Introduction au traitement automatique des langues probabiliste

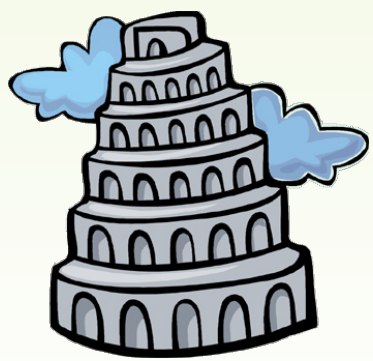
Alexis Nasr - supports adaptés par Carlos Ramisch

Traitement automatique des langues (TAL)

Science pluridisciplinaire à l'intersection entre

- ***informatique***
- ***linguistique***
- ***sciences cognitives***

qui vise à modéliser les ***langues humaines*** dans les ***systèmes informatiques***, pour permettre des interactions en langue naturelle homme-machine et entre humains



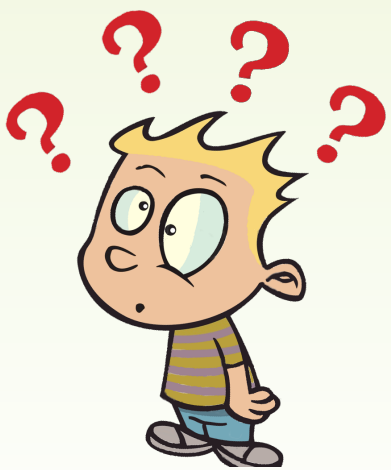
Les langues humaines sont. . .

Omniprésentes !

- Interactions parlées entre personnes (et systèmes informatiques)
- Popularité des applications web et mobiles → . . .
- Très grand volume de texte généré tous les jours
- Naturellement multilingue

Mais les langues humaines sont aussi. . .

- En constante évolution :
  - *non mais allô quoi !*
  - *printemps arabe*
  - *deep learning*
  - *LOL, mdr, ptdr, . . .*
- Ambiguës :
  - *vers (de terre)* $\times$ *vers (préposition)* $\times$ *vert* $\times$ *verre*
  - *il était tard* $\times$ *il est têtard*
  - *avocat (fruit)* $\times$ *avocat (métier)*
- Arbitraires :
  - *pleine lune* $\times$ *?lune entière*
  - *lait entier* $\times$ *?lait complet*
  - *pain complet* $\times$ *?pain plein*



Comment on construit un système de TAL ? I

Approche “experte”

- règles déterministes exprimées explicitement
- représenter notre connaissance sur le fonctionnement des langues
- des années de travail, des linguistes expérimentés
- des milliers d’€
- haute qualité mais peu de robustesse

Comment on construit un système de TAL ? II

Approche “empirique”

- apprendre des modèles automatiquement à partir du texte
- modèles probabilistes à paramètres inférés avec des stats
- indépendant de la langue
- grands volumes de données et ordinateurs puissants
- bruit et silence

Méthodes probabilistes pour le TAL

- Utilisation de statistiques pour choisir entre plusieurs alternatives.
- L'ambiguïté peut être "réelle" :
  - Jean regarde un homme avec un télescope.
  - Jean regarde un homme avec un télescope.
- ou provenir d'un manque de connaissances :
  - Jean commande une glace à la serveuse.
  - Jean commande une glace à la serveuse.
- Le rôle des statistiques est d'aider à faire un choix lorsque l'on n'a pas d'autres moyens.

Ambiguïté

Phénomène omniprésent en TAL :

- reconnaissance de la parole
- étiquetage morpho-syntaxique
- analyse syntaxique
- traduction automatique
- reconnaissance optique de caractères
- ...

Ambiguïté en reconnaissance de la parole

Les	traits	très	tirés
Les	trait	très	tirées
Les	très	très	tirées
Les	trait	trait	tiraient
Les	traits	très	tirait
...			

Ambiguïté en étiquetage morpho-syntaxique

la	belle	ferme	le	voile
Det	N	V	Det	N
N	Adj	N	Pro	V
Pro		Adj	N	

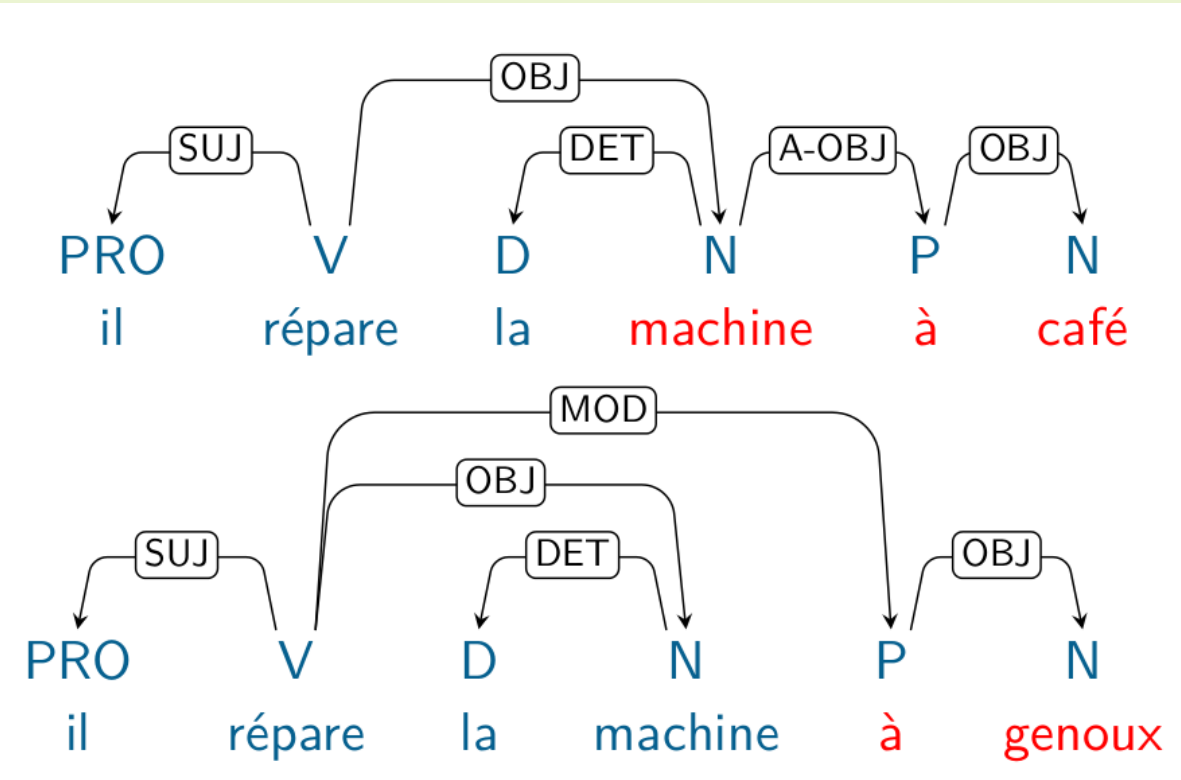
108 combinaisons possibles !

Ambiguïté en étiquetage morpho-syntaxique

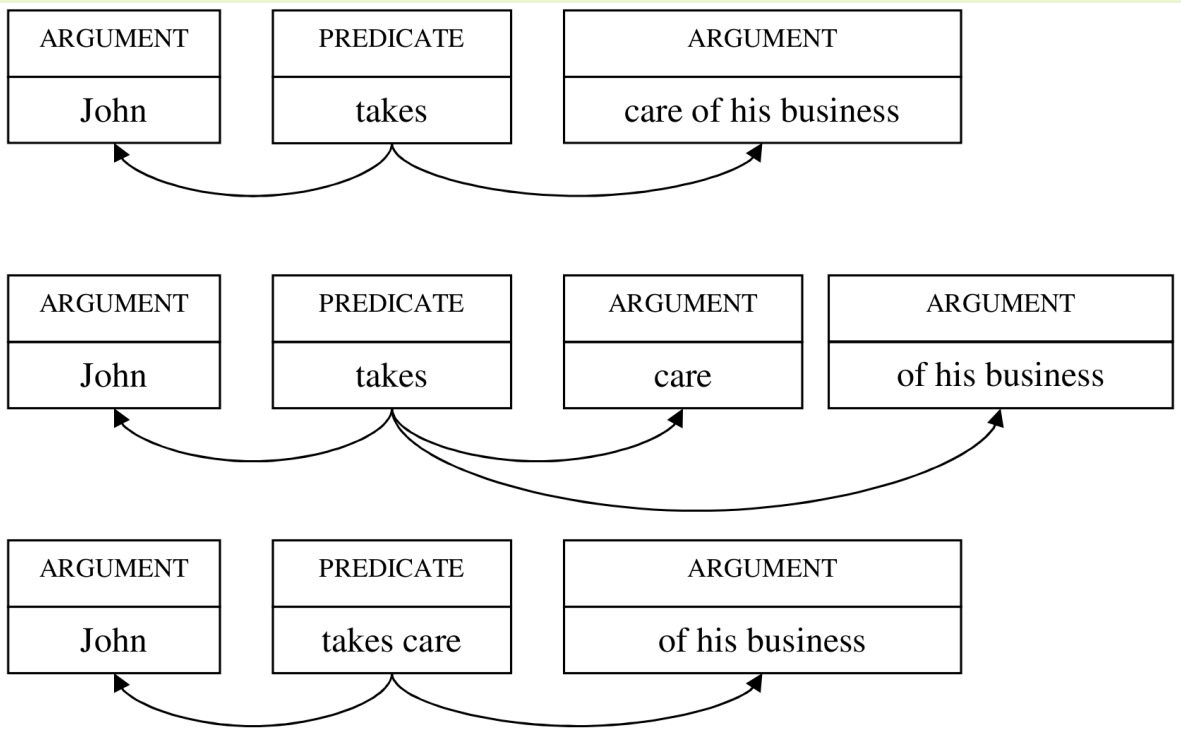
la	belle	ferme	le	voile
Det	N	V	Det	N
N	Adj	N	Pro	V
Pro		Adj	N	

108 combinaisons possibles !

Ambiguïté en analyse syntaxique



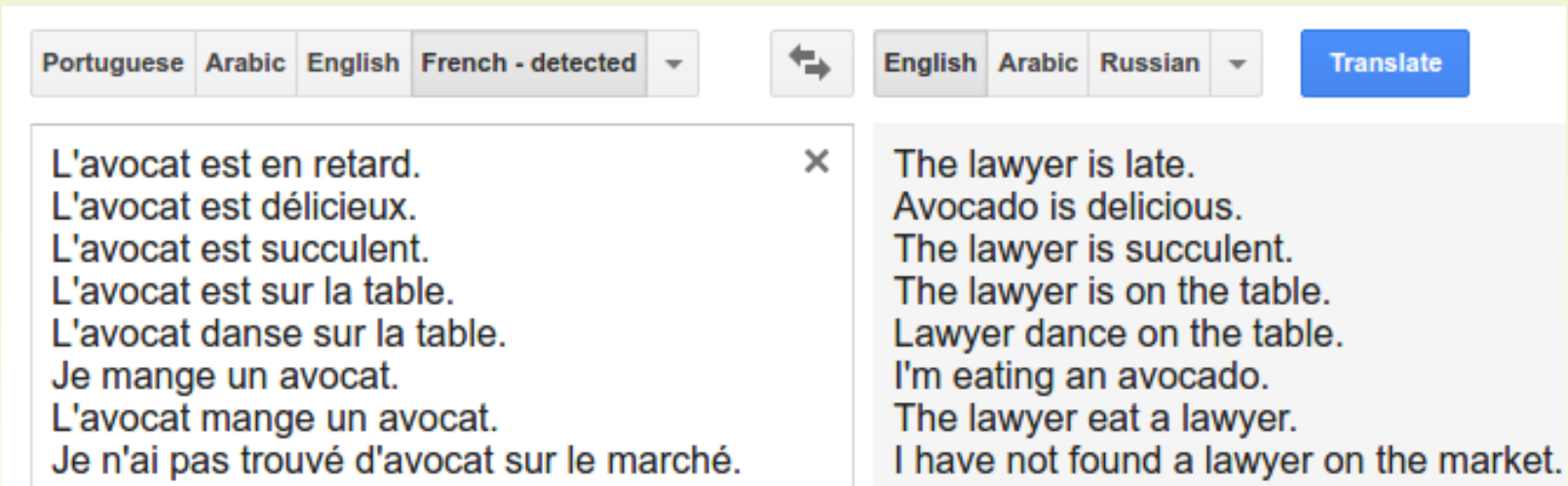
Ambiguïté en analyse sémantique



Ambiguïté en traduction automatique

The river **banks** $\rightarrow$ *les* **berges** *de la rivière*
The **banks** *closed* $\rightarrow$ *les* **banques** *ont fermé*

Ambiguïté en traduction automatique II



Comment lever l'ambiguïté ?

- Ajouter des connaissances :
  - phonétiques / phonologiques
  - lexicales
  - syntaxiques
  - sémantiques
  - pragmatiques
  - encyclopédiques
- Associer un score aux différentes alternatives :
  S(*les traits très tirés*) $>$ S(*les très traits tiraient*)
- Ces scores peuvent être des probabilités, elles sont calculées à l'aide d'un modèle probabiliste ou modèle stochastique.

Exemple : calcul de la probabilité d'une séquence de mots

- Problème récurrent dans plusieurs applications :
  - reconnaissance de la parole
  - reconnaissance optique de caractères
  - traduction automatique
- Qu'attend-t-on du modèle ?
  - attribuer une bonne proba. aux séquences correctes :
    - *les traits très tirés.*
    - *les traits très tirées.*
    - *les très traient tiraient.*
    - *les très très t'iraient.*
  - attribuer une bonne proba. aux séquences attendues :
    - *les traits très tirés.*
    - *lettrés très tirés.*

Modèle unigramme

- On utilise la probabilité d'occurrence des mots de la séquence :

$$P(m_1 \dots m_n) = \prod_{i=1}^{n} P(m_i)$$

- $P(\text{les traits très tirés}) =$
  $P(\text{les}) \times P(\text{traits}) \times P(\text{très}) \times P(\text{tirés})$
- le modèle fait des **hypothèses** évidemment fausses !

séquence	$-\log P$
les très très tirés	14.265
les trait très tiré	15.3102
les traits très tiré	15.4028
les trait très tirés	15.8386
les traits très tirés	15.9312

Modèle unigramme : estimation

- Comment donner une valeur à $P(m_i)$ avec $1 \leq i \leq |V|$?
- On prend une grande quantité de texte : $o_i \ldots o_n$ (o est une *occurrence* de mot)
- On calcule le nombre d'occurrences du mot m :

$$C(m) = \sum_{i=1}^{n} \delta_{o_i,m}$$

- Puis la fréquence relative de m :

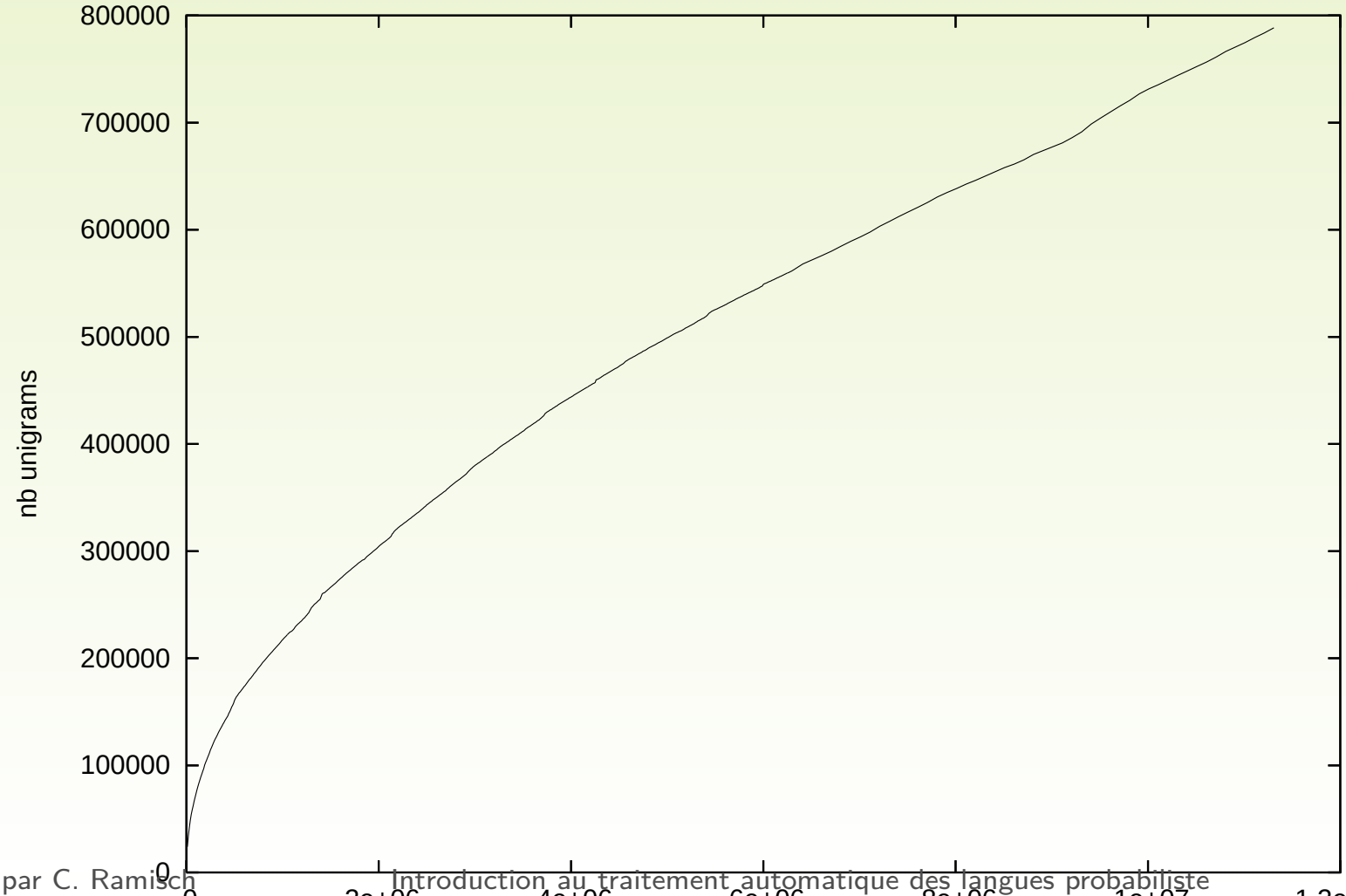
$$P(m) = \frac{C(m)}{\sum_{i=1}^{|V|} C(m_i)}$$

- Il y a $|V|$ probabilités à estimer.

Données d'apprentissage

- Journal Le Monde 1986-2002
- 16 479 270 phrases
- 370 005 285 occurrences

Nombre d'unigrammes différents



Rang, fréq et fréq. de fréq. des unigrammes

rang	fréq	f2f	unigramme
1	13 286 304	1	de
2	6 964 863	1	la
3	5 900 839	1	le
4	5 599 010	1	l’
5	5 017 018	1	et
6	4 762 293	1	les
7	4 208 264	1	des
8	3 856 293	1	d’
9	3 695 434	1	un
10	3 425 787	1	en

Légende

Le mot *de* est le plus fréquent (rang 1). Il apparait 13 286 304 fois. Seulement 1 mot apparaît ce nombre de fois dans le corpus (f2f).

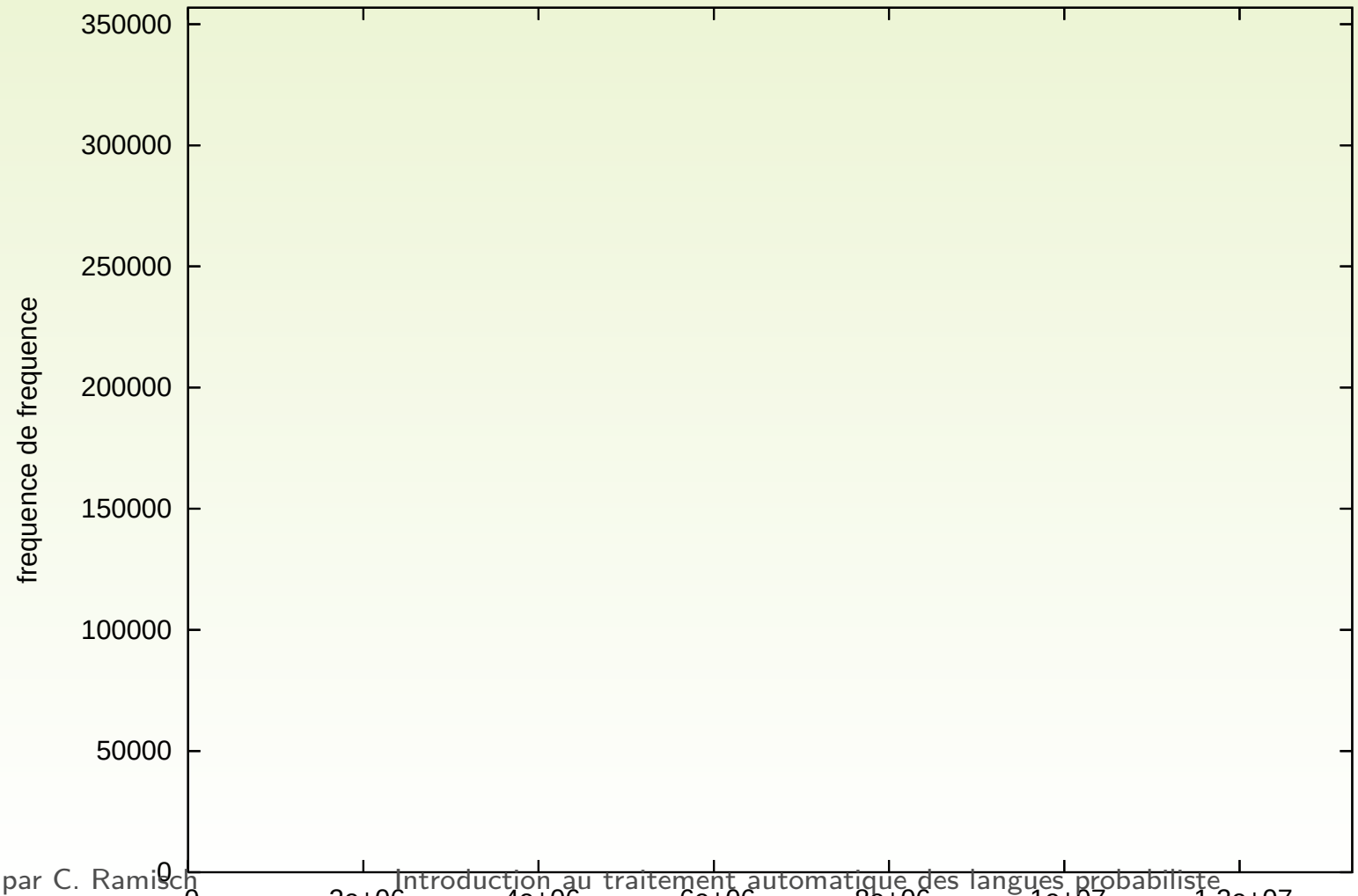
Rang, fréq et fréq. de fréq. des unigrammes

rang	fréq	f2f	unigramme
8532	10	7 992	abaisseront, Abott, antihypertenseur
8533	9	9 369	
8534	8	11 140	
8535	7	13 671	
8536	6	17 351	
8537	5	22 684	
8538	4	31 980	
8539	3	50 311	
8540	2	99 581	
8541	1	356 780	absolumen, absorbable, Ambrozic

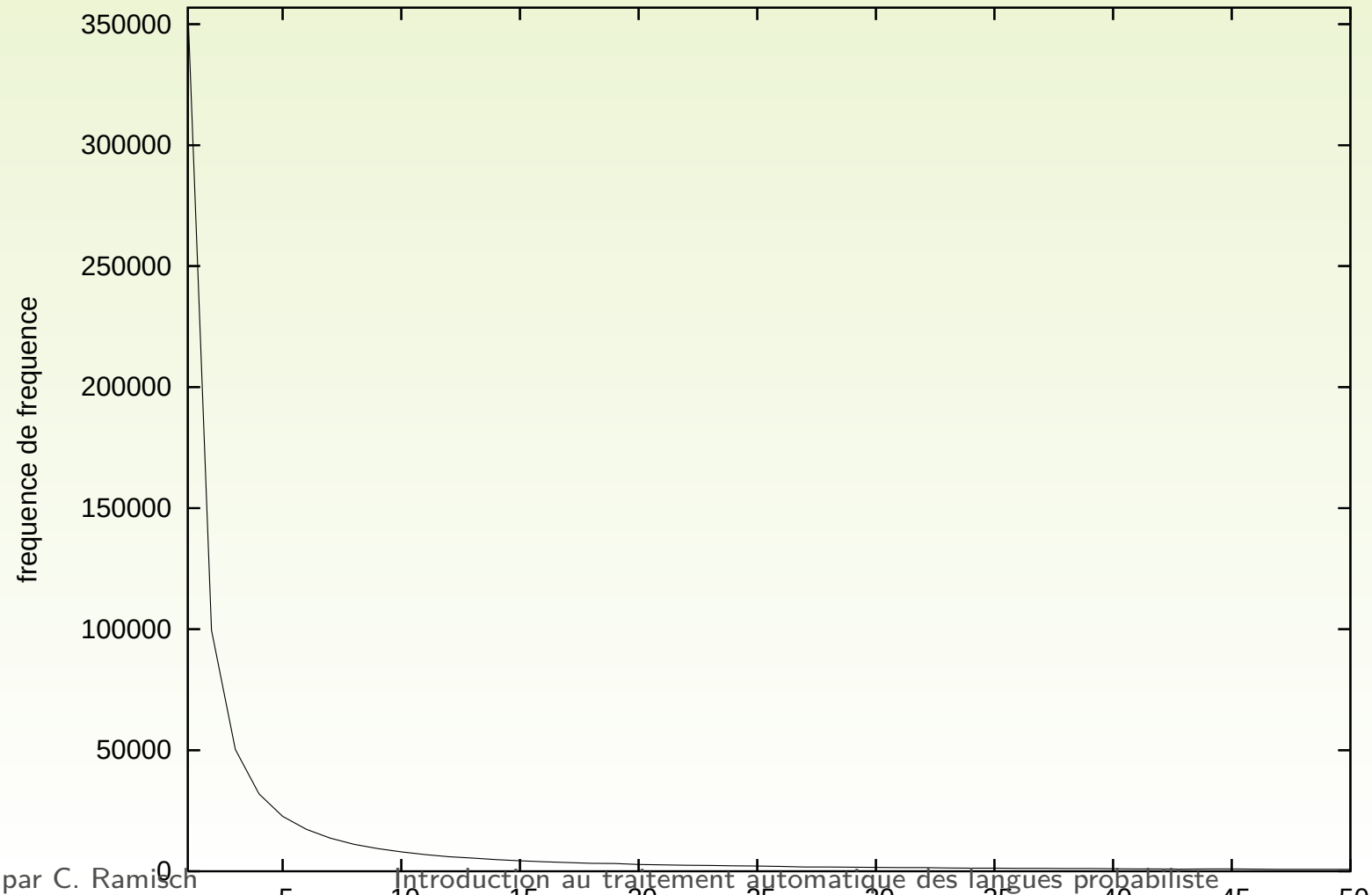
Légende

Il y a 356 780 mots qui apparaissent 1 fois dans le corpus (p. ex. *absolumen, absorbable, Ambrozic*). Ces mots sont les moins fréquents du corpus (rang 8541)

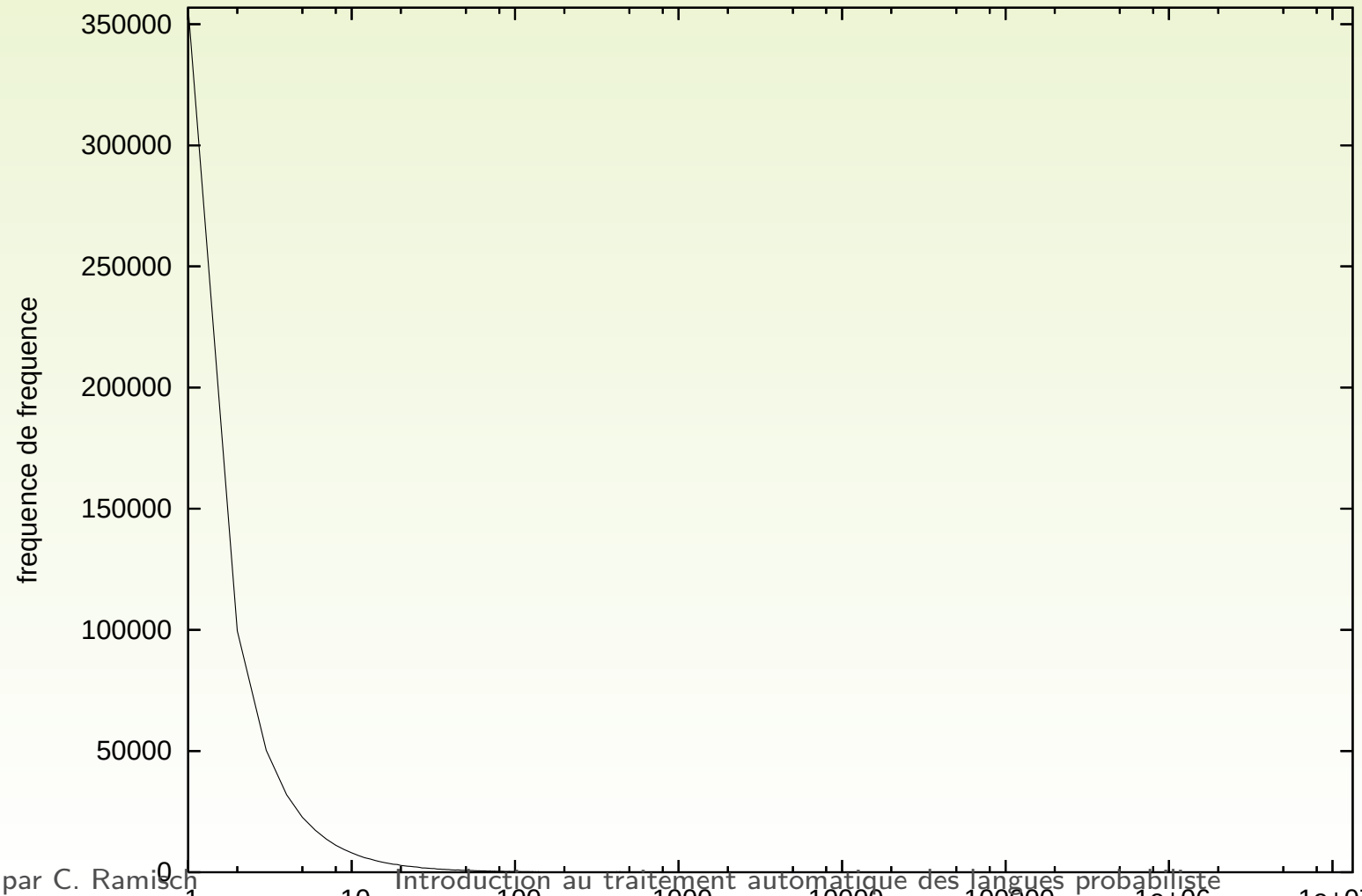
Fréquence des unigrammes, lin-lin



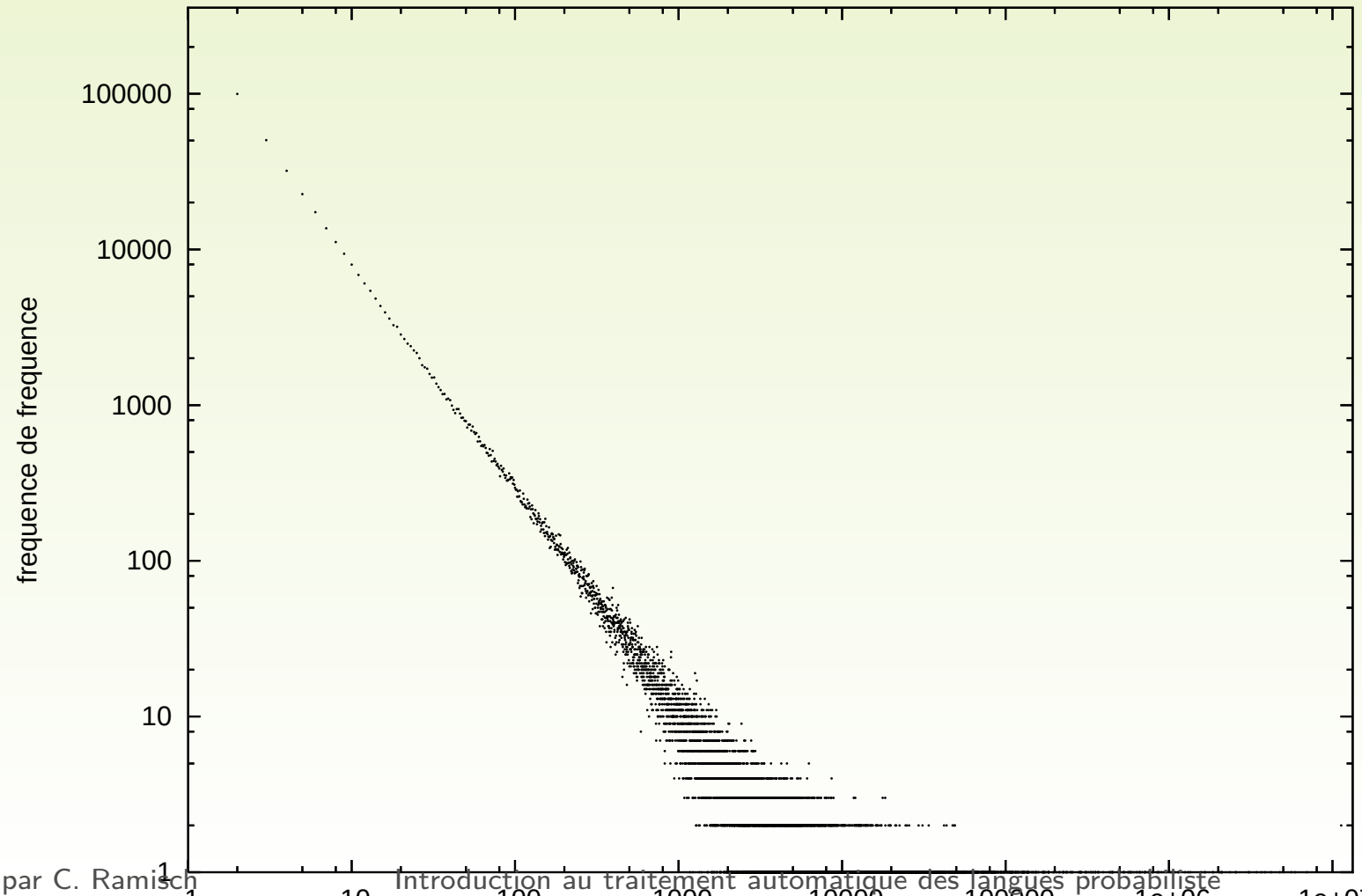
Fréquence des unigrammes, lin-lin



Fréquence des unigrammes, log-lin



Fréquence des unigrammes, log-log



La loi de Zipf

- Dans de nombreux phénomènes humains, il existe une relation log-linéaire entre la fréquence (f) et l'inverse du rang (r) :

$$f(r) = \times \frac{1}{r^k}$$

- Interprétation de Zipf : manière de minimiser l'effort du locuteur et de l'auditeur :
  - le locuteur minimise l'effort en ayant un petit vocabulaire de mots courants
  - l'auditeur minimise l'effort en ayant un vocabulaire important de mots rares (de sorte que les messages sont peu ambigus.)
- De manière grossière : peu de mots très fréquents beaucoup de mots peu fréquents.

Modèle bigramme

- On utilise la probabilité que le mot a suive le mot b :
  $P(m_{i+1} = a | m_i = b)$

$$P(m_1 \dots m_n) = P(m_1) \times \prod_{i=2}^{n} P(m_i | m_{i-1})$$

- $P(\text{les traits très tirés}) =$
  $P(\text{les}) \times P(\text{traits}|\text{les}) \times P(\text{très}|\text{traits}) \times P(\text{tirés}|\text{très})$

séquence	$-\log P$
les très très tirés	13.4767
les traits très tiré	13.8494
les traits très tirés	14.1414
les trait très tirés	18.2462
les trait très tiré	18.5381

Modèle bigramme : estimation

- On calcule le nombre d'occurrences de la séquence ab :

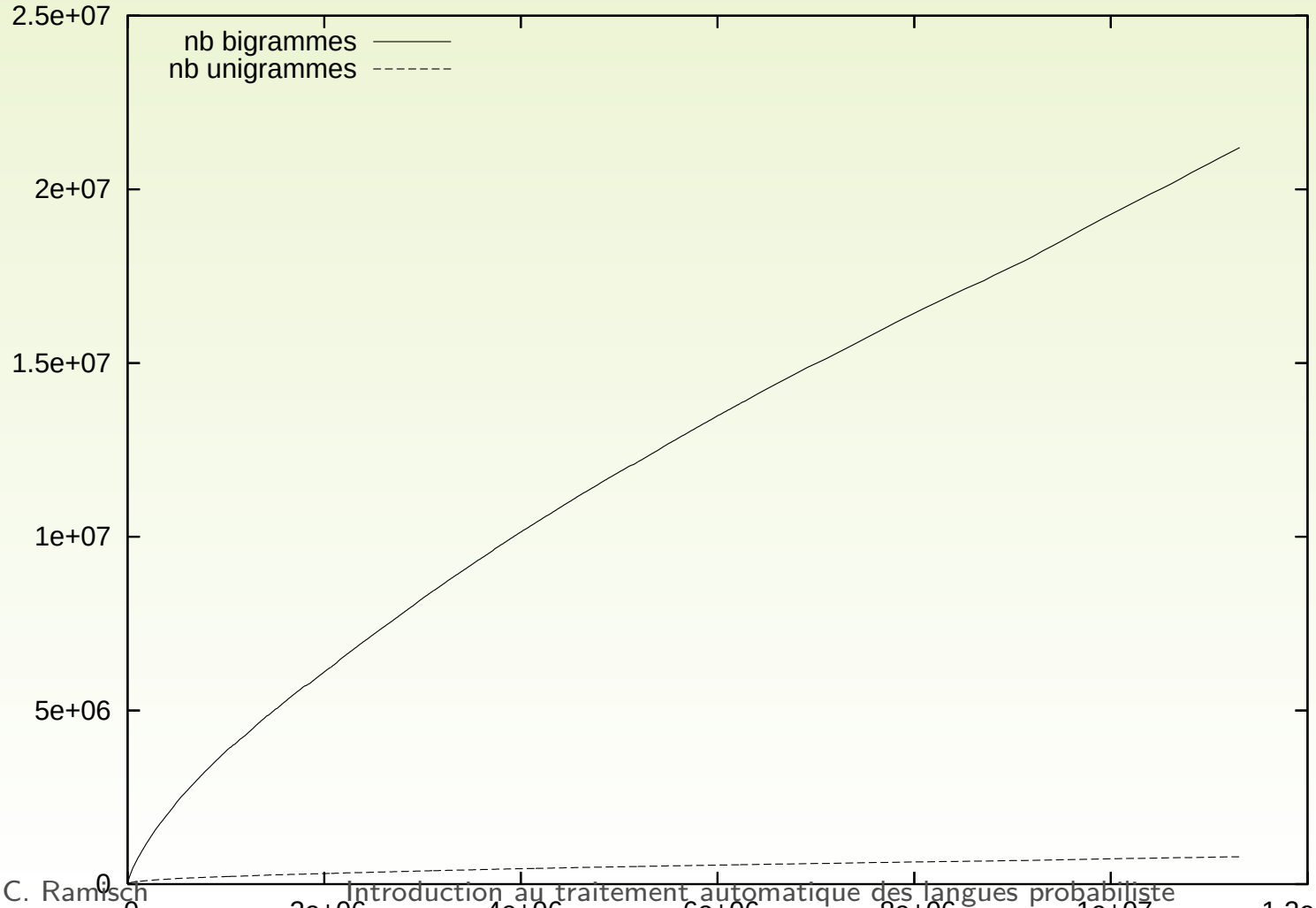
$$C(a,b) = \sum_{i=1}^{N-1} \delta_{o_i,a} \times \delta_{o_{i+1},b}$$

- Puis la fréquence relative :

$$P(b|a) = \frac{C(a,b)}{C(a)}$$

- $|V|^2$ paramètres à estimer.

Nombre de bigrammes différents



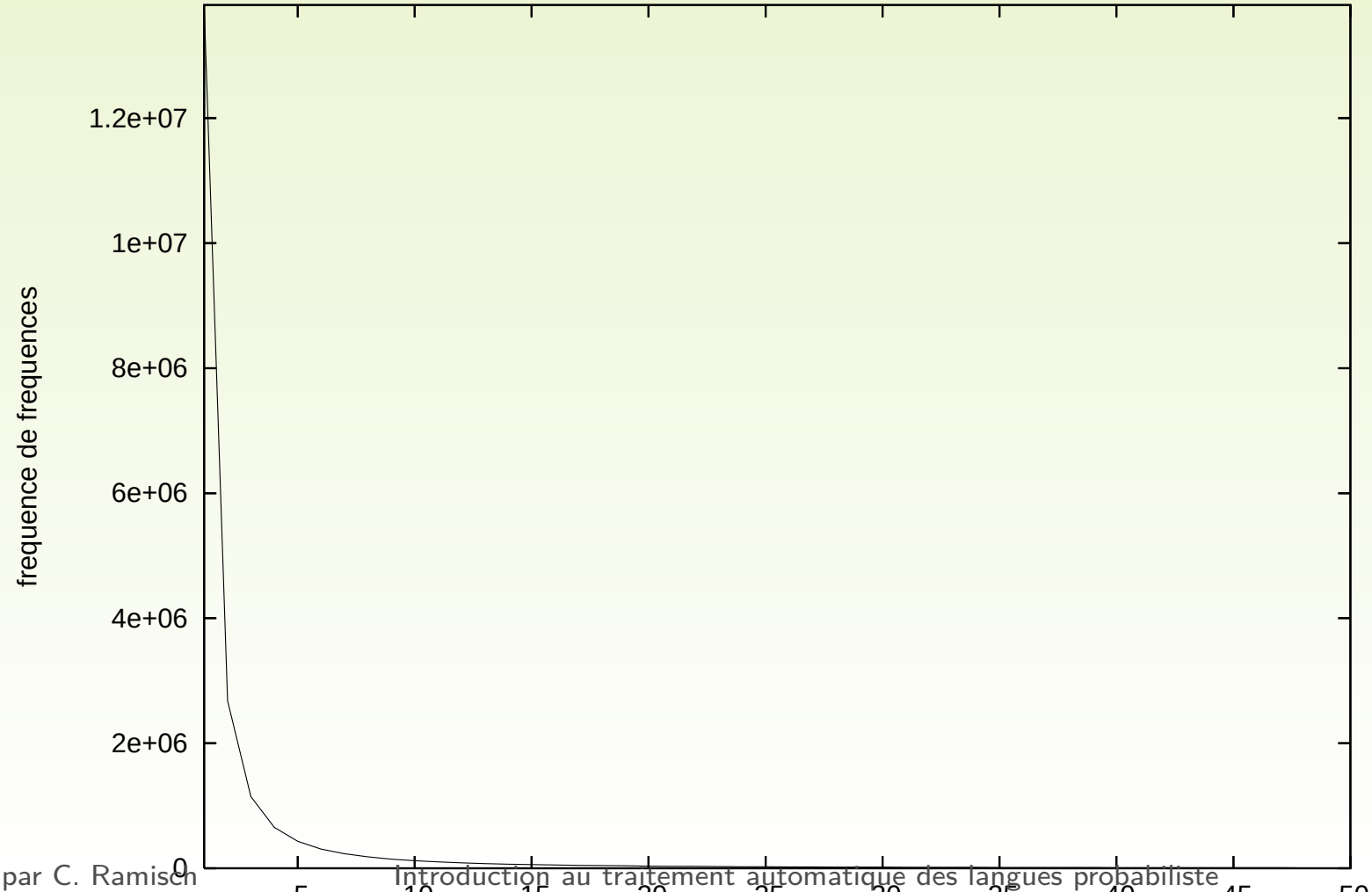
Fréquence des bigrammes

rang	fréq	f2f	bigramme
1	2 091 936	1	de la
2	1 563 496	1	de l'
3	1 156 551	1	neuf cent
4	1 139 331	1	mille neuf
5	921 010	1	cent quatre_vingt
6	744 220	1	d'un
7	578 140	1	d'une
8	571 205	1	c'est
9	556 919	1	et de
10	541 528	1	en mille

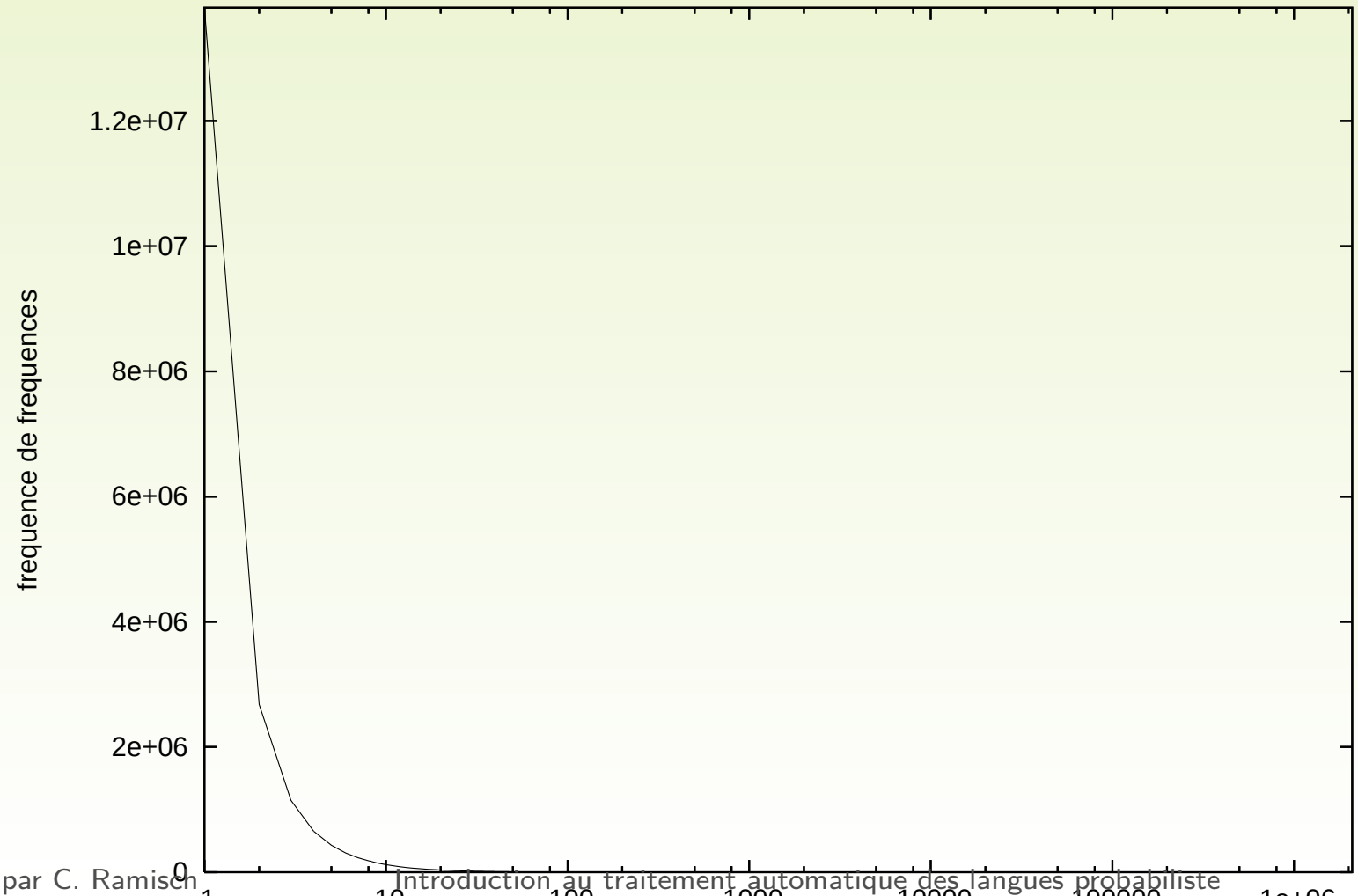
Fréquence des bigrammes

rang	fréq	f2f	bigramme
8817	10	121 362	
8818	9	146 471	
8819	8	181 721	
8820	7	231 738	
8821	6	307 461	
8822	5	429 606	
8823	4	655 244	
8824	3	1 148 479	
8825	2	2 680 674	
8826	1	13 808 569	

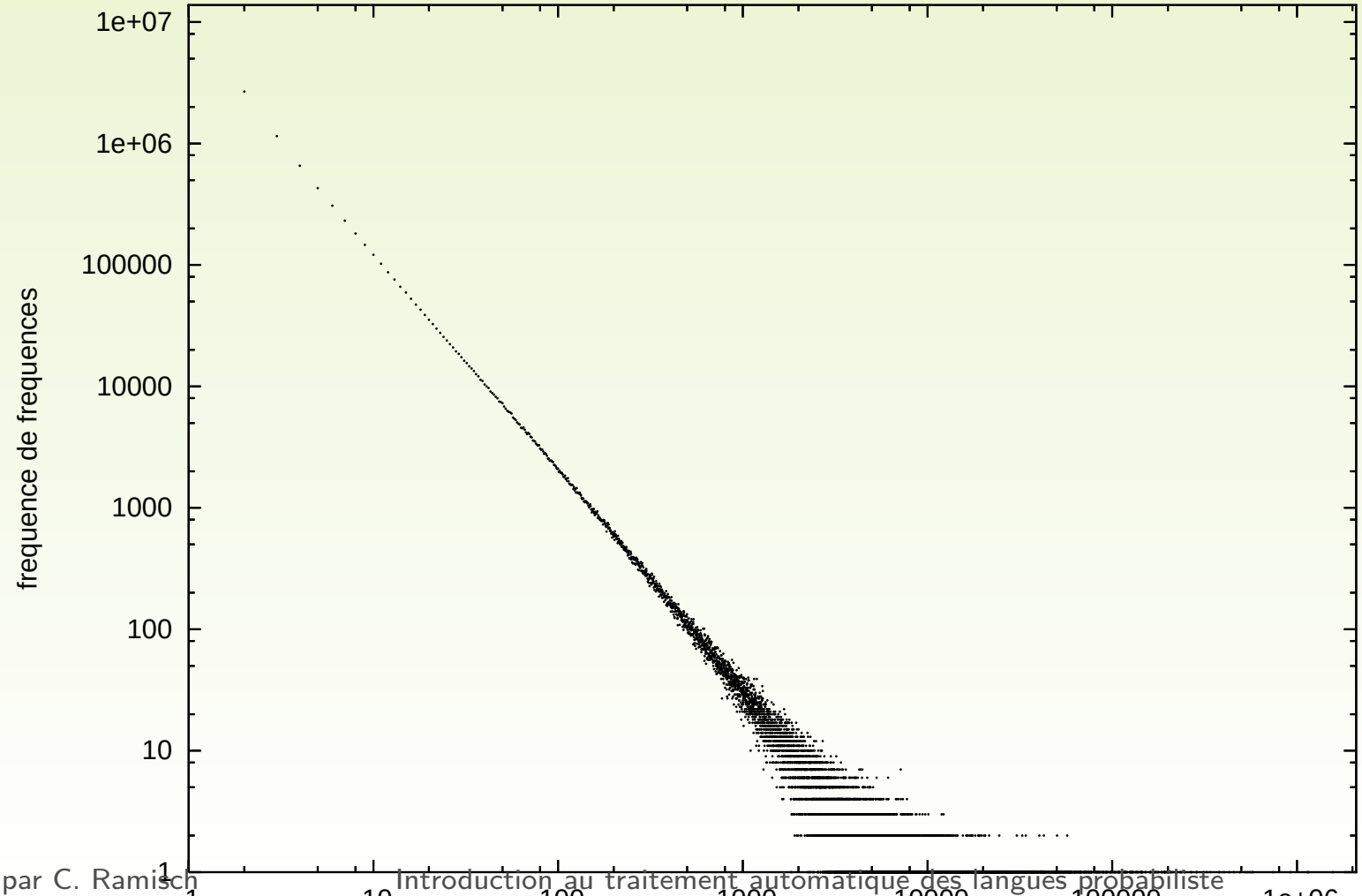
Fréquence des bigrammes, lin-lin



Fréquence des bigrammes, log-lin



Fréquence des bigrammes, log-log



Modèle trigramme

- On utilise la probabilité que c suive la séquence ab :
  $P(m_{i+2} = c | m_i = a, m_{i+1} = b)$

$$P(m_1 \dots m_n) = P(m_1) \times P(m_2|m_1) \times \prod_{i=3}^{n} P(m_i|m_{i-2}m_{i-1})$$

- $P(\text{les traits très tirés}) =$
  $P(\text{les}) \times P(\text{traits}|\text{les}) \times P(\text{très}|\text{les, traits}) \times P(\text{tirés}|\text{traits, très})$

séquence	$-\log P$
les très très tirés	24.0764
les traits très tirés	28.134
les traits très tiré	28.2369
les trait très tirés	31.0759
les trait très tiré	31.1788

Modèle trigramme : estimation

- On calcule le nombre d'occurrences de la séquence *abc* :

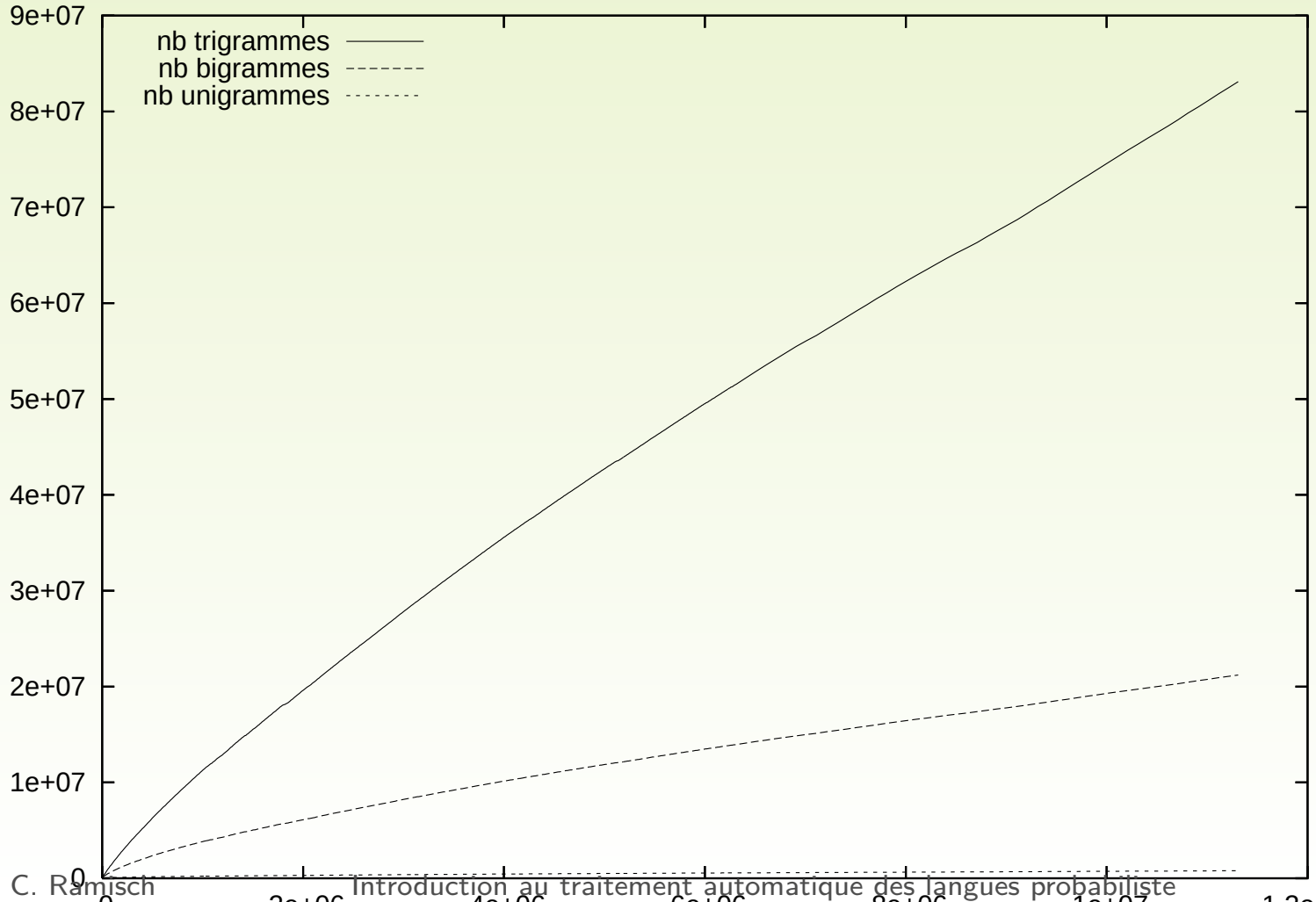
$$C(a,b,c) = \sum_{i=1}^{N-2} \delta_{o_i,a} \times \delta_{o_{i+1},b} \times \delta_{o_{i+2},c}$$

- Puis la fréquence relative :

$$P(c|a,b) = \frac{C(a,b,c)}{C(a,b)}$$

- $|V|^3$ paramètres à estimer.

Nombre de trigrammes différents



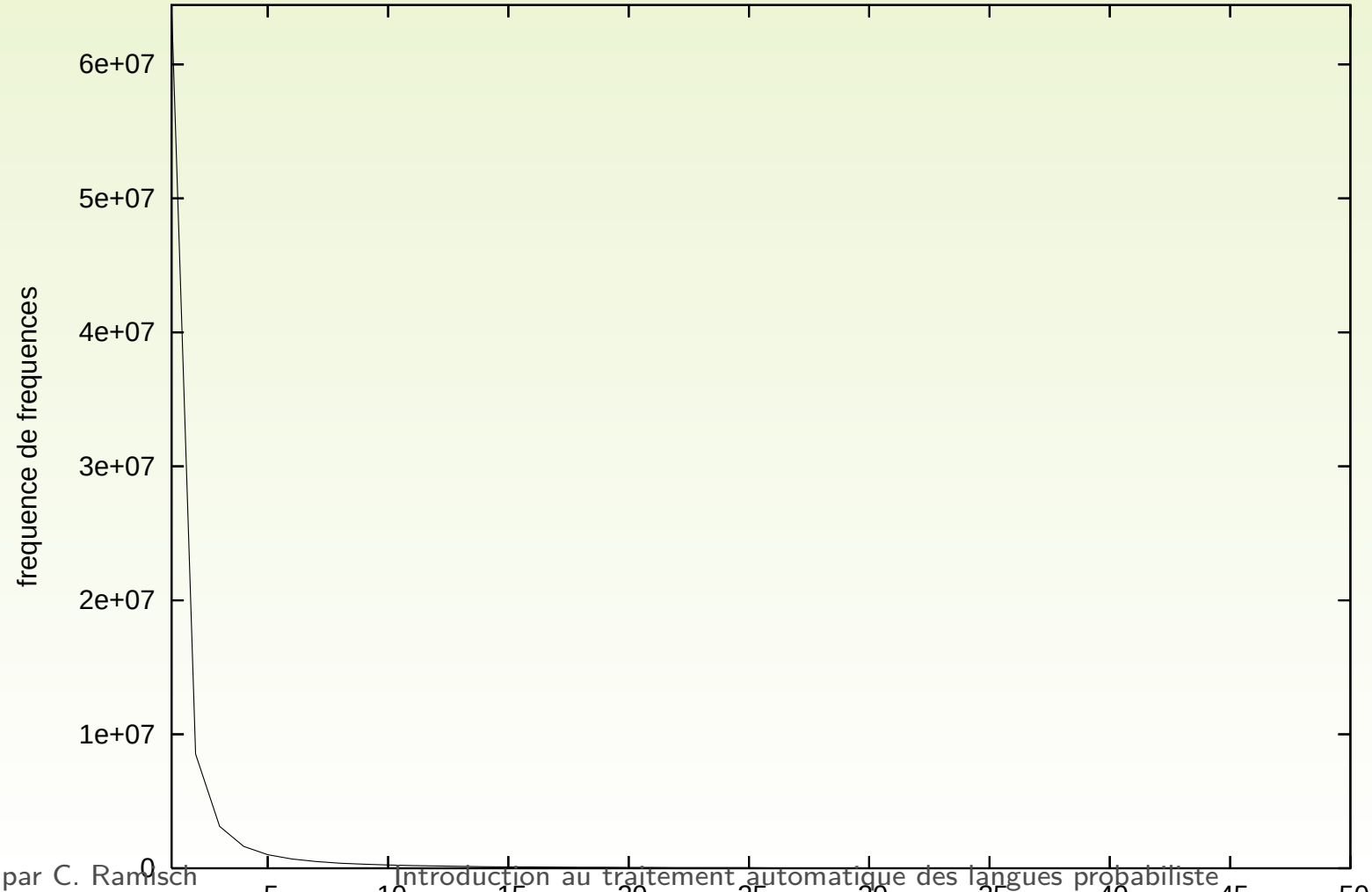
Rang, fréq et fréq. de fréq. des trigrammes

rang	fréq	f2f	trigramme
1	1	1 135 994	mille neuf cent
2	1	792 656	en mille neuf
3	1	379 902	cent quatre_vingt dix
4	1	191 324	n' est pas
5	1	184 626	neuf cent soixante
6	1	167 909	il y a
7	1	160 077	de mille neuf
8	1	145 121	n' a pas
9	1	95 683	et de la
10	1	95 201	cent soixante dix

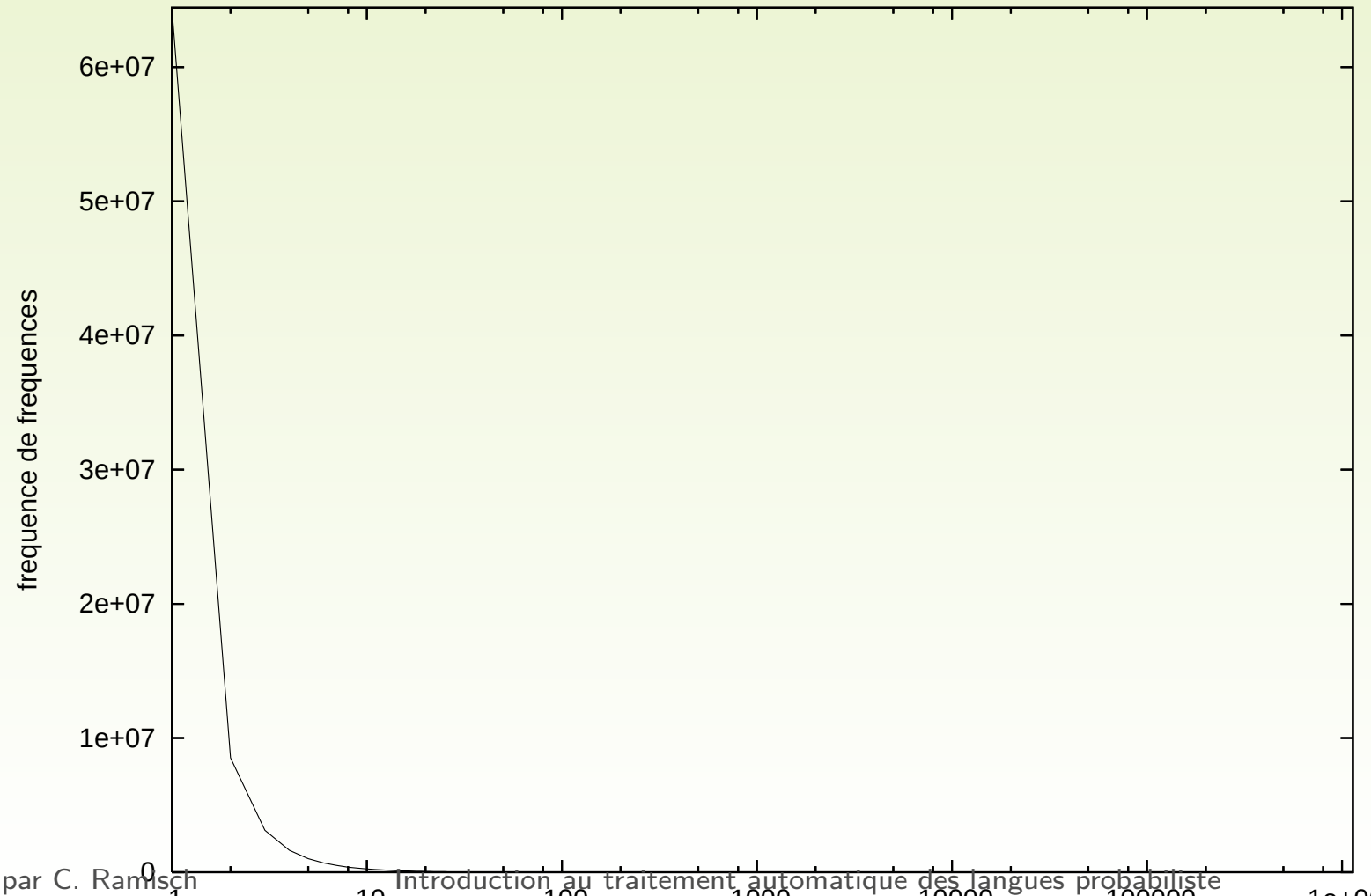
Rang, fréq et fréq. de fréq. des trigrammes

rang	fréq	f2f
5193	10	242 472
5194	9	299 876
5195	8	382 270
5196	7	502 171
5197	6	691 755
5198	5	1 013 295
5199	4	1 641 586
5200	3	3 124 196
5201	2	8 523 984
5202	1	64 425 232

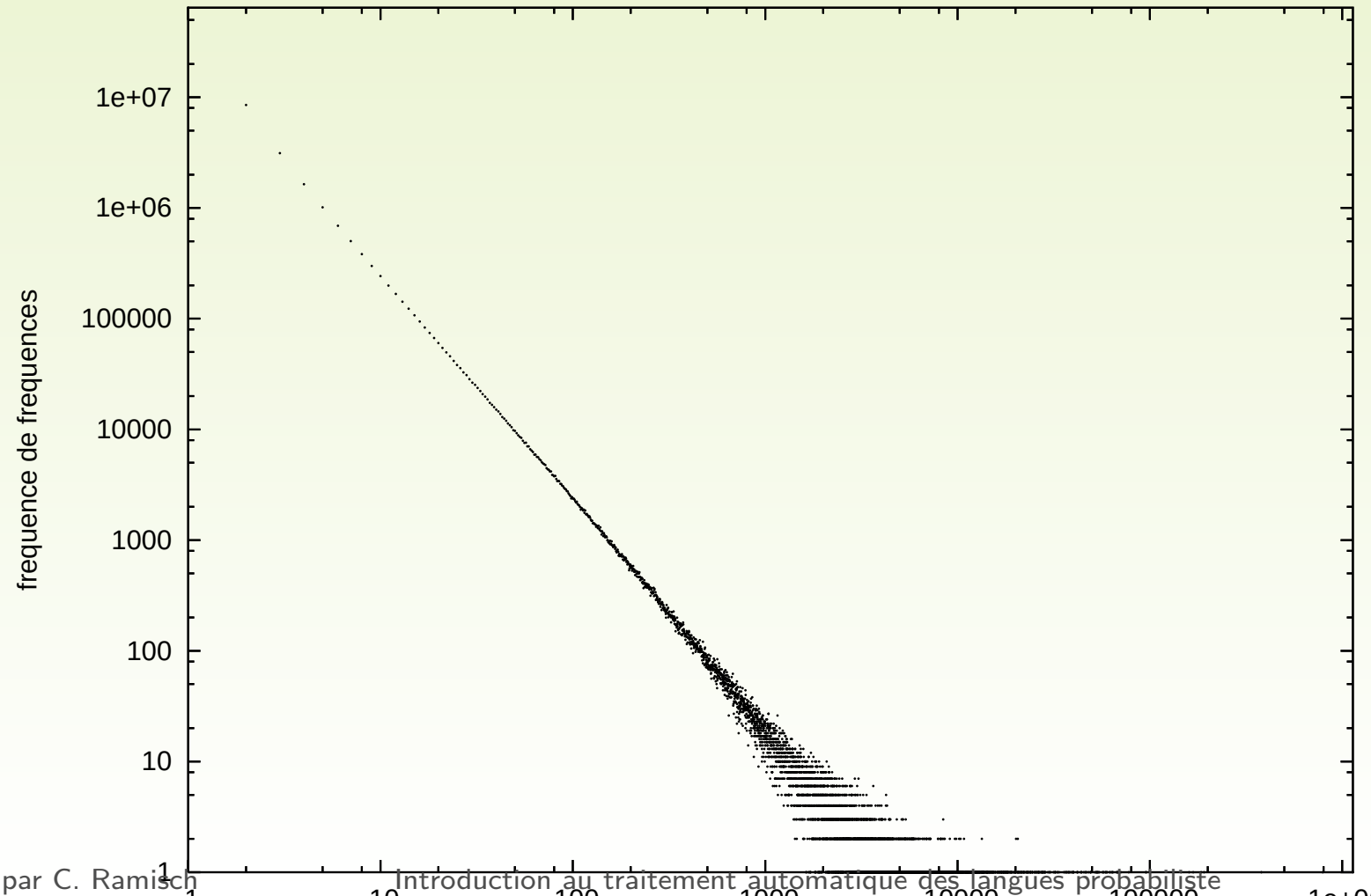
Fréquence des trigrammes, lin-lin



Fréquence des trigrammes, log-lin



Fréquence des trigrammes, log-log



Méthodologie du TAL empirique

Traitement automatique des langues à base de corpus :

1. Définir une tâche
2. En déduire un guide d'annotation
3. Collecter des données
4. Demander à des gens d'annoter ces données
5. Créer un système automatique pour faire la tâche
6. Évaluer la qualité du système

Exemple : segmentation en mots

1. Définir une tâche
   Segmenter le chinois en mots
   jeneconnaispaslechinois $\rightarrow$ je ne connais pas le chinois
2. En déduire un guide d'annotation
   Séparer les séquences formant des mots par des espaces
3. Collecter des données
   Récupérer des textes en chinois sur internet
4. Demander à des gens d'annoter ces données
   Demander à des locuteurs natifs d'effectuer la tâche
5. Créer un système automatique pour faire la tâche
   Utiliser un dictionnaire, reconnaitre les mots les plus longs de manière séquentielle
6. Évaluer la qualité du système
   Calculer le nombre d'espaces insérés à tord par rapport au nombre de mots à trouver

Méthodologie I

1. Définir une tâche
   - Utile, réalisable
2. En déduire un guide d'annotation
   - Augmenter l'accord entre les annotateurs
   - Gérer les cas ambigus, inconnus
   - Anticiper les modifications
3. Collecter des données
   - Quantité suffisante
   - Diversité maximale
4. Demander à des gens d'annoter ces données
   - Faut-il des experts ?
   - Plusieurs annotations
   - Certains tâches sont annotées implicitement

Méthodologie II

5. Créer un système automatique pour faire la tâche
   - Créer des systèmes “baseline” (sorties aléatoire, systématiques...)
   - Minimiser l’erreur sur des données non vues
6. Évaluer la qualité du système
   - Évaluation manuelle empêche de *tricher*
   - Évaluation automatique accélère le développement de système
   - Utilité dans une tâche réelle

Architecture générale des outils de TAL

Structuration du texte ou du signal de parole

Trois opérations formelles :

- Segmentation
  - d'un texte en phrases
  - d'un tour de parole en unités macrosyntaxiques
  - d'une phrase en mots
  - d'une phrase en entités nommées
  - d'un mot en morphèmes
- Etiquetage
  - d'un mot en partie de discours
  - d'une conversation en actes de dialogues
  - d'un document en thèmes
  - d'un mot en sens
- Etablissement de relations
  - morphologiques
  - syntaxiques
  - sémantiques
  - discursives

Architecture générique (version naïve)

Etant donné une entrée X

1. Enumération de toutes les solutions possibles

$$\mathcal{Y} = \{Y_1 \dots Y_n\}$$

2. Pondération des solutions

$$p(Y_i)$$

3. Sélection de la solution de meilleur poids

$$\hat{Y} = \underset{Y \in \mathcal{Y}}{\text{argmax}}\, p(Y)$$

Enumération : Segmentation

$X =$	a	b	c	d
	a |	b |	c |	d
	a	b |	c |	d
$\mathcal{Y} =$	a |	b	c |	d
	a |	b |	c	d
		...		
	a	b	c	d

- 2^{n-1} segmentations possibles

Enumération : Etiquetage

$X =$	a	b	c	d
	1	1	1	1
	1	1	1	2
$\mathcal{Y} =$	1	1	2	1
		...		
	k	k	k	k

- k^n étiquetages possibles
- la segmentation peut être vue comme un cas particulier d'étiquetage (2 étiquettes : {`debut`, `interieur`})

Enumeration : Relations

$X =$	a	b	c	d
	a	a	a	a
	a	a	a	b
$\mathcal{Y} =$	a	a	b	a
		...		
	d	d	d	d

- Fonction *est le fils de*
- n^n graphes possibles

Pondération

- Le poids d'une solution est une fonction des poids des parties : (unités minimales)

$$p(Y) = \sum_{y \in \mathcal{F}(Y)} p(y)$$

- $p(y)$ est le poids de la partie y
- La décomposition de Y en parties est un compromis :
  - trop petites, elles ne sont pas très riches linguistiquement
  - trop grosses, leur poids est difficile à estimer

Estimation des poids

Les poids des parties sont estimés à partir de corpus annotés (X_i, Y_i)

- Modèles génératifs

$$\hat{M} = \underset{M}{\operatorname{argmax}}\, P_M(X, Y)$$

- Modèles discriminants

$$\hat{M} = \underset{M}{\operatorname{argmax}}\, P_M(Y|X)$$

Recherche de la solution de meilleur poids

$$\hat{Y} = \underset{Y \in \mathcal{Y}}{\text{argmax}} \sum_{y \in Y} p(y)$$

Recherche de la solution de meilleur poids

$$\hat{Y} = \underset{Y \in \mathcal{Y}}{\text{argmax}} \sum_{y \in Y} p(y)$$

En pratique $\mathcal{Y}$ n'est jamais énuméré

- Ce n'est pas envisageable d'un point de vue calculatoire
- Ce n'est pas satisfaisant car la plupart des solutions partagent des parties communes
- L'ensemble des solutions est représenté sous la forme d'une structure partagée $\mathcal{F}(\mathcal{Y})$

$$\hat{Y} = \underset{Y \in \mathcal{F}(\mathcal{Y})}{\text{argmax}} \sum_{y \in Y} p(y)$$

Bibliographie

- C. Manning and H. Schütze, *Foundations of Statistical Natural Language Processing*
- D. Jurafsky and J. H. Martin, *Speech and Language Processing*
- H. Baayen, *Word Frequency Distributions*