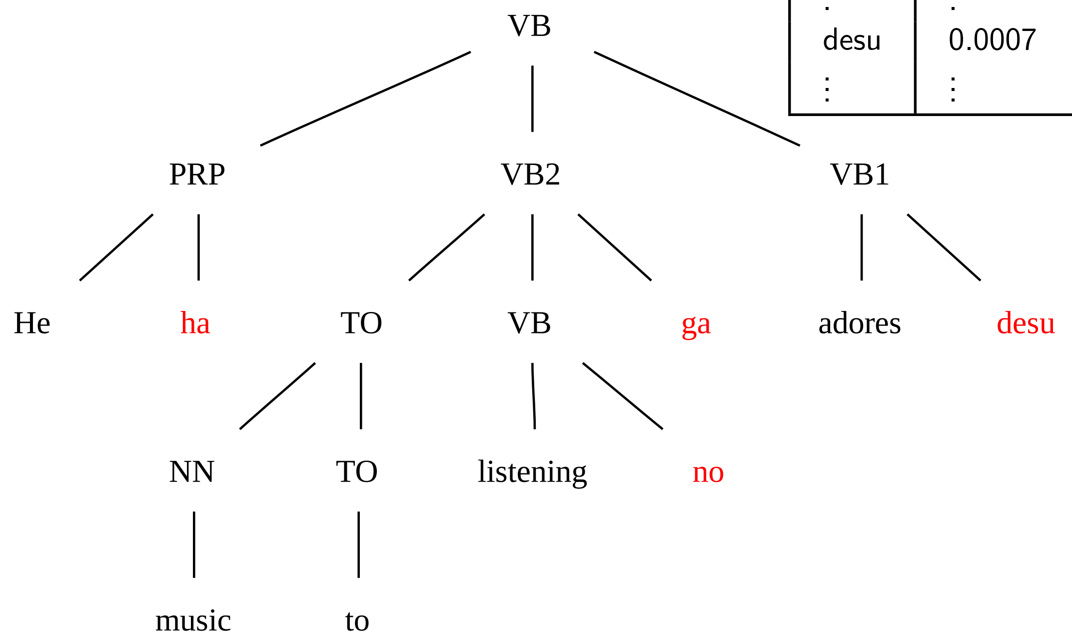
Reordering probability: $0.723 \cdot 0.749 \cdot 0.893 = 0.484$

⇒The insertion of a new node is decided according to the *n-table*

w	P(ins-w)
ha	0.219
ta	0.131
wo	0.099
no	0.094
ni	0.080
te	0.078
ga	0.062
⋮	⋮
desu	0.0007
⋮	⋮

parent	TOP	VB	VB	VB	TO	TO	...
node	VB	VB	PRP	TO	TO	NN	...
P(None)	0.735	0.687	0.344	0.709	0.900	0.800	...
P(Left)	0.004	0.061	0.004	0.030	0.003	0.096	...
P(right)	0.260	0.252	0.652	0.261	0.007	0.104	...



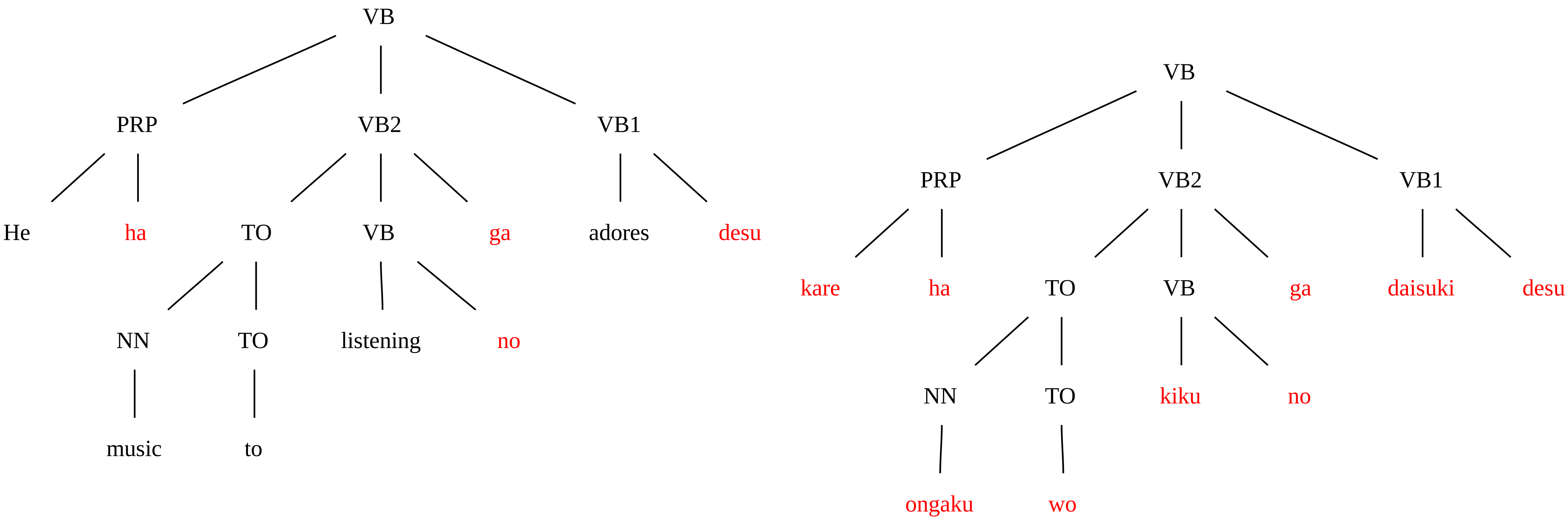


Insertion probability: $(0.652 \cdot 0.219) \cdot (0.252 \cdot 0.094) \cdot (0.252 \cdot 0.062) \cdot (0.252 \cdot 0.0007) \cdot$
$0.735 \cdot 0.709 \cdot 0.900 \cdot 0.800 = 3.498e-9$

5.3 Tree-to-string models

⇒The translation is decided according to the *t-table*

adores		he		listening		music		to		···
daisuki	1.000	kare	0.952	kiku	0.333	ongaku	0.900	ni	0.216	···
		NULL	0.016	kii	0.333	naru	0.100	NULL	0.204	
		nani	0.005	mi	0.333			to	0.133	
		⋮	⋮	⋮	⋮			⋮	⋮	



Translation probability: $0.952 \cdot 0.900 \cdot 0.038 \cdot 1.000 = 0.0108$

Decoder description

- Given a French sentence, the decoder will find the most plausible English parse tree
- Idea: a mechanism similar to normal parsing is used
- Steps:

  1. Start from an English context-free grammar and incorporate to it the channel operations
  2. For each non-lexical rule (such as "VP $\rightarrow$ VB NP PP"), supplement the grammar with reordered rules and probabilities are taken from the r-table
  3. Rules such as "VP $\rightarrow$ VP X" and "X $\rightarrow$ *word*" are added and probabilities are taken from the n-table
  4. For each lexical rule in the English grammar, we add rules such as "englishWord $\rightarrow$ foreingWord"
  5. Parse a string of foreign words
  6. Undo reordering operations and remove leaf nodes with foreign words
  7. Among all possible tree, choose pick the best in which the product of the LM and the TM probability is the highest

Main ideas [Chiang 07]

- It allows to capture difficult reordering
- Hierarchical phrases: phrases that can contain other phrases
- Related to Synchronous CFG: useful for specifying relations between languages.
- Rules are as follows:

$$X \rightarrow \langle \gamma, \alpha, \sim \rangle$$

where

- X is a non-terminal symbol
- γ, α are strings of terminal and non-terminal symbols
- $\sim$ is one-to-one correspondence between non-terminal ocurrences in γ and α

Rule extraction

➢ Rules are extracted from word-alignments sentences

- Extract a rule for each phrase pair
- Replace pharse pairs in each rule by a non-terminal symbol if another rule produces that phrase pair.

➢ The set of rules of two word-aligned sentences $\langle f, e, \sim \rangle$ is the smallest set satisfying the following:

- If $\langle f_i^j, e_{i'}^{j'} \rangle$ is an initial phrase pair, then add the following rule:

$$X \rightarrow \langle f_i^j, e_{i'}^{j'} \rangle$$

- If $(X \rightarrow \langle \gamma, \alpha \rangle)$ is a rule and $\langle f_i^j, e_{i'}^{j'} \rangle$ is an initial phrase pair such that $\gamma = \gamma_1 f_i^j \gamma_2$ and $\alpha = \alpha_1 e_{i'}^{j'} \alpha_2$, then add the following rule:

$$X \rightarrow \langle \gamma_1 X_k \gamma_2, \alpha_1 X_k \alpha_2 \rangle$$

➢ Glue rules:

$$S \rightarrow \langle S_1 X_2, S_1 X_2 \rangle$$
$$S \rightarrow \langle X_1, X_1 \rangle$$

Translation model

- Log-linear model over derivations:

$$P(D) \propto \prod_i \Phi_i(D)^{\lambda_i}$$

where Φ_i are features defined on derivations and λ_i are feature weights

- Features: functions on the rules and and an additonal LM funtion:

$$P(D) \propto P_{LM}(e)^{\lambda_{LM}} \prod_{i \neq LM} \prod_{(X \rightarrow \langle \gamma, \alpha \rangle) \in D} \Phi_i(X \rightarrow \langle \gamma, \alpha \rangle)^{\lambda_i}$$

- Features on rules:
  - $P(\gamma \mid \alpha)$ and $P(\alpha \mid \gamma)$
  - Lexical weights: $P_w(\gamma \mid \alpha)$ and $P_w(\alpha \mid \gamma)$
  - A penalty $\exp(-1)$ to learn a preference for longer or shorter derivations
  - Word penalty: $\exp(-\#T(\alpha))$

Training

- Rules probabilities obtained from frequencies
- λ_i: minimum-error-rate training [Och 02]
- CKY-based algorithm

References

References

[Barrachina 99] S. Barrachina and J.M Vilar. *Bilingual clustering using monolingual algorithms*. TMI. 1999.

[Brown 90] P. F. Brown et al. *A statistical approach to machine translation*. Computational Linguistics, 16, 79–85, 1990.

[Brown 93] P. F. Brown et al. *The mathematics of statistical machine translation: parameter estimation.* Computational Linguistics, 19(2), 263–310, 1993.

[Chiang 07] D. Chiang *Hierarchical phrase-based translation.* Computational Linguistics, 33(2), 201–228, 2007.

[Gascó 10a] G. Gascó and J.A. Sánchez. *Syntax augmented inversion transduction grammars for machine translation*. 11th International Conference on Intelligent Text Processing and Computational Linguistics (CICLING), March, 2010.

[Gascó 10b] G. Gascó, J.A. Sánchez and J.M. Benedí. *Enlarged Search Space for SITG Parsing*, Proc. 11th Annual Conference of the North American Chapter of the Association for Computational Linguistics (NAACL HLT), June, 2010, 653-656.

[Knight 99] K. Knight. *Decoding Complexity in Word-Replacement Translation Models*, Computational Linguistics, Squibs & Discussion, 25(4), 1999.

[Kumar 04] S. Kumar and W. Byrne. *A Weighted Finite State Transducer Implementation of the Alignment Template Model for Statistical Machine Translation*. Proceedings of HLT-NAACL 2003, May 2003.

References

[Ney 00a] H. Ney, S. Nießen, F. Och, H. Sawaf, C. Tillmann and S. Vogel. *Algorithms for Statistical Translation of Spoken Language*. IEEE Transactions on Speech and Audio Processing, vol. 8(1), 24–36, 2000.

[Ney 03a] H. Ney *Statistical Natural Language Processing*, 2003, Canadian Hansard.

[Och 99] F.J. Och. *An Efficient Method for Determining Bilingual Word Classes*. EACL. 1999.

[Och 02] F.J. Och. *Discriminative training and maximum entropy models for statistical machine translation*. Proc. of ACL, 295–302, 2002.

[Tillmann 01] C. Tillmann. *Word re-ordering and DP based search for SMT*. PhD Thesis, 2001.

[Wu 97] D. Wu. *Stochastic Inversion Transduction Grammars and Bilingual Parsing of Parallel Corpora*. Computational Linguisctics, 23(3):377-403, 1997.

[Yamada 01] K. Yamada and K. Knight. *A Syntax-Based Statistical Translation Model*. ACL, 2001.