Columbia University

Statistical NLP for the Web

Machine Translation

Sameer Maskey

Week 10, November 7, 2012

Announcements

- Graded project reports will be returned to you this weekend
- HW3 will be released by end of this weekend
- HW3 due date : Nov 30 (Friday : 11:59pm)
  - You have roughly 3 weeks to finish it

Announcement from Last Lecture : Intermediate Report II

- 20% of the Final project grade
- Intermediate Report II will be Oral
- Everything related with your project is fair game, including theory related with algorithms you are using
- You need to have the first version of end to end system ready including UI
- Prototype Demo should be running in Amazon or your web server
  - Please get help from Morgan if you need help on this
- No report required, just show up!

Intermediate Report II Meeting Signup

- Intermediate Report II signup tonight right after class
- Available dates and times
  - Nov 14 – 10 AM to 4pm
  - Nov 16 – 10 AM to 5pm
  - Nov 21– 10 AM to 4pm
- Send email with 3 preferred times – 20 min slots
  - Email to me smaskey@cs
  - And Morgan mulinski@cs
  - Use the header : "Intermediate Report II Meetings"
- If you are a team you need to sign up for
  - 30 min time slot – 2 person team
  - 40 min time slot – 3 person team
- Random assignments on collision

Heads Up : New Course

Offered jointly across two schools! Columbia Business School + CS@CU

"**Data Science and Technology Entrepreneurship**"

Wednesdays 4:10 to 6pm

B8848-01 – Business School Course ID

CS6998 - Computer Science Course ID

MBA student + CS student pairs/teams

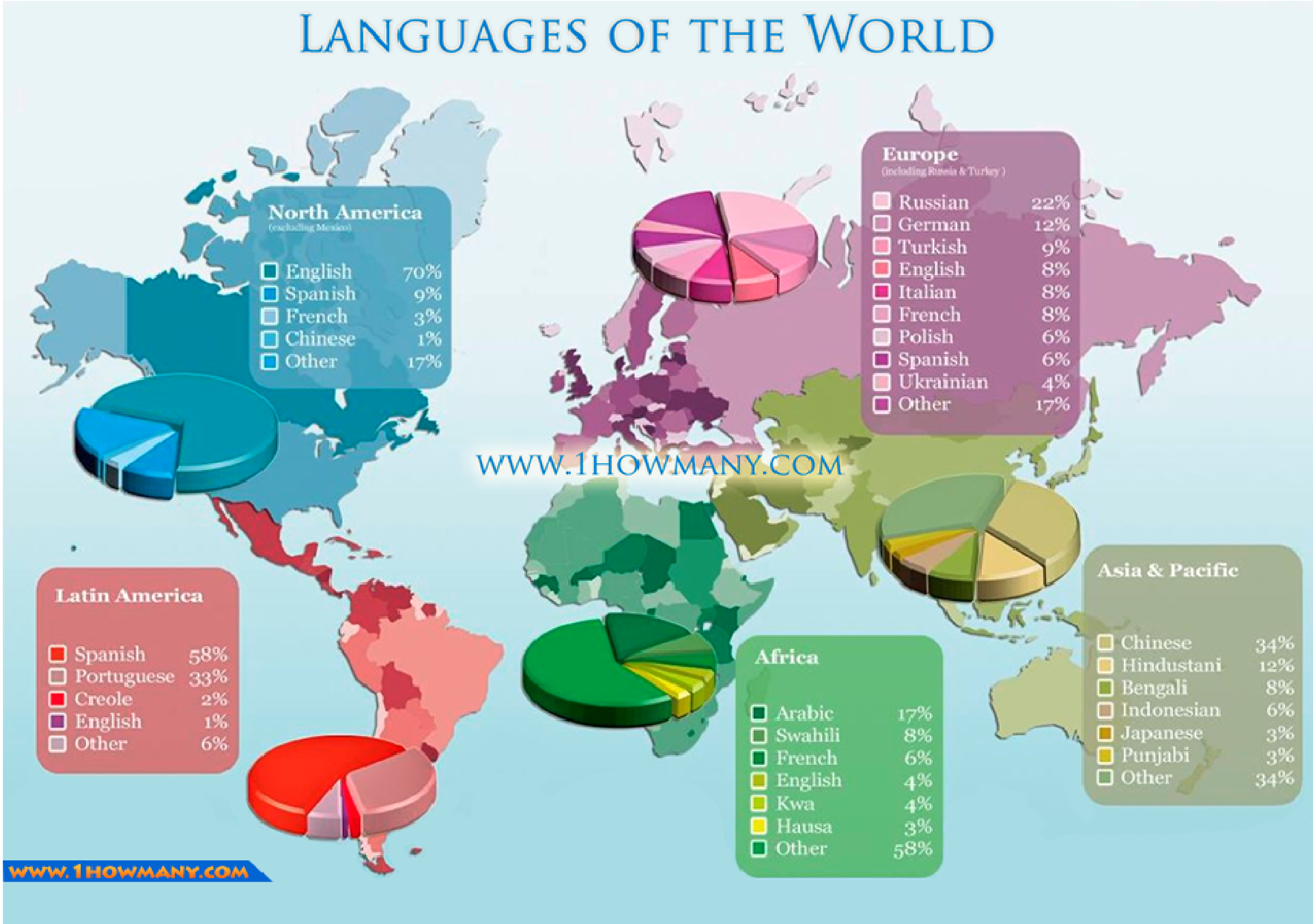
Topic for Today

- Machine Translation

Machine Translation

Machine Translation

Machine Translation



Approximation : may not be 100% accurate

Machine Translation

- Machine Translation is useful
  - Text to Text translation
    - Difficult
  - Speech to Speech translation
    - Very challenging
- Classic NLP problem
- Still an unsolved problem

Bit of History and Early Hopes

- One of the first computer application
- Warren Weaver (1949): “I have a text in front of me which is written in Russian but I am going to pretend that it is really written in English and that it has been coded in some strange symbols. All I need to do is strip off the code in order to retrieve the information contained in the text.”
- 1952 – MIT Conference on MT
- 1959 : IBM’s Mark I
- 1976 : Systran
- Until 1989 rule based approaches dominated
- 1989 : IBM introduces Statistical MT
- 1999 : JHU Workshop – open source SMT model

Why is it Difficult?

- Natural Language is complex
  - Ambiguities
  - Structure
  - Context dependent
  - Domain dependent
  - Word ordering is difficult
- 2 in 1
  - Natural Language Understanding
  - Natural Language Generation

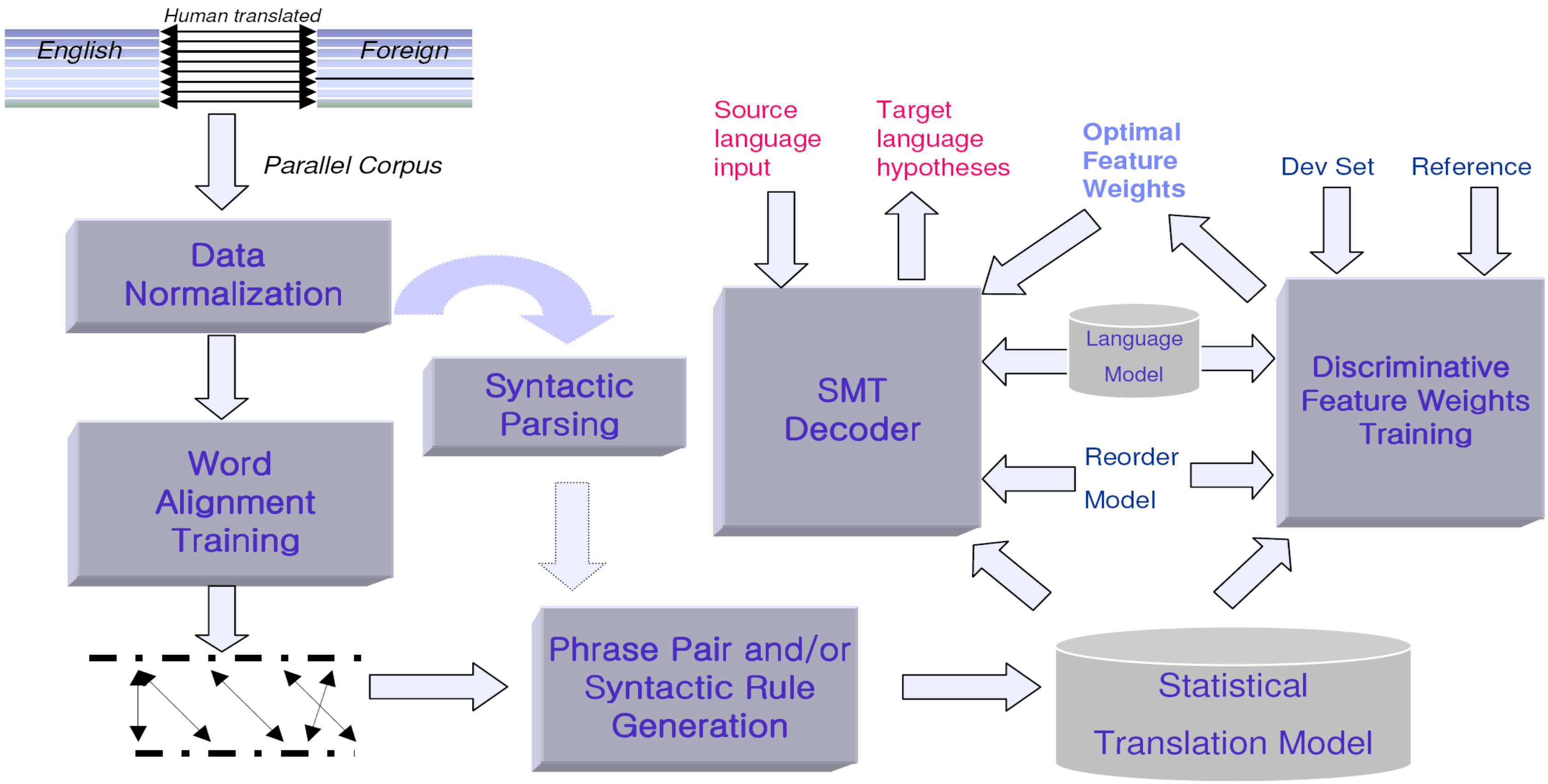
MT Methods

- Rule Based Approaches
  - Manually generated rules
  - Time consuming
  - Expensive
  - Not easily adaptable
- Statistical Data Driven Approach
  - Need parallel corpus
  - Easy adaptation to new languages and domain
  - Difficult to model complex language phenomena
  - Word Based
  - Phrase Based
  - Syntax Based

Rule-Based vs. Statistical MT

- Rule-based MT:
  - Hand-written transfer rules
  - Rules can be based on lexical or structural transfer
  - Pro: firm grip on complex translation phenomena
  - Con: Often very labor-intensive -> lack of robustness
- Statistical MT
  - Mainly word or phrase-based translations
  - Translation are learned from actual data
  - Pro: Translations are learned automatically
  - Con: Difficult to model complex translation phenomena

Corpus Based Statistical MT Architecture



Picture from [2]

Parallel Corpus

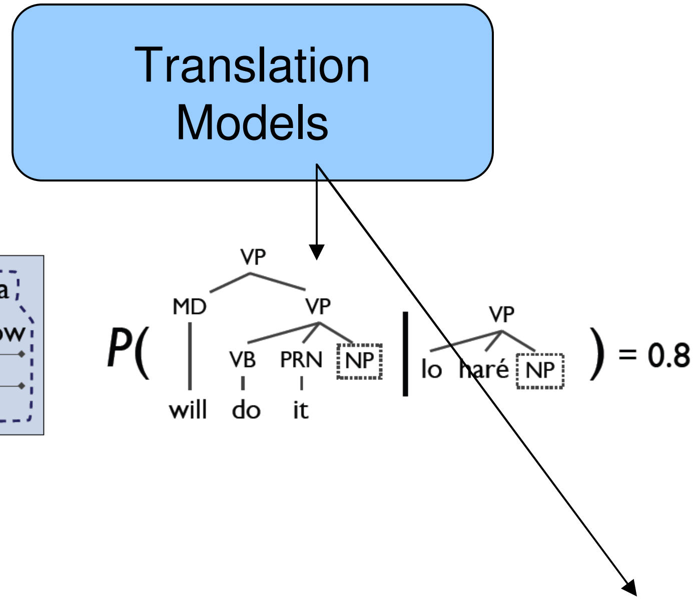
1. i mean what are we doing penny pinching
2. where are you going to get it from
3. clearly a consequence of extending the reference periods would be to increase the flexibility available to companies
4. in the final vote we chose in spite of our hesitations to vote in favour of the report
5. .
6. .
7. .

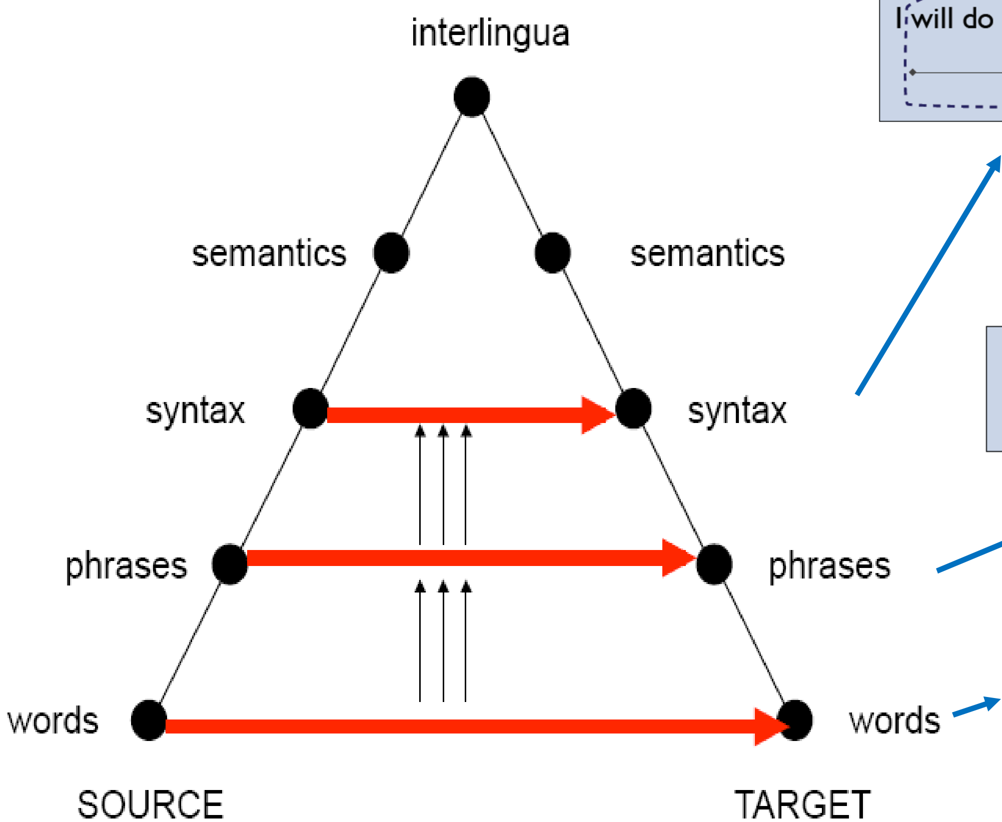
English

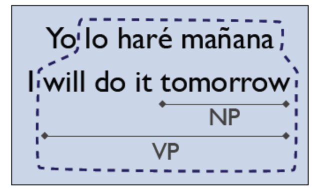
1. ich meine was machen wir denn wollen wir sparen
2. woher wollen sie es nehmen
3. durch die ausweitung der bezugszeitrume wrde den unternehmen deutlich mehr flexibilitt zugestanden
4. in der schlussabstimmung haben wir jedoch trotz unserer zweifel dem bericht zugestimmt
5. .
6. .
7. .

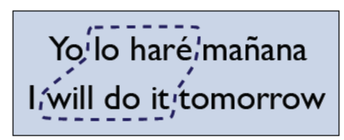
German

Transfer Levels

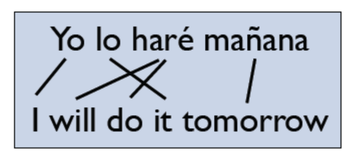








English (E)	*P*(E | lo haré)
will do it	0.8
will do so	0.2



English (E)	*P*(E | mañana)
tomorrow	0.7
morning	0.3

From [1]

Statistical Translation Model

- Given a source sentence you want the best translation in the target language
- We can frame translation as a noisy channel model
- Formally

$$Given E = e_1, e_2, , e_l \text{ and } F = f_1, f_2, f_3, , f_m$$

$$\mathrm{E}$$

$$= \mathrm{argmax}_E P(E|F)$$

$$= argmax_E \frac{P(F|E)P(E)}{P(F)}$$

$$= argmax_E P(F|E)P(E)$$

Translation Models

Language Model

How Do We Get Translation Model?

- We first need to figure out what source word translates to what target word
- Alignment Model
- Introduce a hidden alignment variable
- First proposed by Brown et. al
  - IBM Models

IBM Models 1–5

- Model 1: Bag of words
  - Unique local maxima
  - Efficient EM algorithm (Model 1–2)
- Model 2: General alignment: $a(e_{pos} \mid f_{pos}, e_{length}, f_{length})$
- Model 3: fertility: n(k | e)
- Model 4: Relative distortion, word classes
- Model 5: Extra variables to avoid deficiency

IBM Model 1

$$P(f|e) = \sum_a P(f, a|e)$$

$$P(f, a|e) = \prod_j P(a_j = i|e) P(f_j|e_i)$$

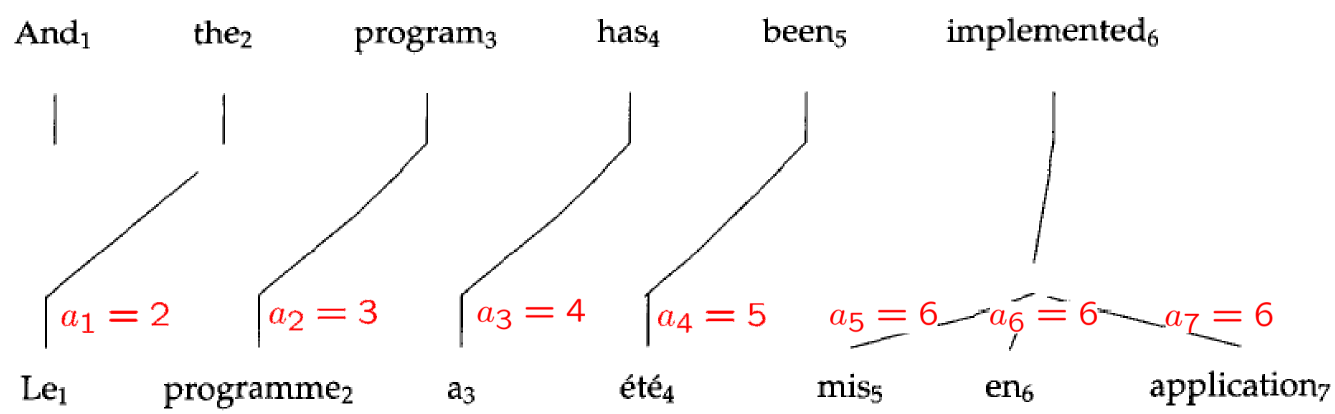
$$P(a_j = i|e, f) = \frac{P(f_j|e_i)}{\sum_{i'} P(f_j|e_{i'})}$$

- Basic idea: pick a source for each word, update co-occurrence statistics, repeat

IBM Model 1 (Brown 93)

- a hidden vector called an *alignment* specifies which English source is responsible for each French target word.

$$a = a_1 \ldots a_J$$



$$P(f, a|e) = \prod_j P(a_j = i) P(f_j|e_i)$$

$$= \prod_j \frac{1}{I+1} P(f_j|e_i)$$

$$P(f|e) = \sum_a P(f, a|e)$$

Word Translation

- Simple Exercise
- How to translate a word “Haus” in German?
  - Dictionary look up:

    Haus: house, building, home, household, shell

Slides 24 to 55 are provided by Dr. Declan Groves [Groves, D.]

- **But there could be Multiple translations**: some more frequent than others
- How do we determine probabilities for possible candidate translations?

Translation of *Haus*	Count
house	8,000
building	1,600
home	200
household	150
shell	50

Estimate Translation Probabilities

Translation of *Haus*	Count
house	8,000
building	1,600
home	200
household	150
shell	50
Total	**10,000**

- Use relative frequencies to estimate probabilities

P(*s*/*t*) = 0.8, if *t* = *house*

0.16, if *t* = *building*

0.02, if *e* = *home*

0.015, if *e* = *household*

0.005, if *e* = *shell*

Alignment

- When identifying lexical translations, given a sentence-aligned parallel text, we align words in one sentence with words in the other

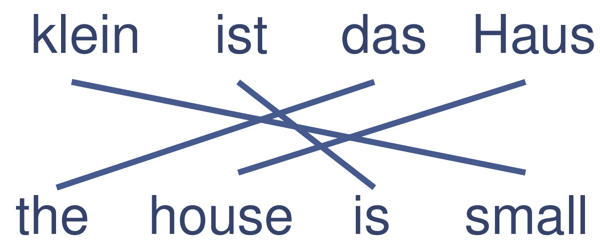
- Alignment can be formalized, mapping English target word at position *i* to German source word at position *j*, with a function $a : i \rightarrow j$:

$$a\{1 \rightarrow 1, 2 \rightarrow 2, 3 \rightarrow 3, 4 \rightarrow 4\}$$

- However, monotone alignments like this are very rare in practice…

Reordering

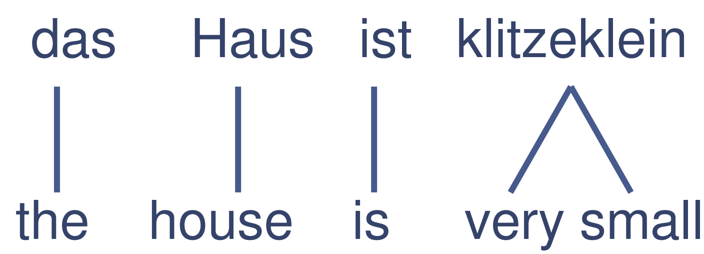
- Words may be **reordered** during translation



$$a\{1 \rightarrow 3, 2 \rightarrow 4, 3 \rightarrow 2, 4 \rightarrow 1\}$$

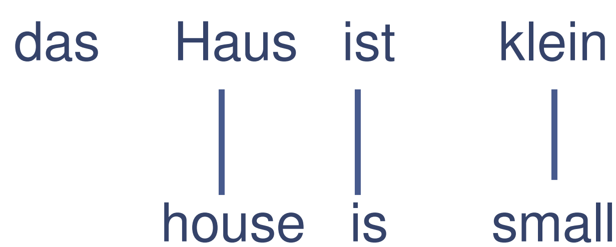
One-to-many, one-to-none

- A source word may translate into multiple target words



$$a\{1 \rightarrow 1, 2 \rightarrow 2, 3 \rightarrow 3, 4 \rightarrow 4, 5 \rightarrow 4\}$$

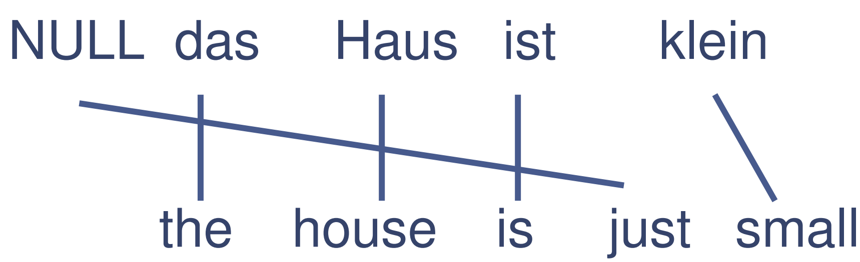
- Words may be dropped when translated



$$a\{1 \rightarrow 2, 2 \rightarrow 3, 3 \rightarrow 4\}$$

Inserting words

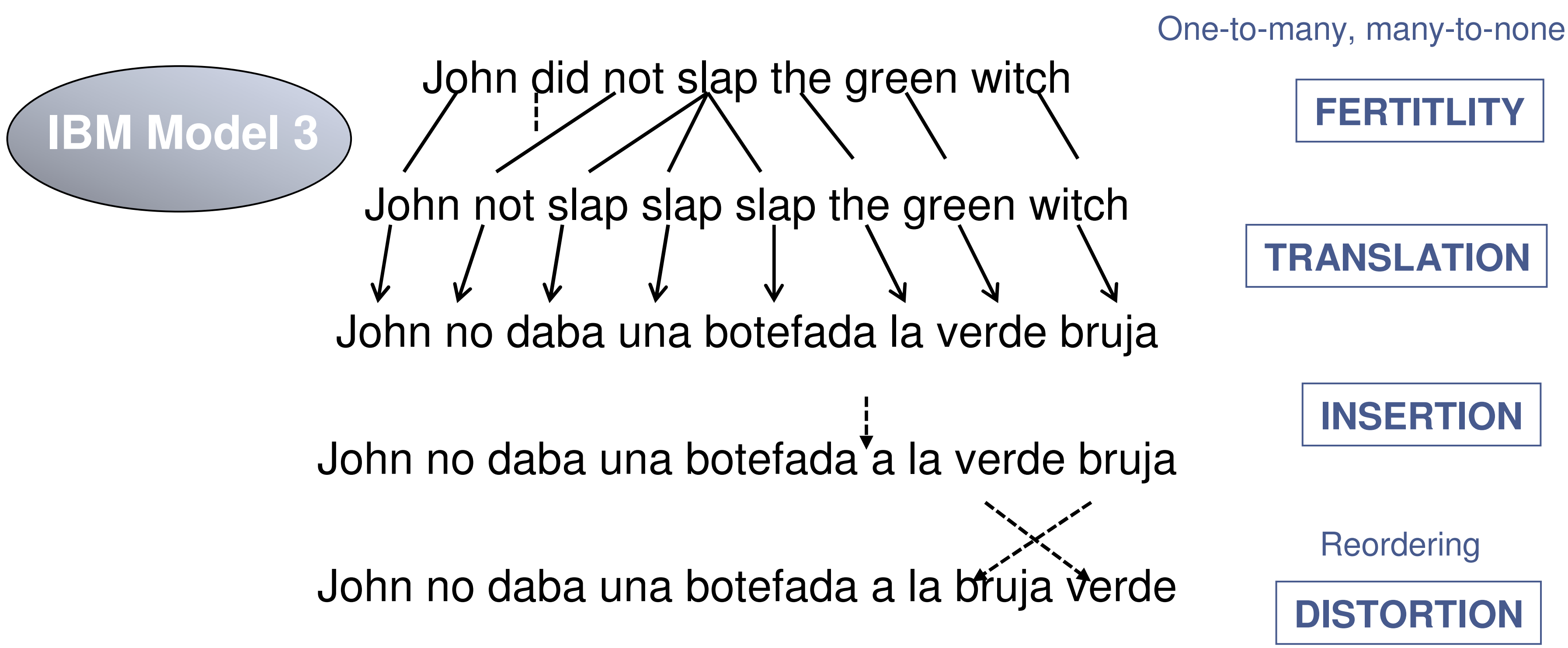
- Words may be added or inserted during translation
  - The English word *just* does not have an equivalent in German
  - We still need to map it to something: special NULL token



$$a\{1 \rightarrow 1, 2 \rightarrow 2, 3 \rightarrow 3, 4 \rightarrow 0, 5 \rightarrow 4\}$$

Translation Process as String Re-Writing

SMT Translation Model takes these alignment characteristics into account:



Translation Model Parameters (1/3)

- Translation Model takes these characteristics into account, modelling them using different parameters.

- **t:** *Lexical* / word-to-word *translation* parameters
  - t(house|Haus)
  - t(building|Haus)…
  - i.e. what is the probability that "Haus" will produce the English word house/building whenever "Haus" appears?

- **n:** *Fertility* parameters
  - n(1|klitzklein)
  - n(2|klitzklein) …
  - i.e. what is the probability that "klitzklein" will produce exactly 1/2… English words?

Translation Model Parameters (2/3)

- **d:** *Distortion* parameters
  - d(3|2)
  - i.e. what is the probability that the German word in position 2 of the German sentence will generate an English word that ends up in position 2/3 of an English translation?

- **p** : We also have word-translation parameters corresponding to *insertions*:
  - t(just | NULL) = ?
  - i.e. what is the probability that the English word just is inserted into the English string?

Translation Model Parameters: Insertion

- **p**: set a single parameter **p1** and use it as follows:
  - Assign fertilities to each word in the German string
  - At this point we are ready to start translating these German words into English words
  - As each word is translated, we insert an English word into the target string with probability **p1**
  - The probability **p0** of not inserting an extra word is given as: **p0 = 1 – p1**

Summary of Translation Model Parameters

FERTITLITY	n	Table plotting source words against fertilities
TRANSLATION	t	Table plotting source words against target words
INSERTION	p1	Single number indicating the probability of insertion
DISTORTION	d	Table plotting source string positions against target string positions

Learning Translation Models

- How can we automatically acquire parameter values for *t, n, d* and *p* from data?
- If we had a set of source language strings (e.g. German) and for each of those strings a sequence of step-by-step rewritings into English… problem solved!
  - Fairly unlikely to have this type of data
- How can collect estimates from non-aligned data?
  - Expectation Maximization Algorithm (EM)
  - We can gather information incrementally, each new piece helping us build the next.

Expectation Maximization Algorithm

- Incomplete Data
  - If we had complete data, we could estimate the *model*
  - If we had a *model* we could fill in the gaps in the data
  - i.e. if we had a rough idea about which words correspond, then we could use this knowledge to infer more data
- EM in a nutshell:
  - Initialise model parameters (i.e. uniform)
  - Assign probabilities to the missing data
  - Estimate model parameters from completed data
  - Iterate

Translation Model & Parameters

- SMT: $argmax\ P(T|S) = argmax\ P(T).P(S|T)$
  - If we carry out, for example, **French→English** translation, then we will have:
    - An **English** language model
    - An **English→French** Translation Model
- Translation Model Parameters:

Translation Model & Parameters:

- SMT: $argmax\ P(T|S) = argmax\ P(T).P(S|T)$
  - If we carry out, for example, **French→English** translation, then we will have:
    - An **English** language model
    - An **English→French** Translation Model
- Translation Model Parameters:
  - Fertility (n): number of target words generated by a particular source word
    - e.g. n(0|house)=?, n(1|house)=?…

Translation Model & Parameters:

- SMT: $argmax\, P(T|S) = argmax\, P(T).P(S|T)$
  - If we carry out, for example, **French→English** translation, then we will have:
    - An **English** language model
    - An **English→French** Translation Model
- Translation Model Parameters:
  - Fertility (n): number of target words generated by a particular source word
    - e.g. n(0|house)=?, n(1|house)=?…
  - Translation (t): (word and/or phrase) translation probabilities
    - e.g. t(maison|house)=?, t(domicile|house)=?, t(merci|house)=?...

Translation Model & Parameters:

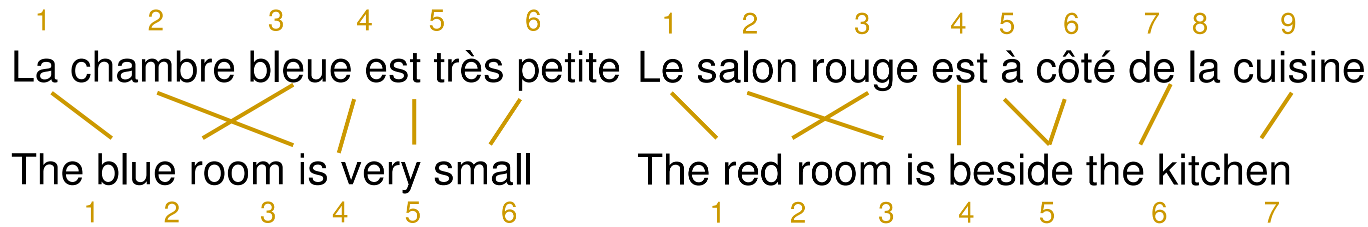
- SMT: $argmax\ P(T|S) = argmax\ P(T).P(S|T)$
  - If we carry out, for example, **French→English** translation, then we will have:
    - An **English** language model
    - An **English→French** Translation Model
- Translation Model Parameters:
  - Fertility (n): number of target words generated by a particular source word
    - e.g. n(0|house)=?, n(1|house)=?…
  - Translation (t): (word and/or phrase) translation probabilities
    - e.g. t(maison|house)=?, t(domicile|house)=?, t(merci|house)=?...
  - Insertion (p1): single number indicating the probablity of an insertion
    - Insertions included as word translation parameters i.e. t(maison|NULL)

Translation Model & Parameters:

- SMT: $argmax\ P(T|S) = argmax\ P(T).P(S|T)$
  - If we carry out, for example, **French→English** translation, then we will have:
    - An **English** language model
    - An **English→French** Translation Model
- Translation Model Parameters:
  - Fertility (n): number of target words generated by a particular source word
    - e.g. n(0|house)=?, n(1|house)=?…
  - Translation (t): (word and/or phrase) translation probabilities
    - e.g. t(maison|house)=?, t(domicile|house)=?, t(merci|house)=?...
  - Insertion (p1): single number indicating the probablity of an insertion
    - Insertions included as word translation parameters i.e. t(maison|NULL)
  - Distortion (d): probabilities for a source word in position *i* ending up in position *j* in the target
    - e.g. d(1|1),d(2|1),d(3|1)…

How to learn TM parameters?

- Answer: from alignments
- French – English, translation model is English-French (i.e. P(F|E))



TRANSLATION

t(chambre | room) =	0.5
t(salon | room) =	0.5
t(le | the) =	0.3
t(la | the) =	0.6

FERTILITY

Number of times “room” is aligned with “chambre” % Number of times “room” occurs

n(2 | beside) = 1.0

DISTORTION

d(2 | 2) =	0.0
d(3 | 2) =	1.0
d(5 | 5) =	1.0
d(6 | 5) =	0.5

- What is we don’t have alignments?
  - Infer alignments automatically; if we have a rough idea about which words correspond, we can use this knowledge to infer more alignments

Expectation Maximization Algorithm (EM)

- EM Algorithm consists of two main steps:
  - **Expectation-Step:** Apply model to the data
    - Parts of the model are hidden (here: alignments)
    - Using the model, assign probabilities to possible values
  - **Maximization-Step:** Estimate the model from the data
    - Take currently assigned values as fact
    - Collect counts (weighted by probabilities)
    - Estimate model from counts
- Iterate these steps until *convergence*

- To apply EM we need to be able to:
  - Expectation-Step: compute probability of alignments
  - Maximization-Step: collect counts

Expectation Maximization Algorithm (EM)

- EM in a nutshell:
  - Initialise model parameters (i.e. uniform)
    - initialisation
  - Assign probabilities to the missing data
    - calculate P(a,f|e) and P(a|ef)
  - Estimate model parameters from completed data
    - calculate new values of **t** from fractional counts
  - Iterate

- Start off ignoring fertility, distortion and insertion parameters and try to estimate translation (lexical) parameters only
  - IBM Model 1(IBM Models 1 – 5)

EM Step 1: Initialise Parameters

- Assume all values of **t** are uniform (i.e. all possible word-translation pairs are equally likely)
  - That is, if we had a French vocabulary consisting of 40,000 words, a given English word e might align with any of these French words. If we assume that all these alignments are equally likely, then for each French word *f*:

$$p(f|ex) = 1/40{,}000$$

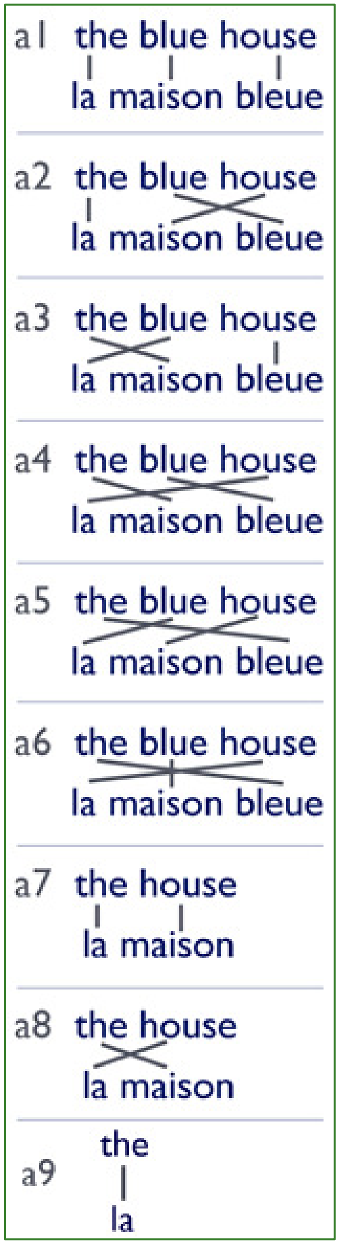
- The EM algorithm then iterates over the distribution, given the possible alignments, and updates the t values after each iteration.
- Parameters produced uniformly will produce a very low $p(f|e)$ but each iteration is guaranteed to improve the estimation of $p(f|e)$.

EM Step 1:
Initialise Parameters

- Given 3 sentence pairs
  - the blue house <-> la maison bleue
  - the house <-> la maison
  - the <-> la
- As we have no seed alignments, we have to consider all possible alignments.
- Set **t** parameters uniformly:

t(la|the) = 1/3
t(maison|the)= 1/3
t(bleue|the)= 1/3
t(la|blue)= 1/3
t(bleue|blue)= 1/3
t(la|house)= 1/3

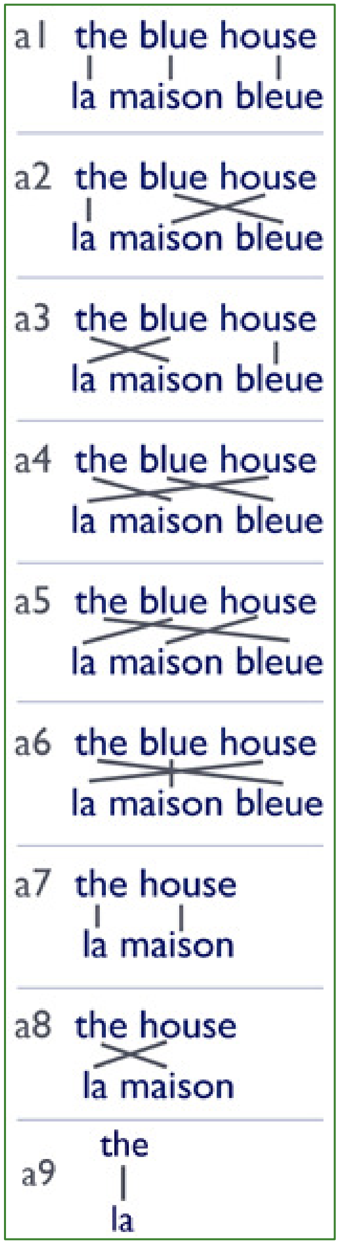
t(la|house)= 1/3
t(maison|house)= 1/3
t(bleue|house) = 1/3



EM Step 2: Compute P(a,f|e)

t(la|the) = 1/3
t(maison|the)= 1/3
t(bleue|the)= 1/3
t(la|blue)= 1/3
t(maison|blue)= 1/3
t(bleue|blue)= 1/3
t(la|house)= 1/3
t(maison|house)= 1/3
t(bleue|house) = 1/3

- Given our initial parameters, compute the probability of each of the possible alignments P(a,f|e) (illustrated in the box to the right):
  - P(a1,f|e) = 1/3 (la|the) × 1/3 (maison|blue) × 1/3 (bleue|house) = 1/27
  - P(a2,f|e) = 1/3 (la|the) × 1/3 (bleue|blue) × 1/3 (maison|house) = 1/27
  - P(a3,f|e) = 1/3 × 1/3 × 1/3 = 1/27
  - P(a4,f|e) = 1/3 × 1/3 × 1/3 = 1/27
  - P(a5,f|e) = 1/3 × 1/3 × 1/3 = 1/27
  - P(a6,f|e) = 1/3 × 1/3 × 1/3 = 1/27
  - P(a7,f|e) = 1/3 (la|the) × 1/3 (masion|house) = 1/9
  - P(a8,f|e) = 1/3 (maison|the) × 1/3 (la|house) = 1/9
  - P(a9,f|e) = 1/3



EM Step 2:
Normalise P(a,f|e) to yield P(a|e,f)

- From previous step:

  P(a1,f|e) = 1/27
  P(a2,f|e) = 1/27
  P(a3,f|e) = 1/27
  P(a4,f|e) = 1/27
  P(a5,f|e) = 1/27
  P(a6,f|e) = 1/27
  P(a7,f|e) = 1/9
  P(a8,f|e) = 1/9
  P(a9,f|e) = 1/3

- Normalize P(a,f|e) values to yield P(a|e,f) (normalize by sum of probabilities of possible alignments for the source string in question):

$$P(a1|e,f) = \frac{1}{27} \div \frac{6}{27} = \frac{1}{6}$$

($\frac{6}{27}$ = sum over a1-a6 as they are possible alignments for the source string "the blue house")

$$P(a2|e,f) = \frac{1}{27} \div \frac{6}{27} = \frac{1}{6}$$

$$P(a3|e,f) = \frac{1}{27} \div \frac{6}{27} = \frac{1}{6}$$

$$P(a4|e,f) = \frac{1}{27} \div \frac{6}{27} = \frac{1}{6}$$

$$P(a5|e,f) = \frac{1}{27} \div \frac{6}{27} = \frac{1}{6}$$

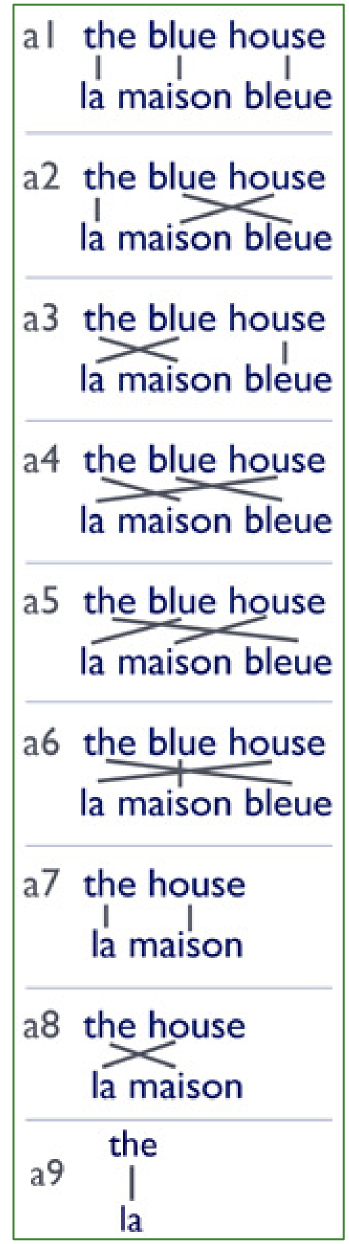
$$P(a6|e,f) = \frac{1}{27} \div \frac{6}{27} = \frac{1}{6}$$

$$P(a7|e,f) = \frac{1}{9} \div \frac{2}{9} = \frac{1}{2}$$

$$P(a8|e,f) = \frac{1}{9} \div \frac{2}{9} = \frac{1}{2}$$

$$P(a9|e,f) = \frac{1}{3} \div \frac{1}{3} = 1$$

only 1 alignment, therefore P(a9|e,f) will always be 1



EM Step 3:
Collect Fractional Counts

$P(a1|e,f) = \frac{1}{6}$

$P(a2|e,f) = \frac{1}{6}$

$P(a3|e,f) = \frac{1}{6}$

$P(a4|e,f) = \frac{1}{6}$

$P(a5|e,f) = \frac{1}{6}$

$P(a6|e,f) = \frac{1}{6}$

$P(a7|e,f) = \frac{1}{2}$

$P(a8|e,f) = \frac{1}{2}$

$P(a9|e,f) = 1$

- Collect fractional counts for each translation pair (i.e. for each translation pair, sum values of P(a|e,f) where the word pair occurs):

$tc(la|the) = \frac{1}{6}$ (from a1) $+ \frac{1}{6}$ (from a2) $+ \frac{1}{2}$ (from a7) $+ 1$ (from a9) $= \frac{11}{6}$

$tc(maison|the) = \frac{1}{6} + \frac{1}{6} + \frac{1}{2} = \frac{5}{6}$

$tc(bleue|the) = \frac{1}{6}$ (from a5) $+ \frac{1}{6}$ (from a6) $= \frac{2}{6}$

$tc(la|blue) = \frac{1}{6} + \frac{1}{6} = \frac{2}{6}$

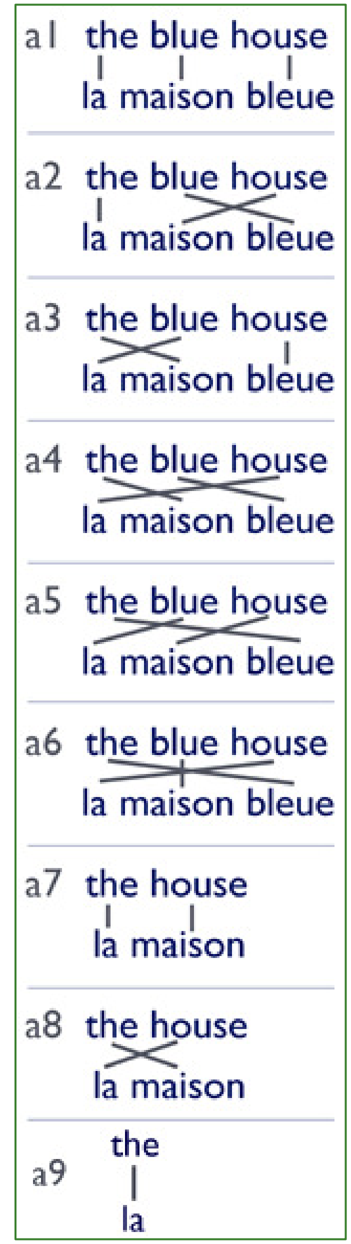
$tc(maison|blue) = \frac{1}{6} + \frac{1}{6} = \frac{2}{6}$

$tc(bleue|blue) = \frac{1}{6} + \frac{1}{6} = \frac{2}{6}$

$tc(la|house) = \frac{1}{6} + \frac{1}{6} + \frac{1}{2} = \frac{5}{6}$

$tc(maison|house) = \frac{1}{6} + \frac{1}{6} + \frac{1}{2} = \frac{5}{6}$

$tc(bleue|house) = \frac{1}{6} + \frac{1}{6} = \frac{2}{6}$



EM Step 3:
Normalize Fractional Counts

$tc(la|the) = \frac{11}{6}$

$tc(maison|the) = \frac{5}{6}$

$tc(bleue|the) = \frac{2}{6}$

$tc(la|blue) = \frac{2}{6}$

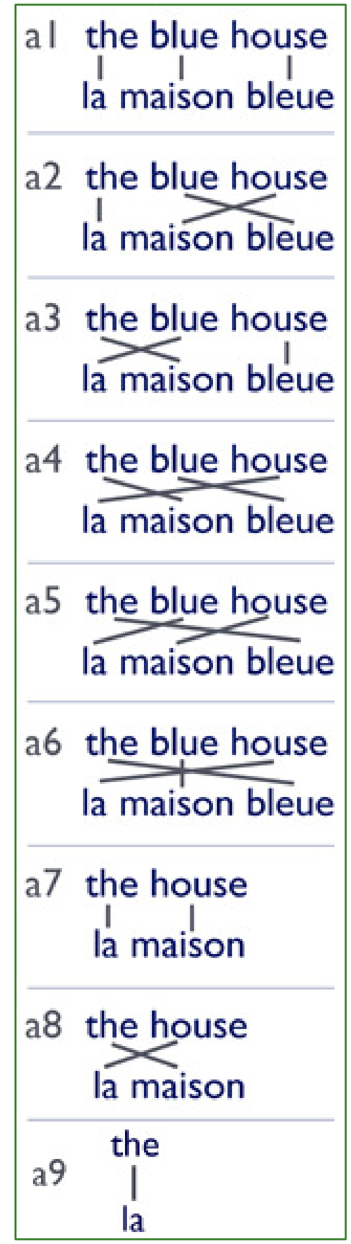
$tc(maison|blue) = \frac{2}{6}$

$tc(bleue|blue) = \frac{2}{6}$

$tc(la|house) = \frac{5}{6}$

$tc(maison|house) = \frac{5}{6}$

$tc(bleue|house) = \frac{2}{6}$



- Normalize fractional counts to get revised parameters for **t**

$t(la|the) = \frac{11}{6} \div \frac{18}{6}$ (sum of counts for translation pairs where "the" occurs) $= \frac{11}{18}$

$t(maison|the) = \frac{5}{6} \div \frac{18}{6} = \frac{5}{18}$

$t(bleue|the) = \frac{2}{6} \div \frac{18}{6} = \frac{2}{18} = \frac{1}{9}$

$t(la|blue) = \frac{2}{6} \div \frac{6}{6} = \frac{2}{6} = \frac{1}{3}$

$t(maison|blue) = \frac{2}{6} \div \frac{6}{6} = \frac{1}{3}$

$t(bleue|blue) = \frac{2}{6} \div \frac{6}{6} = \frac{1}{3}$

$t(la|house) = \frac{5}{6} \div \frac{12}{6} = \frac{5}{12}$

$t(maison|house) = \frac{5}{6} \div \frac{12}{6} = \frac{5}{12}$

$t(bleue|house) = \frac{2}{6} \div \frac{12}{6} = \frac{2}{12} = \frac{1}{6}$

Step 4 Iterate: Repeat Step 2

$t(la|the) = \frac{11}{18}$

$t(maison|the) = \frac{5}{18}$ $t(bleue|blue) = \frac{1}{3}$

$t(bleue|the) = \frac{1}{9}$ $t(la|house) = \frac{5}{12}$

$t(la|blue) = \frac{1}{3}$ $t(maison|house) = \frac{5}{12}$

$t(maison|blue) = \frac{1}{3}$ $t(bleue|house) = \frac{1}{6}$

- Given our new parameter values, re-compute the probability of each of the possible alignments P(a,f|e):

$$P(a1,f|e) = t(la|the) \times t(maison|blue) \times t(bleue|house) = \frac{11}{18} \times \frac{1}{3} \times \frac{1}{6} = \frac{11}{324}$$

$$P(a2,f|e) = \frac{11}{18} \times \frac{1}{3} \times \frac{5}{12} = \frac{55}{648}$$

$$P(a3,f|e) = \frac{5}{18} \times \frac{1}{3} \times \frac{1}{6} = \frac{5}{324}$$

$$P(a4,f|e) = \frac{5}{18} \times \frac{1}{3} \times \frac{5}{12} = \frac{25}{648}$$

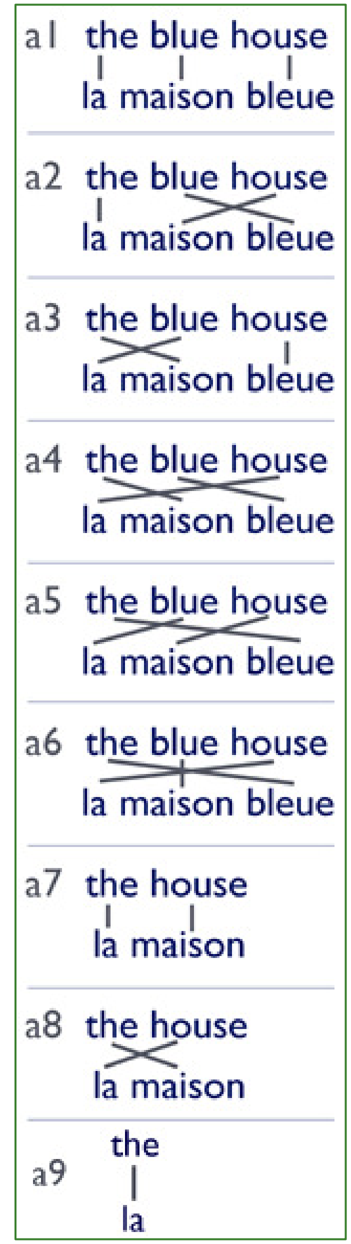
$$P(a5,f|e) = \frac{1}{9} \times \frac{1}{3} \times \frac{5}{12} = \frac{5}{324}$$

$$P(a6,f|e) = \frac{1}{9} \times \frac{1}{3} \times \frac{5}{12} = \frac{5}{324}$$

$$P(a7,f|e) = t(la|the) \times t(maison|house) = \frac{11}{18} \times \frac{5}{12} = \frac{55}{216}$$

$$P(a8,f|e) = \frac{5}{18} \times \frac{5}{12} = \frac{25}{216}$$

$$P(a9,f|e) = \frac{11}{18}$$



2^{nd} Iteration:

Step 2 Normalise P(a,f|e)

$P(a1,f|e) = \frac{11}{324} = \frac{22}{648}$

$P(a2,f|e) = \frac{55}{648}$

$P(a3,f|e) = \frac{5}{324} = \frac{10}{648}$

$P(a4,f|e) = \frac{25}{648}$

$P(a5,f|e) = \frac{5}{324} = \frac{10}{648}$

$P(a6,f|e) = \frac{5}{324} = \frac{10}{648}$

$P(a7,f|e) = \frac{55}{216}$

$P(a8,f|e) = \frac{25}{216}$

$P(a9,f|e) = \frac{11}{18}$

- Normalize P(a,f|e) values to yield P(a|e,f):

$P(a1|e,f) = \frac{22}{648} \div \frac{132}{648}$ (sum a1-a6)

$= \frac{22}{648} \times \frac{648}{132} = \frac{22}{132}$

$P(a2|e,f) = \frac{55}{648} \div \frac{132}{648} = \frac{55}{132}$

$P(a3|e,f) = \frac{10}{648} \div \frac{132}{648} = \frac{10}{132}$

$P(a4|e,f) = \frac{25}{648} \div \frac{132}{648} = \frac{25}{132}$

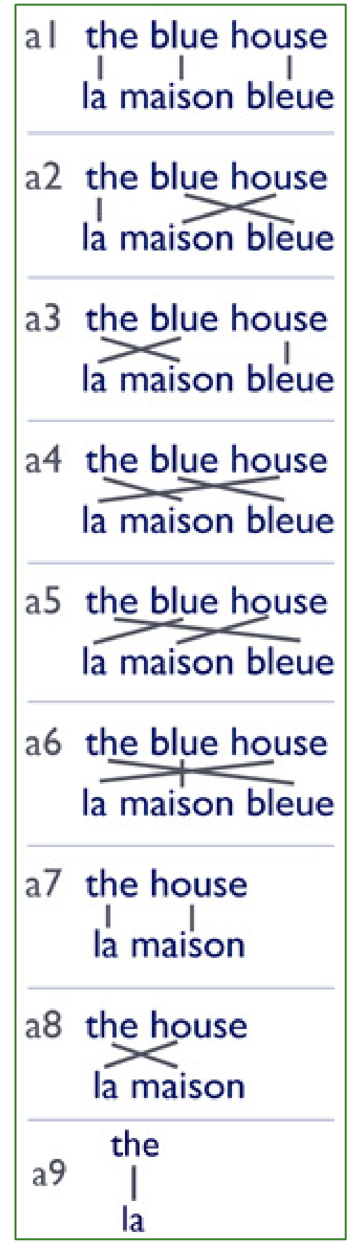
$P(a5|e,f) = \frac{10}{648} \div \frac{132}{648} = \frac{10}{132}$

$P(a6|e,f) = \frac{10}{648} \div \frac{132}{648} = \frac{10}{132}$

$P(a7|e,f) = \frac{55}{216} \div \frac{80}{216} = \frac{55}{80}$

$P(a8|e,f) = \frac{25}{216} \div \frac{80}{216} = \frac{25}{80}$

$P(a9|e,f) = \frac{11}{18} \div \frac{11}{18} = 1$



2^{nd} Iteration:

Step 3 Collect Fractional Counts

$P(a1|e,f) = \frac{22}{132} = \frac{1}{6} = \frac{8}{48} = \frac{88}{528}$

$P(a2|e,f) = \frac{55}{132} = \frac{5}{12} = \frac{20}{48} = \frac{220}{528}$

$P(a3|e,f) = \frac{10}{132} = \frac{40}{528}$

$P(a4|e,f) = \frac{25}{132} = \frac{100}{528}$

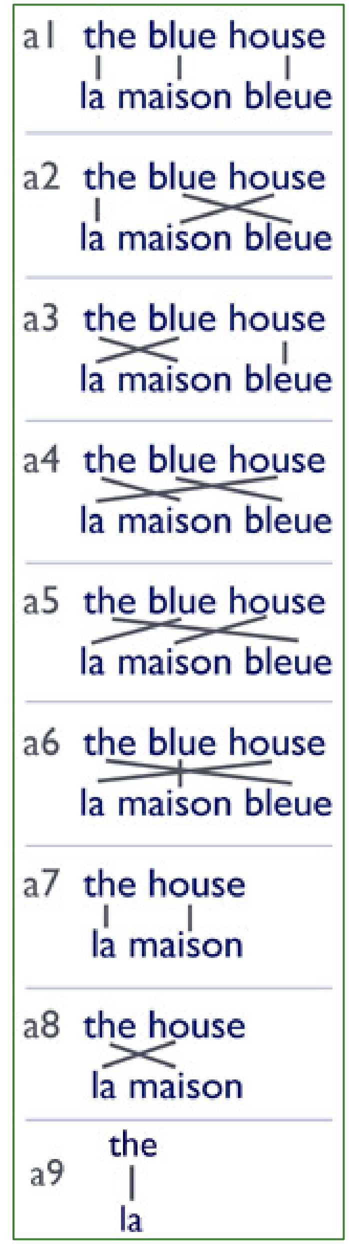
$P(a5|e,f) = \frac{10}{132} = \frac{40}{528}$

$P(a6|e,f) = \frac{10}{132}$

$P(a7|e,f) = \frac{165}{240} = \frac{11}{16} = \frac{33}{48}$

$P(a8|e,f) = \frac{75}{240} = \frac{5}{16} = \frac{165}{528}$

$P(a9|e,f) = 1 = \frac{48}{48}$



- Collect fractional counts for each translation pair

$tc(la|the) = \frac{8}{48} + \frac{20}{48} + \frac{33}{48} + \frac{48}{48} = \frac{109}{48}$ (values from a1, a2, a7 and a9)

$tc(maison|the) = \frac{40}{528} + \frac{100}{528} + \frac{165}{528} = \frac{305}{528}$

$tc(bleue|the) = \frac{40}{528} + \frac{40}{528} = \frac{80}{528}$

$tc(la|blue) = \frac{40}{528} + \frac{40}{528} = \frac{80}{528}$

$tc(maison|blue) = \frac{88}{528} + \frac{40}{528} = \frac{128}{528}$

$tc(bleue|blue) = \frac{220}{528} + \frac{100}{528} = \frac{320}{528}$

$tc(la|house) = \frac{100}{528} + \frac{40}{528} + \frac{165}{528} = \frac{305}{528}$

$tc(maison|house) = \frac{220}{528} + \frac{40}{528} + \frac{33}{48} = \frac{623}{528}$

$tc(bleue|house) = \frac{88}{528} + \frac{40}{528} = \frac{128}{528}$

2nd Iteration:

Step 3 Normalize Fractional Cou

$tc(la|the) = \frac{109}{48}$

$tc(maison|the) = \frac{305}{528}$

$tc(bleue|the) = \frac{80}{528}$

$tc(la|blue) = \frac{80}{528}$

$tc(maison|blue) = \frac{128}{528}$

$tc(bleue|blue) = \frac{320}{528}$

$tc(la|house) = \frac{305}{528}$

$tc(maison|house) = \frac{623}{528}$

$tc(bleue|house) = \frac{128}{528}$

- Normalize fractional counts to get revised parameters for **t**

$t(la|the) = \frac{109}{48} \div \left(\frac{109}{48} + \frac{305}{528} + \frac{80}{528} = \frac{1584}{528}\right) = \frac{1199}{1584} = \frac{109}{144}$

$t(maison|the) = \frac{305}{528} \div \frac{1584}{528} = \frac{305}{1584}$

$t(bleue|the) = \frac{80}{528} \div \frac{1584}{528} = \frac{80}{1584} = \frac{5}{99}$

$t(la|blue) = \frac{80}{528} \div \left(\frac{80}{528} + \frac{128}{528} + \frac{320}{528} = \frac{1}{1}\right) = \frac{80}{528} = \frac{5}{33}$

$t(maison|blue) = \frac{128}{528} \div 1 = \frac{8}{33}$

$t(bleue|blue) = \frac{320}{528} \div 1 = \frac{20}{33}$

$t(la|house) = \frac{305}{528} \div \left(\frac{305}{528} + \frac{623}{528} + \frac{128}{528} = \frac{1056}{528}\right) = \frac{305}{1056}$

$t(maison|house) = \frac{623}{528} \div \frac{1056}{528} = \frac{623}{1056}$

$t(bleue|house) = \frac{128}{528} \div \frac{1056}{528} = \frac{128}{1056}$

EM: Convergence

- After the second iteration, are **t** values are:

$t(la|the) = \frac{109}{144} = \mathbf{0.7569}$

$t(maison|the) = \frac{305}{1584} = 0.1926$

$t(bleue|the) = \frac{5}{99} = 0.0505$

$t(la|blue) = \frac{5}{33} = 0.1515$

$t(maison|blue) = \frac{8}{33} = 0.2424$

$t(bleue|blue) = \frac{20}{33} = \mathbf{0.6061}$

$t(la|house) = \frac{305}{1056} = 0.2888$

$t(maison|house) = \frac{623}{1056} = \mathbf{0.5810}$

$t(bleue|house) = \frac{128}{1056} = 0.1212$

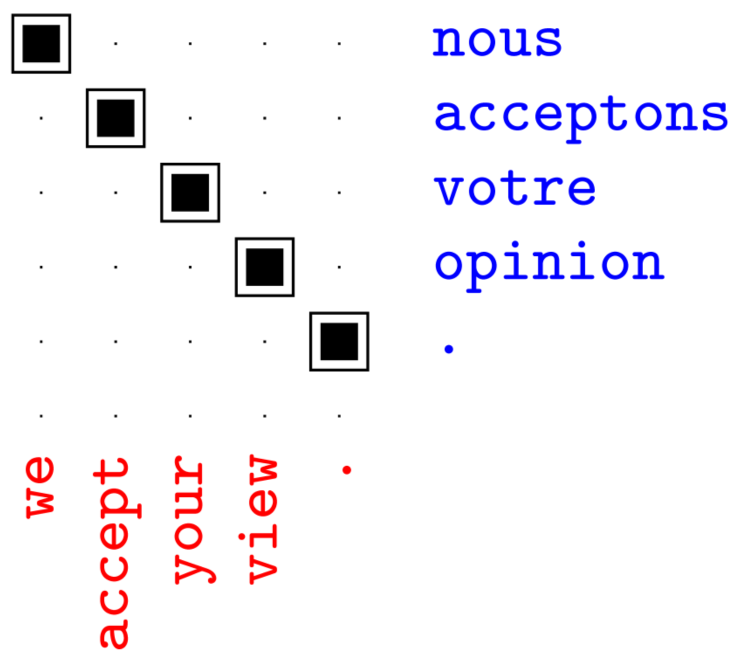
- We continue EM until our **t** values *converge*
- It is clear to see already, after 2 iterations, how some translation candidates are (correctly) becoming more likely then others

Table Representation of Alignment

- Aligned translated sentences

```
nous acceptons votre opinion .

we accept your view .
```



Alignment Error Rate

- Alignment Error Rate

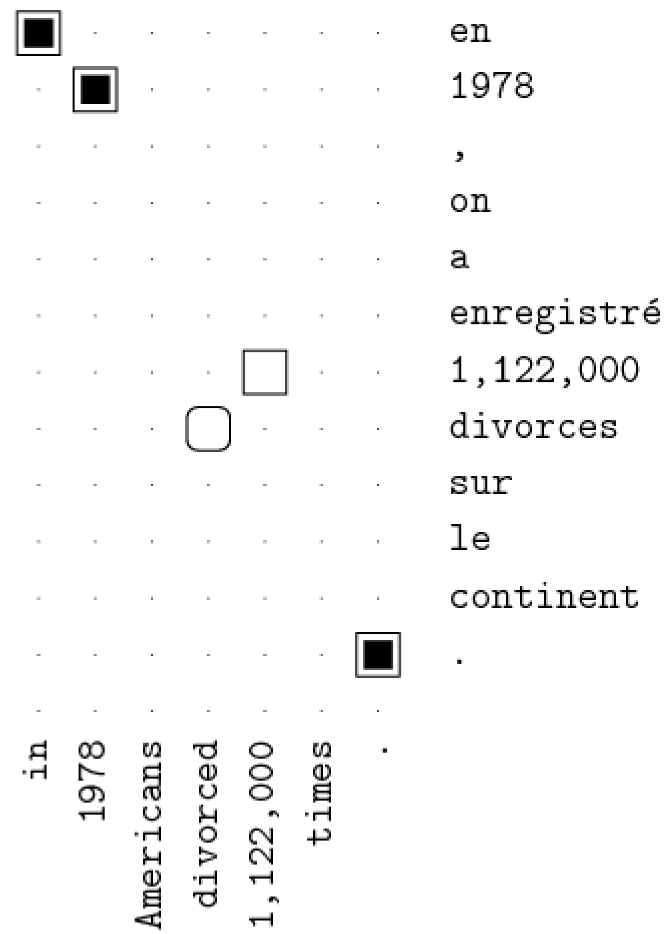
  □ = Sure

  ▢ = Possible

  ■ = Predicted

$$AER(A, S, P) = \left(1 - \frac{|A \cap S| + |A \cap P|}{|A| + |S|}\right)$$

$$= \left(1 - \frac{3+3}{3+4}\right) = \frac{1}{7}$$



From [1]

Beyond IBM Model 1

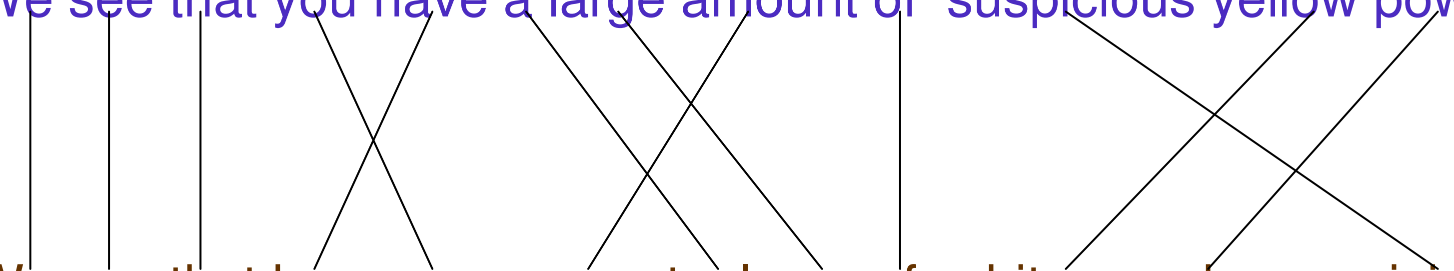
- Previous example shows how EM can be used to calculate lexical parameters
  - IBM Model 1
- But what about fertility, distortion & insertion parameters?
  - Need more complex models (IBM Models 2 – 5)

- IBM Model 2 involves both word translations and adds **absolute reordering (distortion)** model
- IBM Model 3: adds **fertility** model
- IBM Model 4: **relative reordering** model
- IBM Model 5: Fixes **deficiency**

Problems with IBM Model 1

- Improve the reordering of the MT output

We see that you have a large amount of suspicious yellow powder

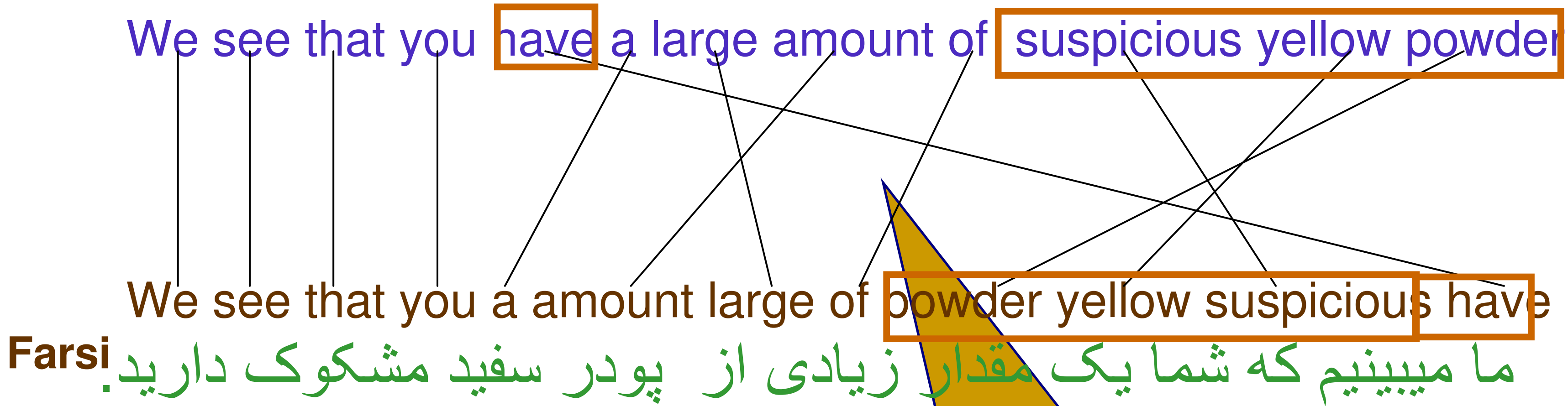


We see that have you amount a large of white powder suspicious

Arabic نحن رأيتى ان يتلقى انت مبلغه كبيره من بيضاء مسحوق مشبوهة

Distortion Model for English/Farsi

- Improve the reordering of the MT output



~ N! possible reordering
(12! = 479 million permutations)
~ Data Sparsity big problem
~ Particularly difficult when source and target language order differs a lot

Adjectives and Possessives

- Adjectives appear after the noun

He has dark skin and dark eyes and dark hair.

او پوست تیره و چشمان تیره و موی تیره دارد.

He skin dark and eyes dark and hair dark has.

- Possessives appear after the nouns they modify

Where is your friend from?

دوستتان از کجا هست؟

Friend your from where is?

Verbs in Farsi/Dari

- Normal Declarative sentences are structured as SOV
  - Subject (S) + Object (O) + Verb (V)

I got here last Friday.

من جمعه گذشته اینجا امدم.

I Friday last here got

- But it's always not the case

We see that you have a large amount of a suspicious yellow powder

ما میبینیم که شما یک مقدار زیادی از یک پودر سفید مشکوک دارید.

We see that you a amount large of a powder white suspicious have

Addressing Reordering

- We can address reordering by weighting the alignments with word jumps

$$P(f, a|e) = \prod_j P(a_j = i|j, I, J)P(f_j|e_i)$$

$$P(dist = i - j\frac{I}{J})$$

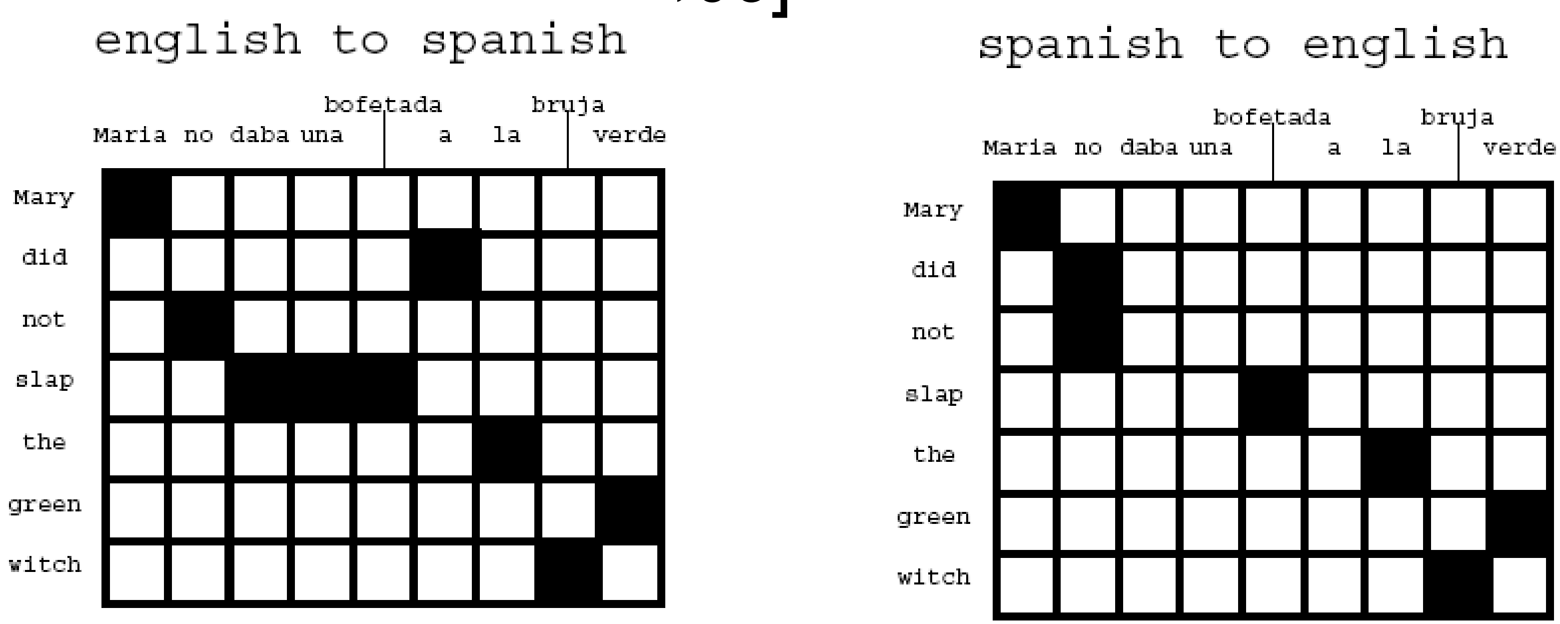
$$\frac{1}{Z}e^{-\alpha(i - j\frac{I}{J})}$$

Many Different Algorithms Have been Proposed for Reordering

- Block orientation [Tillman, 2004]
- Outbound, Inbound, Pair [Al-Onaizan and Papinneni, 2006]
- Distance based – [Berger, 1996]
- Source side reordering with N-best reordered source sentence [Kanthak, et. al. 2005]
- Using syntax on source side [Li, et.al, 2007]
- Hierarchical phrases [Galley and Manning, 2008]

Can We Further Improve Alignments?

- Combining Direction based alignment has been explored [Och and Ney, 2004, Zens et al, 2004; Liang et al. 2006]



Example figures from [Och and Ney, 2004]

Related Work

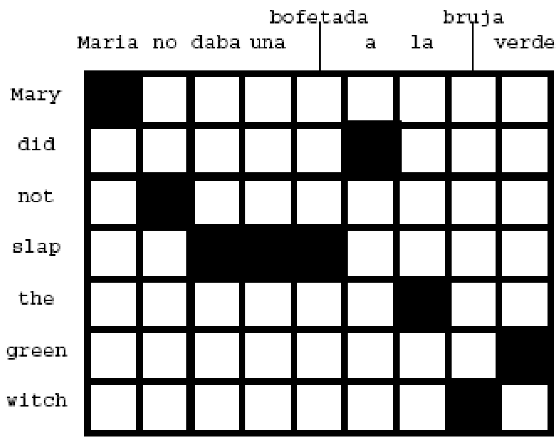
- (Och and Ney, 2003) add links that are adjacent to intersection links
- (Koehn et. al, 2003) add diagonal neighbors
- (Liang et. al, 2006) jointly trained to maximize likelihood and agreemtn of alignments
- (Necip et. al, 2004) combine alignments based on various resources such as POS, dependency and do supervised training
- (Zens et al, 2004) Using statistics from the other direction
- Most of the combination methods are based on heuristics

- Why combination (symmetrization) ?
  - Makes up for model assumption of 1:m
  - Quite simple if heuristic based methods used
  - Works most of the time

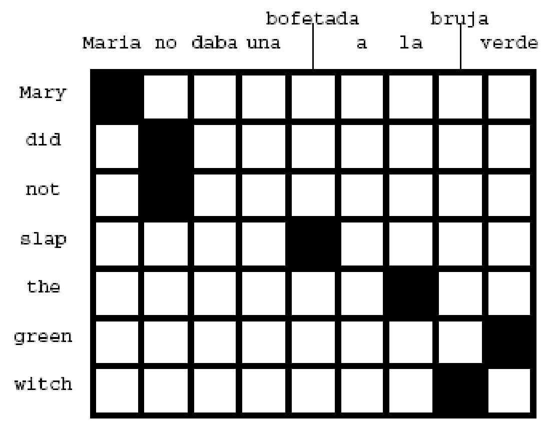
Heuristic Based Methods

- Common practice
  - Combine two sets of alignments
  - Train word alignments in two directions: E→F, F→E
  - Phrase table and/or rule training
- Common Combination Methods
  - Intersection
  - Union
  - Growing Heuristics
  - Och Refined Heuristics

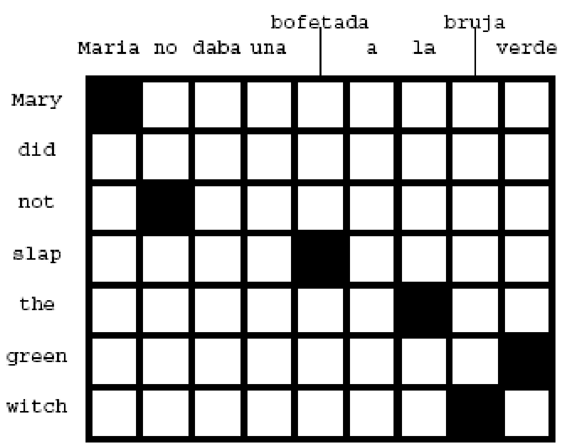
E→F
english to spanish



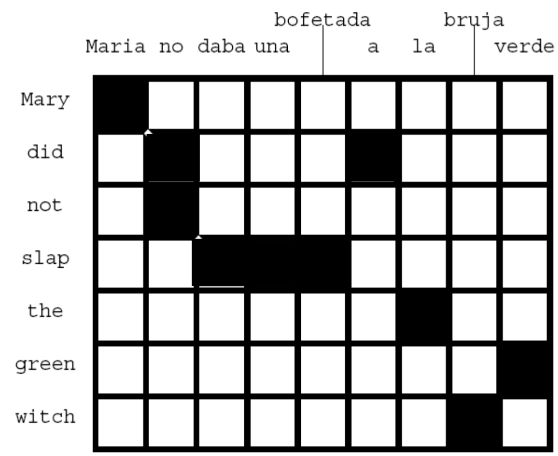
F→E
spanish to english



Intersection (I)

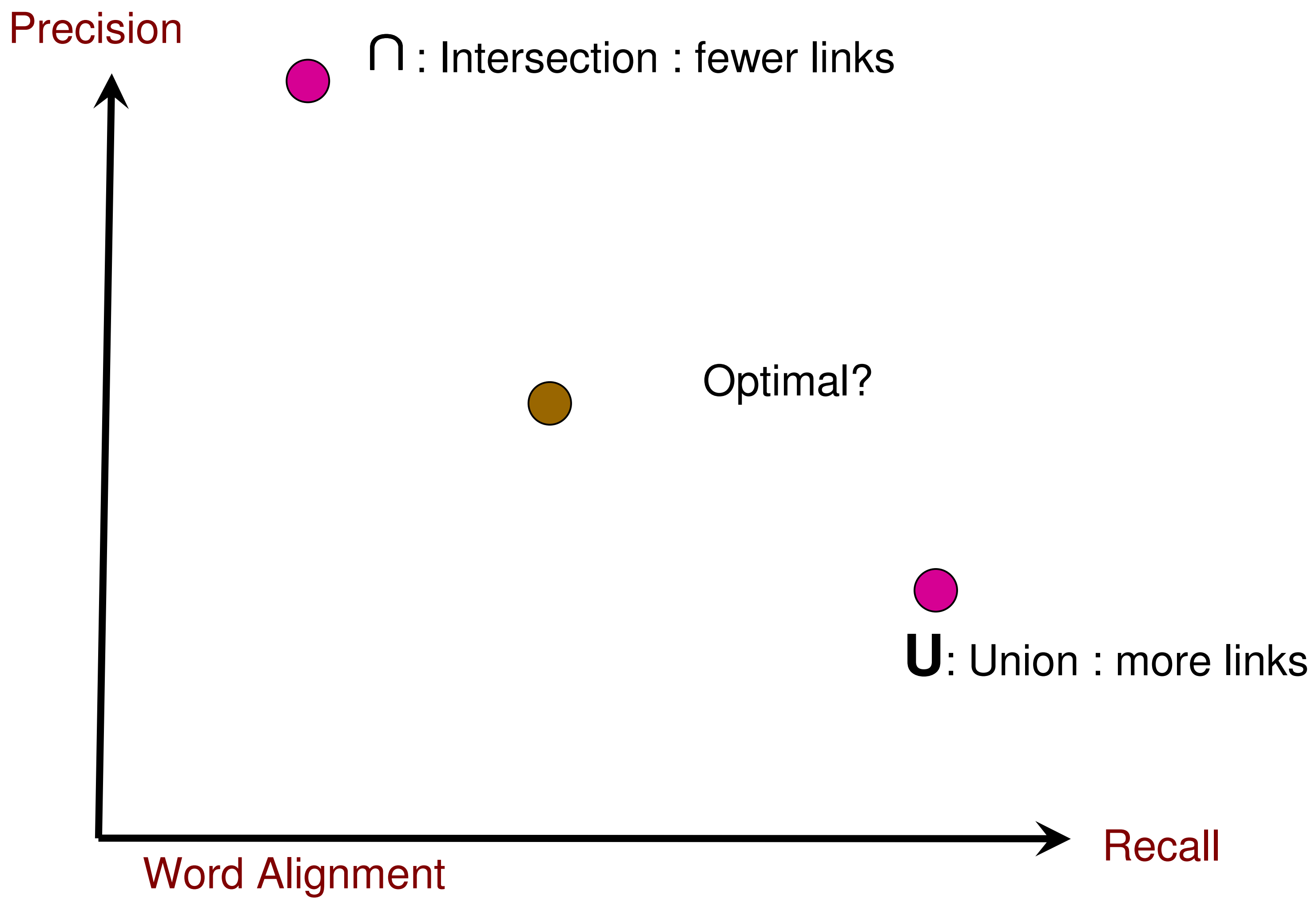


Union (U)



Example figures from [Och and Ney, 2004]

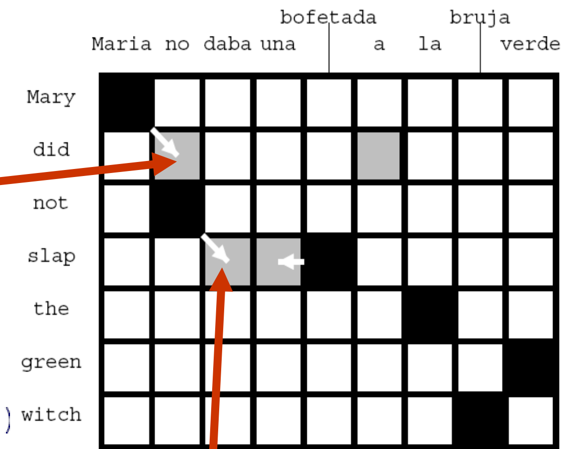
Optimal Combined Alignment



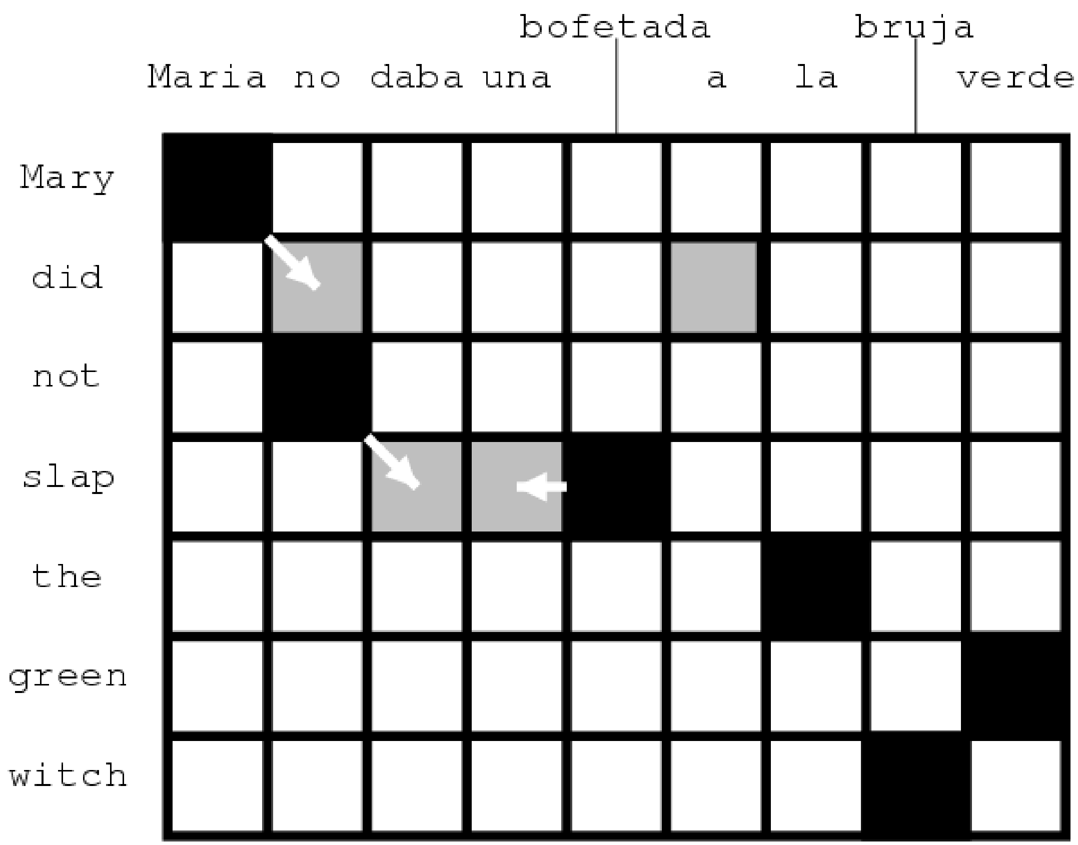
Growing Heuristic [Koehn, et. al, 2003]

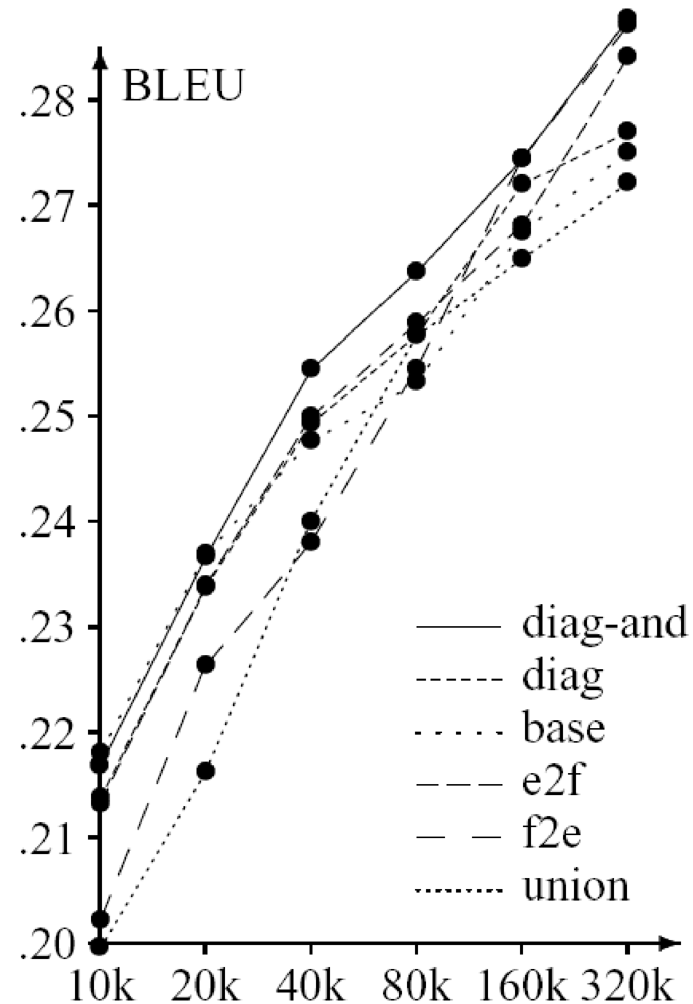
```
GROW-DIAG-FINAL(e2f,f2e):
  neighboring = ((-1,0),(0,-1),(1,0),(0,1),(-1,-1),(-1,1),(1,-1),(1,1))
  alignment = intersect(e2f,f2e);
  GROW-DIAG(); FINAL(e2f); FINAL(f2e);

GROW-DIAG():
  iterate until no new points added
    for english word e = 0 ... en
      for foreign word f = 0 ... fn
        if ( e aligned with f )
          for each neighboring point ( e-new, f-new )
            if ( ( e-new not aligned and f-new not aligned ) and
                 ( e-new, f-new ) in union( e2f, f2e ) )
              add alignment point ( e-new, f-new )
FINAL(a):
  for english word e-new = 0 ... en
    for foreign word f-new = 0 ... fn
      if ( ( e-new not aligned or f-new not aligned ) and
           ( e-new, f-new ) in alignment a )
        add alignment point ( e-new, f-new )
```



Alignment Heuristics





Reference

- [1] http://www.cs.berkeley.edu/~klein/cs294-5/FA05%20cs294-5%20lecture%2010.pdf
- [2] http://www.cs.columbia.edu/~smaskey/CS6998/slides/statnlp_week12.pdf
- [Och and Ney, 2003] Franz Och, Hermann Ney, "A Systematic Comparison of Various Statistical Alignment Models," Computational Linguistics, Vol. 29, 2003
- [Koehn et. al, 03] Philipp Koehn, Franz Josef Och, and Daniel Marcu, " Statistical phrase-based translation. In Proceedings of HLT/NAACL Conference, 2003
- [Groves, D.] http://www.computing.dcu.ie/~dgroves/CA446.html
- [Necip et. al, 04] Necip, Ayan, Bonnie J. Dorr, , and Nizar Habash, Multi-align: Combining linguistic and statistical techniques to improve alignments for adaptable AMTA, 2004
- [Liang et. al, 06] Liang, Percy, Ben Taskar, and Dan Klein, Alignment by agreement. In HLT. ACL, 2006