Chapitre 4.2 :
Modèle de langue et RI
Language model

Plan

- Introduction au modèle de langue
  - Qu'est ce qu'un modèle de langue
  - Estimation d'un modèle de langue
- Modèle de langue et RI
  - Intuition LM et RI
  - Adaptation LM à la RI
  - Modèle vraisemblance de la requête (Query Likelihood)
  - Références bibliographiques

Modèle de langue

- Modèle de langue/language Model (modèle statistique de langue)
  - Capturer la distribution des mots dans une langue (ou d'un texte).
  - Mesure la probabilité d'observer une séquence de mots dans une langue
    - p1=P (un garçon mange une pomme)
    - p2=P (une pomme mange un garçon)
    - p3=P (apple mange un garçon)

- "The goal of a language model is to assign a probability to a sequence of words by means of a probability distribution"

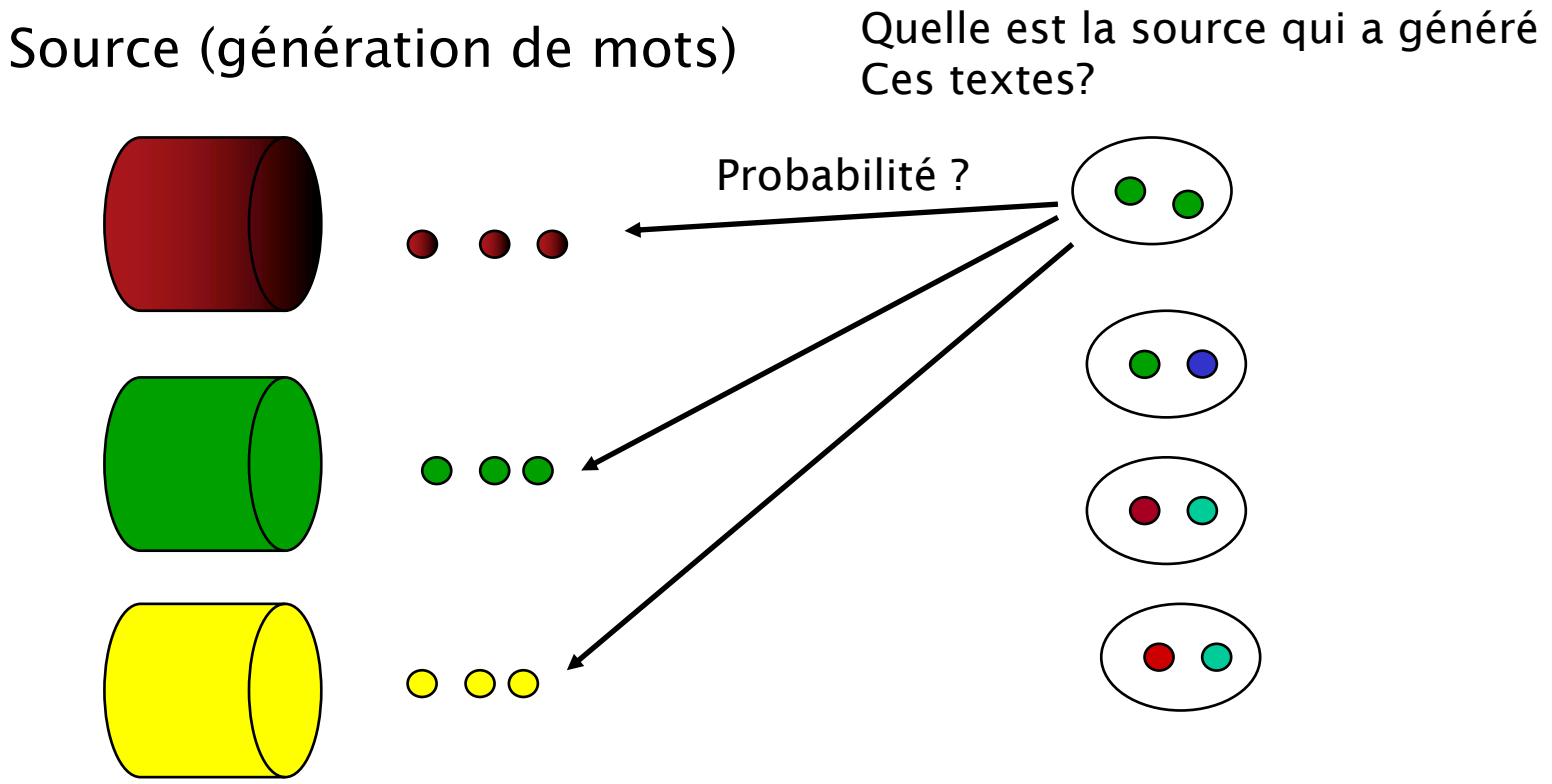
  © wikipédia

Modèle de langue

- Utilisé dans plusieurs applications du traitement automatique de la langue :
  - speech recognition,
  - machine translation,
  - part-of-speech tagging,
  - parsing et information retrieval.

Modèle de langue

- Vu comme une source ou un générateur de textes
  - Mécanisme probabiliste de génération de texte (mots, séquence de mots) →On parle de modèle génératif



Modèle de langue

- Un modèle de langue est défini par son vocabulaire (mots simples, séquence de mots)
- Chaque mot (m)/séquence de mots($m_1 m_2 .. m_n$) a une probabilité d'être généré(e)
- Le but est de calculer → P(s|M)
  - s une observation (séquence de mots/texte) quelconque
  - Probabilité d'observer s dans le modèle (la langue) M

Définir un modèle de langue

- Définir la taille des séquences générées par le modèle ?
  → Séquence de 1 mot, 2 mots, 3 mots, …

- Estimer le modèle → probabilité de chaque séquence générée ?

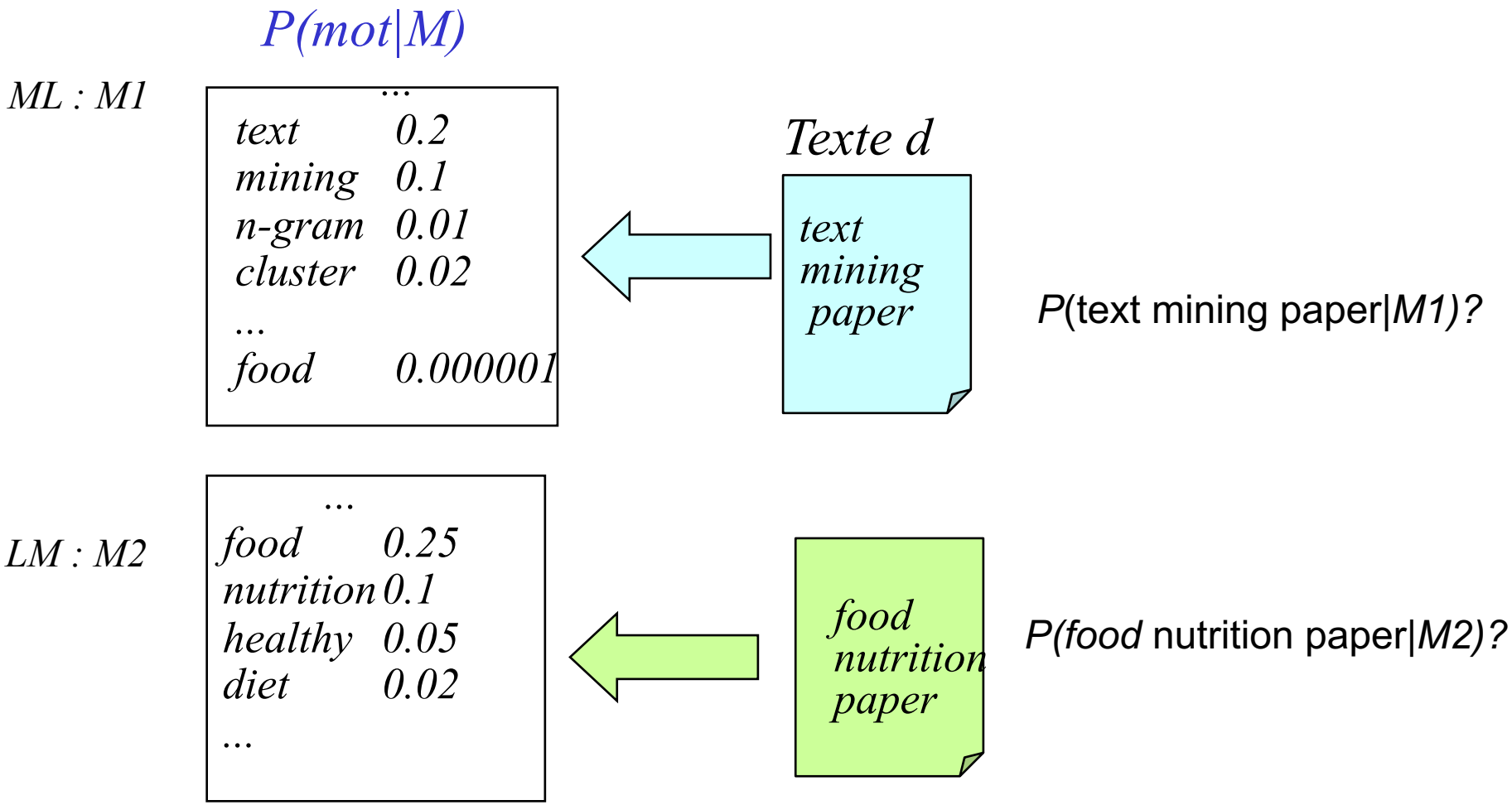
- Calculer la probabilité d'une observation (un texte) quelconque?

Taille de la séquence

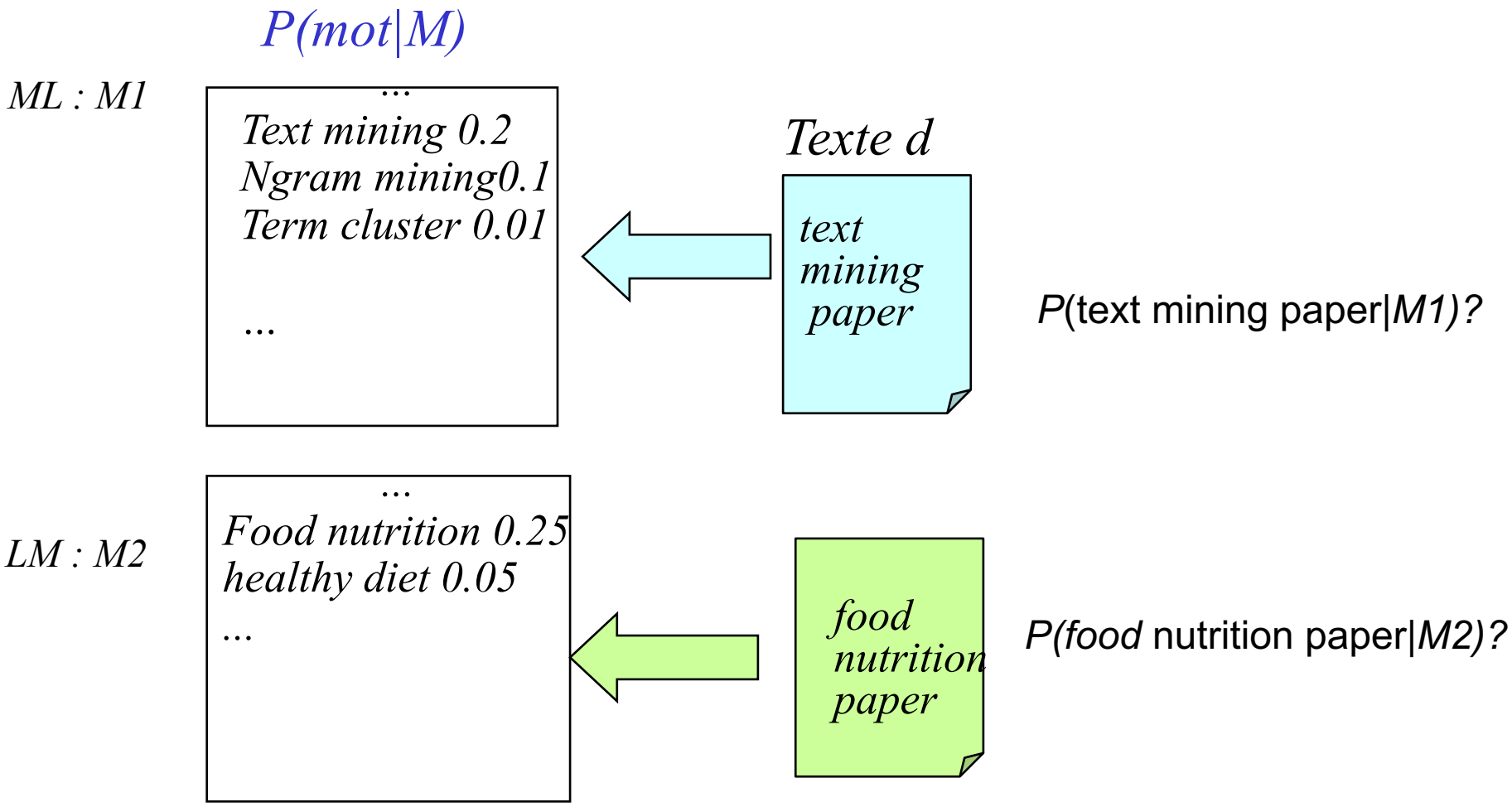
- Différents modèles
  - Séquence d'un mot → modèle *unigram*
  - Séquence de deux mots → modèle *bigram*
  - Séquence de n mots → modèle de *ngram*

- Dans le cas du modèle *unigram* (le plus utilisé en RI)
  - Les textes sont donc « générés » à partir de mots simples générés de manière indépendante les uns des autres
  - Si m_1, m_2, ..m_N est le vocabulaire (les mots acceptés) par le modèle alors
    - Chaque mot m a une probabilité →P(m|M)
    - $P(m_1)+P(m_2)+\ldots P(m_N)= 1$

Exemple de modèle unigram



Exemple de modèle bi-gram



Probabilité d'une observation (séquence)

- Dépend du modèle
  - soit s une observation (un texte) de n mots $s=m_1\ m_2 \ldots m_n$
  - Unigram – (M génère des séquences de 1 mot)

$$P(s \mid M) = P(m_1 m_2 \ldots m_n) = \prod_{i=1}^{n} P(m_i \mid M)$$

  - bigram – (M génère des séquences de deux mots)

$$P(s) = \prod_{i=1}^{n} P(m_i \mid m_{i-1}) = \prod_{i=1}^{l} \frac{P(m_{i-1} m_i)}{P(m_{i-1})}$$

  - ngram – (M génère des séquences de 3 mots)

$$P(s) = \prod_{i=1}^{n} P(m_i \mid m_{i-2} m_{i-1}) = \prod_{i=1}^{n} \frac{P(m_{i-2} m_{i-1} m_i)}{P(m_{i-2} m_{i-1})}$$

Estimation du modèle

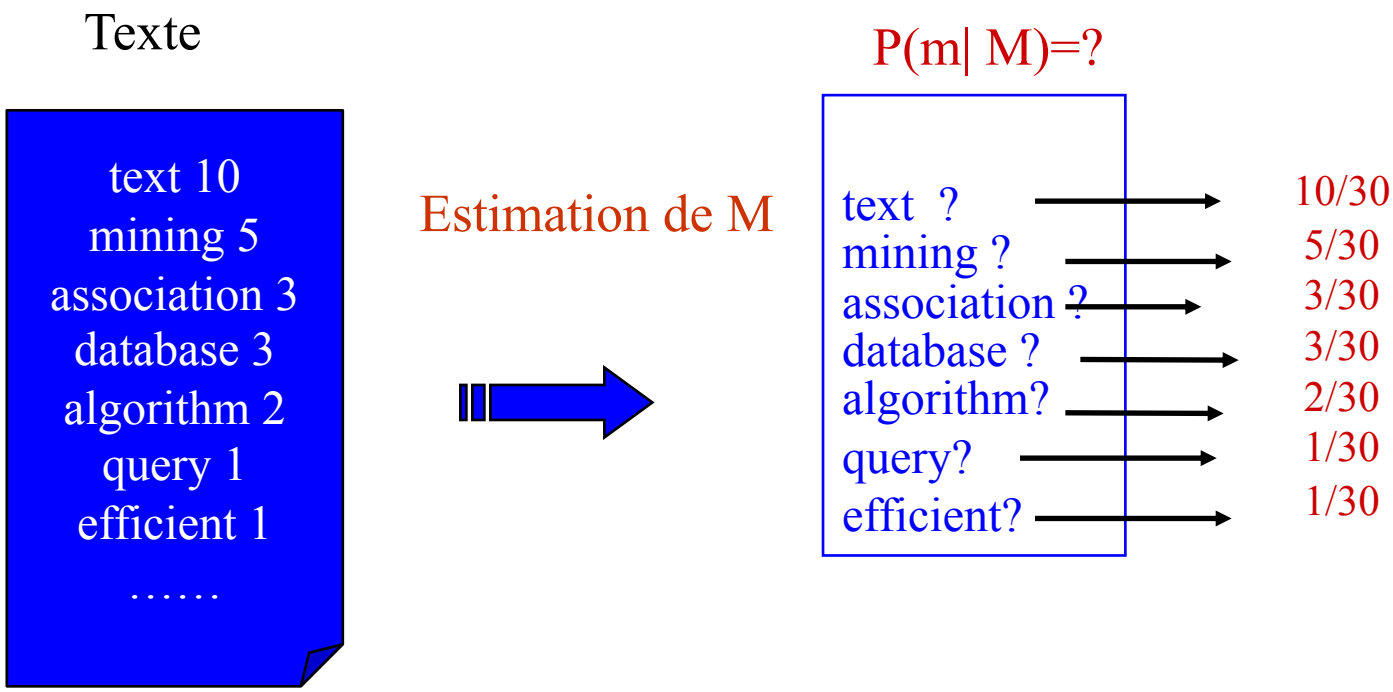
- Selon le modèle il faut estimer
  - $P(m_i)$, $P(m_{i-1}\ m_i)$, $P(m_{i-2}\ m_{i-1}\ m_i)$, ..
- Estimation par Maximum de vraisemblance (*Maximum likelihood*) (la plus fréquente)
  - Compter la fréquence relative de l'événement (m) dans l'échantillon (C)

$$P(m \mid C) = \frac{freq(m)}{\sum_{m \in C} freq(m)}$$

Exemple

- Estimation d'un modèle uni-gram (simple) par ml
  - Compter la fréquence relative des mots m : $P_{ml}(m|M) = \#(m) / N$

Un texte sur
le "text mining"
(total #mots=30)



P("text query")=P(text)*P(query)=(10/30)*(1/30)

P("text retrieval")=P(text)*P(retrieval)=(10/30)*(0)

Problème des fréquences nulles (zéro)

- Si un événement (un mot de la séquence) n'apparait pas dans le modèle, le modèle lui assigne une probabilité 0

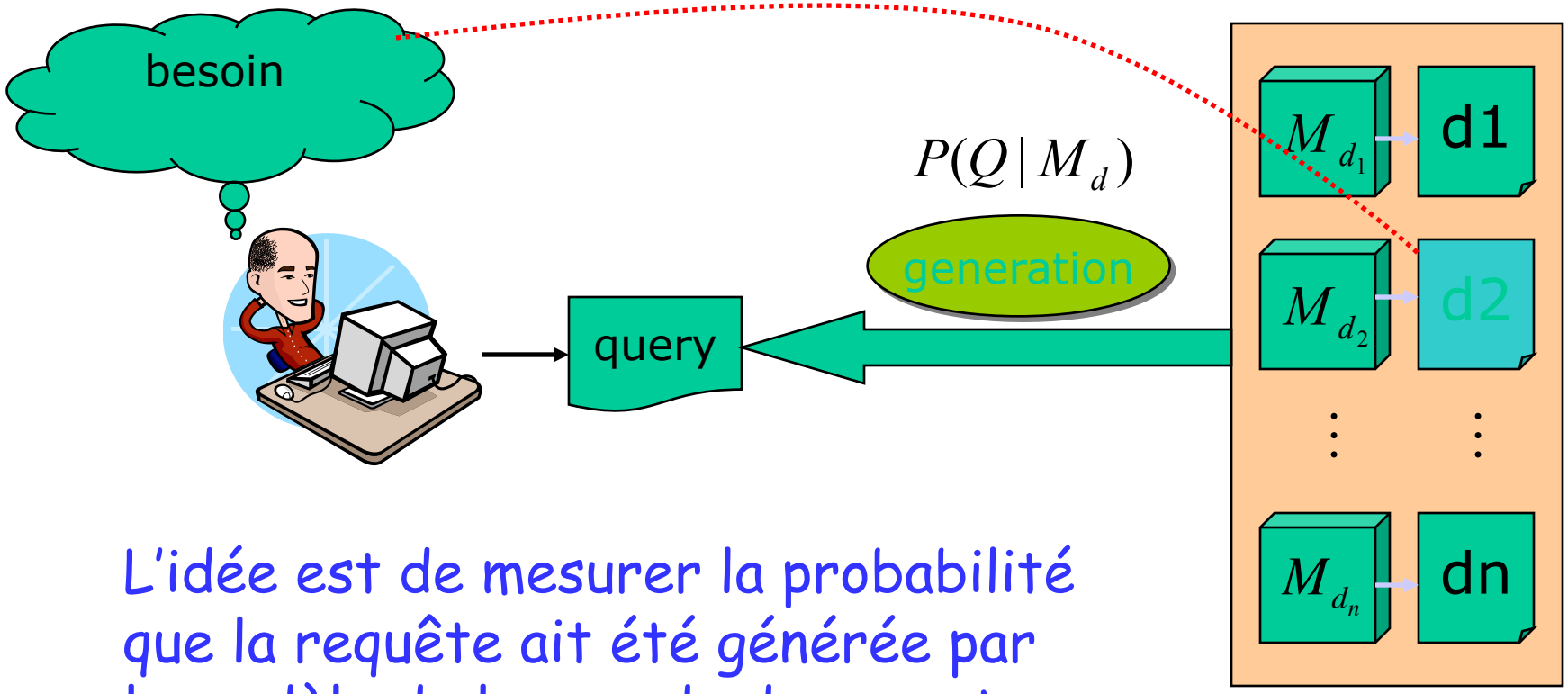
$$P(s \mid M) = \prod_{i=1}^{l} P(m_i \mid M) = 0, \quad si \quad \exists m_i / P(m_i \mid M) = 0$$

- Solution : assigner des probabilités différentes de zéro aux événements (mots) absents
  - → Lissage (Smoothing)

Modèle de Langue en RI

Plusieurs modèles, plusieurs adaptations

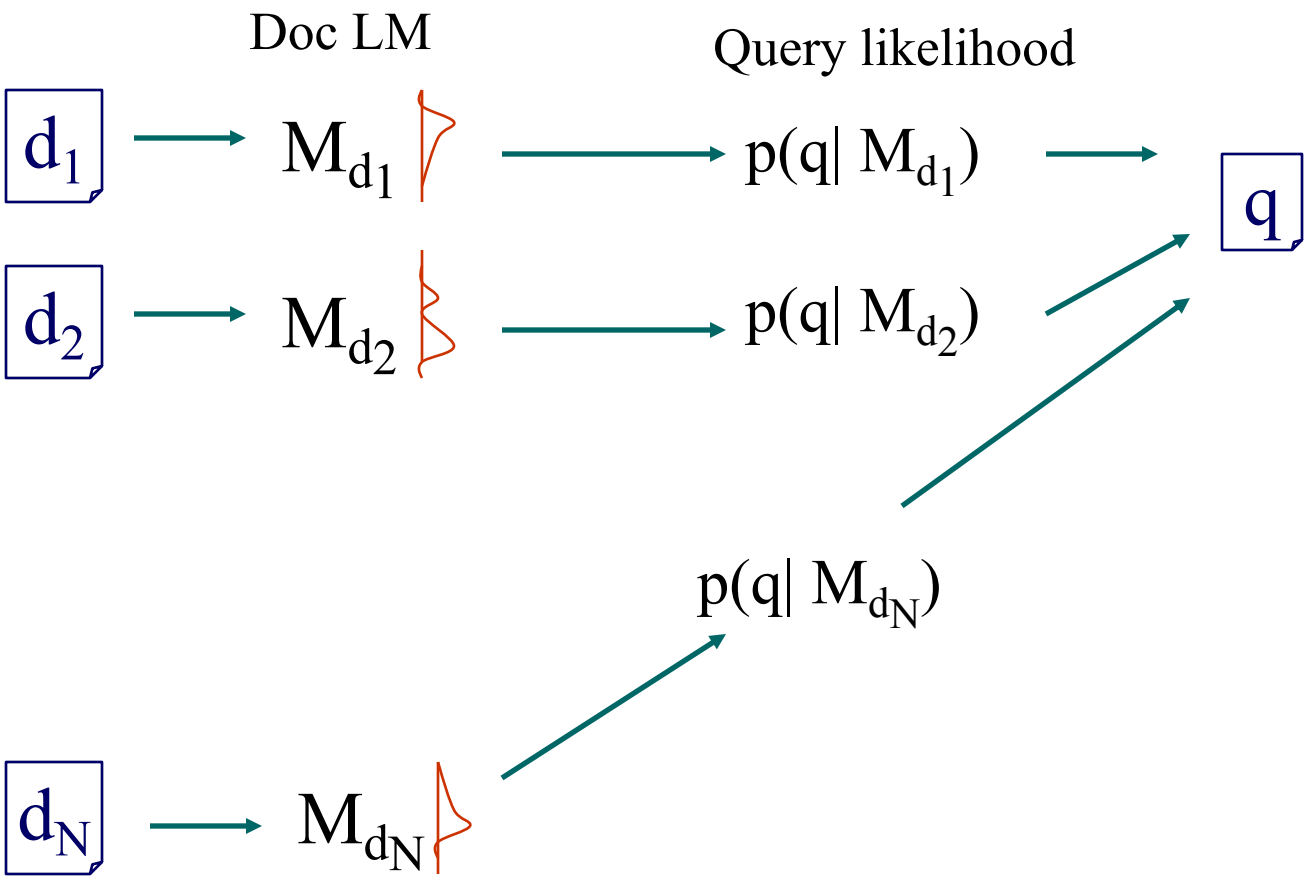
IR et LM : intuition



L'idée est de mesurer la probabilité que la requête ait été générée par le modèle de langue du document considéré

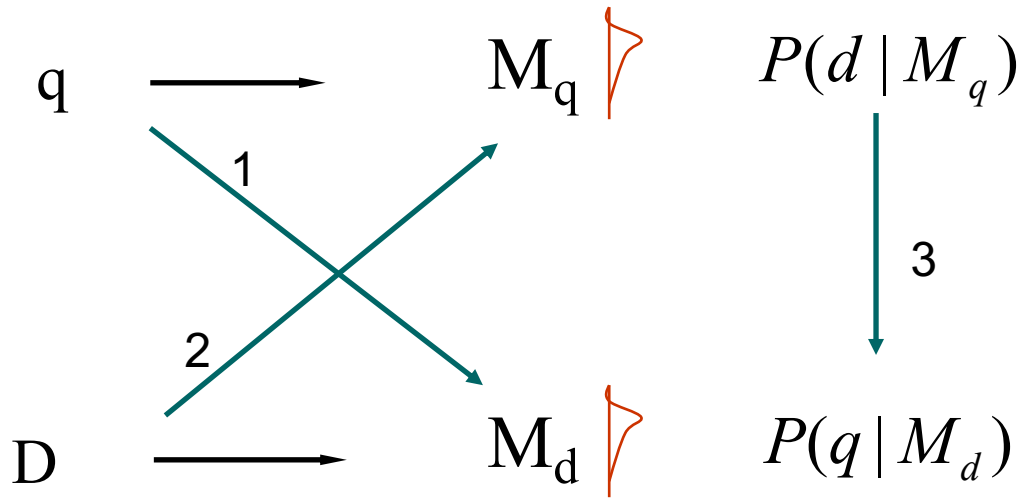
$P(Q|M_d)$

RI et ML : illustration



ML et RI : Plusieurs adaptations possibles

- Il existe plusieurs manières d'adapter les ML à la RI.



3 Principes

- (1) : Probabilité de générer la requête à partir de M_d
- (2) : Probabilité de générer le document à partir de Mq
- (3) : Combinaison (comparaison) des deux modèles

ML et RI : Plusieurs adaptations possibles (suite)

- Principe 1: Vraisemblance de la requête (Query-likelihood)
  - RSV(d,Q)= $P(Q|M_d)$
  - Document d représenté par son ML $P(w|M_d)$
  - Requête Q = séquence ou vecteurs de mots $q_1,q_2,\ldots,q_n$
- Principe 2 : Vraisemblance du document (Document likelihood)
  - RSV(d,Q)= P(D|Mq)
  - Requête Q représentée par son ML P(w|Mq)
  - Document d = séquence ou vecteurs de mots
- Principe 3: comparaison de modèles
  - Document d: LM $P(w|M_D)$
  - Requête Q: LM $P(w|M_Q)$
  - RSV(Q,d): comparer $P(w|M_d)$ and $P(w|M_Q)$

Principe 1 : Vraisemblance de la requête (Query Likelihood)

- Approche standard
  - Estimer le modèle de chaque document
  - Trier les documents selon leur probabilité de générer la requête → $P(Q|M_d)$

$$RSV(Q,D) = P(q|M_D) = P(t_1,t_2,...,t_n|D) = \prod_{t_i \in Q} P(t_i|D)$$

Estimation de M_d ?

- Le modèle de langue est inconnu mais, nous disposons d'un échantillon → le document
- Estimer le modèle à partir du document
  - Maximum de vraissemblance (Maximun Likelihood Estimator)

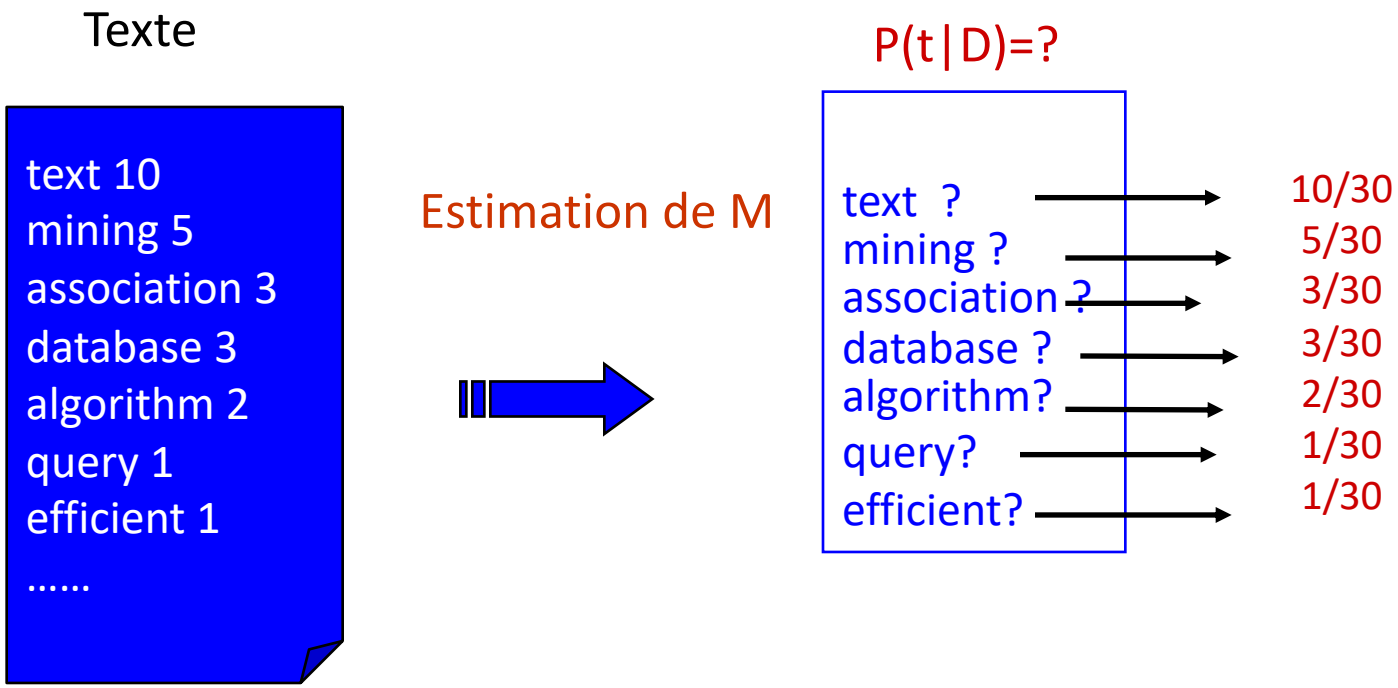
$$P_{ml}(t_i \mid D) = \frac{tf_{(t_i,d)}}{|d|}$$

Les termes de la requête sont générés de manière indépendantes

Les modèles de langue : Exemple

- Estimation d'un modèle uni-gram (simple) par ml
  - Compter la fréquence relative des mots m : $P_{ml}(t|D) = tf(t,d) / dl$

Un texte sur
le "text mining"
(total #mots=30)



P("text query")=P(text)*P(query)=(10/30)*(1/30)

P("text retrieval")=P(text)*P(retrieval)=(10/30)*(0)

Retour sur le problème des fréquences Zéro

- Problème des tf = 0
  - quand un document ne contient pas un ou plusieurs termes de la requête.
- Contraintes
  - On ne peut pas assigner des valeurs différentes de zéro de manière aléatoire
  - La somme des probabilités de l'ensemble des événements doit être égale à 1.
  - Plusieurs solutions ….

Techniques de lissage (Smoothing)

Techniques de lissage

- Méthodes de « discounting »
  - Laplace correction, Lindstone correction, absolute discounting, leave one-out discounting, Good-Turing method
- Techniques d'Interpolation
  - Estimations de Jelinek-Mercer, Dirichlet

Lissage par interpolation

- Interpolation (Jelinek-Mercer)
  - Combiner le modèle M avec un modèle plus général (Modèle de référence)

$$P_{JM}(t \mid M) = \lambda . P_{ML}(t \mid M) + (1-\lambda) P_{ML}(t / REF)$$

  - Pb. “Règlage” de λ

Lissage par interpolation (JM)

- Modèle lissé (JM)

$$P_{JM}(t \mid d) = \alpha \times P_{ML}(t \mid d) + (1-\alpha) P_{ML}(t \mid C)$$

$$P(t \mid M_c) = p(t) = \frac{tf(t,C)}{\sum tf(t',C)}$$

Exemple

P("text retrieval")=P(text)*P(retrieval)=(10/30)*(0)

Collection 100 documents (total tf ds C (2000)
tf(retrieval,C)=6 → P_{ml}(retrieval/C)=6/2000
tf(text,C)=25→ P_{ml}(text/C)= 25/2000

Lissage par interpolation (suite)

- Lissage de Dirichlet
  - Problème avec Jelinek-Mercer
    - Les documents longs seront privilégiés
  - Prendre en compte la taille de l'échantillon
  - Si N est la taille de l'échantillon et μ une constante

$$P_{Dir}(t \mid M) = \left(\frac{N}{N+\mu}\right).P_{ML}(t \mid M) + \left(\frac{\mu}{N+\mu}\right)P_{ML}(t / REF)$$

Lissage par interpolation (suite)

- Lissage de Dirichlet en RI

$$P_{Dir}(t \mid d) = \frac{|d|}{|d| + \mu} \times \frac{tf(t,d) +}{|d|} + \frac{|\mu|}{|d| + \mu} P_{ML}(t \mid C)$$

$$P_{Dir}(t \mid d) = \frac{tf(t,d) + \mu P_{ML}(t \mid C)}{|d| + \mu}$$

$$\alpha = \frac{|d|}{\mu + |d|} \qquad 1 - \alpha = \frac{\mu}{\mu + |d|}$$

P("text retrieval")=[(30/(100+30))P(text/d)+(100/100+30)P(text/C)]*
[(30/(100+30))P(retrieval/d)+(100/(100+30))P(retrieval/C)]

Meilleure méthode de lissage?

- Dépend des données et de la tâche
- Dirichlet semble bien fonctionner pour la RI

Il existe d'autres méthodes de lissage
Voir [Chen & Goodman 98]

Exemple

- (2 documents)
  - d_1: Xerox reports a profit but revenue is down
  - d_2: Lucent narrows quarter loss but revenue decreases further
- Requête: *revenue down*
- MLE unigram;
  - Lissage JM $\lambda = ½$
  - Lissage Dir, $\mu = 1$

Principe 2 : Vraisemblance du document

- Chaque requête est traitée comme un modèle de langage
- Estimer le modèle de langage M_q de chaque requête
- Classer les documents

$$P(D \mid M_Q) = \prod_{t \in Q} P(t \mid M_Q)$$

- M_q peut être vu comme un modèle qui estime le document pertinent type

Principe 3: comparaison de modèles

- Combiner les avantages de deux méthodes de tri des documents
  - Estimer le modèle de requête M_Q et celui du document M_d puis comparer les modèles
  - Mesure naturelle de la similarité entropie croisée

$$H(M_q \| M_d) = -\sum_t P(t / M_q) \log(P(t / M_d))$$

  - Autre mesure Kullback-Leiblar divergence

$$RSV(Q,d) = H(M_Q \| M_d) - H(M_Q \| M_Q)$$

$$RSV(Q,d) = \sum_t P(t / M_q) \log \frac{P(t / M_d)}{P(t / M_q)}$$

Résumé choix LM pour la RI

- Choisir un modèle unigram
  - pas besoin d'aller au-delà des mots simples
- Choisir un modèle multinomial
  - Simple et performant vis-à-vis des autres modèles
- Choisir les modèles basés sur le principe 3
  - Permettent d'intégrer le feedback, expansion
- Estimation M_d et M_q une des questions importantes en LM