TD – Recherche d'information – Recherche et évaluation - CORRECTION

Exercice 1 – Recherche dans le modèle vectoriel

Fournir les résultats des requêtes suivantes pour le corpus de l'exercice 3 des TDs précédents :
Q1 : pomme de terre
Q2 : recherche d'information textuelle
Q3 : domaine du modèle vectoriel

Question 1

Construitre les vecteurs des requêtes. Commencer par analyser les requêtes comme les documents (anti-dictionnaire), et utiliser une pondération des requêtes par le tf.idf .
Calculer les normes.

** Tous nos vecteurs sont 11D, cf la partie sur l'indexation. **

Rappel : terme (idf)
t1 = comprendr (1.10), t2 = domain (1.10), t3 = intér (1.10), t4 = modèl (1.10), t5 = nombreux (1.10), t6 = parl (1.10), t7 = problèm (1.10), t8 = professeur (1.10), t9 = simpl (1.10), t10 = textuel (1.10), t11 = vectoriel (1.10)

Pour Q1 :
Passage en minuscule, extraction des tokens, antidictionnaire (le même que les TD précédent car on utilise le même pour indexer et rechercher):

pomme ~~de~~ terre

Racinisation (troncature) :

pomme – 2 -> pomm /
terre – 2 -> terr /

termes d'indexation ?
pomm => non
terr => non

pas de terme du vocabulaire donc requête vide :
Q1 = (0 0 0 0 0 0 0 0 0 0 0)
donc on n'exécute même pas la requête : pas de document qui répond !

Pour Q2 :
Même processus que Q1

recherche ~~d'~~ information textuel~~le~~

racinisation :

recherch
information
textuel

termes d'indexation ?
recherch => non
information => non
textuel => t10

(t10) tf=1
idf = 1.10 <= celui qui vient de la collection de documents

donc le vecteur requête est Q2 = (0 0 0 0 0 0 0 0 0 1.10 0)

Norme du vercteur requête :
||Q2|| = rac(1.10^2) = 1.10

Pour Q3 :
Traitement : minuscule => tokenisation => antidictionnaire => troncature

domaine ~~du~~ modèle vectoriel

termes d'indexation ?
domain => t2
modèl => t4
vectoriel => t11

Les poids pour les termes :
(t2) tf=1 idf=1.10 (t4) tf=1 idf=1.10 (t11) tf=1 idf = 1.10

donc le vecteur requête est Q3 = (0 1.10 0 1.10 0 0 0 0 0 0 1.10)
||Q3|| = rac(3.63) = 1.91

Question 2
Calculer les similarités RSV(d,q) en reprenant la formule du cours (transparent 10) sans utiliser l'index inversé, avec chacun des documents.
Si on utilise pas l'index inversé, on fait le calcul de cosinus entre les vecteurs requêtes et documents.

***

Q1 : requête vide : c'est fini. On sait qu'il n'y pas de réponse.

***

Q2 = (0 0 0 0 0 0 0 0 0 1.10 0)
RSV(d1,Q2) = RSV((0 0 0 0 0 1.10 0 1.10 0 1.10 0), (0 0 0 0 0 0 0 0 0 1.10 0))

On connait ||D1|| et ||Q2||

RSV(d1,Q2) = (0*0 + 0*0 + …+ 1.10*1.10 + 0*0) / (1.90 * 1.10) = 0.58

d1 répond à Q2 avec un score de 0.58

RSV(d2,Q2) = RSV((0 1.10 1.10 0 1.10 0 1.10 0 0 0 0), (0 0 0 0 0 0 0 0 0 1.10 0))
= 0 / ||d2||*||Q2|| = 0
➔ d2 ne répond pas à Q2

RSV (d3, Q2) = 0
➔ d3 ne répond pas à Q2

Réponse du système à Q2 ?
On sait que d2 et d3 ne répondent pas ➔ pas dans la réponse.
Il ne reste que d1 (0,58), donc la réponse pour Q2 est : d1.

*****

Q3 = (0 1.10 0 1.10 0 0 0 0 0 0 1.10)

RSV(d1,Q3) = RSV((0 0 0 0 0 1.10 0 1.10 0 1.10 0), (0 1.10 0 1.10 0 0 0 0 0 0 1.10))
RSV(d1,Q3) = **0**

RSV(d2,Q3) = RSV((0 1.10 1.10 0 1.10 0 1.10 0 0 0 0), (0 1.10 0 1.10 0 0 0 0 0 0 1.10))
RSV(d2,Q3) = (1.10*1.10) / (2.20 * 1.91) = **0.29**

RSV(d3,Q3) = RSV((1.10 0 0 2.20 0 0 0 0 1.10 0 1.10), (0 1.10 0 1.10 0 0 0 0 0 0 1.10))
RSV(d3,Q3) = (2.20*1.10 + 1.10*1.10) / (2.91 * 1.91) = **0.66**

La réponse pour Q3 est : d3, d2. d3 est devant d2 car son RSV est plus grand.
** d1 NE REPOND PAS A Q3, donc pas dans la réponse.

Question 3
Refaire les calculs avec l'utilisation de l'index inversé.

Pour la **question 3** avec l'index inversé: on doit obtenir exactement les mêmes résultats que pour la question 2.
Pour Q1 (requête vide) rien à faire.
Q2 est facile.

Je corrige pour **Q3** (cf. transparent 12):

Q3 = (0 1.10 0 1.10 0 0 0 0 0 0 1.10)

Rappel, index inversé :

	d1	d2	d3
t1	0	0	1.10
t2	0	1.10	0
t3	0	1.10	0
t4	0	0	2.20
t5	0	1.10	0
t6	1.10	0	0
t7	0	1.10	0
t8	1.10	0	0
t9	0	0	1.10
t10	1.10	0	0
t11	0	0	1.10

1. On garde les t_i <>0 => t2, t4, t11
(Note : ligne_res est un vecteur avec comme indice d1, d2, d3, initialisé à [0, 0, 0], on va modifier ses valeurs au fur et à mesure du traitement des termes de la requête)
2. Boucle sur t2, t4, t11
- pour t2 : (note : la ligne t2 de l'index inversé est [0, 1.10, 0])
  ligne_res [d1] += 0*1.10 ➔ = 0
  ligne_res[d2] += 1.10*1.10 ➔ =1.21
  ligne_res[d3] += 0*1.10 ➔ = 0
  (note : pour le moment, calcul partiel, uniquement en comptant t2, avec tous les docs => [0, 1.21, 0])
- pour t4 : (note, la ligne t4 de l'index inversé est [0, 0, 2.20])
  ligne_res [d1] += 0 *1.10 ➔ = 0
  ligne_res[d2] += 0 *1.10 ➔ = 1.21
  ligne_res[d3] += 2.20 * 1.10 ➔ 2.42
  (note : pour le moment, calcul partiel, en comptant maintenant t2 et t4, avec tous les docs => [0, 1.21, 2.42])
- p- our t11 : (note, la ligne t11 de l'index inversé est [0, 0, 1.10])
  ligne_res [d1] += 0 * 1.10 ➔ = 0
  ligne_res[d2] += 0 * 1.10 ➔ = 1.21
  ligne_res[d3] += 1.10 * 1.10 ➔ 3.63

(note : maintenant, ligne_res ([0, 1.21, 3.63]) contient le *produit scalaire* des docs et de tout Q3 (car on a passé les 3 termes non-nuls de la requête) : le RSV final doit diviser le produit scalaire par le produit des normes, comme vu en cours)

3. Calcul final (obtention du cosinus pour tous les docs): (rappel: ||d1|| = 1.90, ||d2|| = 2.20; ||d3|| = 2,91; ||Q3|| = 1.91)

ligne_res [d1] = ligne_res [d1] / (||d1||*||Q3||) ➔ = **0**
ligne_res[d2] = ligne_res [d2] / (||d2||*||Q3||) ➔ = **0.29**
ligne_res[d3] = ligne_res [d3] / (||d3||*||Q3||) ➔ = **0.66**

(Note : on a maintenant les cosinus pour les 3 docs, en ne regardant QUE les éléments intéressants pour les calculs, c'est beaucoup plus rapide !)
La suite est la même que pour la question 2 :
La réponse pour Q3 est : d3, d2.
** d1 NE REPOND PAS A Q3, donc pas dans la réponse.

Exercice 2 – Evaluation de SRI
Nous réalisons ici une évaluation d'un système de recherche d'information.

Question 1
Supposons que pour une requête Q1 le système de recherche d'information testé renvoie les réponses suivantes:

rang	n° doc	pertinent	rappel	précision
1	588	1	1 / 5 = 0.2	1 / 1 = 1
2	589	1	2 / 5 = 0.4	2 / 2 = 1
3	576	0	2 / 5 = 0.4	2 / 3 = 0.67
4	590	1	3 / 5 = 0.6	3 / 4 = 0.75
5	986	0	3 / 5 = 0.6	3 / 5 = 0.6
6	592	1	4 / 5 = 0.8	4 / 6 = 0.67
7	884	0	4 / 5 = 0.8	4 / 7 = 0.57
8	988	0	4 / 5 = 0.8	4 / 8 = 0.50
9	578	0	4 / 5 = 0.8	4 / 9 = 0.44
10	985	0	4 / 5 = 0.8	4 / 10 = 0.4
11	103	0	4 / 5 = 0.8	4 / 11 = 0.36
12	591	0	4 / 5 = 0.8	4 / 12 = 0.33
13	572	1	5 / 5 = 1	5 / 13 = 0.38
14	990	0	5 / 5 = 1	5 / 14 = 0.36

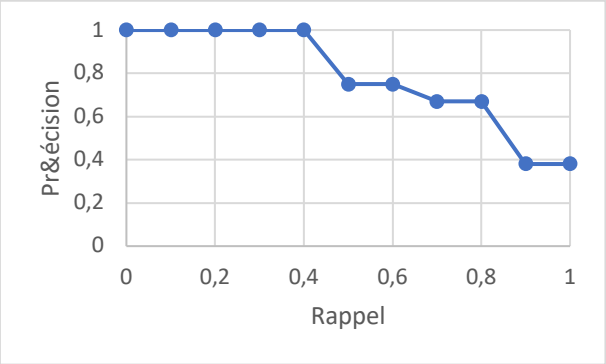
Les documents pertinents pour Q1 sont : 572, 588, 589, 590, 592.
Calculer les taux de précision et de rappel du système à chaque réponse et remplir le tableau ci-dessus.
Donner le tableau de résultats normalisé pour cette requête, et en déduire la courbe de rappel/précision.

Tableau normalisé

Rappel	Précision
0	1
0.1	1
0.2	1
0.3	1

0.4	1
0.5	0.75
0.6	0.75
0.7	0.67
0.8	0.67
0.9	0.38
1.0	0.38

Dessin :



Question 2

Réaliser le même travail pour la requête Q2, avec les réponses suivantes :

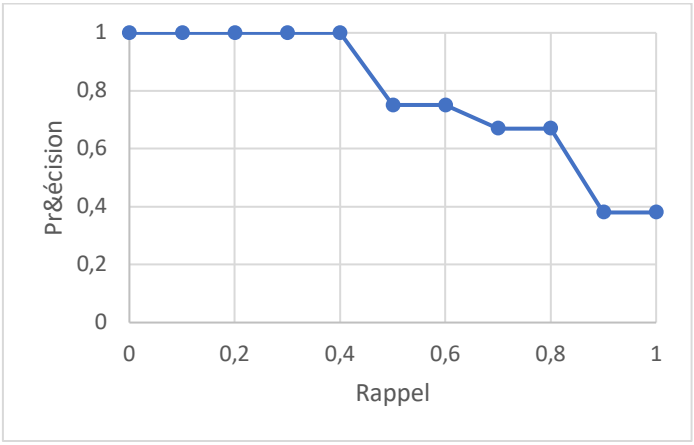
Rang	n° doc	pertinent	Rappel	précision
1	324	X	1/11 = 0.09	1
2	589	X	2/11 = 0.18	1
3	528	X	3/11 = 0.27	1
4	590	X	4/11 = 0.36	1
5	986	X	5/11 = 45	1
6	592	X	6/11 = 55	1
7	899	X	7/11 = 0.64	1
8	988	X	8/11 = 0.73	1
9	578		8/11 = 0.73	8/9 = 0.89
10	985		8/11 = 0.73	8/10 = 0.80
11	537	X	9/11 = 0.82	9/11 = 0.82
12	591	X	10/11 = 0.91	10/12 = 0.83
13	772	X	11/11 = 1	11/13 = 0.85
14	990		11/11 = 1	11/14 = 0.79

La liste des tous les documents pertinents pour la requête Q2 est : 324, 528, 537, 589, 590, 591, 592, 772, 899, 986, 988.

Tableau normalisé

Rappel	Précision
0	1
0.1	1
0.2	1
0.3	1
0.4	1
0.5	1
0.6	1
0.7	1

0.8	0.85
0.9	0.85
1.0	0.85



Question 3

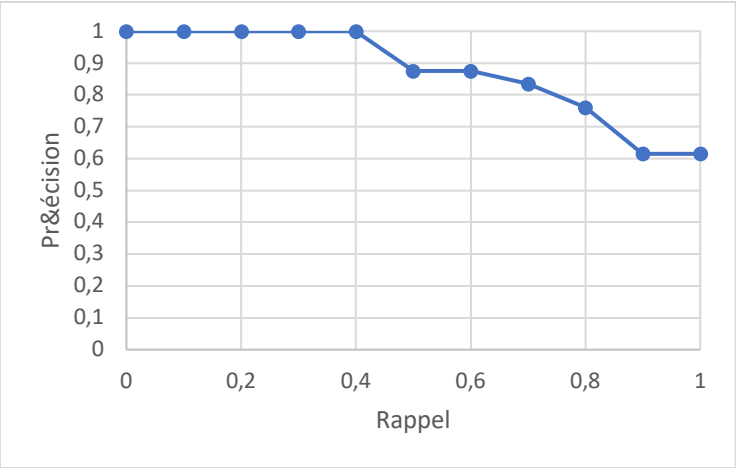
En regardant les courbes, que pouvez-vous déduire de la qualité relative du système pour ces deux requêtes?

Il répond bien aux deux requêtes car les courbes (ou les tableaux) ont des valeurs très élevées.
Il répond mieux à Q2 qu'à Q1.
Dans les 2 cas il est capable de retourner tous les docs pertinents (car on atteint le point de rappel 1 avec une précision non-nulle), ce qui est très bon.

Question 4

Donner le tableau global des résultats du système pour les deux requêtes et dessiner le schéma résultant.
On va moyenner, pour chaque point de rappel, les 2 précisions normalisées :

Rappel	Précision Q1	Précision Q2	Précision
0	1	1	1
0,1	1	1	1
0,2	1	1	1
0,3	1	1	1
0,4	1	1	1
0,5	0,75	1	0,875
0,6	0,75	1	0,875
0,7	0,67	1	0,835
0,8	0,67	0,85	0,76
0,9	0,38	0,85	0,615
1	0,38	0,85	0,615



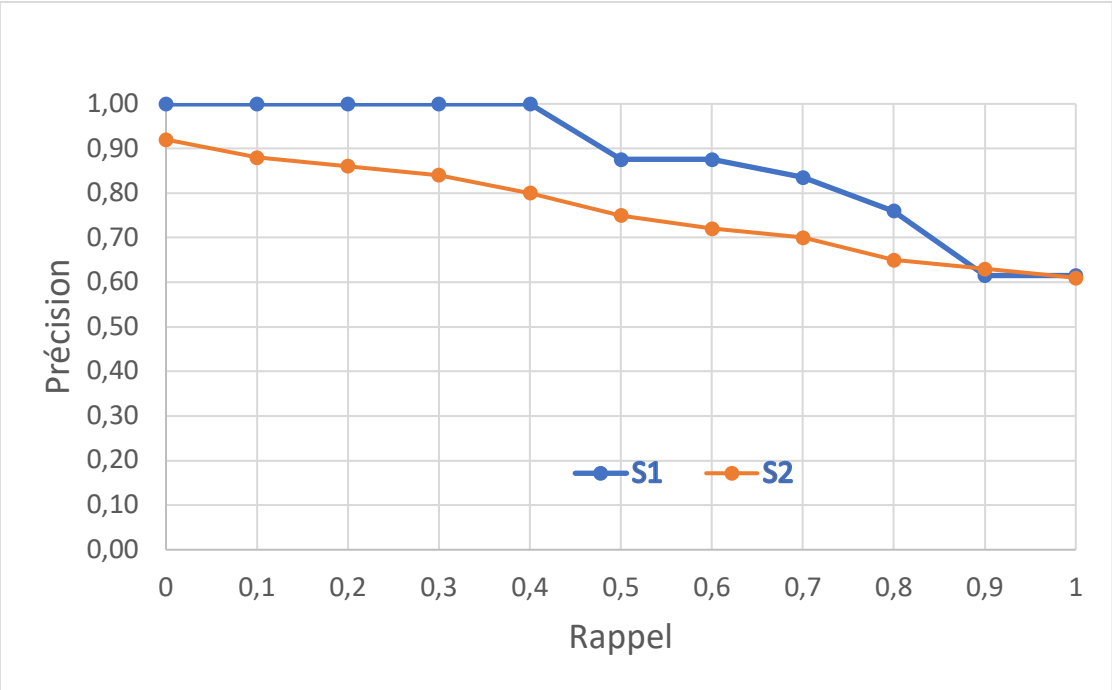
Exercice 3 – Comparaison de SRI

Nous voulons comparer deux systèmes de recherche d'information.
Le premier système S1 est celui de l'exercice 9. Le second système, S2, a pour tableau de rappel/précision pour les deux requêtes Q1 et Q2:

Rappel	Précision
0	0.92
0.1	0.88
0.2	0.86
0.3	0.84
0.4	0.80
0.5	0.75
0.6	0.72
0.7	0.70
0.8	0.65
0.9	0.63
1.0	0.61

Si on compare ces deux systèmes sur le même tableau :

Rappel	Précision S1	Précision 2
0	1	0.92
0,1	1	0.88
0,2	1	0.86
0,3	1	0.84
0,4	1	0.80
0,5	0,875	0.75
0,6	0,875	0.72
0,7	0,835	0.70
0,8	0,76	0.65
0,9	0,615	0.63
1	0,615	0.61



On voit que le système S1 est meilleur (courbe plus haute, en bleu) quasiment pour toutes les valeurs de rappel. Il est très bon au début des réponses (entre 0 et 0,4 de rappel). Les 2 systèmes ont une bonne précision quand le rappel est à 1, ceci veut dire qu'ils sont capables tous les deux de trouver tous les documents pertinents, avec dans cas une précision d'environ 0,6 .