

Méthodes empiriques pour le traitement automatique du langage naturel

Laurent Besacier

- 1. Théorie de l'information et probabilités**
- 2. Modélisation probabiliste du langage**
- 3. Introduction à la traduction automatique probabiliste**

Méthodes empiriques pour le traitement automatique du langage naturel

Laurent Besacier

1. Théorie de l'information et probabilités

Références

- Manning and Schutze: "Foundations of Statistical Language Processing", 1999, MIT Press, available online
- Jurafsky and Martin: "Speech and Language Processing", 2000, Prentice Hall.
- Rajman M. « Speech and language engineering », 2007, EPFL Press.

Le langage comme donnée

- Grandes quantités de textes disponibles sous forme numérique
- Milliards de documents disponibles sur le Web
- Dizaine de milliers de phrases annotées (arbres syntaxiques)
- Centaine de millions de mots traduits entre l'anglais et d'autres langues

Compter les mots

- Statistiques sur le roman « Tom Sawyer » de M. Twain

Word	Count
the	3332
and	2973
a	1775
to	1725
of	1440
was	1161
it	1027
in	906
that	877

Distributions

- 3993 singletons
- La plupart des mots apparaissent peu
- La plus grosse partie du texte correspond à la centaine de mots les plus fréquents

count	count of count
1	3993
2	1292
3	664
4	410
5	243
6	199
7	172
...	...
10	91
11-50	540
51-100	99
> 100	102

Loi de Zipf

Zipf's law: $f \times r = k$

Rank r	Word	Count f	$f \times r$
1	the	3332	3332
2	and	2973	5944
3	a	1775	5235
10	he	877	8770
20	but	410	8400
30	be	294	8820
100	two	104	10400
1000	family	8	8000
8000	applausive	1	8000

Probabilités

- Nous pouvons estimer une distribution de probabilités à partir des fréquences des mots

$$P(w) = \frac{\text{count}(w)}{\sum_{w'} \text{count}(w')}$$

- Cette estimation est appelée par « maximum de vraisemblance »
- Cette distribution répond à la question : « si je tire un mot au hasard d'un texte, quelle est la probabilité que ce mot soit le mot w ? »

Formalisation

- Soit une variable aléatoire W
- Nous définissons la distribution de probabilités p , qui indique avec quelle vraisemblance la variable W prend la 'valeur' w (« est le mot w »)

$$\mathit{prob}(W = w) = p(w)$$

Probabilités conjointes

- On s'intéresse à 2 variables aléatoires en même temps
- Exemple : les mots w_1 et w_2 qui apparaissent l'un après l'autre (un bigramme) ; on modélise cela avec la distribution $p(w_1, w_2)$
- Si l'apparition de 2 mots en bigrammes est indépendante, nous pouvons écrire
 - $p(w_1, w_2) = p(w_1)p(w_2)$; cette hypothèse est probablement fausse !
- On peut estimer la probabilité conjointe de 2 variables de la même façon que ceci est fait pour une seule variable

$$p(w_1, w_2) = \frac{\text{count}(w_1, w_2)}{\sum_{w_1', w_2'} \text{count}(w_1', w_2')}$$

Probabilités conditionnelles

- Notée : $p(w_2|w_1)$
- Répond à la question : si la variable aléatoire $W_1 = w_1$, avec quelle vraisemblance la variable W_2 prend la 'valeur' w_2
- Mathématiquement, on écrit :
$$p(w_2|w_1) = \frac{p(w_1, w_2)}{p(w_1)}$$
- Si W_1 et W_2 sont indépendantes on a alors $p(w_2|w_1) = p(w_2)$

Règle 1 (« chain rule »)

$$p(w_1) p(w_2|w_1) = p(w_1, w_2)$$

$$p(w_1, w_2, w_3) = p(w_1) p(w_2|w_1) p(w_3|w_1, w_2)$$

... Etc.

Règle 2 (Règle de Bayes)

- Règle de Bayes

$$p(x|y) = \frac{p(y|x) p(x)}{p(y)}$$

- Obtenue à partir de la règle précédente

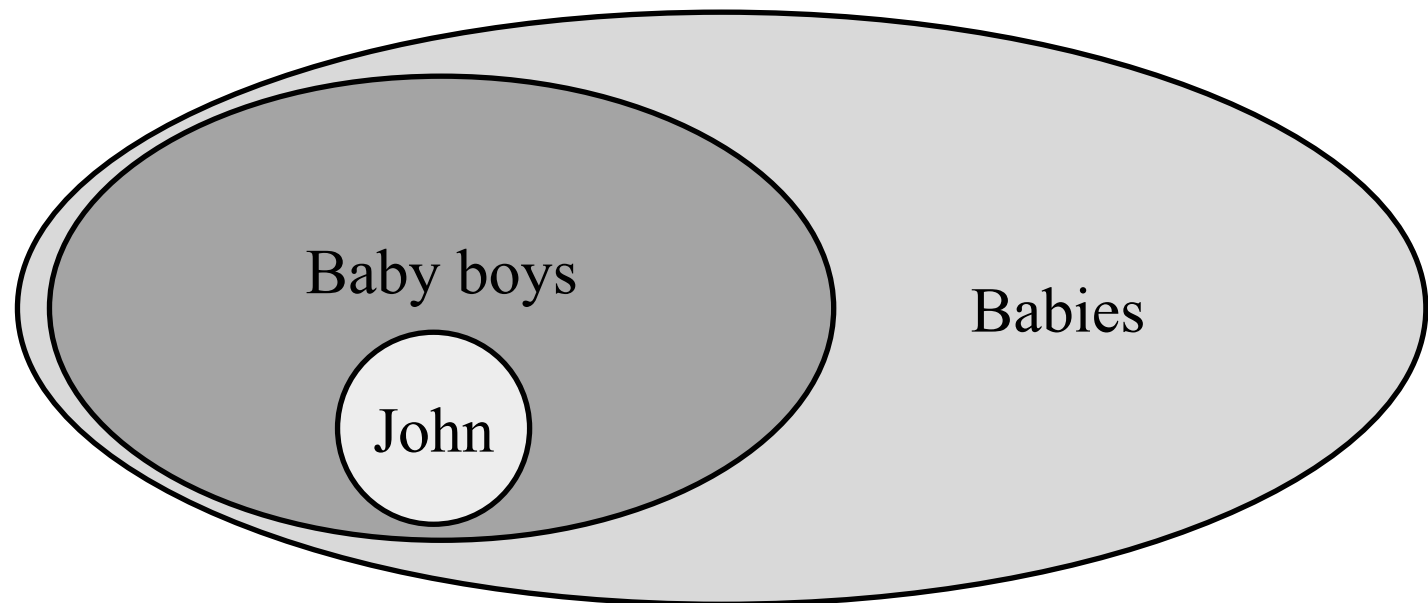
$$p(x, y) = p(x, y)$$

$$p(x|y) p(y) = p(y|x) p(x)$$

$$p(x|y) = \frac{p(y|x) p(x)}{p(y)}$$

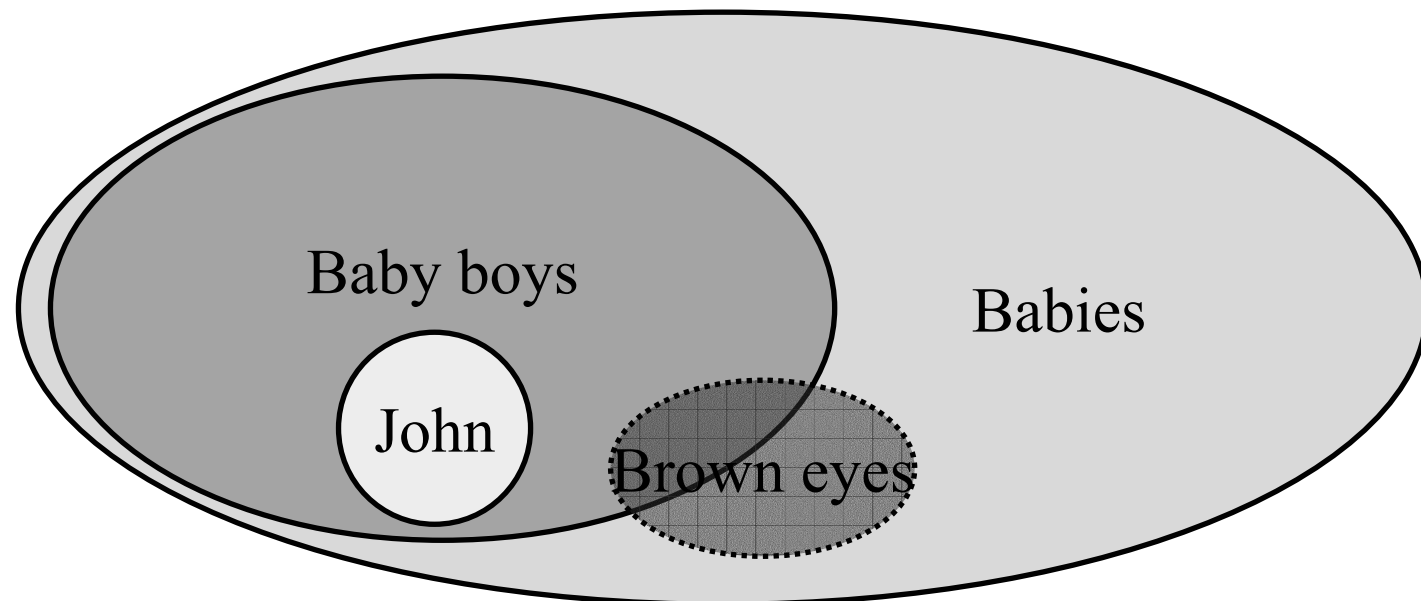
Avec des ensembles...

- $P(X)$ signifie “probabilité que X soit vrai”
 - $P(\text{baby is a boy}) \approx 0.5$ (% of total that are boys)
 - $P(\text{baby is named John}) \approx 0.001$ (% of total named John)



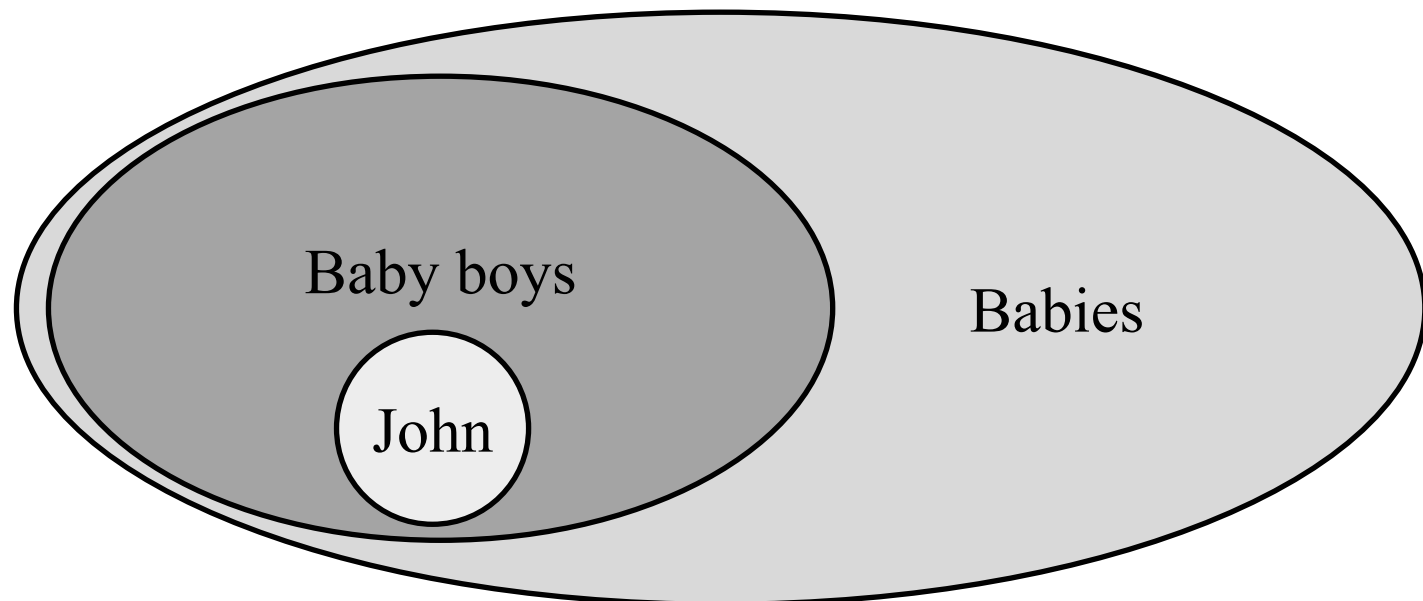
Avec des ensembles...

- $P(X, Y)$ signifie “probabilité que X et Y soient vrais tous les deux”
 - e.g. $P(\text{brown eyes, boy})$



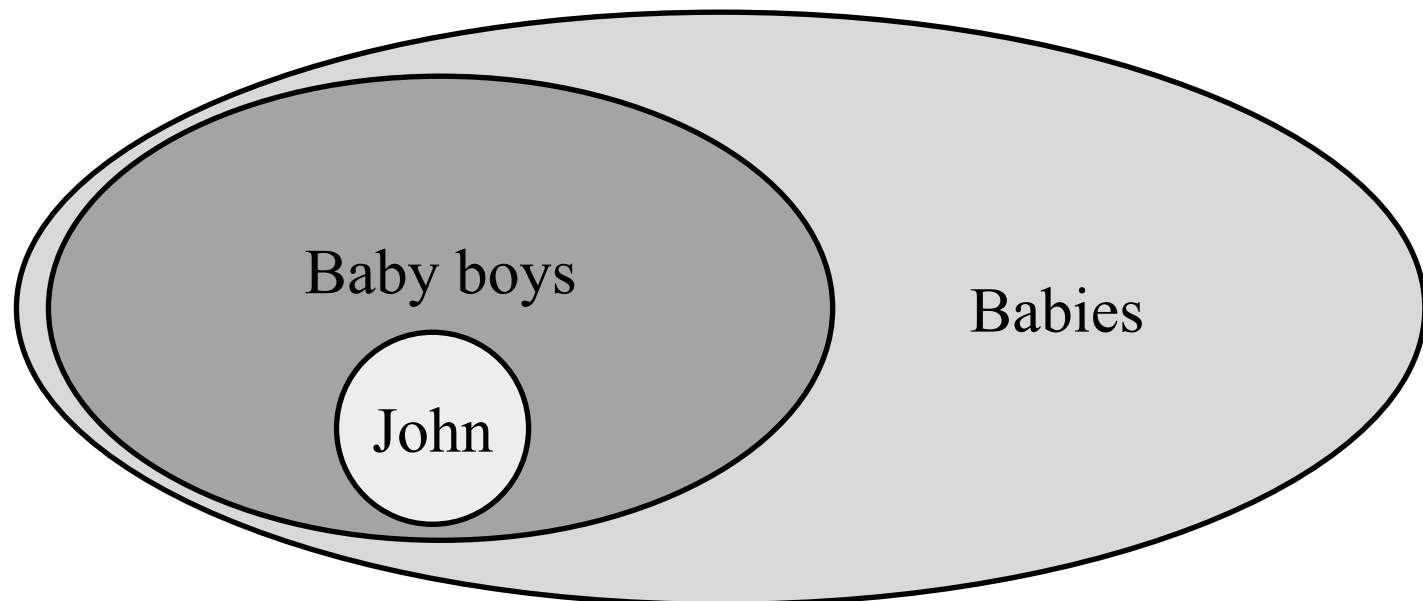
Avec des ensembles...

- $P(X|Y)$ signifie “probabilité que X soit vrai quand on sait déjà que Y est vrai”
 - $P(\text{baby is named John} \mid \text{baby is a boy}) \approx 0.002$
 - $P(\text{baby is a boy} \mid \text{baby is named John}) \approx 1$



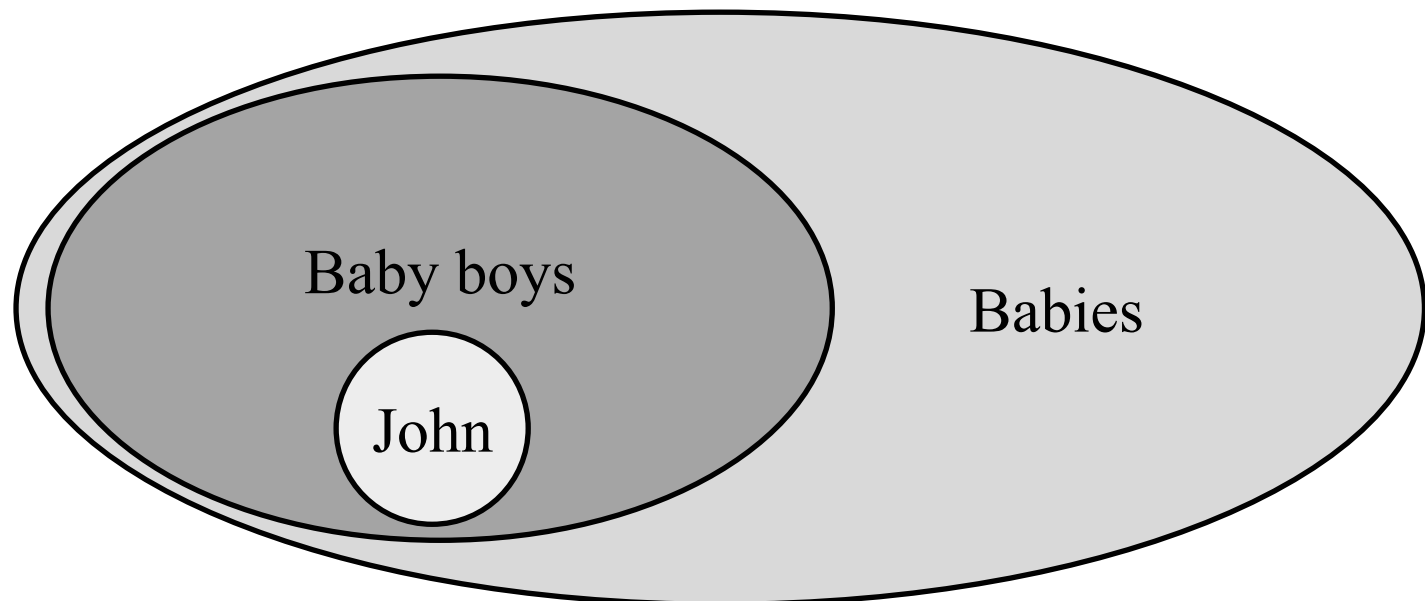
Avec des ensembles...

- $P(X|Y) = P(X, Y) / P(Y)$
 - $P(\text{baby is named John} \mid \text{baby is a boy}) =$
 $P(\text{baby is named John, baby is a boy}) /$
 $P(\text{baby is a boy}) = 0.001 / 0.5 = 0.002$



Avec des ensembles...

- Règle de Bayes : $P(X|Y) = P(Y|X) \times P(X) / P(Y)$
- $P(\text{named John} | \text{boy}) = P(\text{boy} | \text{named John}) \times P(\text{named John}) / P(\text{boy})$



Esperance

- Esperance d'une variable aléatoire X

$$E(X) = \sum_x p(x) x$$

- Exemple du dé à 6 faces

$$E(X) = \frac{1}{6} \times 1 + \frac{1}{6} \times 2 + \frac{1}{6} \times 3 + \frac{1}{6} \times 4 + \frac{1}{6} \times 5 + \frac{1}{6} \times 6 = 3.5$$

Variance

- Variance

$$\begin{aligned} \text{Var}(X) &= E((X - E(X))^2) = E(X^2) - E^2(X) \\ \text{Var}(X) &= \sum_x p(x) (x - E(X))^2 \end{aligned}$$

- Ecart type σ

$$\begin{aligned} \text{Var}(X) &= \sigma^2 \\ E(X) &= \mu \end{aligned}$$

Variance

$$\begin{aligned} \text{Var}(X) &= \frac{1}{6}(1 - 3.5)^2 + \frac{1}{6}(2 - 3.5)^2 + \frac{1}{6}(3 - 3.5)^2 \\ &\quad + \frac{1}{6}(4 - 3.5)^2 + \frac{1}{6}(5 - 3.5)^2 + \frac{1}{6}(6 - 3.5)^2 \\ &= \frac{1}{6}((-2.5)^2 + (-1.5)^2 + (-0.5)^2 + 0.5^2 + 1.5^2 + 2.5^2) \\ &= \frac{1}{6}(6.25 + 2.25 + 0.25 + 0.25 + 2.25 + 6.25) \\ &= 2.917 \end{aligned}$$

Distributions

- Uniforme
 - Tous les événements sont équiprobables
 - $p(x)=p(y)$ pour tout x,y
 - Ex : dé
- Binomiale
 - Une série de tirages avec résultat binaire (ex: succès/échec)
 - Probabilité des événements $p/1-p$; la probabilité d'avoir r succès sur n tirages est

$$b(r; n, p) = \binom{n}{r} p^r (1 - p)^{n-r}$$

Distributions

- Normale (ou gaussienne)
 - Ex : taille des personnes, QI, hauteur des arbres
 - Étant donné une espérance et un écart type, la distribution gaussienne est

$$n(x; \mu, \sigma) = \frac{1}{\sqrt{2\pi\mu}} e^{-\frac{(x-\mu)^2}{2\sigma^2}}$$

!!Erreur : sigma au lieu de mu au dénominateur

Estimation bayésienne

- Modèle M , données D
 - Quel est le modèle le plus probable étant donné les données ?
=> $p(M/D)$

$$p(M|D) = \frac{p(D|M) p(M)}{p(D)}$$

$$\operatorname{argmax}_M p(M|D) = \operatorname{argmax}_M p(D|M) p(M)$$

- $p(M)$: probabilité a priori du modèle
- L'estimation d'un modèle $p(w)$ avec les fréquences des mots correspond à une estimation bayésienne avec une probabilité a priori uniforme (estimation par **maximum de vraisemblance**)

Entropie

- Important concept qui mesure le « degré de désordre »

$$H(X) = \sum_x -p(x) \log_2 p(x)$$

- Exemples

- 1 événement : $P(a)=1$ $\Rightarrow H(X) = -\log_2 1 = 0$
- 2 événements équiprobables : $p(a)=0.5$; $p(b)=0.5$
 - $H(X) = -0.5 \log_2 0.5 - 0.5 \log_2 0.5 = 1$
- 4 événements équiprobables : $H(X) = 2$

Entropie

- 4 évènements dont 1 plus probable que les autres
 - $p(a)=0.7$; $p(b)=0.1$; $p(c)=0.1$; $p(d)=0.1$

$$\begin{aligned}H(X) &= -0.7 \log_2 0.7 - 0.1 \log_2 0.1 \\ &\quad - 0.1 \log_2 0.1 - 0.1 \log_2 0.1 \\ &= -0.7 \log_2 0.7 - 0.3 \log_2 0.1 \\ &= -0.7 \times -0.5146 - 0.3 \times -3.3219 \\ &= 0.36020 + 0.99658 \\ &= 1.35678\end{aligned}$$

Entropie

- Intuition : un bon modèle doit avoir une entropie faible...
- Beaucoup de modèles probabilistes en traitement du langage consistent en une réduction d'entropie

Théorie de l'information et entropie

- Supposons que nous voulons encoder une séquence d'événements X
- Chaque événement est encodé par une séquence de bits
- Exemples
 - Pièce : pile=0 ; face=1
 - 4 événements équiprobables : a=00 ; b=01 ; c=10 ; d=11
 - Codage de huffmann (nombre de bits faible si lettre fréquente)
- **Le nombre de bits nécessaires pour coder les événements de X est inférieur ou égal à l'entropie de X**

Méthodes empiriques pour le traitement automatique du langage naturel

Laurent Besacier

2. Modélisation probabiliste du langage

Vers des modèles de langue

- Nous avons déjà parlé de la probabilité d'un mot $p(w)$
- Mais les mots apparaissent en séquences
- Etant donné une séquence de mots, pouvons nous prévoir le mot suivant ?

$$p(w_n | w_1, \dots, w_{n-1})$$

Modèles de langue

- Etant donnée une chaîne de mots
 $W = w_1 w_2 w_3 w_4 \dots w_n$

$$p(w_1, w_2, w_3, \dots, w_n) = p(w_1) p(w_2|w_1) p(w_3|w_1, w_2) \dots p(w_n|w_1, w_2, \dots, w_{n-1})$$

- Hypothèse de Markov
 - Seule l'historique sur un passé limité compte
 - Seuls les k mots précédents appartiennent à l'historique
 - Modèle d'ordre k
- Exemple : modèle d'ordre 1 (bigramme)

$$p(w_1, w_2, w_3, \dots, w_n) = p(w_1) p(w_2|w_1) p(w_3|w_2) \dots p(w_n|w_{n-1})$$

Estimer les probabilités n-grammes

- Collecter des fréquences de mots et de séquences de mots sur de très grands corpus
- Plusieurs millions de mots

$$p(w_2|w_1) = \frac{\textit{count}(w_1, w_2)}{\textit{count}(w_1)}$$

Taille du modèle

- Pour chaque n-gramme, on doit stocker une probabilité
- Si on suppose un vocabulaire de 20000 mots

Model	Max. number of parameters
0th order (unigram)	20,000
1st order (bigram)	$20,000^2 = 400$ million
2nd order (trigram)	$20,000^3 = 8$ trillion
3rd order (4-gram)	$20,000^4 = 160$ quadrillion

- Les modèles de langage trigrammes sont les plus utilisés

Exemple pratique

- A partir d'un corpus de 275 millions de mots (journaux écrits en anglais type « *Wall Street Journal* »)

1-gram	716,706
2-gram	12,537,755
3-gram	22,174,483

Visualisation

Voir fichiers .pdf

Qualité d'un modèle de langue

- Peut être mesuré avec l'entropie

$$H(W_1^n) = \lim_{n \rightarrow \infty} -\frac{1}{n} \log p(W_1^n)$$

- Ou la perplexité

$$\text{perplexity}(W) = 2^{H(W)}$$

Exemple : unigramme

- Training set

there is a big house
i buy a house
they buy the new house

- Model

$p(\textit{there}) = 0.0714$	$p(\textit{is}) = 0.0714$	$p(\textit{a}) = 0.1429$
$p(\textit{big}) = 0.0714$	$p(\textit{house}) = 0.2143$	$p(\textit{i}) = 0.0714$
$p(\textit{buy}) = 0.1429$	$p(\textit{they}) = 0.0714$	$p(\textit{the}) = 0.0714$
$p(\textit{new}) = 0.0714$		

- Test sentence S : *they buy a big house*

- $p(S) = \underbrace{0.0714}_{\textit{they}} \times \underbrace{0.1429}_{\textit{buy}} \times \underbrace{0.0714}_{\textit{a}} \times \underbrace{0.1429}_{\textit{big}} \times \underbrace{0.2143}_{\textit{house}} = 0.0000231$

Exemple : bigramme

- Training set

there is a big house
i buy a house
they buy the new house

- Model

$p(\text{big} \text{a}) = 0.5$	$p(\text{is} \text{there}) = 1$	$p(\text{buy} \text{they}) = 1$
$p(\text{house} \text{a}) = 0.5$	$p(\text{buy} \text{i}) = 1$	$p(\text{a} \text{buy}) = 0.5$
$p(\text{new} \text{the}) = 1$	$p(\text{house} \text{big}) = 1$	$p(\text{the} \text{buy}) = 0.5$
$p(\text{a} \text{is}) = 1$	$p(\text{house} \text{new}) = 1$	$p(\text{they} \langle s \rangle) = .333$

- Test sentence S : *they buy a big house*

- $p(S) = \underbrace{0.333}_{\text{they}} \times \underbrace{1}_{\text{buy}} \times \underbrace{0.5}_{\text{a}} \times \underbrace{0.5}_{\text{big}} \times \underbrace{1}_{\text{house}} = 0.0833$

Problème des événements inconnus

- Soit la phrase S_2
 - they buy a new house.
- Le bigramme « a new » n'a jamais été vu
 - $P(S_2)=0$!!
 - Alors que la phrase est correcte

Problème des événements inconnus

- Deux types de « zéros »
 - Mots inconnus
 - Problème traité avec une étiquette « INCONNU »
 - La probabilité $p(\text{INCONNU})$ est estimée
 - Tendence à une sur-estimation de cette probabilité
 - » Mécanismes de lissage
 - N-grammes inconnus
 - Lissage en leur donnant une faible probabilité (mais non nulle !)
 - Retour en arrière (backoff) vers un n-gramme d'ordre inférieur
 - Donner une probabilité non nulle à des événements non rencontrés
 - => ce n'est plus une estimation par maximum de vraisemblance !!

Exemple d'application

- Détection de la langue d'un texte (voir TP)
- Reponctuation automatique et restauration de la casse (voir TP)

Restauration de la casse

- Re-ponctuation
- Exemple
 - Restauration de l'entrée « hal 9000 »
- Hypothèse : les variantes de chaque entrée ont été collectées sur un ensemble d'apprentissage

l_w	c_w	$P_m(c_w l_w)$
hal	HAL	$\frac{1}{2}$
hal	Hal	$\frac{1}{4}$
hal	hal	$\frac{1}{4}$
9000	9000	1

Restauration de la casse

- Les hypothèses de restauration sont les suivantes
 - 1. C1 = "HAL 9000"
 - 2. C2 = "Hal 9000"
 - 3. C3 = "hal 9000"
- La séquence restaurée la plus probable est définie par

$$\hat{C} = \arg \max_{C_i} P_{\theta}(C_i) \times \prod_{w \in C_i} P_m(c_w | l_w)$$

Restauration de la casse

- Si on veut calculer le score de $C1 = \text{"HAL 9000"}$

$$P_{\theta}(\text{"HAL 9000"}) \times P_m(\text{HAL} \mid \text{hal}) \times P_m(9000 \mid 9000)$$

- $P(\text{"HAL 9000"})$ peut être calculé avec un modèle de langue classique (n-gramme)
- Si on ne connaît pas P_m , on peut considérer toutes les alternatives comme equiprobables

Restauration de la casse

- Algorithme
- Même problème de complexité que dans le cas de l'étiquetage
 - Algorithme de Viterbi pour calculer

$$\hat{C} = \arg \max_{C_i} P_{\theta}(C_i) \times \prod_{w \in C_i} P_m(c_w | l_w)$$

- Outil *disambig* de la boîte à outils SR† LM

En pratique

- Créer un modèle P_m de la forme suivante

```
w1 w11 p11 w12 p12 ... w1n p1n  
w2 w21 p21 w22 p22 ... w2m p2m  
...
```

$w11, w12, \dots, w1n$ are words from the training data. $w1$ is the lowercase form of $w11, w12, \dots, w1n$ and $p11$ is the probability $P_m(w11 | w1)$ which is computed as follows:

$$p11 = P_m(w11 | w1) = \frac{freq(w11)}{freq(w11) + freq(w12) + \dots + freq(w1n)}$$

En pratique

- Exemple

```
trump Trump 0.975 trump 0.025
```

```
isabella Isabella 1
```

```
uncommitted uncommitted 1
```

```
alberto-culver Alberto-Culver 1
```

Reponctuation

- Hidden-ngram model [Stolcke & Shriberg 1996]
- <http://www.asel.udel.edu/icslp/cdrom/vol2/715/a715.pdf>

$$P(e_1, \dots, e_T) \approx \prod_{t=1}^T P(e_t \mid e_{t-1}, \dots, e_{t-n+1})$$

Ensemble d'événements E: mots et marques de ponctuation

Reponctuation

- Soit une phrase source w_1, w_2, \dots, w_n
- On génère toutes les séquences possibles incluant les différentes marques de ponctuation
 - Chaînes de caractères
 - Automates d'état finis
- La ponctuation trouvée correspond à la chaîne la plus probable selon

$$P(e_1, \dots, e_T) \approx \prod_{t=1}^T P(e_t | e_{t-1}, \dots, e_{t-n+1})$$

Méthodes empiriques pour le traitement automatique du langage naturel

Laurent Besacier

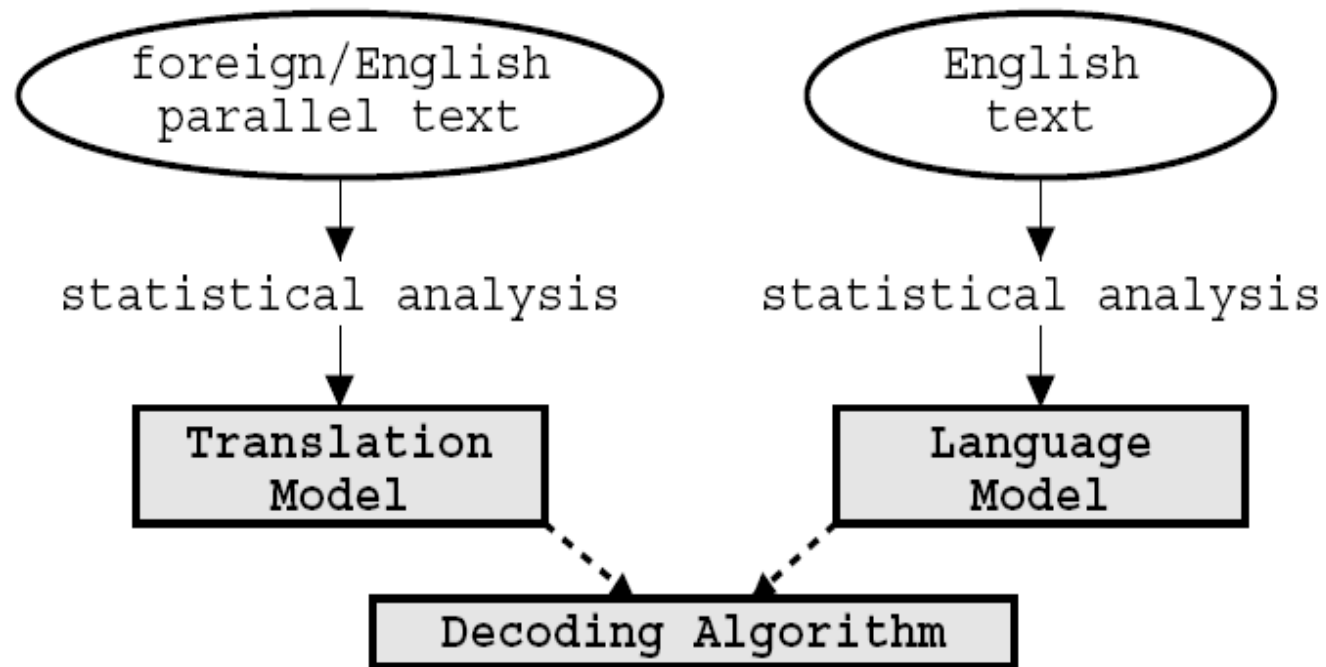
3. Introduction à la traduction automatique probabiliste

Quelques citations...

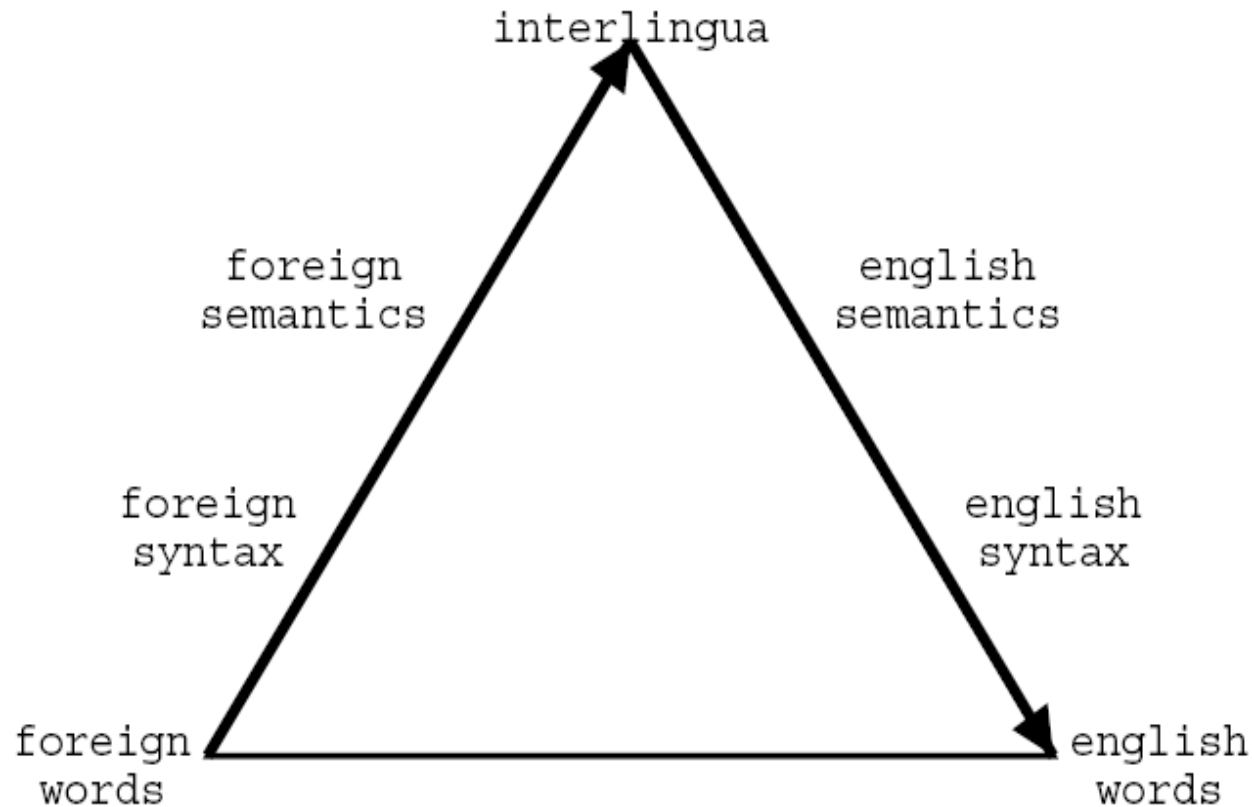
- ***It must be recognized that the notion "probability of a sentence" is an entirely useless one, under any known interpretation of this term. Noam Chomsky, 1969***
- ***Whenever I fire a linguist our system performance improves. Frederick Jelinek, 1988***

Traduction automatique statistique

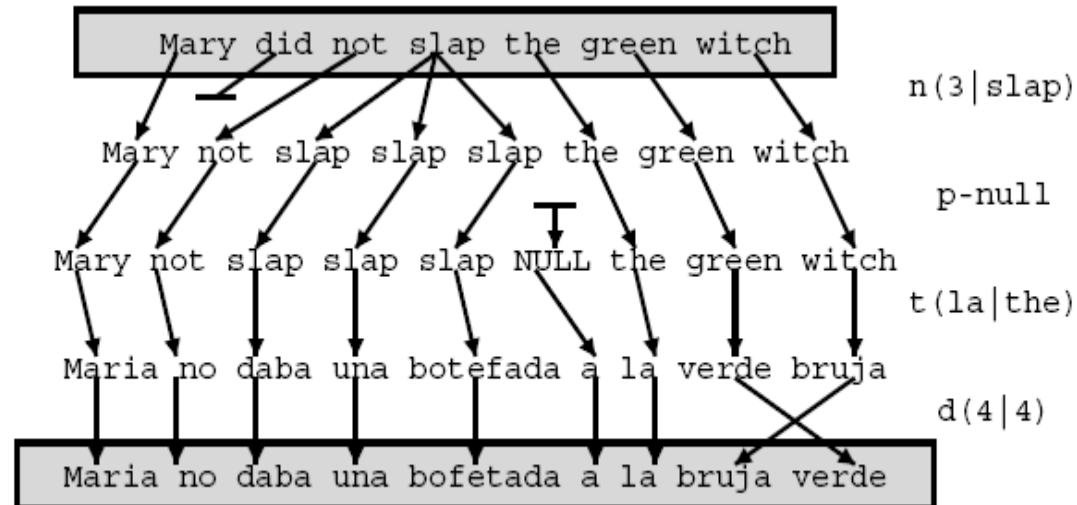
- Components: Translation model, language model, decoder



Le triangle de Vauquois



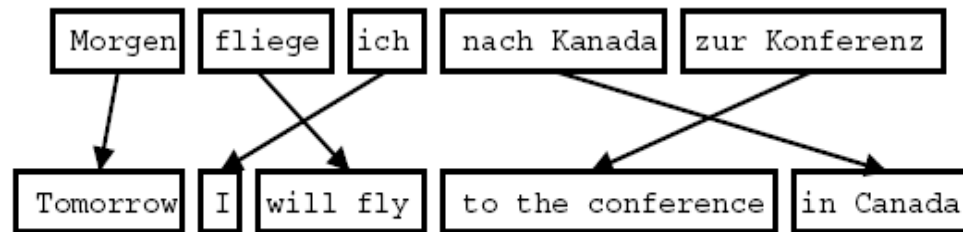
Méthodes à base de mots



[from Knight, 1997]

Premiers modèles pour la TA statistique [Brown et al., 1993]

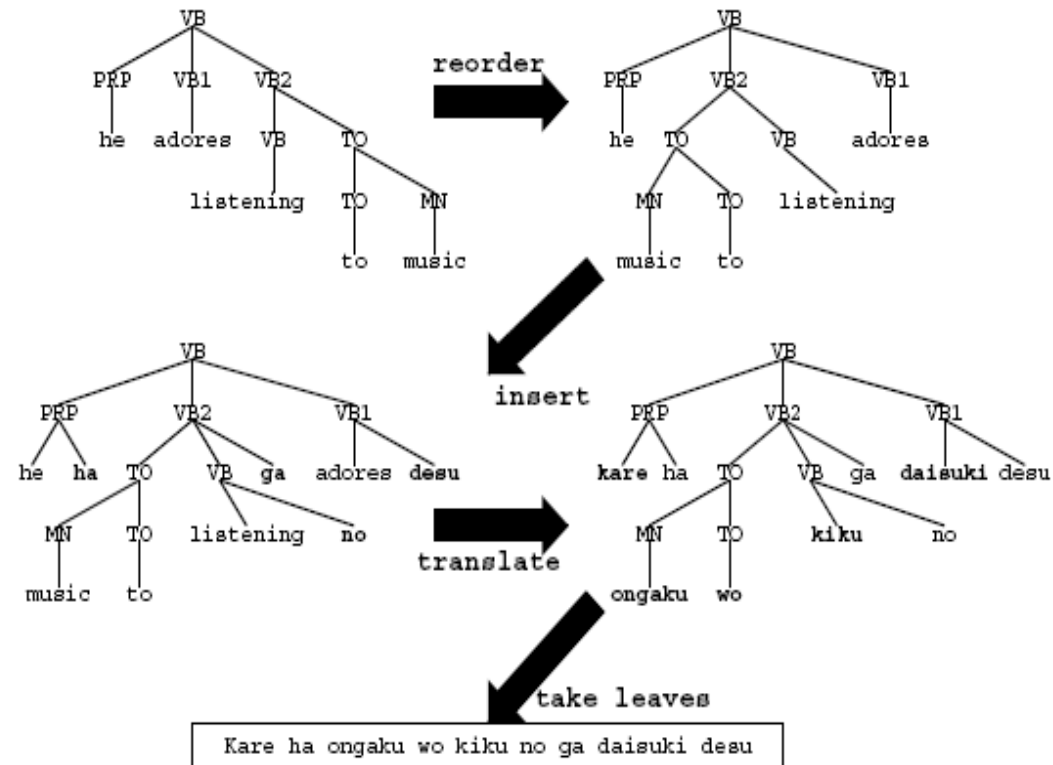
Méthodes à base de séquences



[from Koehn et al., 2003, NAACL]

- Une entrée est segmentée en séquences
- Chaque séquence est traduite
- Les séquences sont ré-ordonnées

Modèles fondés sur la syntaxe



[from Yamada and Knight, 2001]

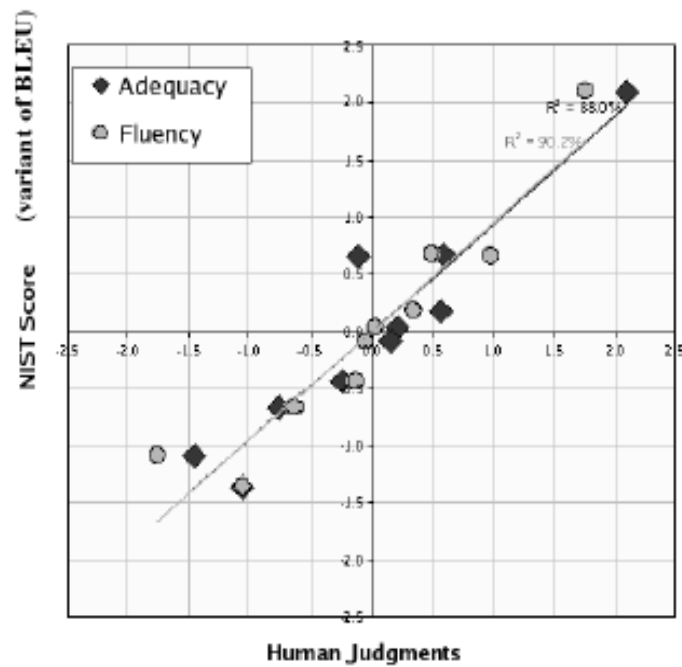
Evaluation automatique

- **Pourquoi une évaluation automatique ?**
 - Evaluation manuelle trop lente
 - Doit être faite sur de larges ensembles de test
 - Permet un tuning automatique des systèmes de TA
- **Historique**
 - Word Error Rate
 - BLEU depuis 2002
- **BLEU : Recouvrement avec des traductions de référence**

Evaluation automatique

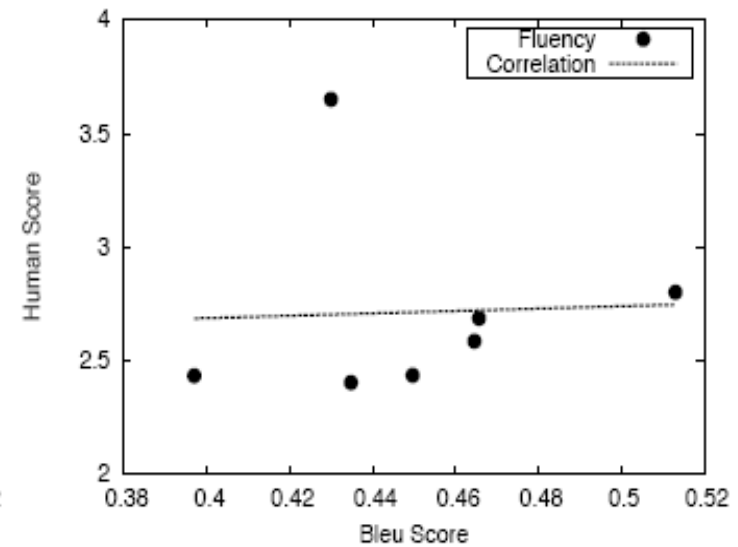
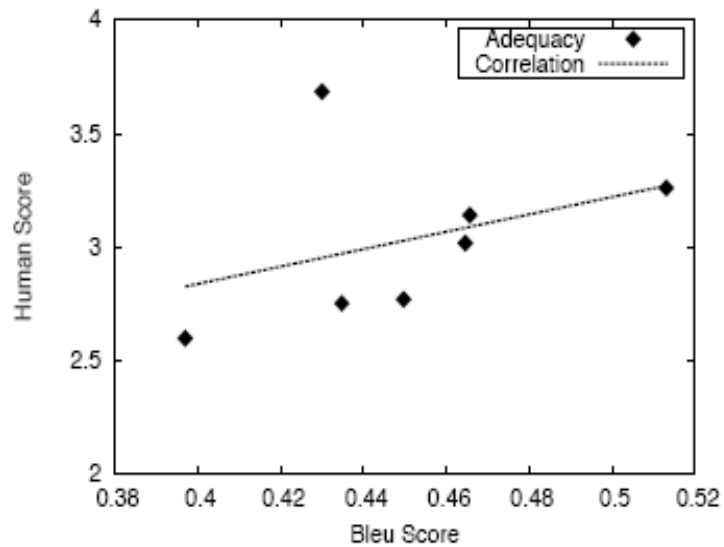
- **Reference Translation**
 - **the gunman was shot to death by the police**
 - the gunman was police kill .
 - wounded police jaya of
 - the gunman was shot dead by the police .
 - the gunman arrested by police kill .
 - the gunmen were killed .
 - the gunman was shot to death by the police .
 - gunmen were killed by police ?SUB>0 ?SUB>0
 - al by the police .
 - the ringer is killed by the police .
 - police killed the gunman .
- vert : 4-gram match => bien
- Rouge : no match => mauvais

Correlation ou pas avec des jugements humains



[from George Doddington, NIST]

Correlation ou pas avec des jugements humains



[from Callison-Burch et al., 2006, EACL]

Campagnes d'évaluation

- **NIST/DARPA:**
 - Campagnes annuelles : Arabic-English, Chinese-English, nouvelles, depuis 2001
- **IWSLT:**
 - Campagnes annuelles : Chinois, Japonais, Italien, Arabe => Anglais , domaine « tourisme » depuis 2003
- **WPT/WMT:**
 - Langues européennes , archives du parlement européen, depuis 2005

Euromatrix

- 110 systèmes !!

	da	de	el	en	es	fr	fi	it	nl	pt	sv
da	-	18.4	21.1	28.5	26.4	28.7	14.2	22.2	21.4	24.3	28.3
de	22.3	-	20.7	25.3	25.4	27.7	11.8	21.3	23.4	23.2	20.5
el	22.7	17.4	-	27.2	31.2	32.1	11.4	26.8	20.0	27.6	21.2
en	25.2	17.6	23.2	-	30.1	31.1	13.0	25.3	21.0	27.1	24.8
es	24.1	18.2	28.3	30.5	-	40.2	12.5	32.3	21.4	35.9	23.9
fr	23.7	18.5	26.1	30.0	38.4	-	12.6	32.4	21.1	35.3	22.6
fi	20.0	14.5	18.2	21.8	21.1	22.4	-	18.3	17.0	19.1	18.8
it	21.4	16.9	24.8	27.8	34.0	36.0	11.0	-	20.0	31.2	20.2
nl	20.5	18.3	17.4	23.0	22.9	24.6	10.3	20.0	-	20.7	19.0
pt	23.2	18.2	26.4	30.1	37.9	39.0	11.9	32.0	20.2	-	21.9
sv	30.3	18.9	22.8	30.2	28.6	29.7	15.3	23.9	21.9	25.9	-

[from Koehn, 2005: Europarl]

Traduire depuis/vers une langue

Language	From	Into	Diff
da	23.4	23.3	0.0
de	22.2	17.7	-4.5
el	23.8	22.9	-0.9
en	23.8	27.4	+3.6
es	26.7	29.6	+2.9
fr	26.1	31.1	+5.1
fi	19.1	12.4	-6.7
it	24.3	25.4	+1.1
nl	19.7	20.7	+1.1
pt	26.1	27.0	+0.9
sv	24.8	22.1	-2.6

[from Koehn, 2005: Europarl]

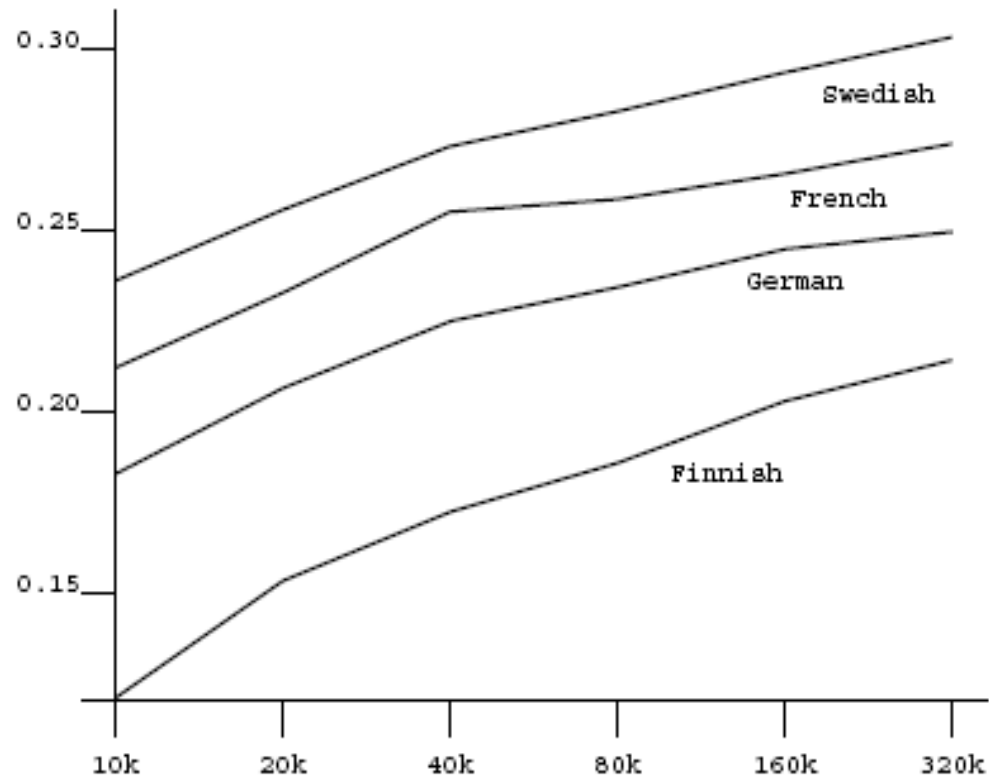
Difficile de traduire vers les langues à riche morphologie
(Allemand, Finnois)

Introduction au TP...

Données disponibles

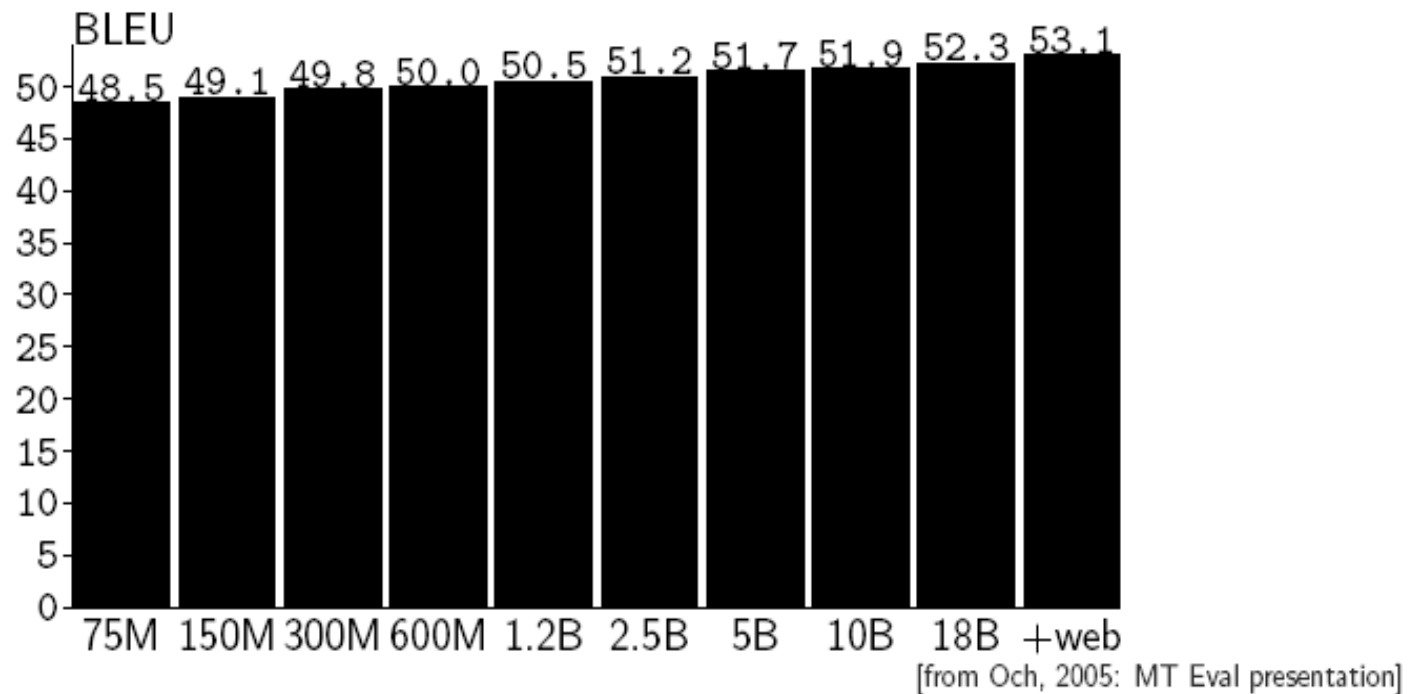
- **Corpus parallèles**
 - **Europarl: 30 millions de mots en 11 langues**
<http://www.statmt.org/europarl/>
 - **Acquis Communautaire: 8-50 million mots (20 langues EU)**
 - **Canadian Hansards: 20 million mots**
 - **Chinese/Arabic to English: plus de 100 million mots (LDC)**
- **Corpus monolingues**
 - **2.8 milliards de mots (Anglais, LDC)**
 - **Web**

Plus de données parallèles...



[from Koehn, 2003: Europarl]

Plus de données en langue cible...



Exemple de sortie d'un système chinois/anglais

In the First Two Months Guangdong's Export of High-Tech Products 3.76 Billion US Dollars

Xinhua News Agency, Guangzhou, March 16 (Reporter Chen Jizhong) - The latest statistics show that between January and February this year, Guangdong's export of high-tech products 3.76 billion US dollars, with a growth of 34.8% and accounted for the province's total export value of 25.5%. The export of high-tech products bright spots frequently now, the Guangdong provincial foreign trade and economic growth has made important contributions. Last year, Guangdong's export of high-tech products 22.294 billion US dollars, with a growth of 31 percent, an increase higher than the province's total export growth rate of 27.2 percent; exports of high-tech products net increase 5.270 billion US dollars, up for the traditional labor-intensive products as a result of prices to drop from the value of domestic exports decreased.

In the Suicide explosion in Jerusalem

Xinhua News Agency, Jerusalem, March 17 (Reporter bell tsui flower nie Xiaoyang) - A man on the afternoon of 17 in Jerusalem in the northern part of the residents of rammed a bus near ignition of carry bomb, the wrongdoers in red-handed was killed and another nine people were slightly injured and sent to hospital for medical treatment.

Systemes de TA statistique à base de mots

- **Traduire un mot**
 - Dictionnaires bilingues
- **Exemple**
 - Haus — house, building, home, household, shell.
 - Traduction multiples

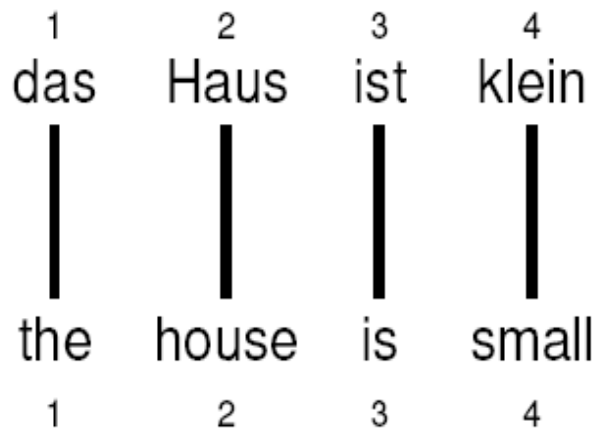
Collecter des statistiques

Translation of <i>Haus</i>	Count
<i>house</i>	8,000
<i>building</i>	1,600
<i>home</i>	200
<i>household</i>	150
<i>shell</i>	50

Maximum likelihood estimation
(Maximum de vraisemblance)

$$p_f(e) = \begin{cases} 0.8 & \text{if } e = \textit{house}, \\ 0.16 & \text{if } e = \textit{building}, \\ 0.02 & \text{if } e = \textit{home}, \\ 0.015 & \text{if } e = \textit{household}, \\ 0.005 & \text{if } e = \textit{shell}. \end{cases}$$

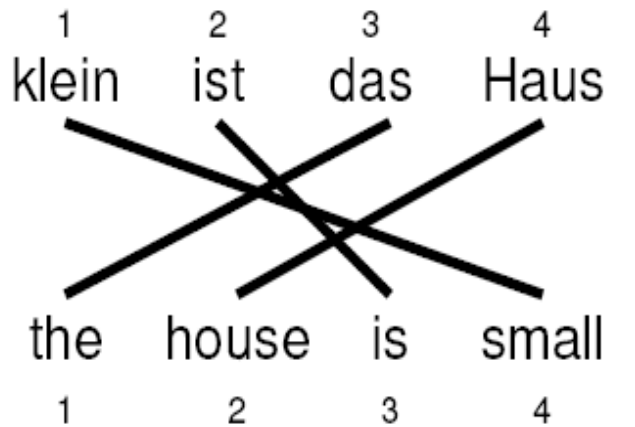
Alignement



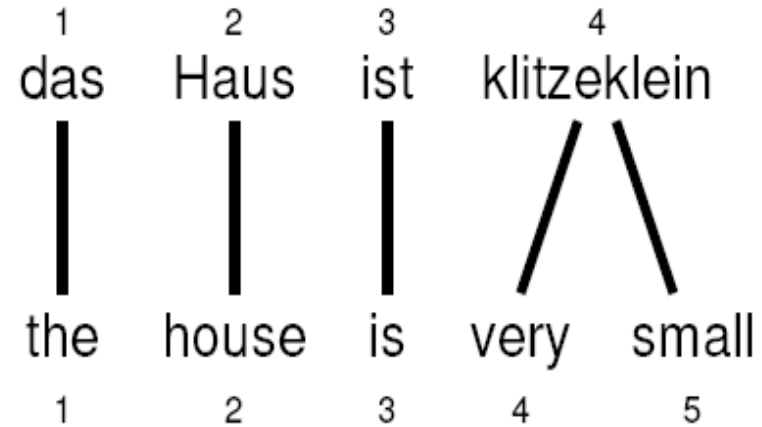
Fonction d'alignement : mot anglais cible en position i associé à mot étranger source en position j par la fonction $a: i \rightarrow j$

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

Exemples d'alignement (1)

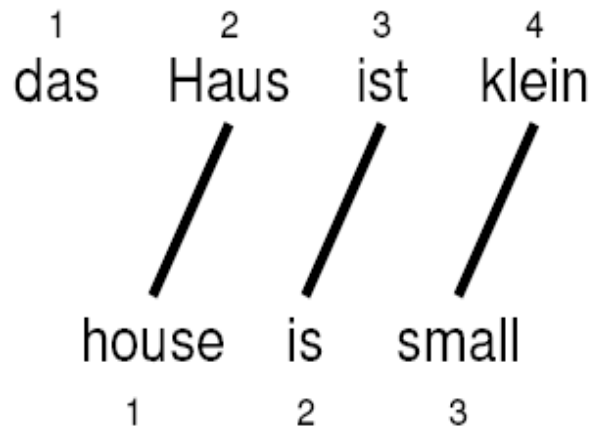


$a : \{1 \rightarrow 3, 2 \rightarrow 4, 3 \rightarrow 2, 4 \rightarrow 1\}$

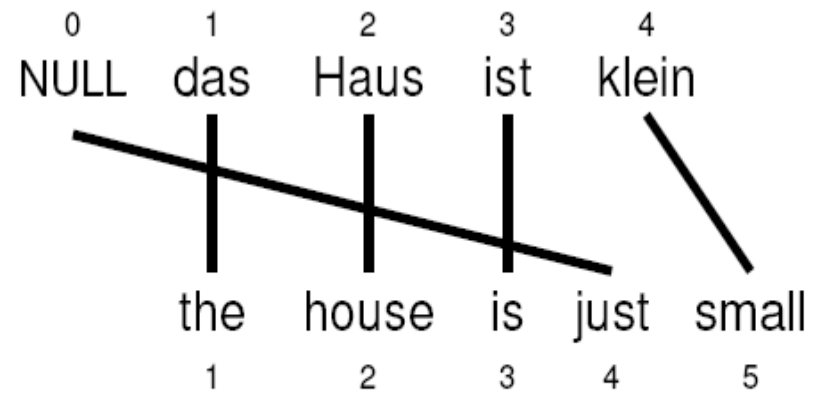


$a : \{1 \rightarrow 1, 2 \rightarrow 2, 3 \rightarrow 3, 4 \rightarrow 4, 4 \rightarrow 5\}$

Exemples d'alignement (2)



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



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

Modèle IBM-1

- *Generative model*: break up translation process into smaller steps
 - **IBM Model 1** only uses *lexical translation*
- Translation probability
 - for a foreign sentence $\mathbf{f} = (f_1, \dots, f_{l_f})$ of length l_f
 - to an English sentence $\mathbf{e} = (e_1, \dots, e_{l_e})$ of length l_e
 - with an alignment of each English word e_j to a foreign word f_i according to the alignment function $a : j \rightarrow i$

$$p(\mathbf{e}, a | \mathbf{f}) = \frac{\epsilon}{(l_f + 1)^{l_e}} \prod_{j=1}^{l_e} t(e_j | f_{a(j)})$$

- parameter ϵ is a *normalization constant*

Example

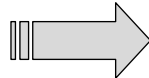
<i>das</i>		<i>Haus</i>		<i>ist</i>		<i>klein</i>	
<i>e</i>	$t(e f)$	<i>e</i>	$t(e f)$	<i>e</i>	$t(e f)$	<i>e</i>	$t(e f)$
<i>the</i>	0.7	<i>house</i>	0.8	<i>is</i>	0.8	<i>small</i>	0.4
<i>that</i>	0.15	<i>building</i>	0.16	<i>'s</i>	0.16	<i>little</i>	0.4
<i>which</i>	0.075	<i>home</i>	0.02	<i>exists</i>	0.02	<i>short</i>	0.1
<i>who</i>	0.05	<i>household</i>	0.015	<i>has</i>	0.015	<i>minor</i>	0.06
<i>this</i>	0.025	<i>shell</i>	0.005	<i>are</i>	0.005	<i>petty</i>	0.04

$$\begin{aligned} p(e, a|f) &= \frac{\epsilon}{4^3} \times t(\text{the}|\text{das}) \times t(\text{house}|\text{Haus}) \times t(\text{is}|\text{ist}) \times t(\text{small}|\text{klein}) \\ &= \frac{\epsilon}{4^3} \times 0.7 \times 0.8 \times 0.8 \times 0.4 \\ &= 0.0028\epsilon \end{aligned}$$

Apprendre un tel modèle (IBM-1)

- A partir d'un corpus parallèle
- Sans avoir les alignements

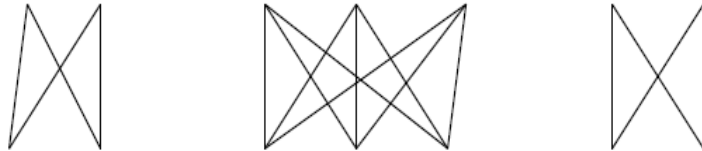
Algorithme EM



Algorithme EM

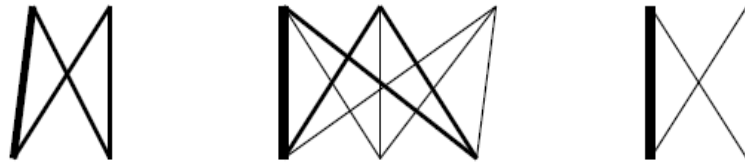
- Au départ tous les alignements sont équiprobables

... la maison ... la maison blue ... la fleur ...



... the house ... the blue house ... the flower ...

... la maison ... la maison blue ... la fleur ...



... the house ... the blue house ... the flower ...

... la maison ... la maison bleu ... la fleur ...



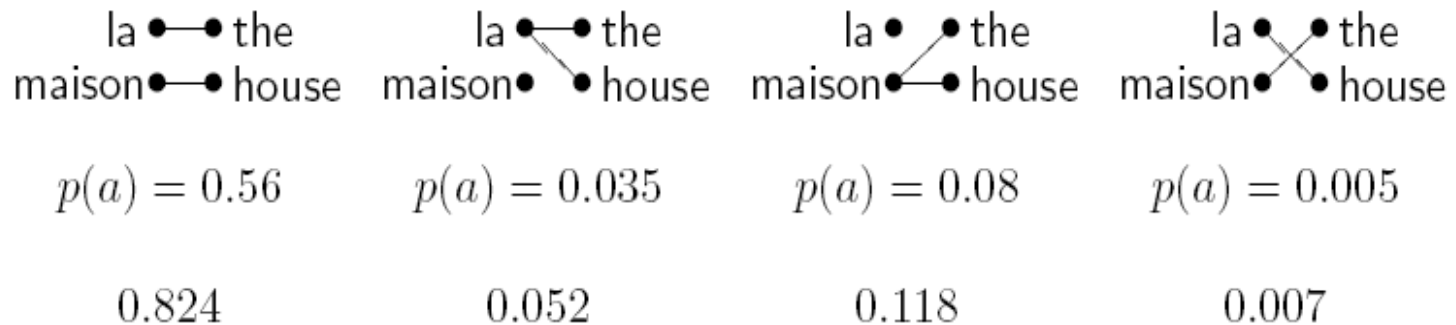
... the house ... the blue house ... the flower ...

Algorithme EM

- Probabilities

$$\begin{array}{ll}
 p(\text{the}|\text{la}) = 0.7 & p(\text{house}|\text{la}) = 0.05 \\
 p(\text{the}|\text{maison}) = 0.1 & p(\text{house}|\text{maison}) = 0.8
 \end{array}$$

- Alignments



- Counts

$$\begin{array}{ll}
 c(\text{the}|\text{la}) = 0.824 + 0.052 & c(\text{house}|\text{la}) = 0.052 + 0.007 \\
 c(\text{the}|\text{maison}) = 0.118 + 0.007 & c(\text{house}|\text{maison}) = 0.824 + 0.118
 \end{array}$$

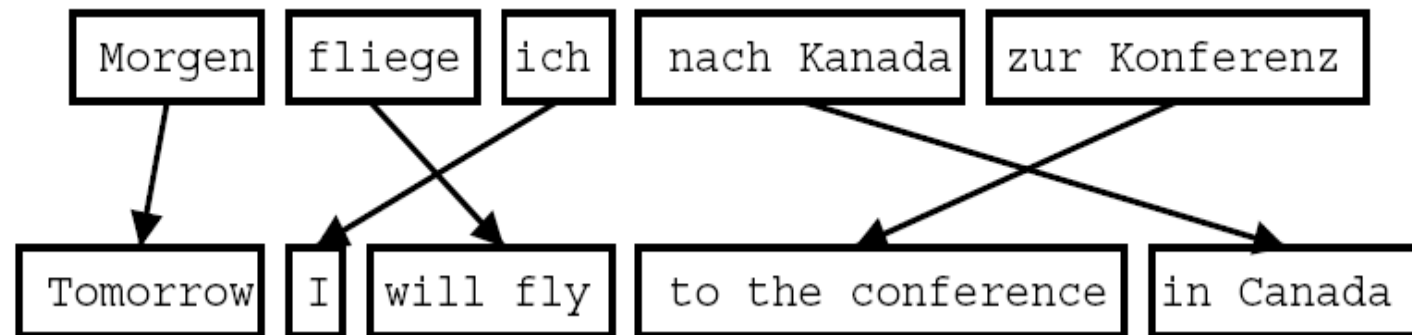
Pseudo-code

```
initialize  $t(e|f)$  uniformly
do
  set  $\text{count}(e|f)$  to 0 for all  $e, f$ 
  set  $\text{total}(f)$  to 0 for all  $f$ 
  for all sentence pairs  $(e\_s, f\_s)$ 
    for all words  $e$  in  $e\_s$ 
       $\text{total\_s} = 0$ 
      for all words  $f$  in  $f\_s$ 
         $\text{total\_s} += t(e|f)$ 
      for all words  $e$  in  $e\_s$ 
        for all words  $f$  in  $f\_s$ 
           $\text{count}(e|f) += t(e|f) / \text{total\_s}$ 
           $\text{total}(f) += t(e|f) / \text{total\_s}$ 
      for all  $f$  in  $\text{domain}(\text{total}(\cdot))$ 
        for all  $e$  in  $\text{domain}(\text{count}(\cdot|f))$ 
           $t(e|f) = \text{count}(e|f) / \text{total}(f)$ 
until convergence
```

Modèles suivants (IBM-2,3,4)

- IBM 2 intègre une loi de distorsion de la forme $p(a_j/j)$:
proba. que le mot cible f_j soit aligné à e_i avec $i=a_j$
- IBM 3 intègre une loi de fertilité qui modélise le nombre de mots dans la phrase cible connectés à un mot dans la phrase source
- IBM 4 intègre un modèle de distorsion plus fin
- Il existe un outil (GIZA++) pour entraîner les paramètres de ces modèles
 - Les paramètres d'un modèle permettent d'initialiser l'apprentissage du modèle suivant

Approche à base de séquences (*phrase-based approach*)



- Entrée en langue source segmentée en séquences (*phrase* en anglais)
- Chaque séquence est traduite en anglais
- Les séquences sont réordonnées

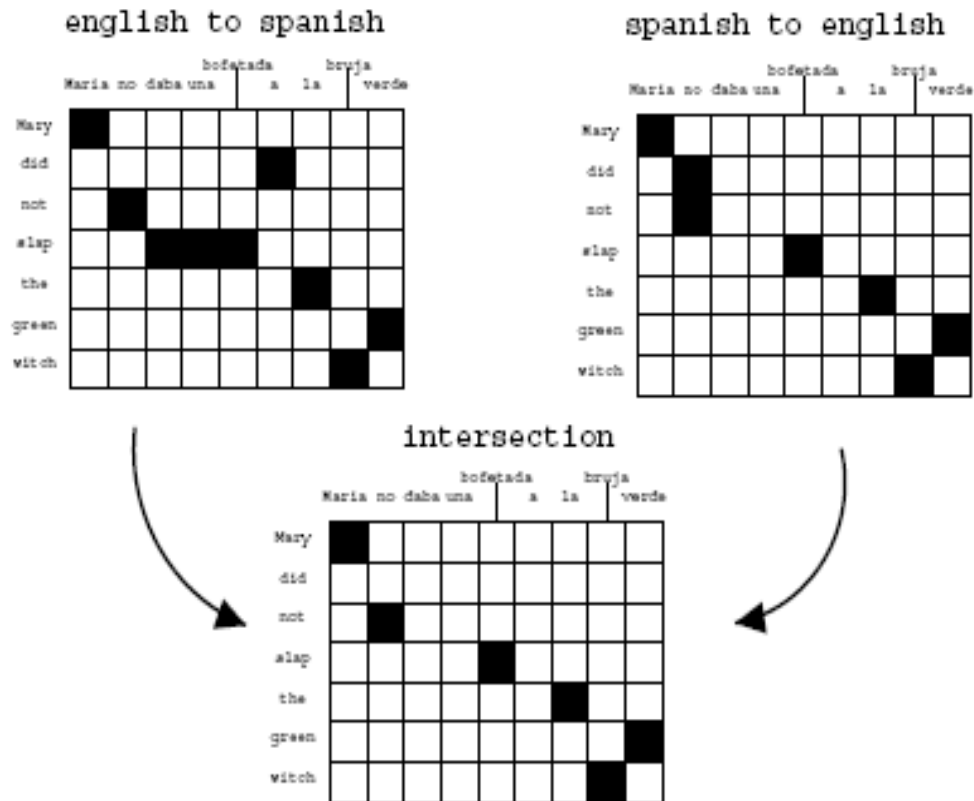
Table de traduction (*phrase table*)

- Table pour la séquence allemande « den Vorschlag »

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

Obtention de la table

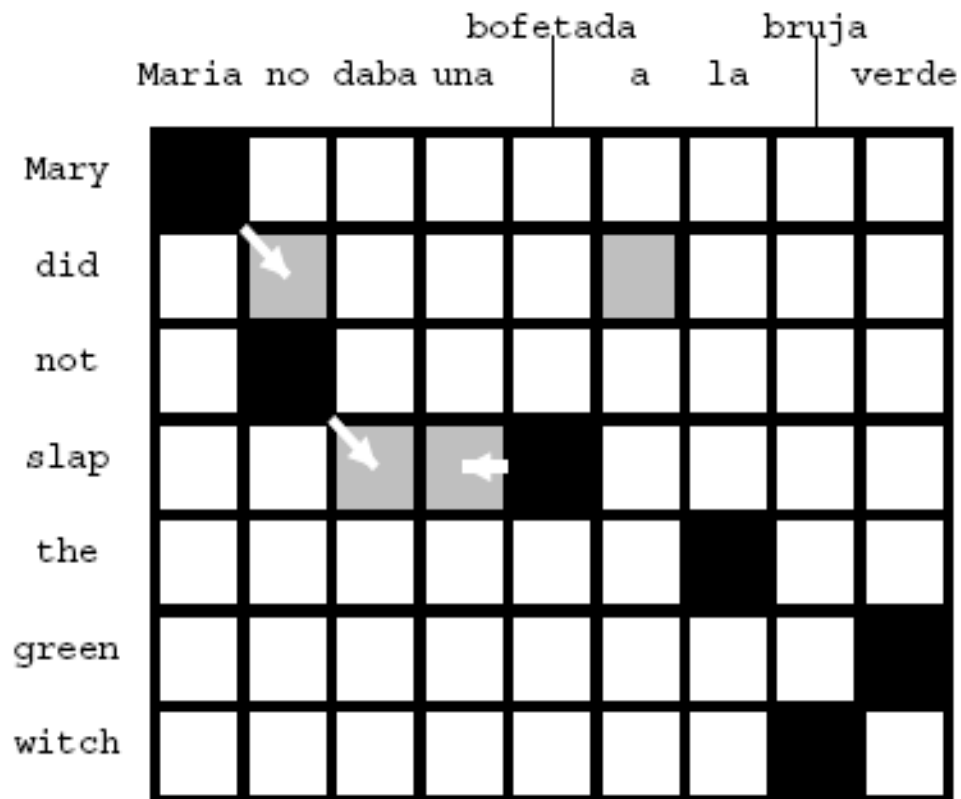
- Obtenue à partir d'alignements bidirectionnels de mots (obtenus avec GIZA++)



Obtention de la table

- Expansion

[Och and Ney, CompLing2003]



Expansion : algo

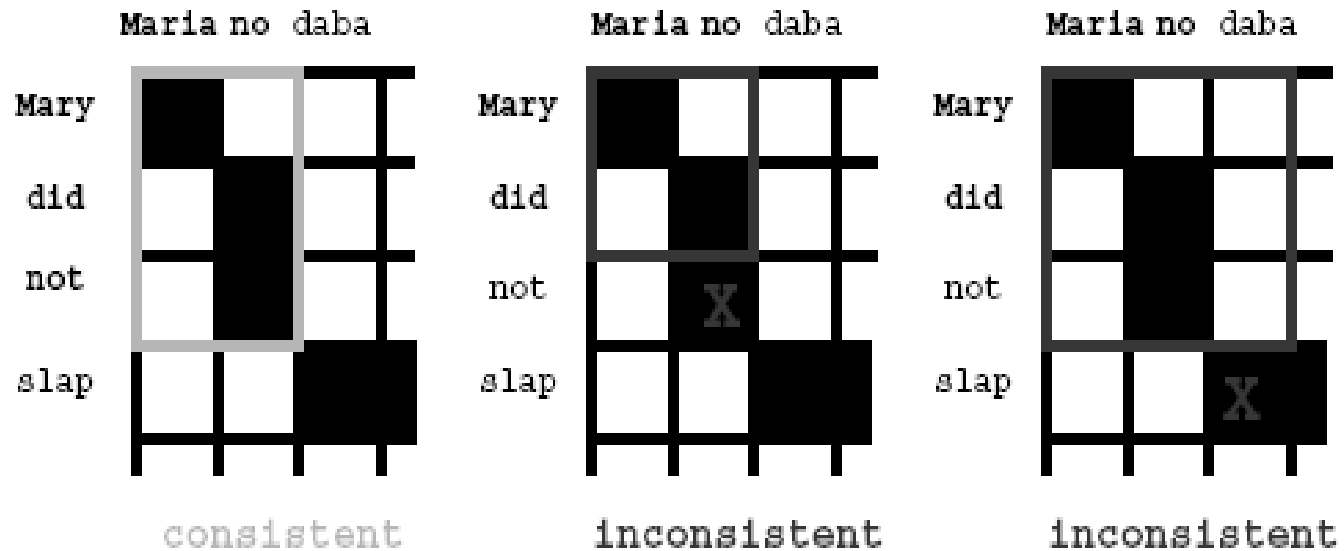
```
GROW-DIAG-FINAL(e2f,f2e):  
  neighboring = ((-1,0),(0,-1),(1,0),(0,1),(-1,-1),(-1,1),(1,-1),(1,1))  
  alignment = intersect(e2f,f2e);  
  GROW-DIAG(); FINAL(e2f); FINAL(f2e);
```

```
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 )
```

Obtention de la table

- Collecter les paires de séquences consistantes avec l'alignement

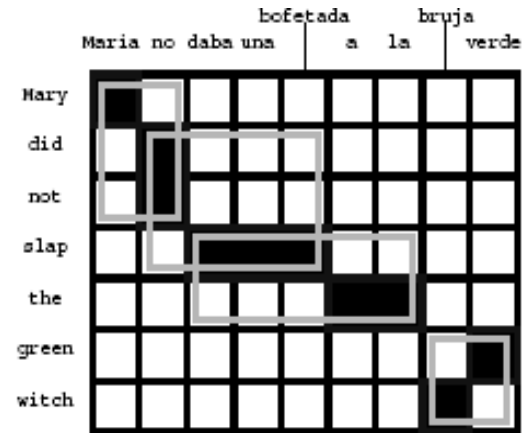


Obtention de la table

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

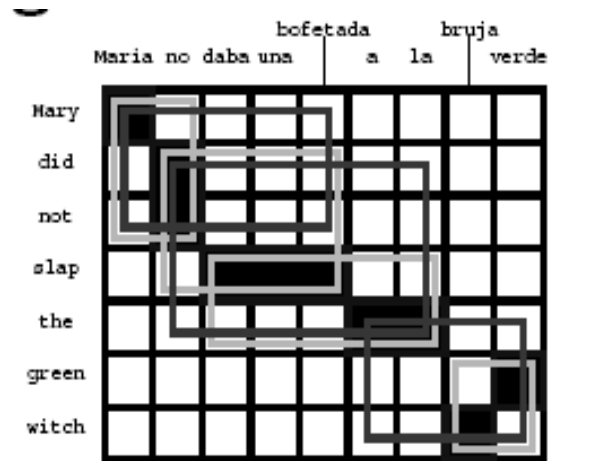
(Maria, Mary), (no, did not), (slap, daba una bofetada), (a la, the), (bruja, witch), (verde, green)

Obtention de la table



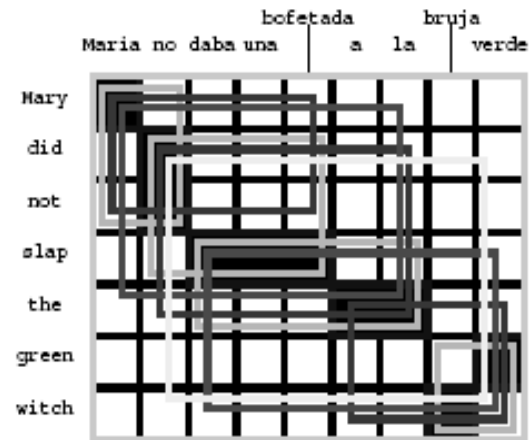
(Maria, Mary), (no, did not), (slap, daba una bofetada), (a la, the), (bruja, witch), (verde, green),
 (Maria no, Mary did not), (no daba una bofetada, did not slap), (daba una bofetada a la, slap the),
 (bruja verde, green witch)

Obtention de la table



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

Obtention de la table



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

Obtention de la table

$$\phi(\bar{f}|\bar{e}) = \frac{\text{count}(\bar{f},\bar{e})}{\sum_{\bar{f}} \text{count}(\bar{f},\bar{e})}$$

Calculer aussi. $\phi(\bar{e}|\bar{f})$

Garder aussi les probabilités lexicales (au niveau mot)

Modèle de traduction par séquences

- Major components of phrase-based model

- **phrase translation model** $\phi(\mathbf{f}|\mathbf{e})$

- **reordering model** $\omega^{\text{length}(\mathbf{e})}$

- **language model** $p_{\text{LM}}(\mathbf{e})$

- Bayes rule

$$\operatorname{argmax}_{\mathbf{e}} p(\mathbf{e}|\mathbf{f}) = \operatorname{argmax}_{\mathbf{e}} p(\mathbf{f}|\mathbf{e})p(\mathbf{e})$$

$$= \operatorname{argmax}_{\mathbf{e}} \phi(\mathbf{f}|\mathbf{e})p_{\text{LM}}(\mathbf{e})\omega^{\text{length}(\mathbf{e})}$$

- Sentence \mathbf{f} is decomposed into I phrases $\bar{f}_1^I = \bar{f}_1, \dots, \bar{f}_I$

- Decomposition of $\phi(\mathbf{f}|\mathbf{e})$

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

Modèles log-linéaires

- IBM Models provided mathematical justification for factoring *components* together

$$p_{LM} \times p_{TM} \times p_D$$

- These may be *weighted*

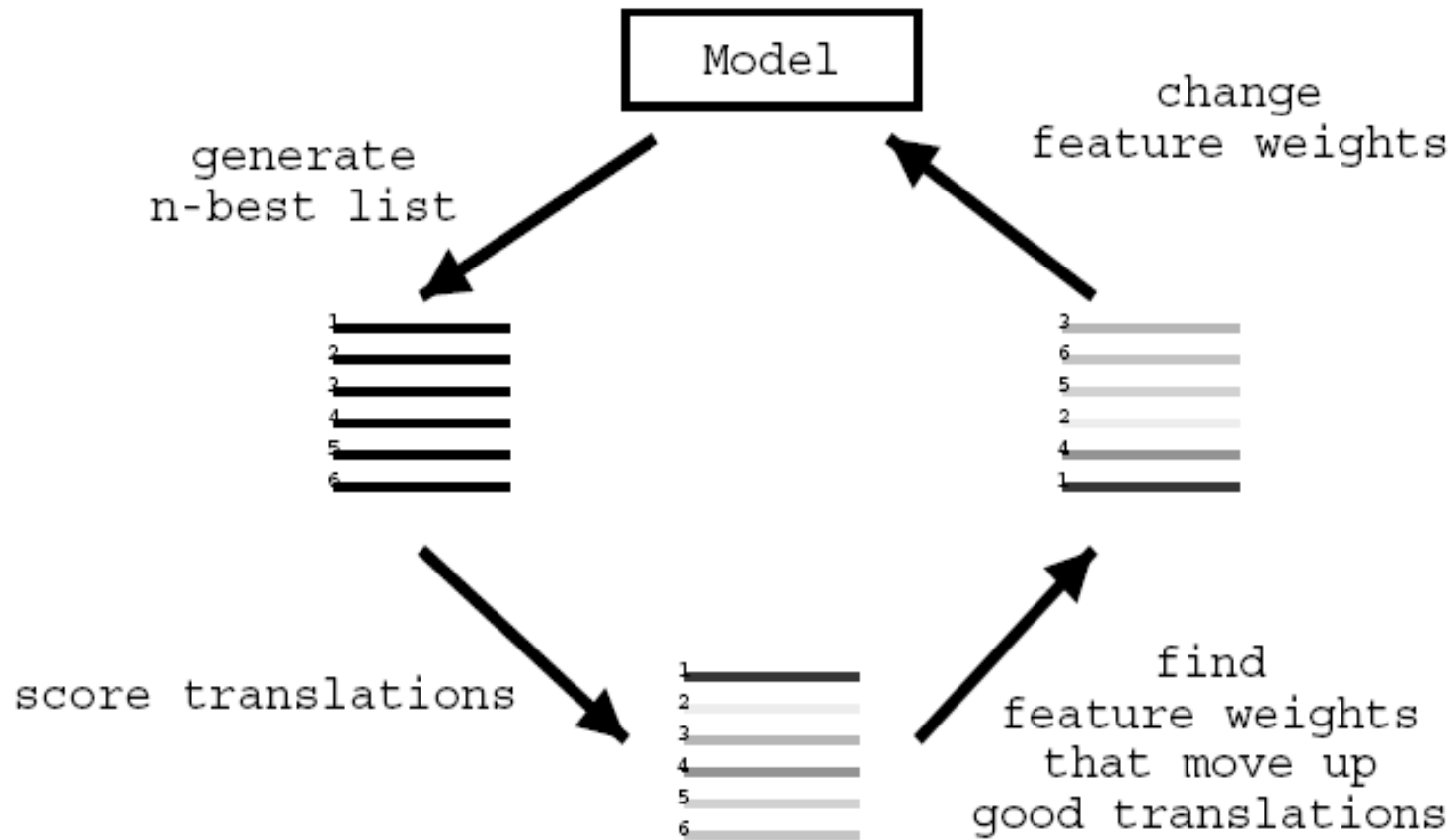
$$p_{LM}^{\lambda_{LM}} \times p_{TM}^{\lambda_{TM}} \times p_D^{\lambda_D}$$

- *Many components* p_i with weights λ_i

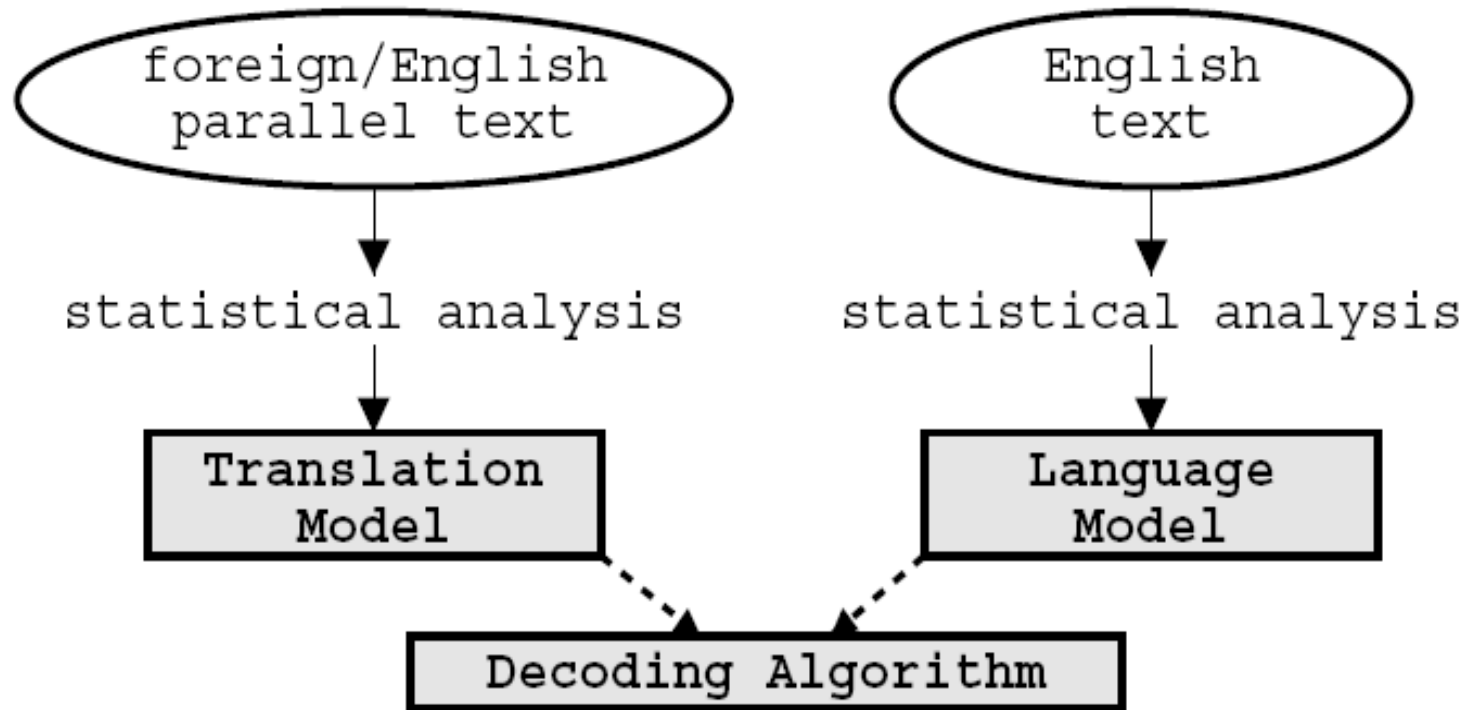
$$\Rightarrow \prod_i p_i^{\lambda_i} = \exp(\sum_i \lambda_i \log(p_i))$$

$$\Rightarrow \log \prod_i p_i^{\lambda_i} = \sum_i \lambda_i \log(p_i)$$

Apprendre les poids



Décodage



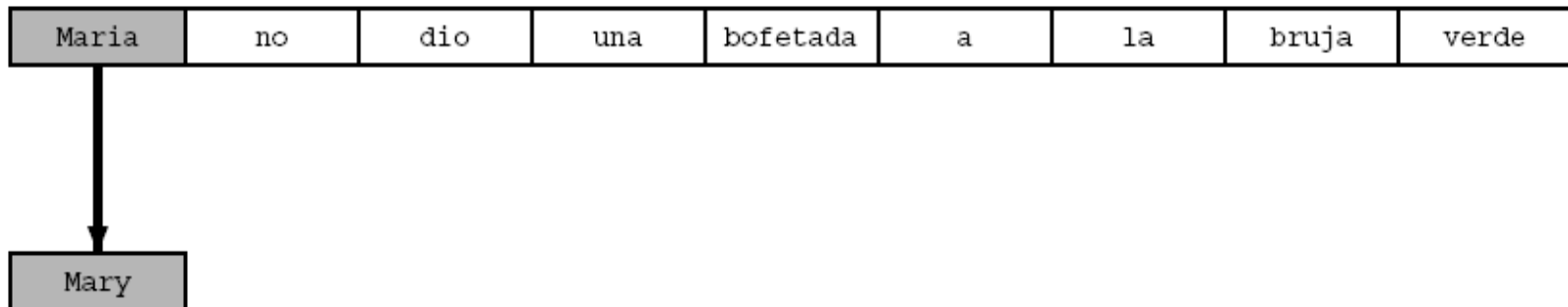
Processus de décodage

- Construit la traduction de gauche à droite
 - Sélectionne les mots source à traduire

Maria	no	dio	una	bofetada	a	la	bruja	verde
-------	----	-----	-----	----------	---	----	-------	-------

Processus de décodage

- Construit la traduction de gauche à droite
 - Sélectionne les mots source à traduire
 - Trouve la séquence anglais correspondante
 - Ajoute la séquence anglais à la fin de la traduction partielle courante



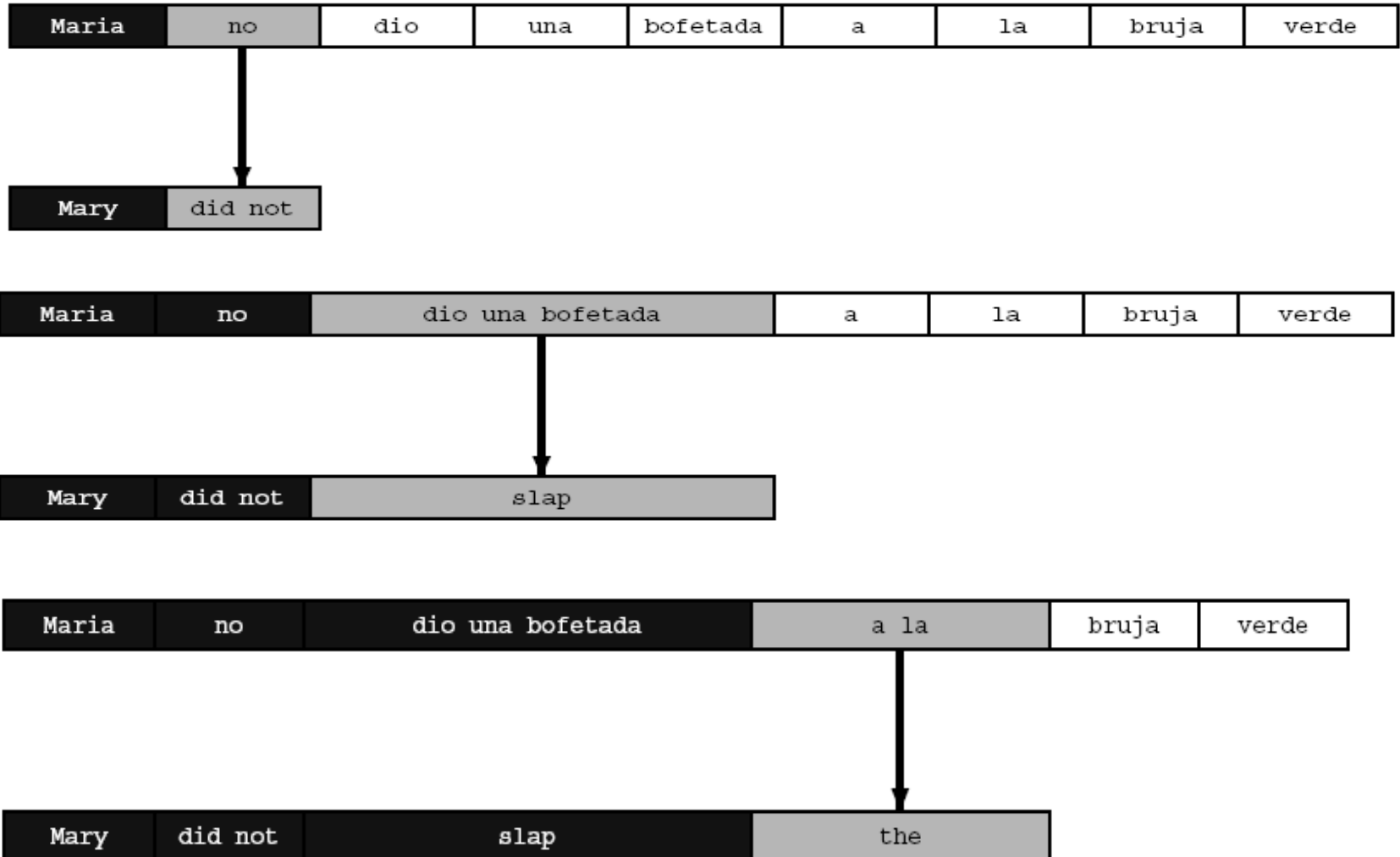
Processus de décodage

- Construit la traduction de gauche à droite
 - Sélectionne les mots source à traduire
 - Trouve la séquence anglais correspondante
 - Ajoute la séquence anglais à la fin de la traduction partielle courante
 - Marque les mots sources « traduits »

Maria	no	dio	una	bofetada	a	la	bruja	verde
-------	----	-----	-----	----------	---	----	-------	-------

Mary

Processus de décodage



Processus de décodage

Maria	no	dio una bofetada	a la	bruja	verde
-------	----	------------------	------	-------	-------

Mary	did not	slap	the	green
------	---------	------	-----	-------

Maria	no	dio una bofetada	a la	bruja	verde
-------	----	------------------	------	-------	-------

Mary	did not	slap	the	green	witch
------	---------	------	-----	-------	-------

Options de traduction

Maria	no	dio	una	bofetada	a	la	bruja	verde
<u>Mary</u>	<u>not</u>	<u>give</u>	<u>a</u>	<u>slap</u>	<u>to</u>	<u>the</u>	<u>witch</u>	<u>green</u>
	<u>did not</u>		<u>a slap</u>		<u>by</u>		<u>green witch</u>	
	<u>no</u>		<u>slap</u>		<u>to the</u>			
	<u>did not give</u>				<u>to</u>			
					<u>the</u>			
			<u>slap</u>			<u>the witch</u>		

- différentes façons de segmenter une phrase source en séquences
- différentes façons de traduire chaque séquence

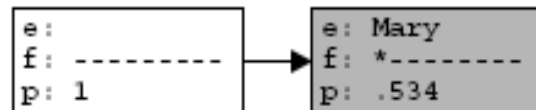
Expansion d'hypothèses

Maria	no	dio	una	bofetada	a	la	bruja	verde
<u>Mary</u>	<u>not</u>	<u>give</u>	<u>a</u>	<u>slap</u>	<u>to</u>	<u>the</u>	<u>witch</u>	<u>green</u>
	<u>did not</u>		<u>a</u>	<u>slap</u>	<u>by</u>		<u>green</u>	<u>witch</u>
	<u>no</u>		<u>slap</u>			<u>to the</u>		
	<u>did not give</u>					<u>to</u>		
						<u>the</u>		
				<u>slap</u>			<u>the</u>	<u>witch</u>

e:
f: -----
p: 1

Expansion d'hypothèses

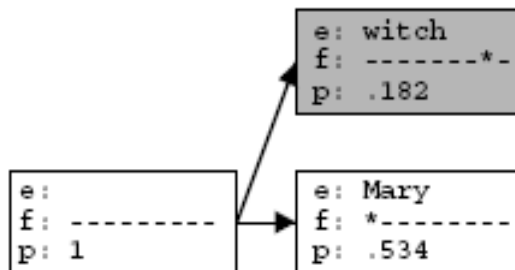
Maria	no	dio	una	bofetada	a	la	bruja	verde
Mary	not	give	a	slap	to	the	witch	green
	did not		a	slap	by		green	witch
	no		slap		to	the		
	did not give				to			
				slap		the		
							the	witch



Expansion d'hypothèses

Maria	no	dio	una	bofetada	a	la	bruja	verde
-------	----	-----	-----	----------	---	----	-------	-------

Mary not give a slap to the witch green
did not a slap by green witch
no slap to the
did not give to
slap the witch



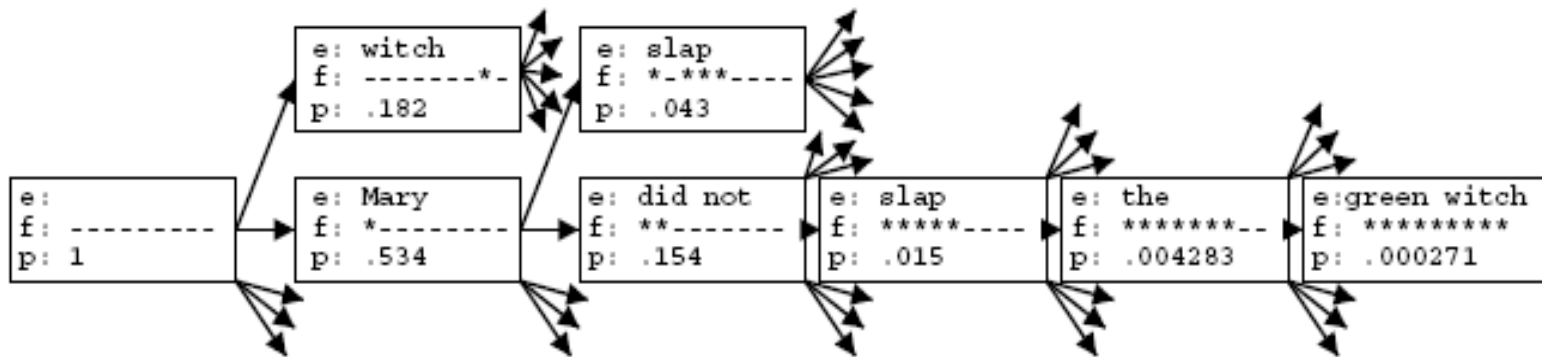
Expansion d'hypothèses



Expansion d'hypothèses

Maria	no	dio	una	bofetada	a	la	bruja	verde
-------	----	-----	-----	----------	---	----	-------	-------

Mary not give a slap to the witch green
did not a slap by green witch
no slap to the
did not give to
slap the witch

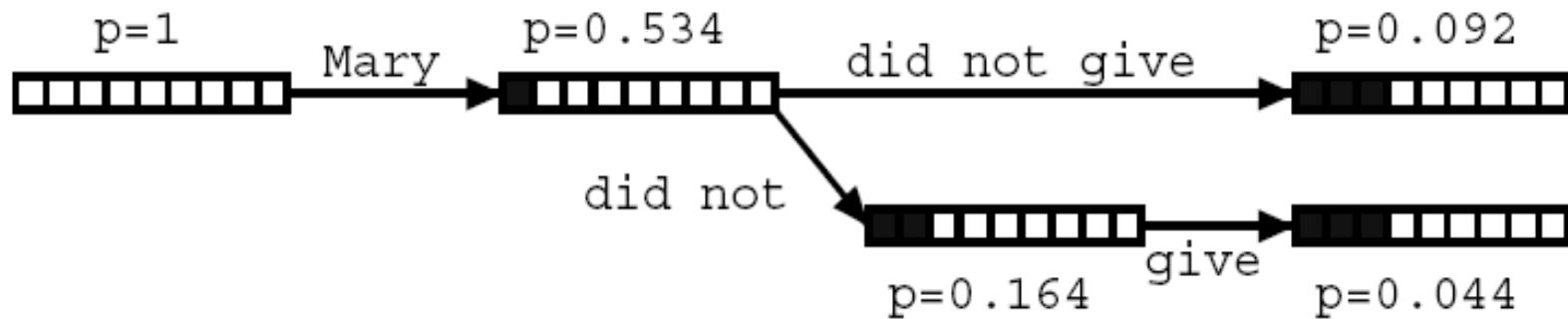


Explosion de l'espace de recherche

- Le nombre d'hypothèses croît exponentiellement avec le nombre de mots dans la phrase source
- Le processus de décodage est un problème NP complet [Knight, 1999]
- Besoin de réduire l'espace de recherche
 - Recombinaison d'hypothèses
 - Elagage (pruning)

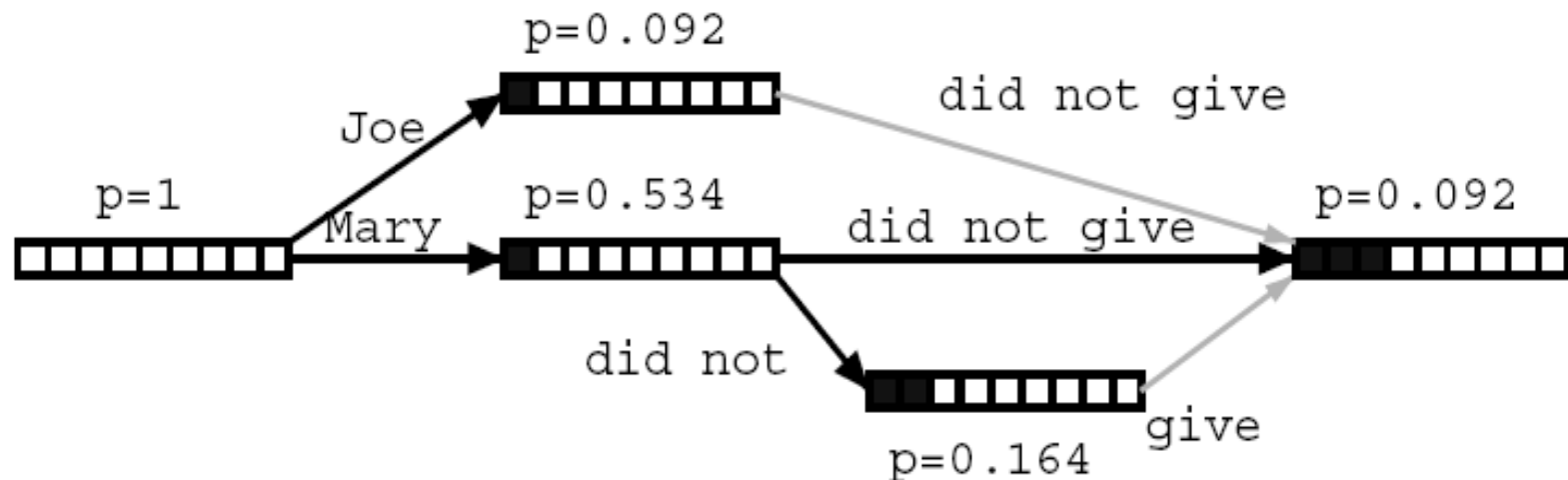
Recombinaison d'hypothèses

- Différents chemins mènent à la même hypothèse partielle
 - Supprimer le chemin le moins probable



Recombinaison d'hypothèses

- Les chemins n'ont pas besoin d'être strictement identiques
- On peut supprimer un chemin si
 - Les 2 derniers mots cible (anglais) correspondent (ML)
 - La couverture en mots source correspond (futurs chemins)

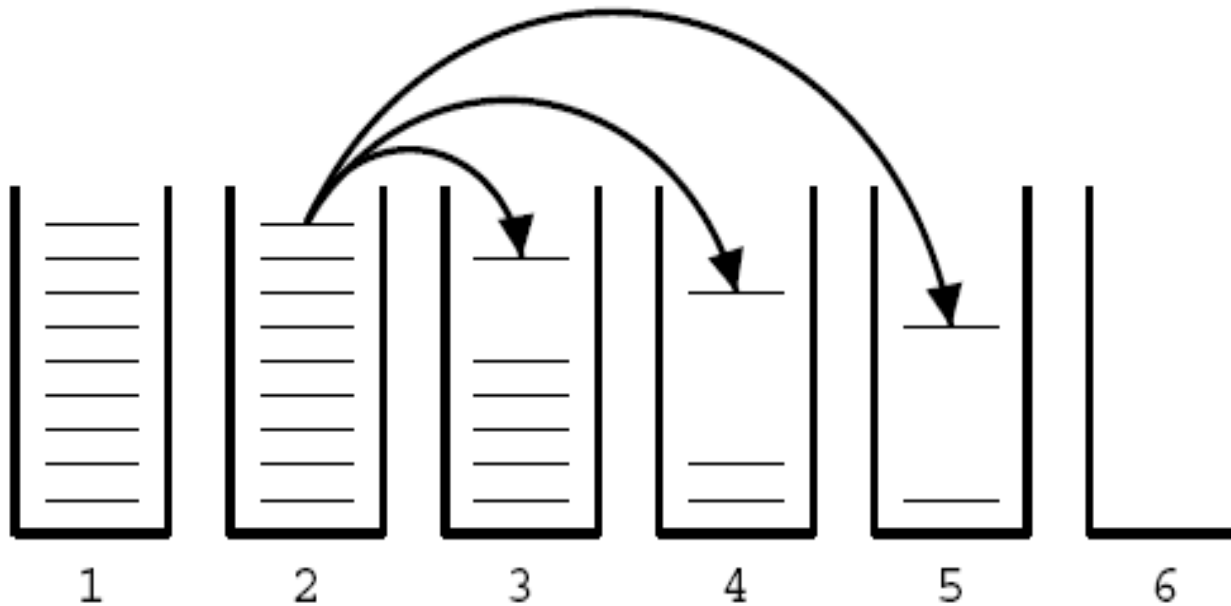


Elagage (pruning)

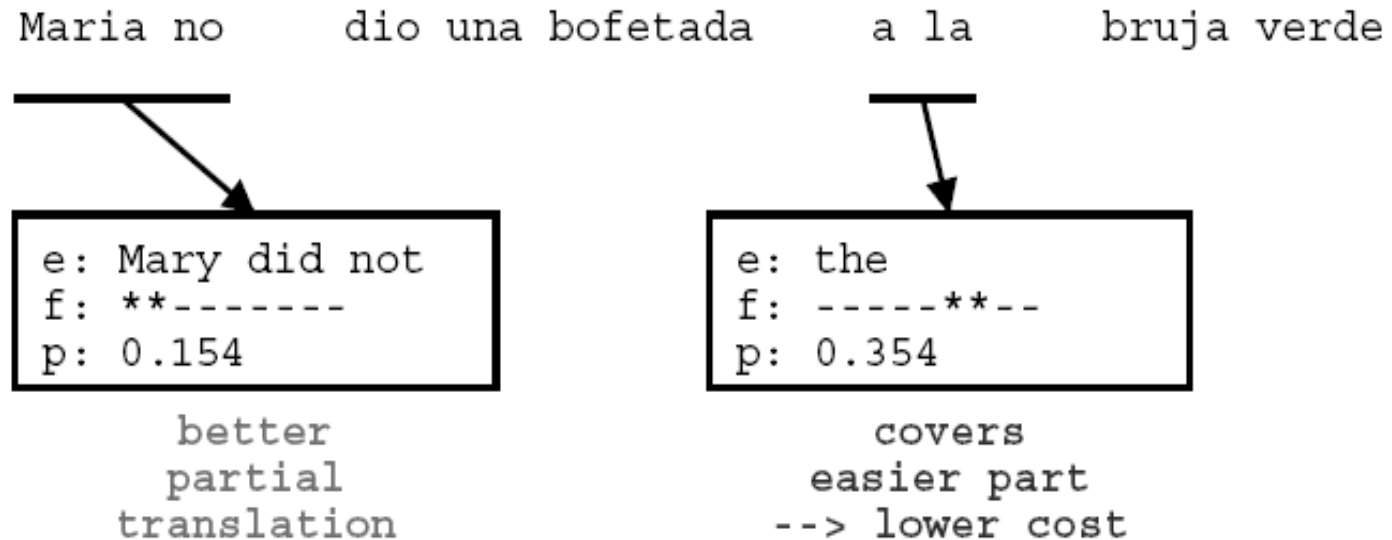
- **Organiser les hypothèses en piles par**
 - mêmes mots sources couverts
 - même nombre de mots sources couverts
 - même nombre de mots cibles traduits
- **Comparer les hypothèses dans les piles, supprimer les mauvaises**
 - *histogram pruning*: garder les n meilleures hypothèses pour chaque pile (e.g., n=100)
 - *threshold pruning*: garder les hypothèses qui ont au plus un score égal à x fois le score de la meilleure hypothèse (e.g. x= 0.001)

Exemple

- Piles fondées sur le nombre de mots étrangers traduits

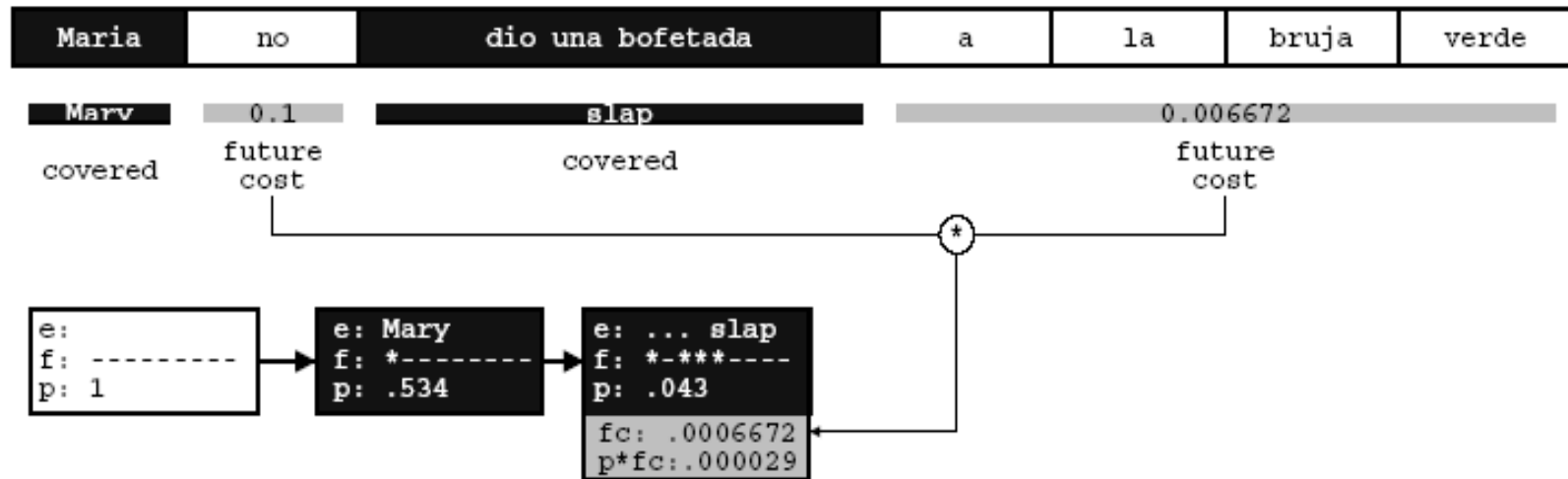


Comparer les hypothèses



Une hypothèse qui couvre une partie « facile » risque d'être préférée au détriment d'une bonne traduction partielle
=> Il faut considérer les coût futur des parties non traduites

Estimer les score futur



- Utiliser le score futur pour élaguer les hypothèses
- Ajouter score futur & score passé pour décider d'élaguer une hypothèse ou pas

Outils disponibles...

- GIZA++
- Moses
- SRI-LM