INF8007 – Languages de scripts
Similarité de textes

Michel Desmarais

Hiver 2018

Définitions

Quelques définitions préalables auquelles nous référerons en RI :

t : nombre de termes distincts dans la collection.

tf_{ij} : *fréquence du terme (term frequency)* qui correspond au nombre d'occurences du terme t_j dans le document D_i.

df_j : *fréquence des documents (document frequency)* qui correspond au nombre de documents qui contiennent le terme t_j.

$idf_j = log(\frac{d}{df_j})$: *fréquence inverse des documents (inverse document frequency)* qui représente une mesure de la spécifité du terme j. d est le nombre total de documents.

$tfidf = tf \times idf$ valeur pondérée de la fréquence de termes

Modèle de l'espace vectoriel

- On a n termes différents pour un ensemble de documents et une requête
- On a d documents
- Les termes de la requête est représenté dans un vecteur $< q_0, q_1, q_2, \ldots, q_n >$
- Les documents sont représentés dans une matrice $n \times d$

Exemple d'un modèle de l'espace vectoriel

- Deux termes, 'A' et 'B'
- Trois documents, D1, D2 et D3

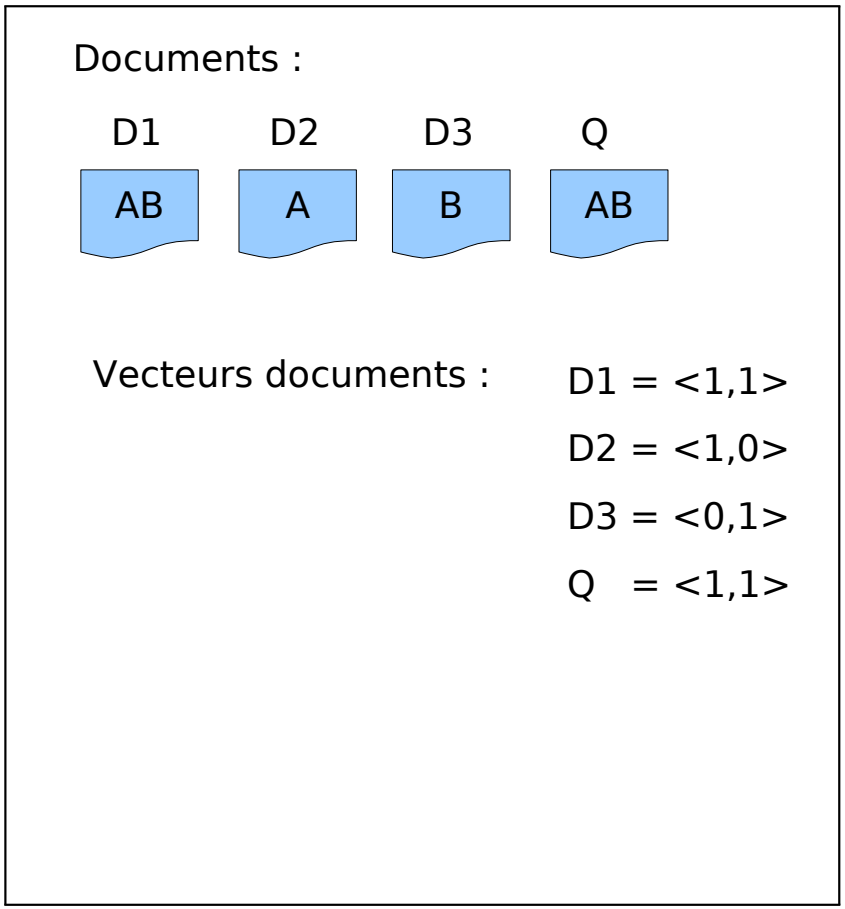
$$D_i(tf_{i1}, tf_{i2}, \ldots, tf_{it})$$

- Une requête Q

$$Q_q(tf_{q1}, tf_{q2}, \ldots, tf_{qt})$$

Matrice termes-documents+requête

	D_1	D_2	D_3	Q
A	1	1	0	1
B	1	0	1	1



Exemple d'un modèle de l'espace vectoriel

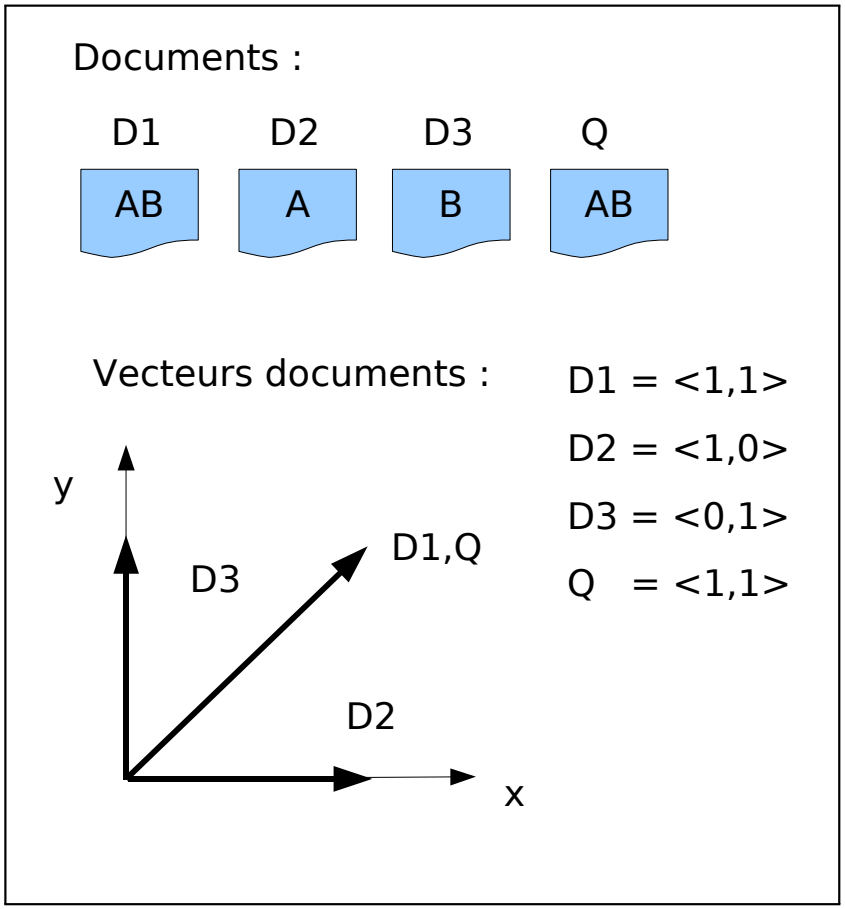
Calcul de similitude avec le produit scalaire :

$$CS(Q, D_i) = \sum_{j=1}^{t} tf_{qj} \times tf_{ij}$$

en utilisant une transformation des fréquences brutes :

$$d_{ij} = tf_{ij} \times idfj$$

(voir Définitions)



Cosinus comme mesure de similitude

Le produit scalaire n'est toutefois pas l'idéal. Le coefficient de similitude, ou CS, le plus commun est le cosinus de l'angle entre la requête et le document :

$$CS(Q, D_i) = \frac{\sum_{j=1}^{t} tf_{qj} tf_{ij}}{\sqrt{\sum_{j=1}^{t} (tf_{ij})^2} \sqrt{\sum_{j=1}^{t} (tf_{qj})^2}}$$

Toutefois, comme le terme $\sum_{j=1}^{t} (w_{qj})^2$ est identique pour tous les documents, on obtient le même résultat en divisant simplement le produit scalaire par la taille du document.
Il existe aussi d'autres variantes qui utilisent différents poids tout en normalisant selon la taille du document et autres facteurs.

Exemple de Grossman et Frieder

Voir l'exemple 2.1.1, p.15, dans Grossman et Frieder (2004)

Q : "gold silver truck"
D_1 : "Shipment of gold damaged in a fire"
D_2 : "Delivery of silver arrived in a silver truck"
D_3 : "Shipment of gold arrived in a truck"

Dans cet exemple et en utilisant le produit scalaire comme coefficient de similitude, l'ordre de pertinence des documents est :
D_2, D_3 et D_1

Exemple Grossman et Frieder

Voir l'exemple, p. 15 de Grossman et Frieder (2004).

	D_1	D_2	D_3	Q
a	1	1	1	0
arrived	0	1	1	0
damaged	1	0	0	0
delivery	0	1	0	0
fire	1	0	0	0
gold	1	0	1	1
in	1	1	1	0
of	1	1	1	0
shipment	1	0	1	0
silver	0	2	0	1
truck	0	1	1	1

Q : "gold silver truck"

D_1 : "Shipment of gold damaged in a fire"

D_2 : "Delivery of silver arrived in a silver truck"

D_3 : "Shipment of gold arrived in a truck"

Exemple, matrice des coefficients

Matrice du `tfidf` : $w_{ij} = tf_{ij} \times idf_j$ (voir Définitions)

Terme	**idf**	$\mathbf{D_1}$	$\mathbf{D_2}$	$\mathbf{D_3}$	**Q**
a	0	0	0	0	0
arrived	0,176	0	0,176	0,176	0
damaged	0,477	0,477	0	0	0
delivery	0,477	0	0,477	0	0
fire	0,477	0,477	0	0	0
gold	0,176	0,176	0	0,176	0,176
in	0	0	0	0	0
of	0	0	0	0	0
shipment	0,176	0,176	0	0,176	0
silver	0,477	0	**0,954**	0	0,477
truck	0,176	0	0,176	0,176	0,176
$Q \cdot D_i$		*0,031*	*0,486*	*0,062*	
Cosinus		*0,080*	*0,825*	*0,327*	

Quelques variations autour du poids w_{ij}

- Un seul terme comportant une grande fréquence dans le document (tf) peut influencer le poids w_{ij} de façon trop importante. On utilise donc la transformation

$$\log(tf) + 1$$

- On peut aussi appliquer des variations du calcul pour la requête

Variations autour du calcul du coefficient de similitude

- Tout comme pour le calcul du poids des termes, il existe plusieurs variations autour du calcul du coefficient. Entre autre, la longueur du document est un facteur qui est normalisé dans le calcul du cosinus (peu importe la longueur des vecteurs, leur angle est toujours le même).
- Par exemple, la formule suivante fournit de meilleurs résultats selon les tests faits pour TREC en normalisant pour tenir compte du fait que les documents plus long sont souvent plus pertinents car ils contiennent plus d'information :

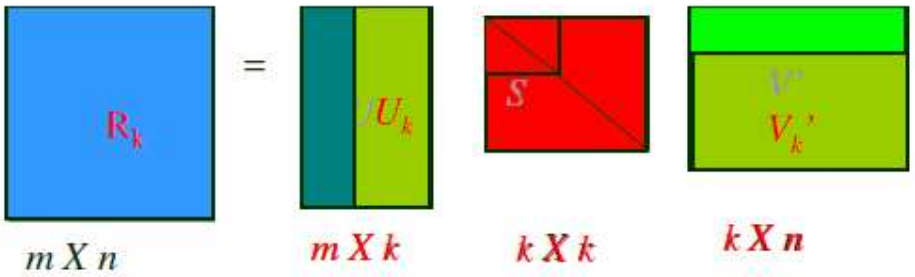
$$CS(Q, D_i) = \frac{\sum_j^t w_{qj} d_{ij}}{((1-s)p + (s)(|d_i|))}$$

où p, le *pivot*, et s, la *pente*, représentent des facteurs de correction qui diminue le poids des petits documents et augmente celui des longs

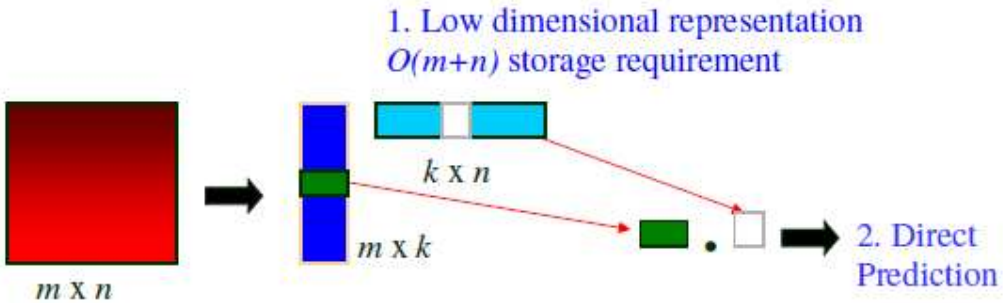
Révision

- Plus un terme est commun, plus son poids, $w_{\bullet j}$, sera petit, vrai ou faux ?
- Donnez une définition mathématique de “terme commun”.
- Euclide se dit que deux documents sont similaires s’ils sont près l’un de l’autre dans l’espace vectoriel défini par la matrice termes-documents, il conclut donc qu’une bonne mesure de similarité serait la distance euclédienne plutôt que le cosinus. A-t-il raison ? Expliquez votre réponse par un schéma.
- Tim tente d’améliorer la recherche de pages web et il se dit qu’il pourrait utiliser la matrice d’adjacence, qui indique quelle page réfère à quelle autre, pour trouver les pages similaires et appliquer le cosinus comme coefficient de similitude. Discutez de son idée.

Principes de l'indexation sémantique



The reconstructed matrix $R_k = U_k.S_k.V_k'$ is the closest *rank-k* matrix to the original matrix *R*.



Principes de l'indexation sémantique (suite)

- Permet d'éliminer les dimensions les moins pertinentes et d'effectuer une projection autour des dimensions restantes par la méthode de décomposition des valeurs singulières.
- Supposons une matrice termes-documents, A. Cette matrice se décompose un produit de trois matrices :

$$A = U \Sigma V^T$$

- La matrice diagonale Σ représente les valeurs singulières et sont triées par leur magnitude. Puis, on ne conserve que les premières valeurs k de cette matrice, les premières k colonnes de U et les premières k lignes de la matrice V^T. On obtient une nouvelle matrice, A' dont on se sert pour transformer les vecteurs originaux.

Exemple

Voir l'exemple Grossman et Frieder, p. 71.

	D_1	D_2	D_3
a	1	1	1
arrived	0	1	1
damaged	1	0	0
fire	1	0	0
gold	1	0	1
in	1	1	1
of	1	1	1
shipment	1	0	1
silver	0	2	0
truck	0	1	1

Q : "gold silver truck"

D_1 : "Shipment of gold damaged in a fire"

D_2 : "Delivery of silver arrived in a silver truck"

D_3 : "Shipment of gold arrived in a truck"

Décomposition SVD : $U\,\Sigma\,V_T$

U		
-0.4201	**-0.0747**	-0.0459
-0.2994	**0.2000**	0.4078
-0.1206	**-0.2748**	-0.4538
-0.1575	**0.3046**	-0.2006
-0.1206	**-0.2748**	-0.4538
-0.2625	**-0.3794**	0.1546
-0.4201	**-0.0747**	-0.0459
-0.4201	**-0.0747**	-0.0459
-0.2625	**-0.3794**	0.1546
-0.3151	**0.6092**	-0.4012
-0.2994	**0.2000**	0.4078

$\times$

Σ		
4.098	0.000	0.000
0.000	**2.361**	0.000
0.000	0.000	1.273

$\times$

V_T		
-0.4944	**-0.6491**	**-0.5779**
-0.6458	**0.7194**	**-0.2555**
-0.5817	-0.2469	0.7749

Projections dans l'espace réduit

Les matrices réduites, U_2 et Σ_2, on transforme la requête et tous les documents dans l'espace transformé. Par exemple, la requête correspond à $q^T\ U_2\ \Sigma_2^{-1}$:

$$\begin{bmatrix} 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 1 \\ 0 \\ 0 \\ 0 \\ 1 \\ 1 \end{bmatrix}^T \begin{bmatrix} -0.42 & -0.07 \\ -0.29 & 0.20 \\ -0.12 & -0.27 \\ -0.15 & 0.30 \\ -0.12 & -0.27 \\ -0.26 & -0.37 \\ -0.42 & -0.07 \\ -0.42 & -0.07 \\ -0.26 & -0.37 \\ -0.31 & 0.60 \\ -0.29 & 0.20 \end{bmatrix} \begin{bmatrix} 0.24 & 0 \\ 0 & 0.42 \end{bmatrix} = \begin{bmatrix} -0.21 & -0.18 \end{bmatrix}$$

Projections dans l'espace réduit

En ne conservant que deux des trois dimensions, la requête et les documents prennent les valeurs suivantes :

$$\begin{aligned} Q &= [-0.2140 \quad -0.1821] \\ D_1 &= [-0.4945 \quad 0.6492] \\ D_2 &= [-0.6458 \quad -0.7194] \\ D_3 &= [-0.5817 \quad 0.2469] \end{aligned}$$

et les coefficient de similitude avec la requête basé sur le cosinus :

$$\begin{aligned} D_1 &= -0.0541 \\ D_2 &= 0.9910 \\ D_3 &= 0.4478 \end{aligned}$$

Voir *http://www.miislita.com/information-retrieval-tutorial/svd-lsi-tutorial-4-lsi-how-to-calculations.html*