# Traitement AUtomatique de la Langue

## Cours « Document et Web Sémantique »

Nicolas Delestre

INSA ROUEN NORMANDIE

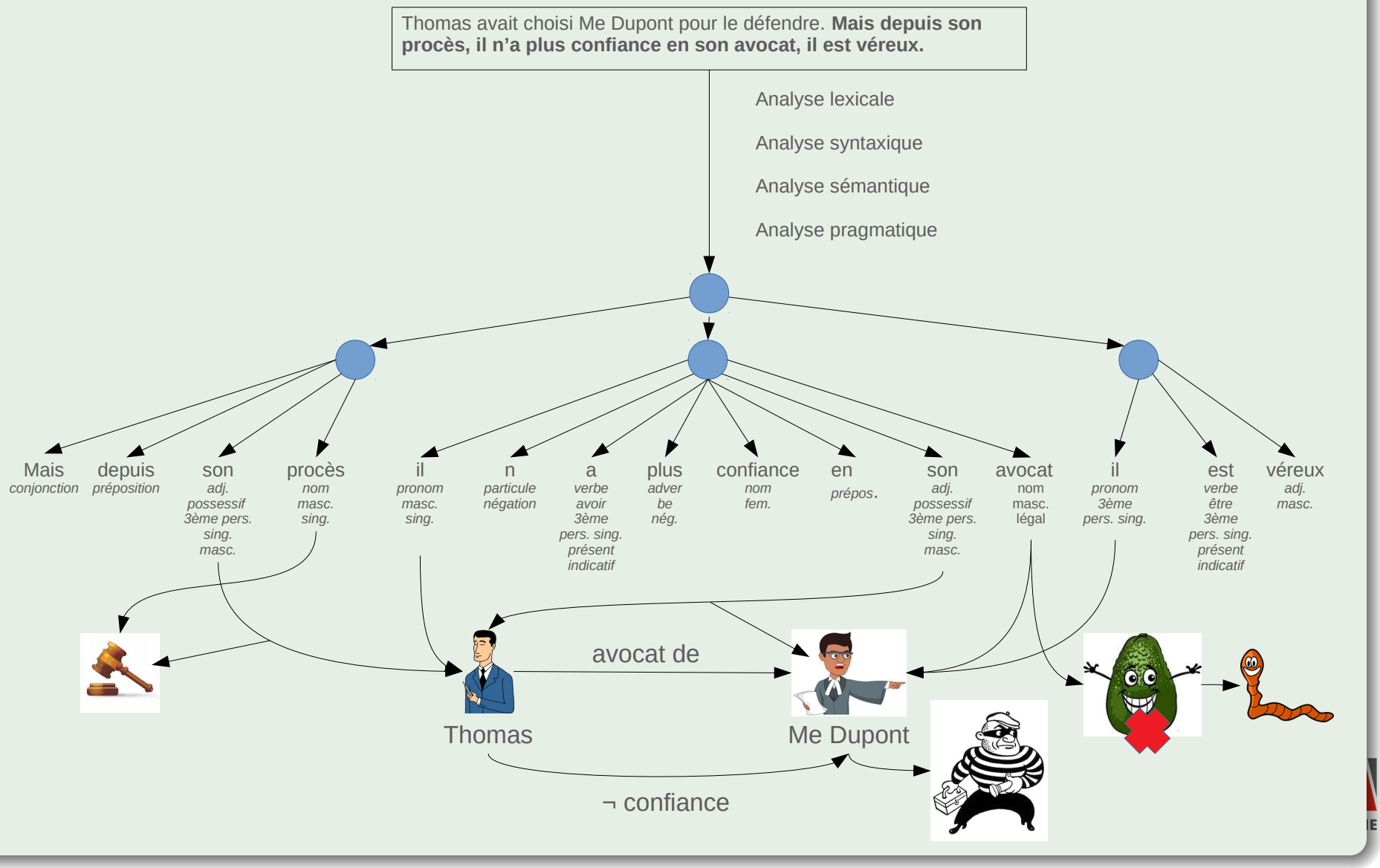
INSA ROUEN NORMANDIE

# Le TAL

- Le TAL a pour objectif d'interpréter des documents textuels, pour :
  - les classer
  - les traduire
  - les résumer
  - les corrigers
  - etc.

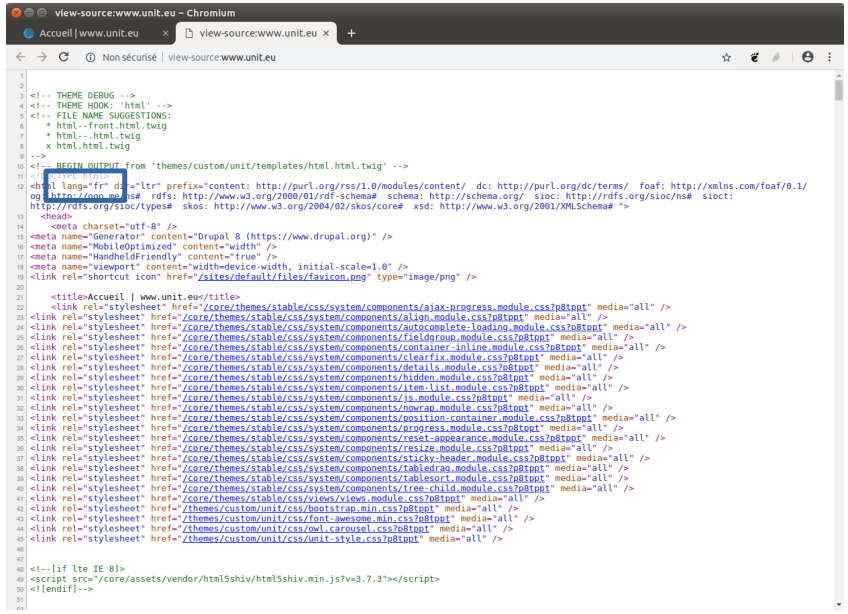
INSA ROUEN NORMANDIE

# Une chaîne de traitements



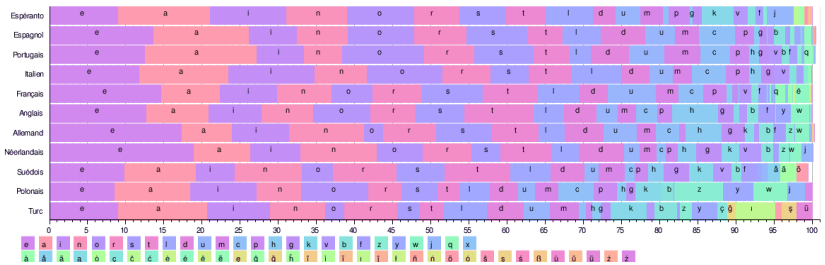
# Identification de la langue

- Information présente sous forme de métadonnées
- Inféré à partir de statistique
  - Taux compression
  - Calculer la fréquence d'apparition de n-gram (succession de n caractères) : méthode tolérante aux fautes d'orthographe
- Difficultés lorsque le document
  - est court
  - est multilingue
  - contient du texte qui n'est pas de la langue



# Identification de la langue

- Information présente sous forme de métadonnées
- Inféré à partir de statistique
  - Taux compression
  - Calculer la fréquence d'apparition de n-gram (succession de n caractères) : méthode tolérante aux fautes d'orthographe
- Difficultés lorsque le document
  - est court
  - est multilingue
  - contient du texte qui n'est pas de la langue




https://fr.wikipedia.org/wiki/Fréquence_d'apparition_des_lettres_en_français

# Identification de la langue

- Information présente sous forme de métadonnées
- Inféré à partir de statistique
  - Taux compression
  - Calculer la fréquence d'apparition de n-gram (succession de n caractères) : méthode tolérante aux fautes d'orthographe
- Difficultés lorsque le document
  - est court
  - est multilingue
  - contient du texte qui n'est pas de la langue

sticef .org
Sciences et Technologies de l'Information et de la Communication pour l'Éducation et la Formation

version pleine page
version à télécharger (pdf)
Volume 14, 2007
Article de recherche

Sommaire
Rechercher
auteur
année
titre
résumé
abstract
rubrique

Contact : infos@sticef.org

**Analyse et représentation en deux dimensions de traces pour le suivi de l'apprenant**

Nicolas DELESTRE, Nicolas MALANDAIN [LITIS, INSA de Rouen]

RÉSUMÉ : Le suivi d'apprenants lors de la résolution de problèmes est difficile, surtout lorsque le nombre d'apprenants est important ou lorsque la résolution de problèmes se fait à distance. Nous proposons ici une représentation graphique en deux dimensions des traces de ces apprenants qui pourrait être utilisée dans un logiciel de « monitoring ». Pour arriver à ce résultat nous avons adapté et combiné des algorithmes d'analyse numérique (principalement des algorithmes de réduction de dimensions : carte de Kohonen et SNE). Nous avons aussi abordé la problématique de distance entre ensembles en proposant une nouvelle mesure de similarité lorsque leurs éléments sont sémantiquement proches. Enfin nous avons validé et amélioré notre approche à l'aide tout d'abord de données simulées, puis de données réelles issues d'une expérimentation.

MOTS CLÉS : Visualisation de traces, projection 2D de données symboliques, distance/similarité entre ensembles, cartes conceptuelles, carte de Kohonen, algorithme du SNE.

ABSTRACT : The learner follow-up in problem solving is a hard issue. It is more difficult when there are a lot of learners or when those learners use distance learning. We propose in this paper a two-dimensional graphic representation of student's traces. To achieve this goal, we use and modify numerical analysis algorithms (automatic dimensionality reduction algorithms like Self Organizing Map and Stochastic Neighbour Embedding). We also propose a new distance between sets whose elements have semantic similarity. Finally, we validate and improve our algorithm with simulated data and experimental data.

KEYWORDS : Display of student traces, symbolic data 2D

INSA ROUEN NORMANDIE

# Identification de la langue

- Information présente sous forme de métadonnées
- Inféré à partir de statistique
  - Taux compression
  - Calculer la fréquence d'apparition de n-gram (succession de n caractères) : méthode tolérante aux fautes d'orthographe
- Difficultés lorsque le document
  - est court
  - est multilingue
  - contient du texte qui n'est pas de la langue

Dès lors, l'algorithme d'apprentissage d'une carte de Kohonen devient un algorithme d'initialisation (Cf. figure 12).

**Données en entrée** :

$X$ : les cartes conceptuelles d'apprentissage (de l'enseignant)

$N$ : les neurones

**Données en sortie** :

$W$ : les prototypes de la carte de Kohonen

**début**

Initialiser les $W_i$ avec $\emptyset$

$W_{N(X_i)} \leftarrow \{X_i\}, i \in [1..|X|]$

**Pour chaque** neurone $n \notin N$ **faire**

Calculer les cartes influentes $C_i \subset X$ avec $i > 0$

Calculer les attributs $att$ de l'ensemble des cartes $C_i$

Calculer le nombre d'attributs $nb_{att}$ des cartes pour $W_n$

$W_n \leftarrow$ l'ensemble des cartes générées à partir de $att$ et $nb_{att}$

**fin**

**Figure 12 • Phase d'initialisation d'une carte de Kohonen de cartes conceptuelles**

Précisons quelques points de cet algorithme :

- pour un neurone *n* de coordonnées *(i,j)*, les cartes d'influence sont les cartes se trouvant dans le cercle de centre *(i,j)* et de rayon *r*. Ce rayon est par défaut fixé au nombre maximal d'attributs que possèdent les cartes d'apprentissage,

INSA ROUEN NORMANDIE

# Segmentation

## Découper le texte en mot

- utiliser les séparateurs de mots
  - espaces, ponctuations, apostrophe
    - l'enfant
    - aujourd'hui
  - trait d'union
    - Mont-Saint-Michel
    - qu'en est-il ?
- Cela peut être difficile avec des langues acceptant des mots composés
  - ≪ Rindfleischetikettierungsüberwachungsaufgabenübertragungsgesetz ≫
  - ≪ loi sur le transfert des obligations de surveillance de l'étiquetage de la viande bovine ≫

Thomas avait choisi Me Dupont pour le défendre. Mais depuis son procès, il n'a plus confiance dans son avocat, il est véreux.

⇓

Thomas avait choisi Me Dupont pour le défendre Mais depuis son procès il n a plus confiance dans son avocat il est véreux

INSA ROUEN NORMANDIE

# Pré-étiquetage

## Hypothèse quant au rôle des mots

- mots connus (dictionnaire)
- mots inconnus :
  - les terminaisons : noms (-eur), adjectifs (-able), adverbes (-ment), verbes (-er, -ir, -aient), ...
  - genre/nombre fonction des terminaisons
  - statistiques
- problèmes avec les ≪ homographes ≫

Thomas avait choisi Me Dupont pour le défendre Mais depuis son procès il n a plus confiance dans son avocat il est véreux

⇓

- Thomas : *EN*
- avait : *avoir 3ème pers imparfait*
- choisi : *choisir p. passé* ou *adj.*
- Me Dupont : *EN*
- pour : *prep.*
- le : *art. def.*
- défendre : *défendre inf.*
- Mais : *conj.* ou *nom* ou *adv.*
- depuis : *prep.* ou *adv.*
- son : *adj. pos.* ou *nom*
- procès : *nom*
- il : *pron.*
- n : *particule*
- a : *pron. pers.* ou *avoir, 3ème pers. sing. présent*
- plus : adv. ou conj.
- confiance : *nom* ou *confiancer, 3ème 3ème pers. sing. présent indic.* ou *subj.* ou *imper.*
- dans : *prép.*
- son : *adj. pos.* ou *nom*
- avocat : nom
- il : *pron.*
- est : *être 3ème sing. présent* ou *nom*
- véreux : *adj.*

# Treetagger

Logiciel d'analyse lexicale gratuit (non opensource) multi-plateformes, multi-lingues

- Site Web : `http://www.cis.uni-muenchen.de/~schmid/tools/TreeTagger/`
- Tagset français : `http://www.cis.uni-muenchen.de/~schmid/tools/TreeTagger/data/french-tagset.html`

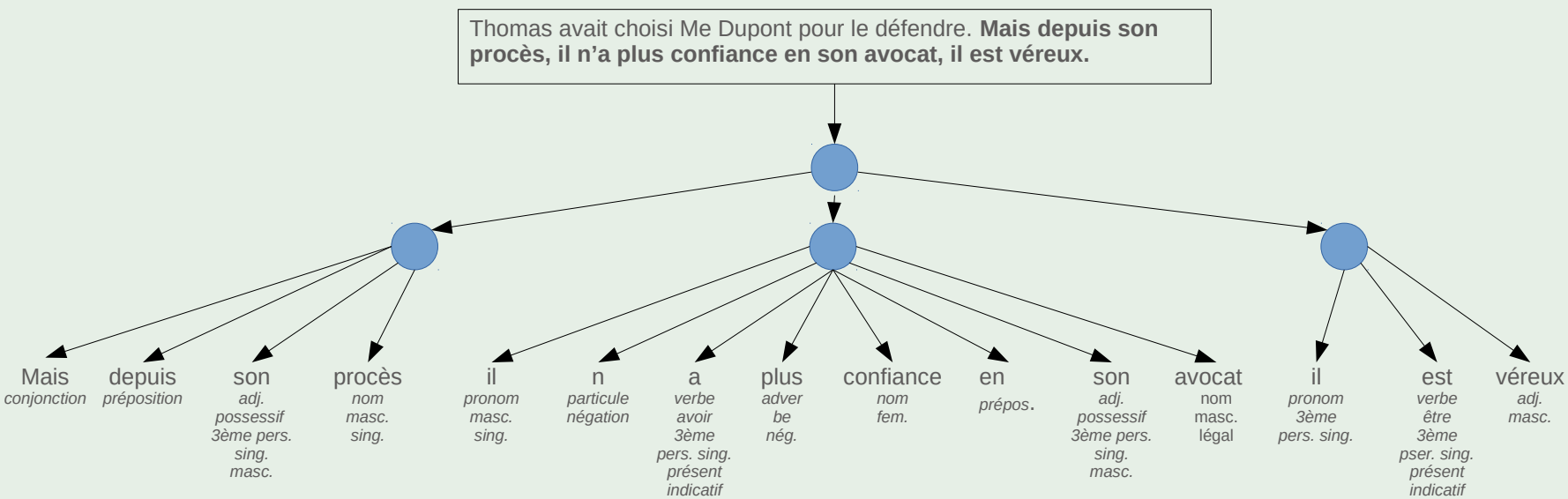
```
$echo "Thomas avait choisi Me Dupont pour le défendre Mais depuis son procès il n a plus confiance
dans son avocat il est véreux" | cmd/tree-tagger-french
 reading parameters ...
 tagging ...
Thomas NAM Thomas
avait VER:impf avoir
choisi VER:pper choisir
Me ABR Me
Dupont NAM Dupont
pour PRP pour
le PRO:PER le
défendre VER:infi défendre
```

Treetagger peut proposer plusieurs étiquetages (ex : « Je **suis** allé au cinéma »)

# Analyse syntaxique

## Objectifs

- Créer un arbre syntaxique
- Préciser l'étiquetage des mots
  - Trois méthodes : automatique à partir d'un corpus étiqueté et avec un algorithme d'étiquetage, à partir d'une grammaire formelle, méthode mixte

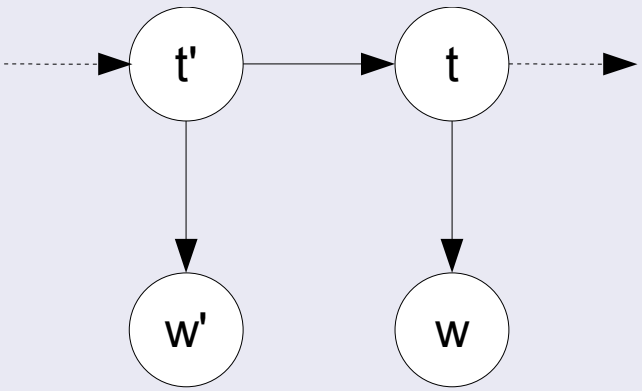


# Exemple d'étiquetage automatique 1 / 3

## Hidden Markov Model / Chaînes de Markov Cachées

On cherche l'étiquette $t$ du mot $w$ :

**Apprentissage :**




$$p(w|t) = \frac{|(w,t)|}{\sum_x (x,t)} = \frac{|(w,t)|}{|t|}$$

$(w,t)$ signifie que $w$ est étiqueté par le tag $t$

$$p(t|t') = \frac{|t't|}{\sum_y |t'y|} = \frac{|t't|}{|t'|}$$

$t't$ représente la séquence des deux tags

**Décision :** Choix de la séquence de tags de plus forte probabilité pour une séquence $w_1^n$ de $n$ mots :

$$\hat{t}_1^n = \underset{t_1^n}{argmax} \prod_{i=1}^{n} \underbrace{p(w_i|t_i)}_{\text{émission}} \times \underbrace{p(t_i|t_{i-1})}_{\text{transition}} \qquad \textit{(taggeur bigramme)}$$

# Exemple d'étiquetage automatique 2 / 3

## Tagger bigramme pour « les beaux avions »

- « les » : article ou pronom ?
- « beaux » : nom, adjectif ou un adverbe (« il a beau partir tôt, il arrivera en retard ») ?
- « avions » : verbe ou nom ?

## « les »

- $P(les|article) \times P(article|debutDePhrase)$
- $P(les|pronom) \times P(pronom|debutDePhrase)$

$\Rightarrow$ *les* est un *article*

## « beaux »

- $P(beaux|nom) \times P(nom|article)$
- $P(beaux|adverbe) \times P(adverbe|article)$
- $P(beaux|adjectif) \times P(adjectif|article)$

$\Rightarrow$ *beaux* est un *adjectif*

# Exemple d'étiquetage automatique 3 / 3

## ≪ avions ≫

- $P(avions|nom) \times P(nom|adjectif)$
- $P(avions|verbe) \times P(verbe|adjectif)$

$\Rightarrow$ *avions* est un *nom*

## Attention

- Certains cas peuvent nécessiter des taggeur trigramme (voire plus)

≪ il la fatigue ≫ , ≪ de la fatigue ≫ [Éric Laporte]

# Analyse sémantique

## Objectif : ≪ extraire ≫ le sens d'un texte

Analyse du sens des mots → de la phrase → du texte

## Réalité d'une machine

Analyse ≪ automatique ≫ limitée à un domaine

- création d'un modèle du domaine
  - logique
  - réseaux sémantiques
  - graphes conceptuels (représentation des connaissances basée sur la logique)
- filtrage du ≪ sens ≫ des mots liés au domaine
  (réduit les problèmes comme celui de la polysémie)

INSA ROUEN NORMANDIE

# Analyse pragmatique

## Objectif

Re-situer l'analyse sémantique dans le contexte d'énonciation
« interprétation »

- de l'implicite (ex : « Je vais à la poste », laquelle ? certainement la plus proche)
- des présupposés (culture commune, contexte commun, etc.)
- des actes de langage (la façon de dire des choses, d'insister sur certains, figure de style)

Par exemple savoir que le procès de Thomas se déroule en France, indique que la justice « est inquisitoire. C'est l'enquête qui est au centre de la procédure, et non l'accusation. » (cf. `http://www.politique.net/2011060602-differences-justice-francaise-justice-americaine.htm`), alors qu'aux états unis elle est accusatoire.

# Représentation vectorielle des documents

## Pourquoi ?

- Calculer des similarités (par ex. cosinus), des distances (par ex. euclidienne) entre documents
- Calculer des nouveaux vecteurs à l'aide d'opérations vectorielles donnant du sens
- Utiliser les outils du *machine learning*

INSA ROUEN NORMANDIE

# Représentation simple

## Définition

- $n$ documents $d$ et $m$ termes $t$
- $v_{i,j}$ = nombre d'occurences du terme $t_j$ dans un document $d_i$

## Inconvénients

- matrice sparce
- non prise en compte du nombre d'occurrences d'un terme au sein du corpus
- non prise en compte des synonymes, des homonymes
- ajout d'un nouveau document

| | $d_1$ | $d_2$ | ... | $d_i$ | ... | $d_n$ |
|---|---|---|---|---|---|---|
| $t_1$ | 2 | 0 | ... | | ... | 5 |
| $t_2$ | 0 | 4 | ... | ... | ... | 1 |
| ⋮ | ⋮ | ⋮ | | | | ⋮ |
| $t_j$ | | | | $v_{i,j}$ | | |
| ⋮ | ⋮ | ⋮ | | | | ⋮ |
| $t_m$ | 0 | 0 | ... | | ... | 1 |

# Représentation *tf.idf* (années 70)

## Définition

- $tf_{i,j}$ : nombre d'occurrences du terme $t_j$ dans dans le document $d_i$
- $idf_j$ : $log(\frac{|D|}{\{d_i : t_j \in d_i\}})$
- $v_{i,j} = tf_{i,j} \times idf_j$

## Inconvénients

- matrice sparce
- non prise en compte des synonymes, des homonymes
- ajout d'un nouveau document

$$
\begin{array}{c} \\ t_1 \\ t_2 \\ \vdots \\ t_j \\ \vdots \\ t_m \end{array}
\begin{array}{c}
\begin{array}{cccccc} d_1 & d_2 & \dots & d_i & \dots & d_n \end{array} \\
\left[\begin{array}{cccccc}
0.3 & 0 & \dots & & \dots & 0.3 \\
0 & 0.9 & \dots & \dots & \dots & 0.1 \\
\vdots & \vdots & & & & \vdots \\
 & & & v_{i,j} & & \\
\vdots & \vdots & & & & \vdots \\
0 & 0 & \dots & & \dots & 0.8
\end{array}\right]
\end{array}
$$

# Représentation LSA (1990)

## Définition

- *Latent Semantic Analysis* : réunir les termes qui sont corrélés (concept $c_j$)
- Reprendre la matrice $tf.idf$ $V$, calculer les valeurs et vecteurs singuliers : $V = U\Sigma W^t$
- Retenir uniquement les vecteurs ayant les valeurs singulières supérieures $\alpha$ : $V = U_k \Sigma_k W_k^t$ avec $k << m$

$$
\begin{array}{c}
 \\
c_1 \\
c_2 \\
\vdots \\
c_j \\
\vdots \\
c_k
\end{array}
\begin{array}{c}
\begin{array}{ccccccc} d_1 & d_2 & \dots & d_i & \dots & d_n & \end{array} \\
\begin{bmatrix}
0.3 & 0.3 & \dots & & \dots & 0.3 & \\
0.5 & 0.6 & \dots & \dots & \dots & 0.1 & \\
\vdots & \vdots & & & & \vdots & \\
 & & & w_{i,j} & & & \\
\vdots & \vdots & & & & \vdots & \\
0.1 & & 0.7 & \dots & & \dots & 0.8
\end{bmatrix}
\end{array}
$$

## Inconvénients

- non prise en compte des homnymes
- ajout d'un nouveau document

# Vers une représentation des mots

- Pour une meilleur compréhension des documents, ne plus représenter les documents par un ≪ ensemble ≫ de termes : prendre en compte l'ordre
- Donner ≪ sens ≫ aux mots, aux phrases, aux documents
  - Qu'est ce que donner du sens ?
    - Relier les mots, phrases et documents entre eux par des liens sémantiques
    - **Permettre des traitements sur ces mots, phrases, ou documents**
  - Word2vec
    - Apprentissage automatique à partir d'un corpus de documents
    - Chaque terme est représenté par un vecteur, tels que : deux vecteurs proches sont sémantiquement proches des opérations sont possibles sur ces vecteurs, per ex. $roi - homme + femme \approx reine$

INSA ROUEN NORMANDIE

# word2vec (2013) et doc2vec (2014) 1 / 2

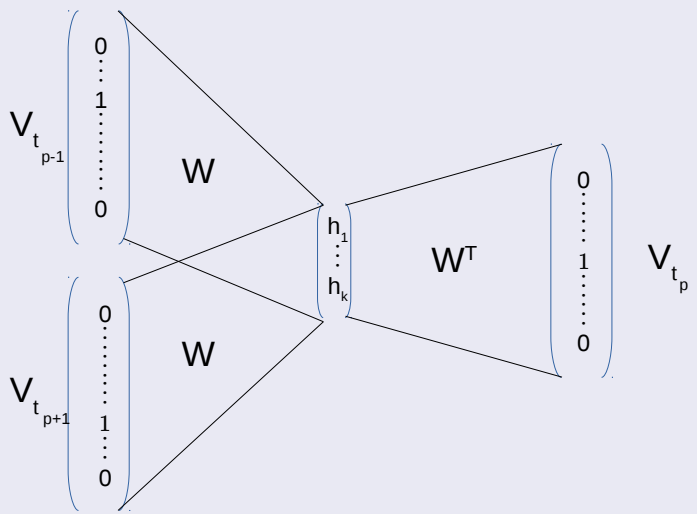
## word2vec : principe

cf. https://www.youtube.com/watch?v=_YYQNpjvvLE

- Construction d'une matrice carrée symétrique $M$ (de taille $t$) du nombre de co occurrences de termes $t_i$ du corpus
- Calcul d'une matrice $W$ (de taille $t, k$) tel que $W.W^t \simeq M$, $W$ est la représentation des termes $t_i$

## word2vec dans la pratique : utilisation d'un réseau de neurones

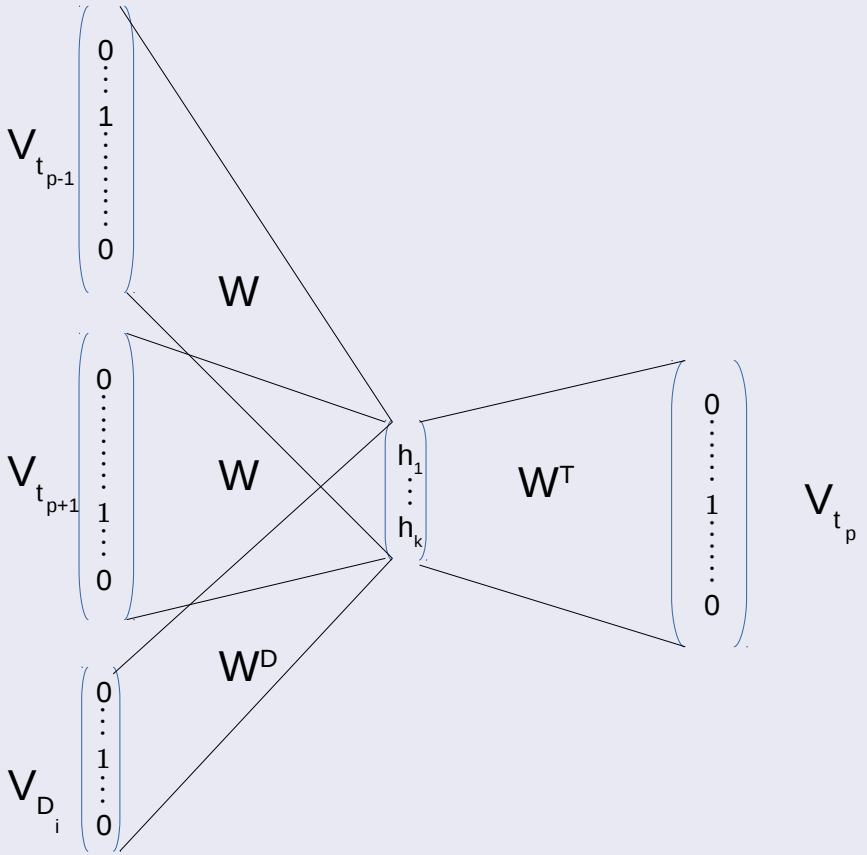
- Chaque terme $t_i$ est représenté par un vecteur $v_{t_i}$ de taille $t$ (un seul 1, le reste à 0)
- Pour chaque document, on présente au réseau les représentations vectorielles de termes entourant le terme $t_p$ à reconnaître (ici fenêtre de 3)
- La représentation finale du terme $t_p$ est $W^T.V_{t_p}$



# word2vec (2013) et doc2vec (2014) 2 / 2

## doc2vec : une extension de word2vec

- Lorsque l'on apprend le réseau, on ajoute en entrée le document pour lequel on apprend la représentation du terme $t_p$
- La représentation du document $D_i$ est $W^{D^T}.V_{D_i}$
- Plus adapté à la thématisation des documents plutôt qu'à la recherche d'information
-



# BERT (2018)

## *Bidirectional Encoder Representations from Transformers*

- BERT repose sur :
  - la tokenisation des phrases (30K tokens). Token = suite de caractères statistiquement représentatif, avec en plus `CLS` (début de texte) et `SEP` (séparateur de phrase)

    `https://gptforwork.com/tools/tokenizer`
  - l'architecture des **Transformers** (2017) qui utilise un mécanisme d'attention (attribution de poids plus importants aux tokens les plus pertinents) pour traiter une séquence (512 tokens dans le cas de BERT)
- L'apprentissage du réseau porte sur deux tâches principales :
  - **Masked Language Model (MLM)** : prédire un mot masqué dans une phrase (apprentissage de l'attention)
  - **Next Sentence Prediction (NSP)** Prédire si une phrase suit une autre (meilleur compréhension des relations entre phrases)
- Il produit des représentations contextuelles des mots selon leur contexte gauche et droit, la représentation des mots/tokens n'est plus statique, elle est dynamique ⇒ gestion des synonymes et homonymes

# S-BERT (2019)

## *Sentence BERT*

- **S-BERT** est une version de BERT adaptée pour la similarité de phrases :
  - Son architecture repose sur deux encoders BERT (un par phrase) et il utilise la représentation du token CLS ou la moyenne des embeddings de tous les tokens (méthode *pooling*) pour représenter les phrases
  - Il est entraîné avec des paires de phrases et des valeurs indiquant leur similarité

## CamenBERT, FlauBERT, S-CamenBERT, etc.

- Il existe des variantes nationales de BERT et S-BERT

INSA ROUEN NORMANDIE

# Conclusion

- Évolution des méthodes : méthodes *top-down* $\Rightarrow$ *bottom-up*
- Application : **Recherche d'information**, systèmes de recommandations, traduction automatique, résumés, chatbots, etc.
- Défis : consommation énergétique

INSA ROUEN NORMANDIE