

INF5820/INF9820

LANGUAGE TECHNOLOGICAL APPLICATIONS

Jan Tore Lønning, Lecture 4, 14 Sep. 2016

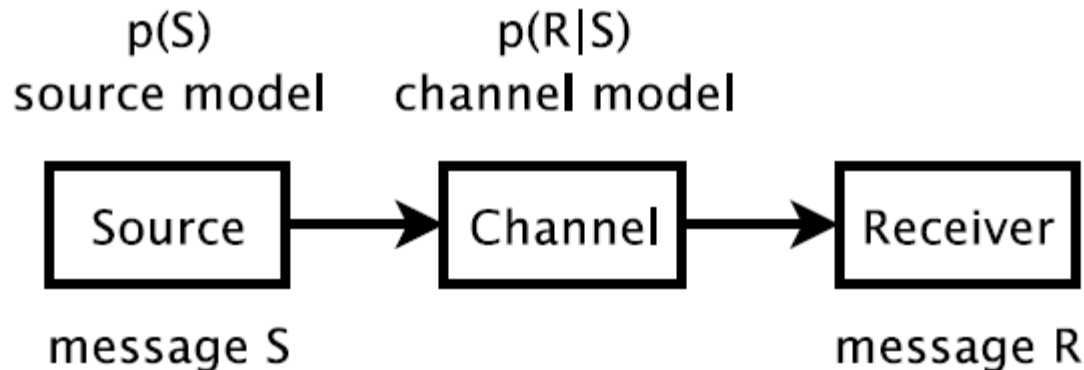
jtl@ifi.uio.no

Today

2

- Statistical machine translation:
 - ▣ The noisy channel model
 - Word-based
 - IBM model 1
- Training

Noisy Channel Model



- Applying Bayes rule also called noisy channel model
 - we observe a distorted message R (here: a foreign string **f**)
 - we have a model on how the message is distorted (here: translation model)
 - we have a model on what messages are probably (here: language model)
 - we want to recover the original message S (here: an English string **e**)

SMT example

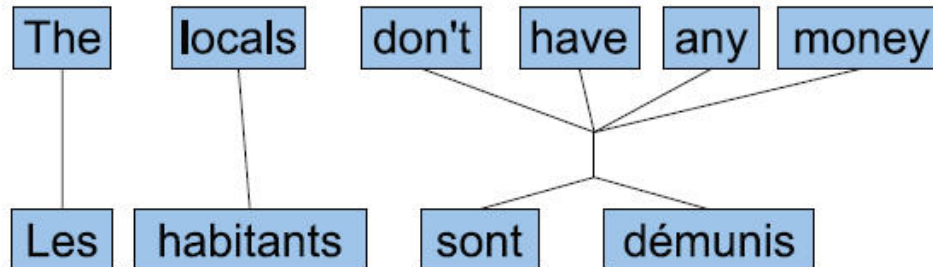
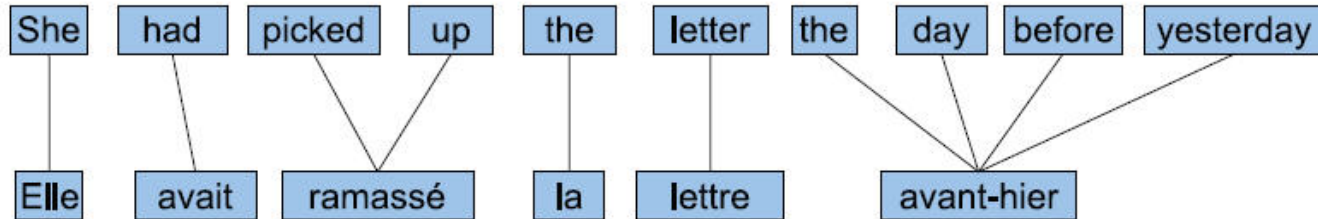
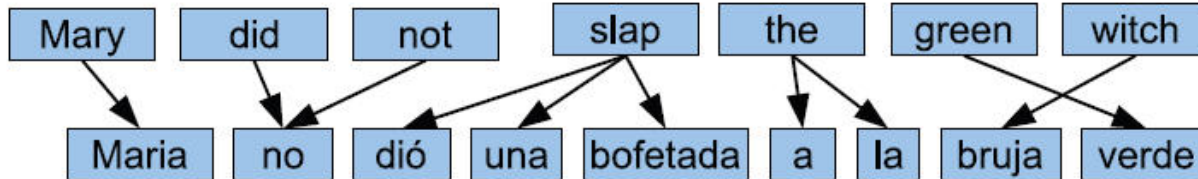
4

En	kokk	lagde	en	rett	med	bygg	.
a 0.9	chef 0.6	made 0.3	a 0.9	right 0.19	with 0.4	building 0.45	
...	cook 0.3	created 0.25	...	straight 0.17	by 0.3	construction 0.33	
	...	prepared 0.15		court 0.12	of 0.2	barley 0.11	
		constructed 0.12		dish 0.11	
		cooked 0.05		course 0.07			
				

Similarly for:

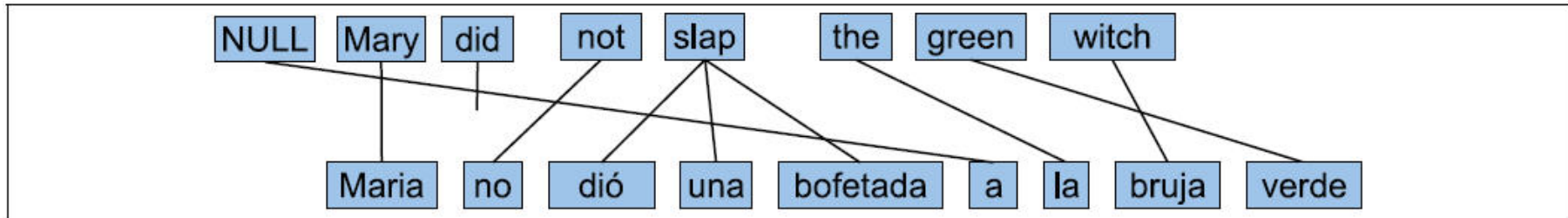
- pos 0-2 (2x3)
- pos 1-3
- pos 2-4
- pos 3-5 (4x5)
- pos 6-8

Pos4 – pos 6 (1x3x3 many)		Pos5 – pos 7 (5x3x3 many)	
a right with	2.7×10^{-12}	right with building	1.7×10^{-18}
a right of	1.5×10^{-10}	right with construction	5.4×10^{-18}
a right by	9.7×10^{-12}	right with barley	8.7×10^{-19}
...		...	
a course of	1.5×10^{-14}	course of barley	1.5×10^{-16}



Alignment

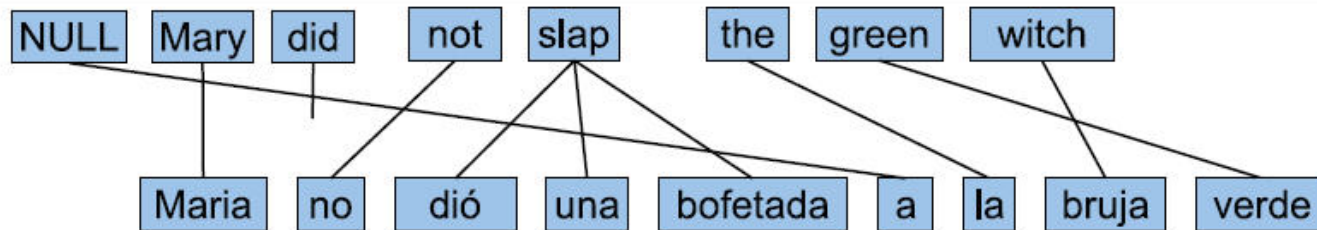
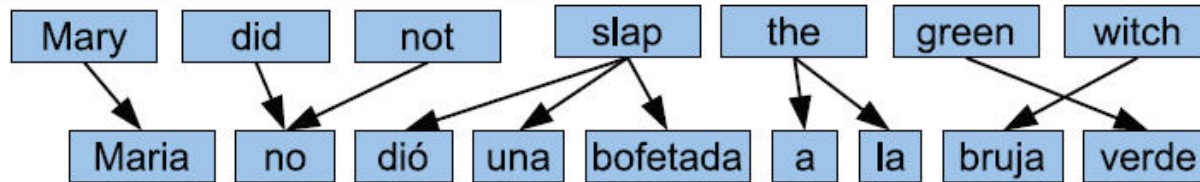
6



- Length of English string: k ($=7$)
- Length of foreign string: m ($=9$)
- An alignment is a vector of length m , each entry a number between 0 and k
- The example:
 - ▣ $\langle a_1, a_2, \dots, a_9 \rangle = \langle 1, 3, 4, 4, 4, 0, 5, 7, 6 \rangle$

Alignment

7



- Artificial restrictions:
 - ▣ Several foreign words may be aligned with the same E word
 - ▣ A foreign word cannot be aligned to more than one E word

IBM Model 1

8

- Consider all possible alignments \mathbf{a} :

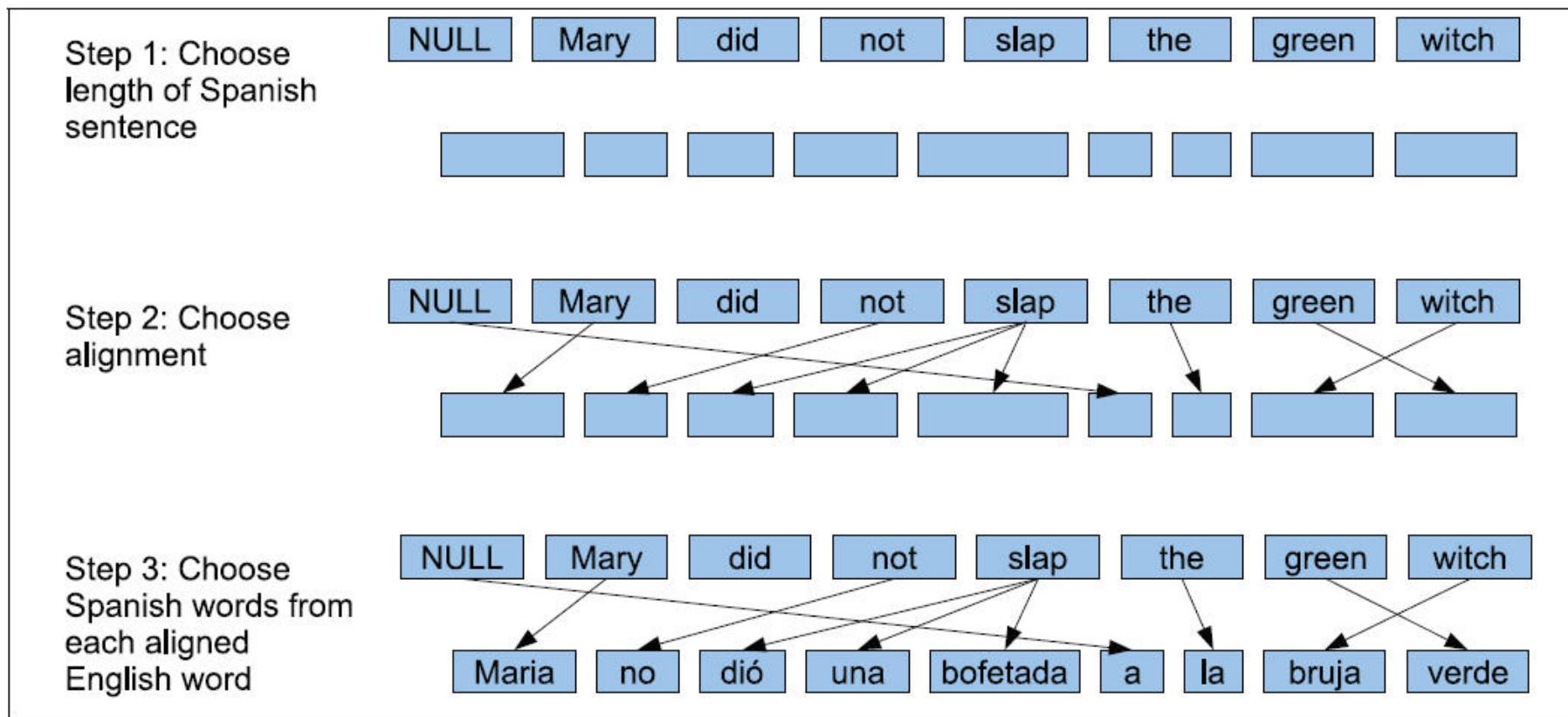
$$P(\mathbf{f} | \mathbf{e}) = \sum_{\mathbf{a}} P(\mathbf{f}, \mathbf{a} | \mathbf{e})$$

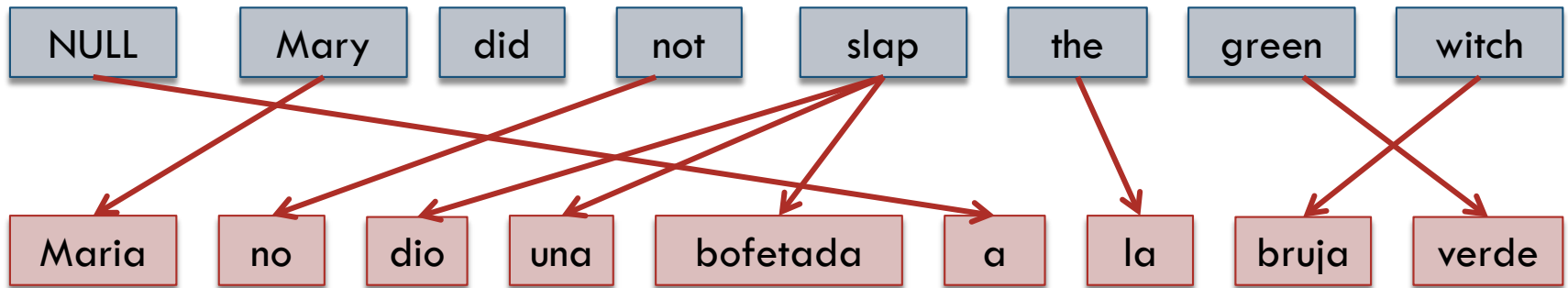
- For each alignment use the generative model:

$$P(\mathbf{f}, \mathbf{a} | \mathbf{e}) = P(m | \mathbf{e}) \prod_{j=1}^m P(a_j | a_1^{j-1}, f_1^{j-1}, m, \mathbf{e}) P(f_j | a_1^j, f_1^{j-1}, m, \mathbf{e})$$

- Simplify the model – make assumptions

Figure 25.23





- Choose the length of the foreign string

$$P(m | \mathbf{e})$$

- Which E word translates to the first F word?

$$P(a_1 | m, \mathbf{e})$$

- What is the translation of this word?

$$P(f_1 | a_1, m, \mathbf{e})$$

- Which E word translates to the j -th F word given the choices so far?

$$P(a_j | a_1^{j-1}, f_1^{j-1}, m, \mathbf{e})$$

- What is the translation of this word given the choices so far?

$$P(f_j | a_1^j, f_1^{j-1}, m, \mathbf{e})$$

Assumptions, approximations

11

$$P(\mathbf{f}, \mathbf{a} | \mathbf{e}) = P(m | \mathbf{e}) \prod_{j=1}^m P(a_j | a_1^{j-1}, f_1^{j-1}, m, \mathbf{e}) P(f_j | a_1^j, f_1^{j-1}, m, \mathbf{e})$$

- $P(m | \mathbf{e})$ is a constant, independent of m and E
- $P(a_j | a_1^{j-1}, f_1^{j-1}, m, \mathbf{e}) = (k + 1)^{-1}$
 - ▣ all alignments the same probability (adds to 1)
- $P(f_j | a_1^j, f_1^{j-1}, m, \mathbf{e}) = t(f_j | e_{a_j})$
 - ▣ the word translation probability only depends on source word

IBM model 1

12

$$P(\mathbf{f}, \mathbf{a} | \mathbf{e}) = P(m | \mathbf{e}) \prod_{j=1}^m P(a_j | a_1^{j-1}, f_1^{j-1}, m, \mathbf{e}) P(f_j | a_1^j, f_1^{j-1}, m, \mathbf{e})$$

□ Simplifies to

$$P(\mathbf{f}, \mathbf{a} | \mathbf{e}) = \varepsilon \prod_{j=1}^m (k+1)^{-1} t(f_j | e_{a_j})$$

$$P(\mathbf{f}, \mathbf{a} | \mathbf{e}) = \frac{\varepsilon}{(k+1)^m} \prod_{j=1}^m t(f_j | e_{a_j})$$

- ε is a normalisation factor
- Formula 4.7 in the SMT book
 - (The book goes $f \rightarrow e$, not $e \rightarrow f$)

Parameter estimation

13

- If the training corpus was aligned, the model could be learned by counting:

$$t(f_j | e_{a_j}) = \frac{C(f_j, e_{a_j})}{\sum_f C(f, e_{a_j})}$$

- If we had known the translation probabilities, we could have found the most probable alignment.
- We neither know word probabilities nor alignment: Chicken and egg problem
- EM-algorithm: we may learn the two simultaneously

Training – the idea

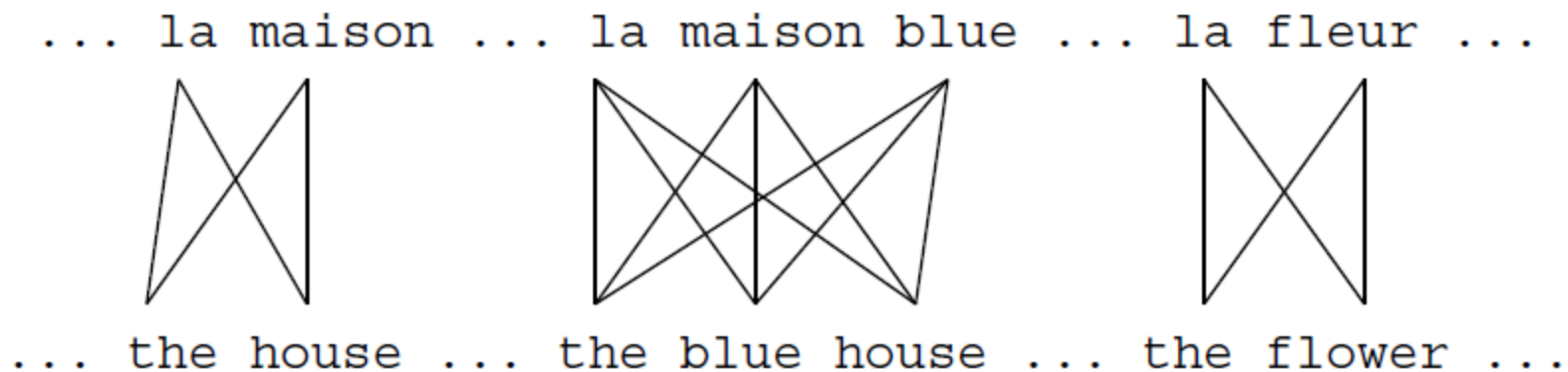
14

1. From the translation probabilities, we may estimate probabilities for the various alignments
 - ▣ (We do not choose only the best alignment)
2. From alignment probabilities, we may recalculate translation probabilities
 - ▣ By alternating between (1) and (2), the numbers converge towards better results
 - ▣ For IBM Model 1 it may be proved that they converge towards a global optimum

EM Algorithm

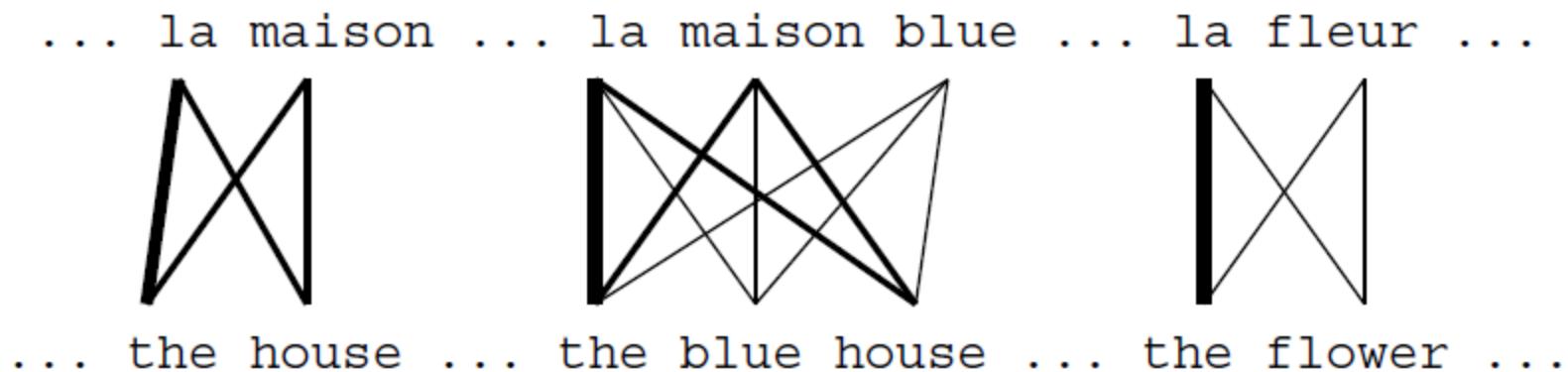
- Incomplete data
 - if we had *complete data*, would could estimate *model*
 - if we had *model*, we could fill in the *gaps in the data*
- Expectation Maximization (EM) in a nutshell
 1. initialize model parameters (e.g. uniform)
 2. assign probabilities to the missing data
 3. estimate model parameters from completed data
 4. iterate steps 2–3 until convergence

EM Algorithm



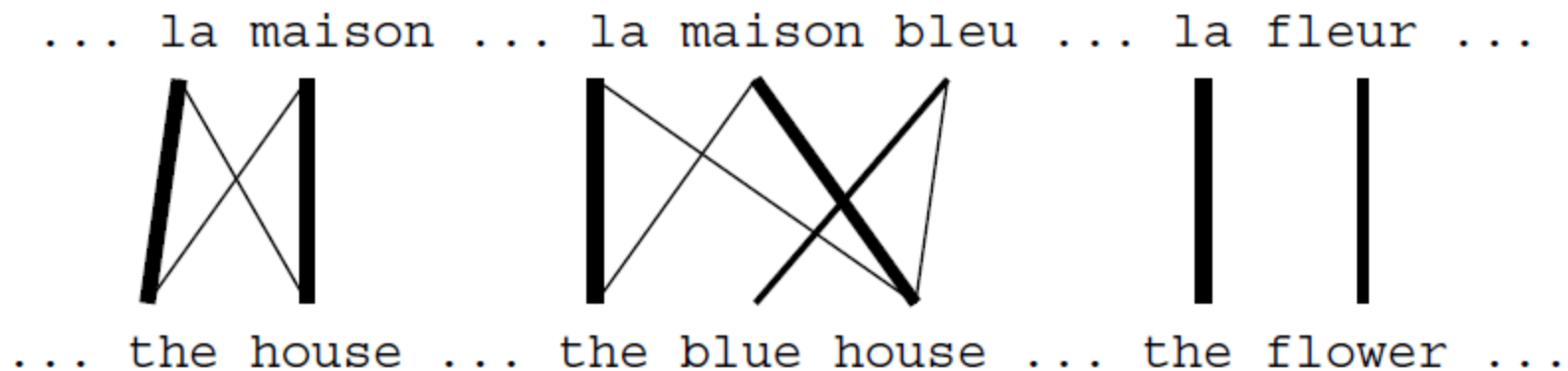
- Initial step: all alignments equally likely
- Model learns that, e.g., **la** is often aligned with **the**

EM Algorithm



- After one iteration
- Alignments, e.g., between **la** and **the** are more likely

EM Algorithm



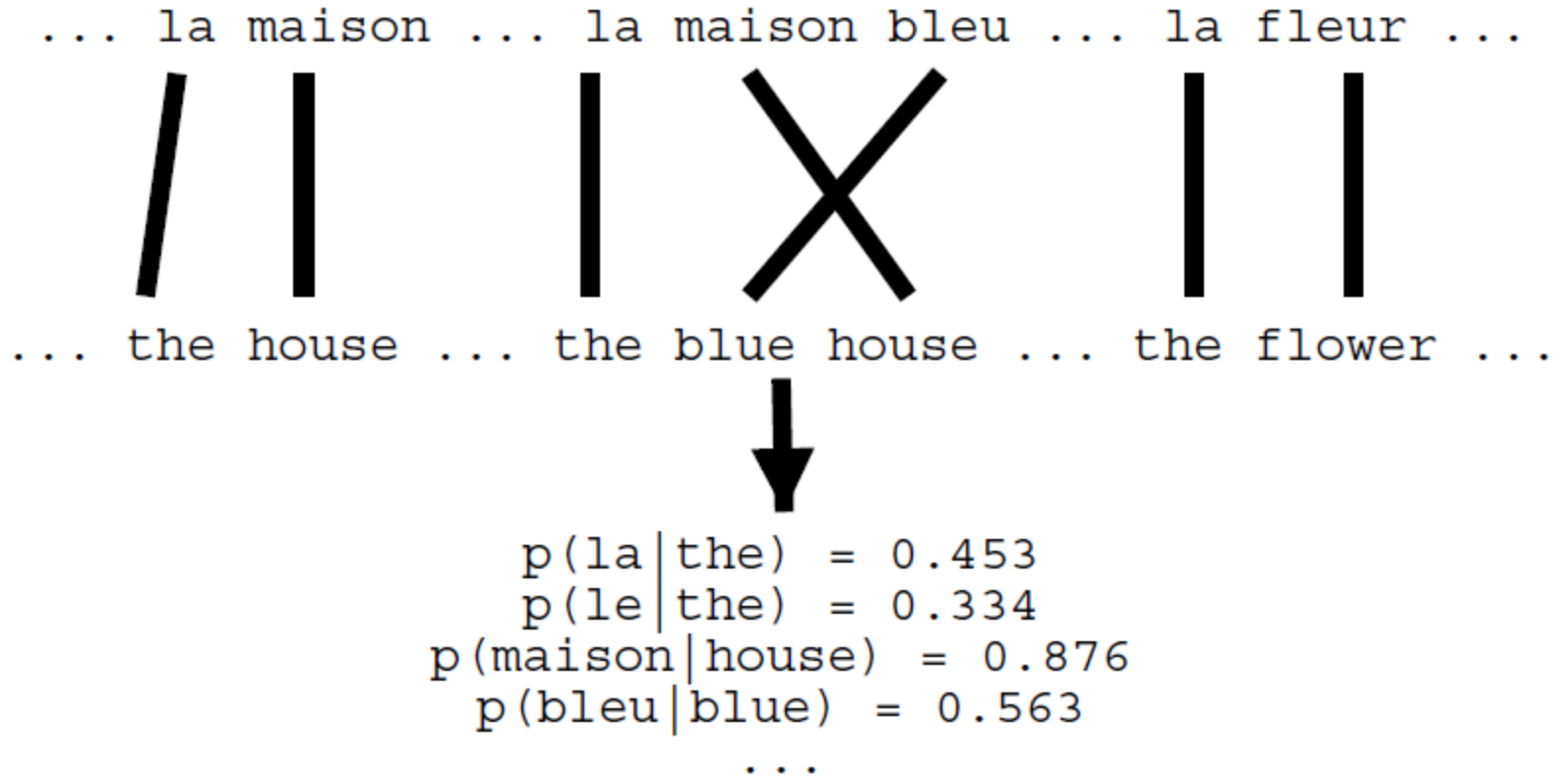
- After another iteration
- It becomes apparent that alignments, e.g., between fleur and flower are more likely (pigeon hole principle)

EM Algorithm

... la maison ... la maison bleu ... la fleur ...
/ | | X | |
... the house ... the blue house ... the flower ...

- Convergence
- Inherent hidden structure revealed by EM

EM Algorithm



- Parameter estimation from the aligned corpus

Two ways to describe the algorithm

21

Intuitive

- Proceed
 - ▣ 1. Translation prob
 - ▣ 1. Alignment prob
 - ▣ 2. Translation prob
 - ▣ 2. Alignment prob
 - ▣ 3. Translation prob
 - ▣ Etc
- J&M, sec 25.6.1, example
- Intractable in practice

Efficient

- Sidestep alignment probs:
 - ▣ 1. Translation prob
 - ▣ 2. Translation prob
 - ▣ 3. Translation prob
 - ▣ Etc
- K:SMT, sec 4.2.3, example
- This is how it gets implemented

Training – the intuitive approach

22

1. Initialize the parameter values $t(f/e)$ for pairs of words f and e .
 - ▣ With no info, initialize them uniformly:
Each word f in the foreign language is an equally likely translation of the word e .
2. For each pair f, e of sentences in the corpus, use t to calculate the probabilities $P(a/f, e)$ to all possible alignments a of the two sentences.
 - ▣ (Called the **expectation** step, apply model to data)

Training – the intuitive approach

23

3. Collect fractional counts, $tc(f|e)$:

(«How many times e is translated as f »)

1. First, calculate this, $c(f|e; \mathbf{f}, \mathbf{e})$ for each sentence \mathbf{f}, \mathbf{e} , where we count:

- how many times e is aligned to f by each alignment,
- weighed by the probability of the alignment.

2. Then add over all sentences to get

$$tc(f|e) = \sum_{(\mathbf{f}, \mathbf{e})} c(f|e; \mathbf{f}, \mathbf{e})$$

Training – the intuitive approach

24

4. Calculate the new translation probabilities

$$t(f|e) = \frac{tc(f|e)}{\sum_{f'} tc(f'|e)}$$

Errors in formula
4.14 in K:SMT

- ▣ where f' varies over all foreign words
 - ▣ (Called the **maximization** step, estimate model from counts)
5. Repeat from 2 as long as you like

Assign probabilities to alignments

25

□ Goal: compute $P(\mathbf{a} | \mathbf{f}, \mathbf{e})$

□ Since $P(\mathbf{f}, \mathbf{a} | \mathbf{e}) = P(\mathbf{a} | \mathbf{f}, \mathbf{e})P(\mathbf{f} | \mathbf{e})$

▣ we have

$$P(\mathbf{a} | \mathbf{f}, \mathbf{e}) = \frac{P(\mathbf{f}, \mathbf{a} | \mathbf{e})}{P(\mathbf{f} | \mathbf{e})}$$

□ We know

$$P(\mathbf{f}, \mathbf{a} | \mathbf{e}) = \frac{\varepsilon}{(k+1)^m} \prod_{j=1}^m t(f_j | e_{a_j})$$

□ Hence:

$$P(\mathbf{f} | \mathbf{e}) = \sum_{\mathbf{a}} P(\mathbf{f}, \mathbf{a} | \mathbf{e})$$

$$P(\mathbf{a} | \mathbf{f}, \mathbf{e}) = \frac{\frac{\varepsilon}{(k+1)^m} \prod_{j=1}^m t(f_j | e_{a_j})}{\frac{\varepsilon}{(k+1)^m} \sum_{\mathbf{a}'} \prod_{j=1}^m t(f_j | e_{a_j})} = \frac{\prod_{j=1}^m t(f_j | e_{a_j})}{\sum_{\mathbf{a}'} \prod_{j=1}^m t(f_j | e_{a_j})}$$

Example – the intuitive way

26

□ Corpus

e_1 : Dog barked
 f_1 : Hund bjeffet

e_2 : Dog bit dog
 f_2 : Hund bet hund

3 English words: dog bit barked
3 foreign words: hund bjeffet bet

Step 1 initialization

27

$t(\text{hund} \text{dog}) = 1/3$	$t(\text{bet} \text{dog}) = 1/3$	$t(\text{bjeffet} \text{dog}) = 1/3$
$t(\text{hund} \text{bit}) = 1/3$	$t(\text{bet} \text{bit}) = 1/3$	$t(\text{bjeffet} \text{bit}) = 1/3$
$t(\text{hund} \text{barked}) = 1/3$	$t(\text{bet} \text{barked}) = 1/3$	$t(\text{bjeffet} \text{barked}) = 1/3$
$t(\text{hund} 0) = 1/3$	$t(\text{bet} 0) = 1/3$	$t(\text{bjeffet} 0) = 1/3$

- Uniform
- Observe that we include the last line since an f-word may be aligned to 0.

Step 2: Alignment probabilities

28

e_1 : Dog barked
 f_1 : Hund bjeffet

e_2 : Dog bit dog
 f_2 : Hund bet hund

- Sentence pair 1:
 - ▣ 9 possible alignments:
 - $\langle 0,0 \rangle, \langle 0,1 \rangle, \langle 0,2 \rangle, \langle 1,0 \rangle, \langle 1,1 \rangle, \langle 1,2 \rangle, \langle 2,0 \rangle, \langle 2,1 \rangle, \langle 2,2 \rangle$
 - ▣ Each equally probable: $1/9$
 - ▣ (call this a_1 : e.g. $a_1(\langle 0,1 \rangle) = 1/9$)
- Sentence pair 2:
 - ▣ 64 possible alignments:
 - $\langle 0,0,0 \rangle, \langle 0,0,1 \rangle, \dots \langle 3,3,3 \rangle$
 - Each equally probable: $1/64$
 - (call this a_2 .)
 - Or, the hard way (next slide)

Step 2: The hard way

29

e_2 : Dog bit dog
 f_2 : Hund bet hund

□ Sentence pair 2:

▣ 64 possible alignments:

■ $\langle 0,0,0 \rangle, \langle 0,0,1 \rangle, \dots \langle 3,3,3 \rangle$

▣ Each translation probability: $1/27$

$$P(\mathbf{f}_2, \langle 1,2,0 \rangle | \mathbf{e}_2) = \frac{\varepsilon}{(k+1)^m} \prod_{j=1}^m t(f_j | e_{a_j}) = \frac{\varepsilon}{(3+1)^3} \prod_{j=1}^3 t(f_j | e_{a_j}) = \frac{\varepsilon}{4^3} t(f_1 | e_1) \times t(f_2 | e_2) \times t(f_3 | e_0) =$$
$$\frac{\varepsilon}{4^3} t(\text{hund} | \text{dog}) \times t(\text{bet} | \text{bit}) \times t(\text{hund} | 0) = \frac{\varepsilon}{4^3} \frac{1}{3} \times \frac{1}{3} \times \frac{1}{3} = \frac{\varepsilon}{4^3 \times 3^3}$$

$$P(\langle 1,2,0 \rangle | \mathbf{f}_2, \mathbf{e}_2) = \frac{P(\mathbf{f}_2, \langle 1,2,0 \rangle | \mathbf{e}_2)}{P(\mathbf{f}_2 | \mathbf{e}_2)} = \frac{P(\mathbf{f}_2, \langle 1,2,0 \rangle | \mathbf{e}_2)}{\sum_{\mathbf{a}} P(\mathbf{a}, \mathbf{f}_2 | \mathbf{e}_2)} = \frac{\frac{\varepsilon}{64 * 27}}{64 * \frac{\varepsilon}{64 * 27}} = \frac{1}{64}$$

Step 3.1: Collect fractional counts

30

Calculate $c(f/e ; f, e)$ for each sentence f, e :

□ Example: $f = \text{hund}, e = \text{dog}, f_1, e_1$:

▣ There are 3 alignments that connect them:

$\langle 1,0 \rangle, \langle 1,1 \rangle, \langle 1,2 \rangle$

▣ $c(\text{hund} | \text{dog}; f_1, e_1) =$

$$\alpha_1(\langle 1,0 \rangle) + \alpha_1(\langle 1,1 \rangle) + \alpha_1(\langle 1,2 \rangle) = 3 * (1/9) = 1/3$$

e_1 : Dog barked
 f_1 : Hund bjeffet

$c(\text{hund} \text{dog}; f_1, e_1) = 1/3$	$c(\text{bjeffet} \text{dog}; f_1, e_1) = 1/3$
$c(\text{hund} \text{barked}; f_1, e_1) = 1/3$	$c(\text{bjeffet} \text{barked}; f_1, e_1) = 1/3$
$c(\text{hund} 0; f_1, e_1) = 1/3$	$c(\text{bjeffet} 0; f_1, e_1) = 1/3$

Step 3.1: Collect frac. counts ctd

31

f_2, e_2 :

e_2 : Dog bit dog
 f_2 : Hund bet hund

□ $f = \text{bet}, e = \text{bit}$

▣ 16 alignments connect them: $\langle x, 2, z \rangle$ for x, z in $\{0, 1, 2, 3\}$

▣ $c(\text{bet} | \text{bit}; f_2, e_2) = 16/64 = 1/4$

□ $f = \text{bet}, e = \text{dog}$

▣ all alignments $\langle x, 1, z \rangle$ and $\langle x, 3, z \rangle$ for x, z in $\{0, 1, 2, 3\}$

▣ $c(\text{bet} | \text{dog}; f_2, e_2) = 2 \cdot 16/64 = 1/2$

$c(\text{hund} \text{dog}; f_2, e_2) = 1$	$c(\text{bet} \text{dog}; f_2, e_2) = 1/2$
$c(\text{hund} \text{bit}; f_2, e_2) = 1/2$	$c(\text{bet} \text{bit}; f_2, e_2) = 1/4$
$c(\text{hund} 0; f_2, e_2) = 1/2$	$c(\text{bet} 0; f_2, e_2) = 1/4$

Step 3.2: Total counts

32

$$tc(f | e) = \sum_{(\mathbf{f}, \mathbf{e})} c(f | e; \mathbf{f}, \mathbf{e})$$

$tc(\text{hund} \text{dog}) = 1 + 1/3$	$tc(\text{bet} \text{dog}) = 1/2$	$tc(\text{bjeffet} \text{dog}) = 1/3$	$tc(* \text{dog}) = 4/3 + 1/2 + 1/3 = 13/6$
$tc(\text{hund} \text{bit}) = 1/2$	$tc(\text{bet} \text{bit}) = 1/4$	$tc(\text{bjeffet} \text{bit}) = 0$	$tc(* \text{bit}) = 3/4$
$tc(\text{hund} \text{barked}) = 1/3$	$tc(\text{bet} \text{barked}) = 0$	$tc(\text{bjeffet} \text{barked}) = 1/3$	$tc(* \text{barked}) = 2/3$
$tc(\text{hund} 0) = 1/2 + 1/3$	$tc(\text{bet} 0) = 1/4$	$tc(\text{bjeffet} 0) = 1/3$	$tc(* 0) = 17/12$

Step 4: new trans. probabilities

33

$$t(f|e) = \frac{tc(f|e)}{\sum_{f'} tc(f'|e)}$$

e	f	t(f e)	exact	decimal
0	hund	(5/6)/(17/12)	10/17	0.588235
0	bet	(1/4)/(17/12)	3/17	0.176471
0	bjeffet	(1/3)/(17/12)	4/17	0.235294
dog	hund	(4/3)/(13/6)	8/13	0.615385
dog	bet	(1/2)/(13/6)	3/13	0.230769
dog	bjeffet	(1/3)/(13/6)	2/13	0.153846
bit	hund	(1/2)/(3/4)	2/3	0.666667
bit	bet	(1/4)/(3/4)	1/3	0.333333
barked	hund	(1/3)/(2/3)	1/2	0.5
barked	bjeffet	(1/3)/(2/3)	1/2	0.5

Repeat: Step 2, sentence 1

34

- 9 different alignments
- $P'(\mathbf{a}) = c P(\mathbf{a}, \mathbf{f}_1 \mid \mathbf{e}_1)$
- $P(\mathbf{a}) = P(\mathbf{a} \mid \mathbf{e}_1, \mathbf{f}_1)$

\mathbf{e}_1 : Dog barked
 \mathbf{f}_1 : Hund bjeffet

			P'	P=P'/1,4145436
$P'(<0,0>) =$	$t(\text{hund} 0)*t(\text{bjeffet} 0)=$	$(10/17)*(3/17)=$	0,103806	0,0733848
$P'(<0,1>)=$	$t(\text{hund} 0)*t(\text{bjeffet} \text{dog})=$	$(10/17)*(2/13)=$	0,0904977	0,0639766
$P'(<0,2>)=$	$t(\text{hund} 0)*t(\text{bjeffet} \text{barked})=$	$(10/17)*(1/2)=$	0,294118	0,207924
$P'(<1,0>) =$	$t(\text{hund} \text{dog})*t(\text{bjeffet} 0)=$	$(8/13)*(3/17)=$	0,108597	0,0767718
$P'(<1,1>) =$	$t(\text{hund} \text{dog})*t(\text{bjeffet} \text{dog})=$	$(8/13)*(2/13)=$	0,0946746	0,0669294
$P'(<1,2>) =$	$t(\text{hund} \text{dog})*t(\text{bjeffet} \text{barked})=$	$(8/13)*(1/2)=$	0,307692	0,217520
$P'(<2,0>) =$	$t(\text{hund} \text{barked})*t(\text{bjeffet} 0)=$	$(1/2)*(3/17)=$	0,0882352	0,06237715
$P'(<2,1>)=$	$t(\text{hund} \text{barked})*t(\text{bjeffet} \text{dog})=$	$(1/2)*(2/13)=$	0,0769231	0,05438015
$P'(<2,2>)=$	$t(\text{hund} \text{barked})*t(\text{bjeffet} \text{barked})=$	$(1/2)*(1/2)=$	0,25	0,176735
Sum of P's			1,4145436	

Repeat: Step 2, sentence 2

35

- 64 different alignments
- Home work to next week! 😊
- How many alignments if the sentences are 10 words long?

Words, $m=k$	2	3	4	5	6	7	8	9	10
Align.	9	64	625	1296	117 649	168 070	43mill	138 413	25 billions

- That's why we need a smarter way.
- To be continued ...