

# Latent Semantic Analysis

[Landauer *et al.*, 98]

**Mathieu Roche**

**Cours ECD (Recherche  
d'Information et Langage Naturel)**

**2007/2008**

# Plan du cours

- Introduction
- Méthode
- Mesure de similarité utilisée
- Exemple
- Perspectives : pré-traitement des données
- Discussion

# Introduction

- ***Motivations*** : trouver la similarité entre deux mots (ou deux textes).
- ***Cadre de travail*** : ensemble de documents textuels.

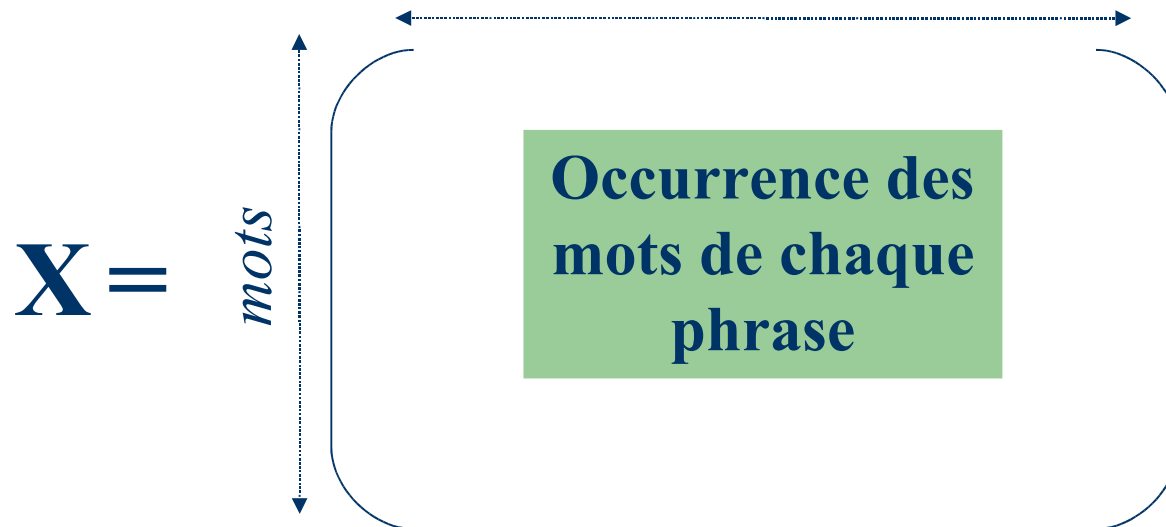
# Type de méthode

- Méthode non supervisée
- Méthode qui s'appuie sur le contexte des mots.

# Méthode (1/6)

- Matrice relative aux mots du texte

- *phrases*
- *paragraphes*
- *documents*



# Méthode (2/6)

- Normalisation (1ère méthode) :

$$\log(1+x_i)$$

+



# Méthode (3/6)

- Normalisation (2ème méthode) :

Utilisation de la méthode du «  $TF \times IDF$  » [Salton, 89] pour normaliser [Turney, 01].

Une formule  $tf*idf$  combine deux critères :

- l'importance du terme pour un document (par  $tf$ )
- le pouvoir de discrimination de ce terme (par  $idf$ ).

Ainsi, un terme qui a une valeur de  $tf*idf$  élevée doit être à la fois important dans ce document, et aussi il doit apparaître peu dans les autres documents. C'est le cas où un terme correspond à une caractéristique importante et unique d'un document.

# Méthode (4/6)

- Normalisation (2ème méthode) : *TF X IDF*

$$w_{ij} = tf_{ij} \cdot \log_2 \frac{N}{n}$$

- $w_{ij}$  = poids du terme  $T_j$  dans le document  $D_i$
- $tf_{ij}$  = fréquence du terme  $T_j$  dans le document  $D_i$
- $N$  = nombre de documents dans la collection
- $n$  = nombre de documents où le terme  $T_j$  apparaît au moins une fois

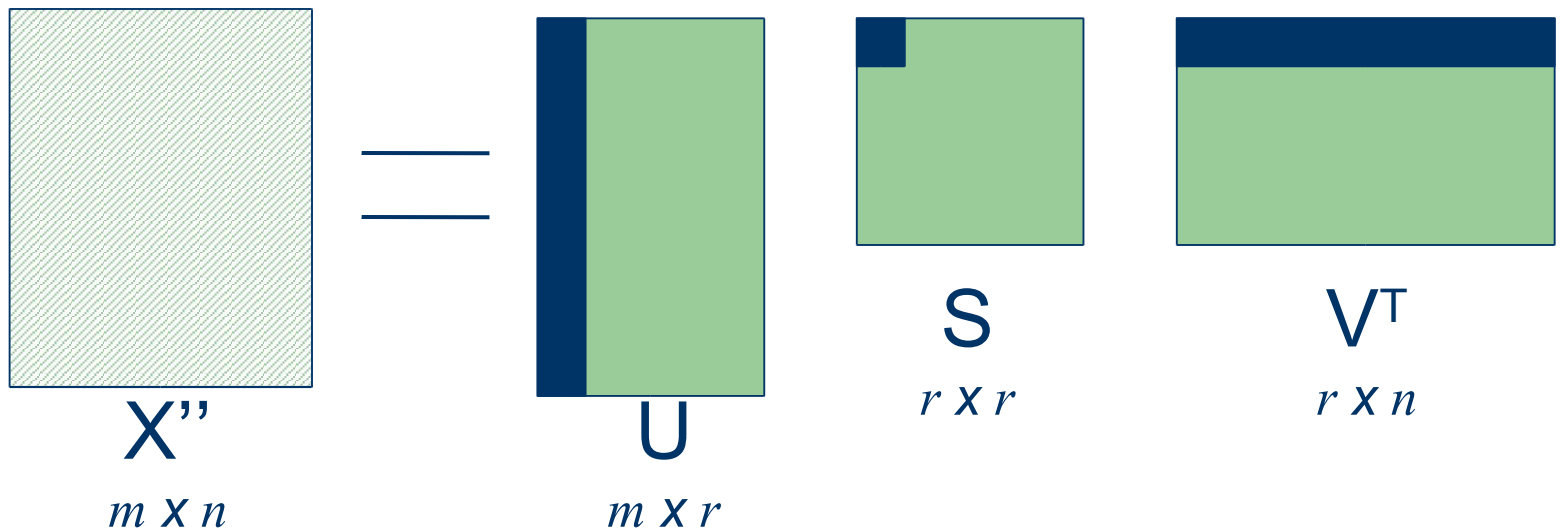
# Méthode (5/6)

- Décomposition en valeurs propres : *une matrice de rang  $r$  peut se décomposer de la manière suivante*

$$\begin{matrix} \text{[Dark Green Box]} \\ X' \\ m \times n \end{matrix} = \begin{matrix} \text{[Light Green Box]} \\ U \\ m \times r \end{matrix} \begin{matrix} \text{[Light Green Box]} \\ S \\ r \times r \end{matrix} \begin{matrix} \text{[Light Green Box]} \\ V^T \\ r \times n \end{matrix}$$

# Méthode (6/6)

- Approximation de la matrice  $X'$  : *construction sur seulement  $d$  dimensions d'une matrice  $X''$  qui est une approximation de la matrice d'origine.*



# Mesure de similarité utilisée

- Mesure de Spearman (tendance des données à varier ensemble)

$$R = \frac{\frac{\sum_{i=1}^n (x_i - \bar{x})(y_i - \bar{y})}{n-1}}{\sqrt{\frac{\sum_{i=1}^n (x_i - \bar{x})^2}{n-1}} \sqrt{\frac{\sum_{i=1}^n (y_i - \bar{y})^2}{n-1}}}$$

où  $-1 \leq R \leq 1$

$x = (x_1 \dots x_n)$  et  $y = (y_1 \dots y_n)$

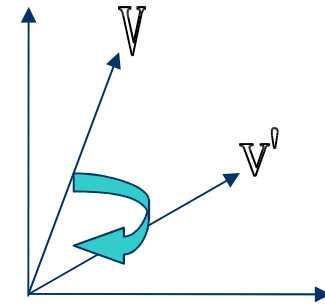
# Mesure de similarité utilisée

- Mesure de Spearman
  - Si  $R = -1$  alors *corrélation négative parfaite*
  - Si  $R = 1$  alors *corrélation positive parfaite*

# Mesure de similarité utilisée

- D'autres mesures, par exemple, le *cosinus* ...

$$a_1, \dots, a_j, \dots, a_L$$
$$b_1, \dots, b_j, \dots, b_L$$



$$\cos \alpha = \cos(\mathbf{V}, \mathbf{V}') = \frac{\sum_{j=1}^L a_j b_j}{\sqrt{\sum_{j=1}^L a_j^2} \sqrt{\sum_{j=1}^L b_j^2}}$$

# Exemple d'utilisation de LSA (1/8)

- c1: Human machine interface for ABC computer applications
  - c2: A survey of user opinion of computer system response time
  - c3: The EPS user interface management system
  - c4: System and human system engineering testing of EPS
  - c5: Relation of user perceived response time to error measurement
- 
- m1: The generation of random, binary, ordered trees
  - m2: The intersection graph of paths in trees
  - m3: Graph minors IV: Widths of trees and well-quasi-ordering
  - m4: Graph minor: A survey

# Exemple d'utilisation de LSA (1/8)

- c1: *Human* machine *interface* for ABC *computer* applications
- c2: A *survey* of *user* opinion of *computer system response time*
- c3: The *EPS user interface* management *system*
- c4: *System* and *human system* engineering testing of *EPS*
- c5: Relation of *user* perceived *response time* to error measurement
  
- m1: The generation of random, binary, ordered *trees*
- m2: The intersection *graph* of paths in *trees*
- m3: *Graph minors* IV: Widths of *trees* and well-quasi-ordering
- m4: *Graph minors*: A *survey*

# Exemple d'utilisation de LSA (2/8)

$X =$

	c1	c2	c3	c4	c5	m1	m2	m3	m4
human	1	0	0	1	0	0	0	0	0
interface	1	0	1	0	0	0	0	0	0
computer	1	1	0	0	0	0	0	0	0
user	0	1	1	0	1	0	0	0	0
system	0	1	1	2	0	0	0	0	0
response	0	1	0	0	1	0	0	0	0
time	0	1	0	0	1	0	0	0	0
EPS	0	0	1	1	0	0	0	0	0
survey	0	1	0	0	0	0	0	0	1
trees	0	0	0	0	0	1	1	1	0
graph	0	0	0	0	0	0	1	1	1
minors	0	0	0	0	0	0	0	1	1

$R = -0.29$  (between 'human' and 'minors')

$R = -0.38$  (between 'human' and 'user')

# Exemple d'utilisation de LSA (3/8)

## *Intuition de l'approximation :*

- m1: The generation of random, binary, ordered trees
- m2: The intersection graph of paths in trees
- m3: Graph minors IV: Widths of trees and well-quasi-ordering
- m4: Graph minors: A survey

	c1	c2	c3	c4	c5	m1	m2	m3	m4
	...	...	...	...	...	...	...	...	...
survey	0	1	0	0	0	0	0	0	1
trees	0	0	0	0	0	1	1	1	0
graph	0	0	0	0	0	0	1	1	1
minors	0	0	0	0	0	0	0	1	1

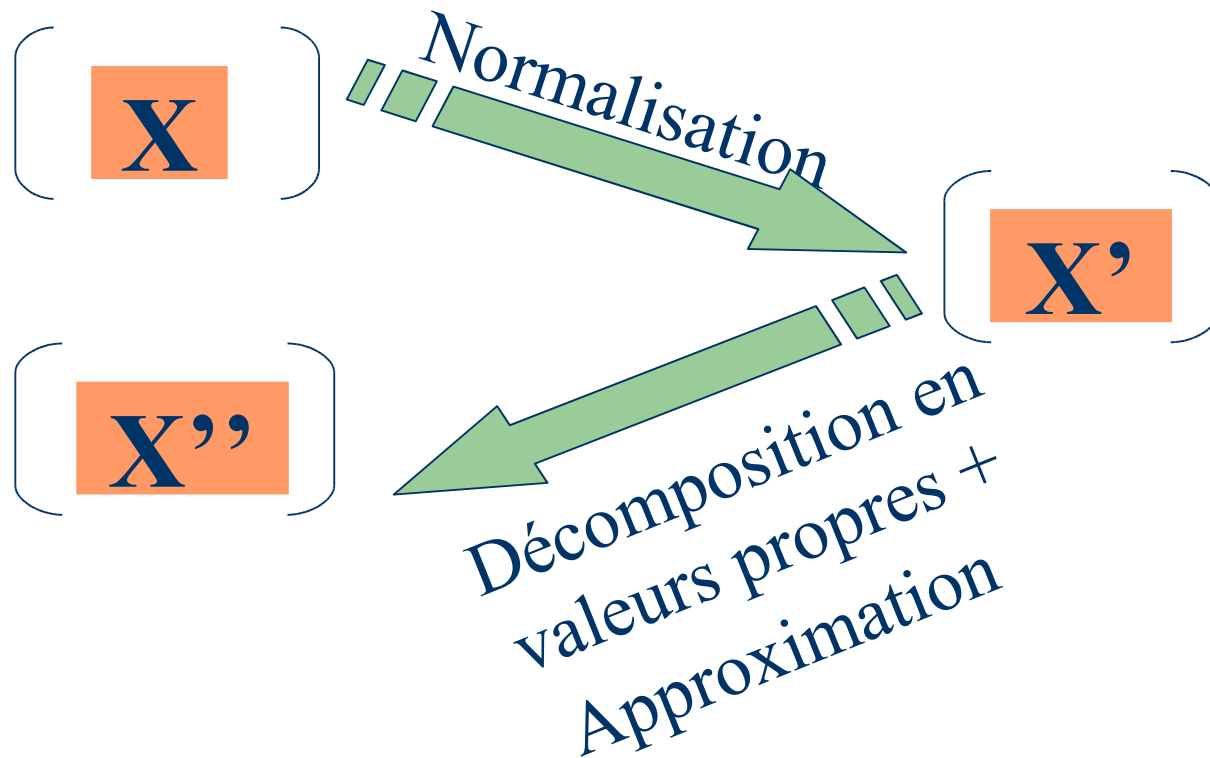
# Exemple d'utilisation de LSA (4/8)

## Intuition de l'approximation :

- m1: The generation of random, binary, ordered *trees*
- m2: The intersection *graph* of paths in *trees*
- m3: *Graph minors* IV: Widths of *trees* and well-quasi-ordering
- m4: *Graph minors*: A survey

	c1	c2	c3	c4	c5	m1	m2	m3	m4	
	...	...	...	...	...	...	...	...	...	
survey	0	1	0	0	0	0	0	0	1	
<u>trees</u>	0	0	0	0	0	1	1	1	<del>0</del>	→ 0.66
<u>graph</u>	0	0	0	0	0	1	1	1		
<u>minors</u>	0	0	0	0	0	0	1	1		

# Exemple d'utilisation de LSA (5/8)



# Exemple d'utilisation de LSA (6/8)

	c1	c2	c3	c4	c5	m1	m2	m3	m4
human	0.16	0.40	0.38	0.47	0.18	-0.05	-0.12	-0.16	-0.09
interface	0.14	0.37	0.33	0.40	0.16	-0.03	-0.07	-0.10	-0.04
computer	0.15	0.51	0.36	0.41	0.24	0.02	0.06	0.09	0.12
user	0.26	0.84	0.61	0.70	0.39	0.03	0.08	0.12	0.19
system	0.45	1.23	1.05	1.27	0.56	-0.07	-0.15	-0.21	-0.05
$X'' =$ response	0.16	0.58	0.38	0.42	0.28	0.06	0.13	0.19	0.22
time	0.16	0.58	0.38	0.42	0.28	0.06	0.13	0.19	0.22
EPS	0.22	0.55	0.51	0.63	0.24	-0.07	-0.14	-0.20	-0.11
survey	0.10	0.53	0.23	0.21	0.27	0.14	0.31	0.44	0.42
trees	-0.06	0.23	-0.14	-0.27	0.14	0.24	0.55	0.77	0.66
graph	-0.06	0.34	-0.15	-0.30	0.20	0.31	0.69	0.98	0.85
minors	-0.04	0.25	-0.10	-0.21	0.15	0.22	0.50	0.71	0.62

# Exemple d'utilisation de LSA (7/8)

	c1	c2	c3	c4	c5	m1	m2	m3	m4
human	0.16	0.40	0.38	0.47	0.18	-0.05	-0.12	-0.16	-0.09
interface	0.14	0.37	0.33	0.40	0.16	-0.03	-0.07	-0.10	-0.04
computer	0.15	0.51	0.36	0.41	0.24	0.02	0.06	0.09	0.12
user	0.26	0.84	0.61	0.70	0.39	0.03	0.08	0.12	0.19
system	0.45	1.23	1.05	1.27	0.56	-0.07	-0.15	-0.21	-0.05
$X'' =$ response	0.16	0.58	0.38	0.42	0.28	0.06	0.13	0.19	0.22
time	0.16	0.58	0.38	0.42	0.28	0.06	0.13	0.19	0.22
EPS	0.22	0.55	0.51	0.63	0.24	-0.07	-0.14	-0.20	-0.11
survey	0.10	0.53	0.23	0.21	0.27	0.14	0.31	0.44	0.42
trees	-0.06	0.23	-0.14	-0.27	0.14	0.24	0.55	0.77	0.66
graph	-0.06	0.34	-0.15	-0.30	0.20	0.31	0.69	0.98	0.85
minors	-0.04	0.25	-0.10	-0.21	0.15	0.22	0.50	0.71	0.62

$R=0.94$

$R=-0.83$

# Exemple d'utilisation de LSA (8/8)

$X =$

	c1	c2	c3	c4	c5	m1	m2	m3	m4
human	1	0	0	1	0	0	0	0	0
interface	1	0	1	0	0	0	0	0	0
computer	1	1	0	0	0	0	0	0	0
user	0	1	1	0	1	0	0	0	0
system	0	1	1	2	0	0	0	0	0
response	0	1	0	0	1	0	0	0	0
time	0	1	0	0	1	0	0	0	0
EPS	0	0	1	1	0	0	0	0	0
survey	0	1	0	0	0	0	0	0	1
trees	0	0	0	0	0	1	1	1	0
graph	0	0	0	0	0	0	1	1	1
minors	0	0	0	0	0	0	0	1	1

$R = -0.29$  (between human and minors)

$R = -0.38$  (between human and user)

# Plan de l'exposé

- Introduction
- Méthode
- Mesure de similarité utilisée
- Exemple
- Perspectives : pré-traitement des données
- Discussion

# Perspectives : pré-traitement des données

- Corpus étudié
  - corpus (3784 Ko) en français, propriété de la société *PerformanSe*, commentant les résultats d'un test de psychologie dans le domaine des Ressources Humaines [Roche & Kodratoff, 03]

# Perspectives : pré-traitement des données

- Lemmatisation et LSA

<b>Lemmatisation</b>	<b>non</b>	<b>oui</b>
Similarité (cosinus)	<b>0.3</b>	
% de termes correctement associés (c-à-d. % de couples corrects)	<b>17.7 %</b> (28/158)	<b>19.2 %</b> (31/161)
Similarité (cosinus)	<b>0.4</b>	
% de termes correctement associés (c-à-d. % de couples corrects)	<b>25.5 %</b> (12/47)	<b>32.1 %</b> (9/28)
Similarité (cosinus)	<b>0.5</b>	
% de termes correctement associés (c-à-d. % de couples corrects)	<b>20.0 %</b> (3/15)	<b>42.9 %</b> (3/7)
Similarité (cosinus)	<b>0.6</b>	
% de termes correctement associés (c-à-d. % de couples corrects)	<b>75.0 %</b> (3/4)	<b>75.0 %</b> (3/4)

# Perspectives : pré-traitement des données

- Suppression des mots vides
  - similarité de **0.3**, une Précision de 17.1% en prenant en compte les mots « vides » --> 19.2% sans les prendre en compte.
  - similarité de **0.4**, nous obtenons une Précision de 24% avec prise en compte des mots « vides » --> 32.1% sans leur prise en compte.

# Discussions

- Rehder *et al.* ont montré que si les contextes (documents) possèdent moins de 60 mots alors la méthode LSA se révèle décevante [Rehder *et al.*, 98]
- Ordre des mots non pris en compte
- Modification de l'algorithme : Wiemer-Hastings [Wiemer-Hastings, 00]
- Travaux de Peter Turney [Turney, 01]

# Construction des classes sémantiques

Les distances entre les vecteurs représentant les mots ou les termes peuvent être regroupés en utilisant des algorithmes classiques de Fouille de Données :

- K plus proches voisins
- K moyennes

# Description de deux méthodes de classification

**K plus proches voisins (KPPV)** - approche supervisée :

- **But** : déterminer les K plus proches voisins des textes à prédire.
- **La classe majoritaire propre à ces K plus proches voisins est choisie pour les textes à prédire** (ou la classe majoritaire après pondération avec la mesure de similarité).
- Cette méthode utilise deux paramètres : la **valeur K** et la **mesure de similarité** (par exemple, la mesure cosinus)

# Description de deux méthodes de classification

## **Les K-moyennes** - approche non supervisée :

- Choix d'un nombre  $K$  de groupes à constituer.
- $K$  textes constitueront les  $K$  centres initiaux des classes.
- Le but est alors d'associer chaque texte représenté sous forme vectorielle au groupe qui est le plus proche.
- Pour chaque nouveau groupe constitué, le nouveau centre est calculé (grâce au calcul de la moyenne)
- Lorsque les groupes sont stables (aucun texte n'a changé de groupe), l'algorithme prend fin.

*Méthode sensible au choix des centres initiaux*

# Ajout de la syntaxe à LSA <sup>(1/4)</sup>

- **Associer la syntaxe à LSA** [Wiemer-Hastings, 1999].
- Chaque phrase est **décomposée** en (sujet, verbe, objet). Moyenne calculée avec les matrices LSA formées à partir de chacune des composantes
- **Avantages :**
  - **Prise en compte de mots “vides”** (« if », « because », « have », etc.).
  - Décomposition des phrases ayant un **même verbe** associé à plusieurs sujets ou plusieurs objets.

# Ajout de la syntaxe à LSA (2/4)

## Autres pistes :

- Ajouter un poids aux mots partageant les **mêmes structures syntaxiques**.
- Donner un **poids plus importants** à certaines structures syntaxiques (par exemple, les verbes).
- Mettre en oeuvre des **méthodes d'apprentissage supervisé** pour déterminer ces poids.

# Ajout de la syntaxe à LSA (3/4)

## Autres pistes : ajout de connaissances syntaxico-sémantiques (Nicolas Béchet, LIRMM)

- Enrichissement du contexte avec une mesure de proximité comme Asium

- Exemple (calculer le cosinus avant et après expansion) :

1. *Le chat poursuit la souris*

2. *Le chaton joue avec le mulot*

Expansion: souris -> (souris ou mulot)

mulot -> (souris ou mulot)

# Ajout de la syntaxe à LSA (3/4)

## Autres pistes : ajout de connaissances syntaxico-sémantiques

- Expansion grâce aux relations syntaxiques (voir cours Asium) :

→ *chasser souris, chasser mulot, poursuivre souris, poursuivre rat*

- **Limite** : problème de polysémie (exemple : souris !)
- **Possibilités** : enrichissement grâce à des connaissances du domaine ou des mesures fondées sur les chaînes de caractères (exemple : chat/chaton)