République Algérienne Démocratique et Populaire
Ministère de l'Enseignement Supérieur et de la Recherche Scientifique
Université Mouloud Mammeri de Tizi-Ouzou
Faculté de : Génie électrique et d'informatique
Département : Informatique

MEMOIRE DE FIN D'ETUDE

En vue de l'obtention du diplôme de

Master en Informatique

Spécialité : Ingénierie des systèmes informatiques

Présenté par : NABIL LARBI

Thème

'' Exploitation des signaux sociaux de twitter afin d'améliorer la recherche d'information ''

Proposée et dirigée par **Mme FELLAG Samia**

Présidente	Mme G. SINI
Examinatrice	Mme F. ACHEMOUKH
Promotrice	Mme S. FELLAG

.M

Remerciements

Je tiens à témoigner de toute ma reconnaissance et ma gratitude à ma directrice de mémoire, Madame FELLAG Samia. Que je remercie infiniment pour sa patience et sa disponibilité et surtout de m'avoir encadré, orienté, aidé et conseillé.
Mes remerciements s'adressent également aux membres du jury pour avoir bien voulu examiner et juger ce travail.
Et Je tiens aussi à exprimer mes sincères remerciements à tout le corps professoral et administratif de l'université Mouloud Mammeri de Tizi-Ouzou.
Enfin, je remercie tous ceux qui, de près ou de loin, ont contribué à la réalisation de ce travail.

Dédicaces

Je dédie ce modeste travail à mes très chers parents pour tous leurs sacrifices et encouragements perpétuels

À mes frères et sœurs pour leurs soutiens et encouragement, à toute ma famille et aussi mes très chers amis qui sont restés auprès de moi tout au long de mon travail et qui m'ont aidé à l'accomplir

Résumé

Nous présentons dans ce mémoire une contribution à la modélisation en Recherche d'information en proposant une fonction de reclassement basé sur la pertinence sociale qui considère la dimension sociale du Web. Cette dimension sociale est toute information sociale entourant des documents avec le contexte social des utilisateurs extraite dans notre cas exclusivement du réseau social Twitter. Actuellement, notre approche repose sur l'utilisation des métadonnées sociales, par exemple retweet, commentaire, like et ce afin de déterminer le classement de popularité des hashtags. L'évaluation effectuée selon notre approche montre ses avantages pour le reclassement de résultats (re-ranking).

Mots clés :

Recherche d'information, réseaux sociaux, Twitter, hashtag, re-Ranking

Abstract

In this thesis, we present a contribution to Information Retrieval modeling by proposing a reclassification function based on social relevance that considers the social dimension of the Web. This social dimension is any social information surrounding documents with the social context of users extracted in our case exclusively from the social network Twitter. Currently, our approach is based on the use of social metadata, for example retweet, comment, like, in order to determine the popularity ranking or trendline of hashtags. The evaluation carried out according to our approach shows its advantages for the re-ranking of search results.

Keywords:

Information retrieval, social networks, Twitter, hashtag, re-ranking

Tables des matières

I N T R O D U C T I O N

1- Contexte et problématique :

La Recherche d'Information est un domaine Dont le but est de sélectionner un ensemble de documents à un utilisateur en fonction de son besoin en informations exprimé à l'aidé d'une requête. Le défi est de pouvoir, parmi le volume important de documents disponibles, trouver ceux qui correspondent au mieux aux besoin d'information de l'utilisateur.

Le web 2.0 a remis la RI face à de nouveaux défis d'accès à l'information, il s'agit cette fois de retrouver une information pertinente dans un espace diversifié et de taille considérable, ceci a donné naissance à un nouveau domaine appelé la **recherche d'Information sociale**. Ce domaine permet de combiner entre une pertinence textuelle classique et une pertinence sociale [Badache.2016] qui est basé sur les réactions et traces laissé par l'utilisateur sur les ressources du Web. La motivation derrière l'exploitation de ces contenus, en particulier les signaux sociaux est d'essayer de tirer profit de ces traces provenant des actions collectives des utilisateurs pour améliorer la RI par rapport à un besoin en information. Les principales problématiques liées à cette discipline consistent d'abord, à identifier les ressources sociales issues des réseaux sociaux pouvant répondre aux exigences de l'utilisateur et comment les exploiter pour améliorer le processus de la RI.

Nos travaux se situent dans la recherche d'information sociale, plus précisément, dans le cadre du micro blog *Twitter* pour améliorer la RI.

2- Contribution :

Notre approche dans le domaine de la RI Sociale, s'intéresse particulièrement au hashtag qui est un mot ou phrase suivi du caractère dièse(#) utilisé premièrement sur Twitter pour exprimer le contexte d'un message ou tweet , notre technique se situe autour de l'extraction des hashtags des tweets et de calculé leurs popularité en prenant en compte leurs occurrences et le nombre de signaux sociaux (retweets,commentaire,like) dans les tweets qui les contiennent, ainsi on aura une liste des hashtags classés par ordre de popularités, que nous comparons avec l'index inverse des documents de notre collection , ainsi un document contenant un terme a haute pondération équivalent à un hashtag , se verra bénéficier du score de popularité de ce dernier .

3- Organisation :

Notre document est organisé selon le plan suivant :

- **le chapitre 1** introduit les concepts de base de la RI, Nous commençons par quelques définitions puis ensuite nous allons présenter le processus de la RI en détaillant ses principales étapes. Enfin, nous concluons de la RI et les mesures d'évaluation des systèmes de recherche d'information SRI.
- **Le chapitre 2** présente la recherche d'information sociale (RIS). Nous décrivons d'abord l'information sociale dans le Web. Ensuite, la notion de RI sociale sera définie en mettant en évidence ses concepts de base. Puis, nous présentons un aperçu sur les travaux de l'état de l'art consacrés à l'exploitation des informations sociales dans le processus de RIS. Enfin, nous présentons les principales collections de tests qui sont utilisées pour évaluer les SRI sociale.
- **Le chapitre 3** présente notre contribution qui concerne l'intégration et l'exploitation des signaux sociaux de *Twitter* au sein du processus de la RI et ainsi son expérimentation.

Partie 1

RECHERCHE D'INFORMATION CLASSIQUE

1- Recherche d'information

1.1- Définition

La Recherche d'Information (RI) est un vaste domaine de l'informatique initialement destiné à fournir aux utilisateurs un accès facile à l'information dont ils ont besoin, l'une des premières définitions est celle de [*Salton*,1968] :

" La recherche d'information est un domaine qui concerne la structure, l'analyse, l'organisation, le stockage, la recherche et la récupération d'informations «

En termes de recherche académique, la RI peut être étudiée en deux axes principaux complémentaire l'un à l'autre : machine et utilisateur. Du point de vue machine, la RI doit construire des indexes efficaces et traiter les requêtes utilisateurs avec une haute performance et développer des algorithmes de classement pour améliorer les résultats. Concernant le point de vue utilisateurs la RI doit étudier les habitudes des utilisateurs et comprendre leurs besoins en informations, et en essayant aussi de comprendre la nature « floue » de la pensés humaine

2- Processus de la RI

Pour décrire un SRI ou « Système de Recherche d'information » on utilise une architecture d'application très simple comme le montre la **figure 1.1**, la première étape consiste à assembler une collection de documents puis viendra la deuxième étape plus importante et complexe qui est basé sur trois notions : **l'indexation, le requêtage et l'appariement.**

2.1- Collection de documents :

Un système RI classique dispose d'une collection de document alimenté manuellement

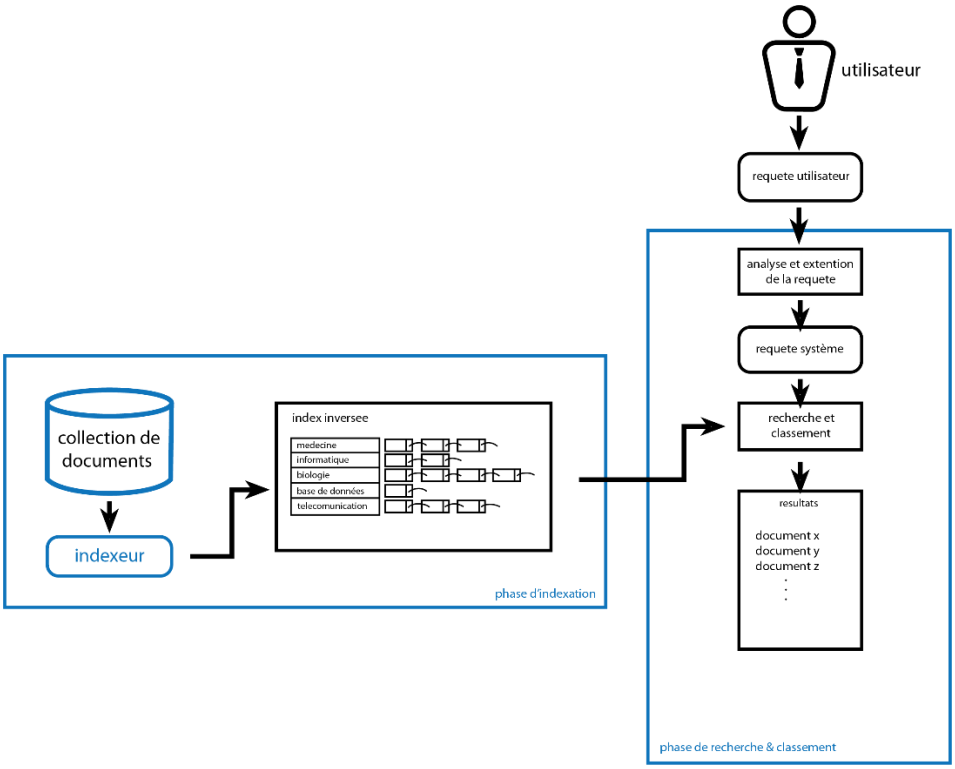


Figure1.1 **:** *les 3 processus de la RI : indexation, recherche et classement (Ranking) [Baeza et Ribeiro]*

2.2- Indexation, appariement et requêtage :

2.2.1- L'indexation

l'indexation est un processus qui **« transforme les documents en substituts capables de représenter leurs contenus »** [Salton et McGill,1983] ,cette étape consiste à identifier pour chaque document ces termes importants , puis à exploiter ces termes comme index pour avoir accès plus rapidement au document , l'index dans son sens général est composé de mots distincts (après suppression des répétitions et des mots vides) de chaque document et pour chaque mot les document qui en contient une instance ou plus. Le plus utilise des index est **« l'index inversé »** que l'on va définir ultérieurement, il existe 3 types d'indexation : automatique, manuelle et semi-automatique

- **L'indexation automatique :** on l'utilise pour l'indexation des documents, c'est un processus totalement automatisé.
- **Indexation manuelle :** chaque document est analysé par un expert qui identifie les mots clés qui y sont contenus et ensuite classé dans un index.
- **Indexation semi-automatique :** basé sur un processus automatique assisté par des personnes expertes du domaine qui ont la décision finale sur le choix des mots clés

Def. Index inverse : Un index inversé est un mécanisme dont la structure est composée de deux éléments : **vocabulaire et occurrences**, on peut aussi y ajouter d'autres paramètres si besoin, le vocabulaire ou le dictionnaire est composé des différents mots contenus dans le document, pour chaque mot les documents le contenant, avec un index inverse vu qu'on peut reconstruire les documents d'une manière inverse en utilisant l'index.
L'index inverse est construit sous forme matricielle, le problème avec cette simple solution est qu'elle requiert beaucoup d'espace

L'indexation automatique comprend plusieurs traitements automatisés internes, appliqué aux documents et au requêtes utilisateurs :

- **L'extraction des mots (tokenisation),**
- **L'élimination des mots vides,**
- **La normalisation**
- **La pondération**.

2.2.1.1- Extraction des mots (tokenisation) :

Comme son nom l'indique cette phase consiste à identifier les unités lexicales élémentaires (tokens et mots) du texte du document, cette étape est indispensable pour tout travail sur le texte vu qu'elle permet la sélection des tokens qui seront candidats à la l'indexation et à la recherche dans une requête, ces tokens sont délimités par des séparateurs (espaces, signes de ponctuation …)

2.2.1.2- Elimination des mots vides

Les textes contiennent souvent des termes non significatifs appelés mots vides (pronoms personnels, prépositions, etc.) l'objectif est de les identifier et ainsi les supprimer, ces mots constitues jusqu'à 80 % du contenu d'un texte et les supprimer permet d'économiser un espace considérable dans un index et d'optimiser la recherche,

2.2.1.3- Normalisation

Cette phase est liée à la lemmatisation ou racinisation bien que légèrement différent, il s'agit d'un traitement morphologique des mots permettant de regrouper les variantes d'un mot. En effet, dans un texte, il peut y avoir différentes formes d'un mot désignant le même sens, le but est de les regrouper sous un seul mot portant leur sens commun, les documents contenant différentes formes d'un même mot auront les mêmes chances d'être restitués et elle réduit la taille de l'index et améliore le rappel mais elle peut réduire la précision

- **Lemmatisation vs racinisation :**

  Ces deux procédés bien que très proches ont plusieurs différences fondamentales, La Lemmatisation a pour objectif de retrouver le lemme d'un mot, par exemple l'infinitif pour les verbes et la forme au singulier pour les noms et adjectifs. La racinisation consiste à supprimer la fin des mots et d'en extraire une racine.

  Aussi la normalisation fait appel à des algorithmes ou « racinicateurs » dont le plus connu dans le monde anglophone est l'algorithme **Porter** « simple et élégant » [*Martin Porter*.1980] spécialisé pour la langue anglaise il a été plus tard adapté pour la langue française,

2.2.1.4- Pondération des mots

Cette étape vient après identification des termes représentant les documents et leurs normalisations, ces termes n'ont pas la même importance, donc la pondération est une phase clé dans le processus de l'indexation puisqu'elle traduit l'importance des termes dans les documents

Parmi les formes de pondération utilisées :

- **TF×IDF (Term Frequency and inverse document frequency)**

La fréquence du terme **TF** et la fréquence inverse du document **IDF** sont les fondements de la plus rependue des fonctions de poids de termes dans la RI qui est appelles la **TF-IDF** Les facteurs TF et IDF sont définie comme suit :

1. Fréquence du Terme (TF) : ce facteur prend en compte le nombre d'occurrences d'un terme dans un document qui est base sur les travaux de [*Hans Peter Luhn*,1957] ou **l'hypothèse de Luhn** qui dit que plus un terme k_i est présent dans un texte de document d_j plus la fréquence de terme $TF_{i,j}$ est haute , cette hypothèse est fondée sur le fait que les mots à haute fréquence d'apparition sont importants pour représenter les sujets clés d'un document .On peut formuler TF aussi comme suit :

$$TF_{i,j} = 1 + \log(f_{i,j})$$

Avec $f_{i,j}$ est le nombre d'occurrences du terme k_i dans le document d_j

2. Fréquence Inverse du Document (IDF) : ce facteur mesure la fréquence d'un terme dans toute la collection ou la pondération globale, plus un terme est rare dans la collection plus son importance augmente et aussi le contraire, l'IDF est un facteur complémentaire de la fréquence de terme (TF)

$$IDF_i = \log\frac{N}{n_i}$$

Avec N le nombre de documents dans la collection et n_i et le nombre de documents contenant le terme k_i

2.2.1.5- Normalisation de la longueur des documents :

Dans une collection, la longueur des documents peut varier énormément, ceci peut poser un problème vu que les documents trop longs seront les plus retournés pour une requête donnée pour la simple raison qu'ils contiennent plus de mots, pour atténuer cet effet indésirable on peut diviser le rang ou classement de ce document sur sa longueur.

2.2.2- Requêtage :

La recherche vise à sélectionner les documents pertinents qui couvrent les besoins en informations de l'utilisateur. Cette phase dépend de l'information recherchée par l'utilisateur et de la représentation des documents et les préférences de l'utilisateur (la langue, la date, la localisation, le format, etc.).

Cette étape s'intéresse à l'expression des besoins de l'utilisateur, souvent à travers une liste de mots-clés représentant la requête, Ainsi la requête soumise par l'utilisateur subit les mêmes traitements que ceux réalisés précédemment (correction, élimination des mots-vide, etc.) Sur les documents au cours de leur indexation. Toutefois, la requête peut être étendue où reformulée pour renforcer les préférences des utilisateurs et le retour de pertinence.

2.2.3- Appariement :

Le système de RI procède à la mesure de pertinence de chaque document vis-à-vis du besoin d'information (requête) selon une fonction de correspondance relative au modèle de recherche, et à renvoyer ensuite à l'utilisateur une liste de résultats. Cette mise en correspondance génère un score de pertinence reflétant le degré de similarité entre la requête et le document. Ce score est calculé à partir d'une valeur appelée $RSV(q,d)$ **Retrieval Status Value**, où q représente une requête et d un document. Cette mesure de pertinence système, que l'on essaye de rapprocher le plus possible du jugement de pertinence de l'utilisateur vis-à-vis du document, prend en compte les pondérations de termes calculés au moment de l'indexation. Le score final permet d'ordonner les documents retournés. Certains de ces documents parmi les résultats retournés par le système de RI peuvent potentiellement satisfaire les besoins de l'utilisateur. Ces documents sont appelés documents **pertinents**.

La pertinence est l'un des concepts fondamentaux de la recherche d'information, c'est une métrique utilisée pour mesurer l'utilité d'un résultat en dépendant de la requête de l'utilisateur, [Bradford.1950] le premier à utiliser le terme « relevant » ou pertinent en anglais pour caractériser des articles ayant relation avec un sujet

« Le premier but d'un SRI et de trouver tous les documents pertinents et en même temps aucun document non pertinent »

La difficulté ne réside pas seulement de savoir comment extraire l'information des documents mais aussi de savoir comment décider de leur utilité ou précisément leur pertinence pour ceci on utilise de multiple paramètres « la location, la date … »

Un système parfait ne doit retourner que des documents pertinents, en rejetant les non-pertinents, Aujourd'hui les systèmes retournent généralement une liste classée de documents,

dans lesquels les documents en tête de liste sont ceux qui sont les plus susceptibles d'intéresser les utilisateurs, ou les plus susceptibles d'être pertinents. En effet, l'utilisateur ne consulte généralement que les premiers documents renvoyés

2.2.3.1- Modèles de la RI

Les modèles proposés en recherche d'information dans la littérature comportent trois caractéristiques principales de documents, Dans ce qui suit, nous présentons les principaux modèles de recherche d'information sur la base de ces propriétés

1- Les modèles ensemblistes : ces modèles trouvent leurs fondements théoriques dans la théorie des ensembles. On distingue le modèle booléen pur ou de base (Boolean model), le modèle booléen étendu (extended Boolean model) et le modèle basé sur les ensembles flous (fuzzy set model).
2- Les modèles vectoriels : basés sur l'algèbre, plus précisément le calcul vectoriel. Ils englobent le modèle vectoriel (vector model), le modèle vectoriel généralisé (generalized vector model), Latent Semantic Indexing (LSI) et le modèle connexionniste (neural network model).
3- Les modèles probabilistes : se basent sur les probabilités. Ils comprennent le modèle probabiliste général, le modèle de réseau de document ou d'inférence (Document Network) et le modèle de langue

Ces différents types de modèles de systèmes de recherche se distinguent par le processus d'indexation ou de formulation de requêtes mais surtout par le processus d'appariement des résultats

- **Les modèles ensemblistes :**

Les modèles ensemblistes ont été les premiers à avoir été mis en place, de par leur simplicité, basé sur la théorie des ensembles et l'algèbre de Boole, dans ces modèles le document est représenté comme l'ensemble de ses termes. Les requêtes elles sont représentées par un ensemble de mots-clés exprimant le besoin en information, cette catégorie est constituée du modèle booléen de base, le modèle booléen étendu et le modèle ensembliste flou (fuzzy-set model)

1- Le modèle booléen de base : ce modèle est très simpliste (l'un des premiers dans le domaine de la RI), basé sur l'algèbre de Boole, c'est un modèle assez intuitif avec des paramètre très précis car il ne restitue que les documents répondant exactement aux termes de la requête, qui considère qu'un terme d'index est soit présent ou absent dans un document, c'est à dire que les fréquences des termes-document dans le document dans la *matrice des termes-document* sont des valeurs binaires (0 ;1),

- **Le modèle vectoriel :**

Ce modèle propose d'assigner à chaque terme document ou requête issu de l'index un vecteur dans un espace à n-dimension ou n est le nombre de termes dans l'index. Plusieurs formules ont été propose dont

1- **Le produit scalaire :**

$$RSV(d,q) = \sum_{i=1}^{n} d_i . q_i$$

2- **La mesure Cosinus :**

$$RSV(d,q) = \frac{\sum_{i=1}^{n} d_i . q_i}{\sqrt{\sum_{i=1}^{n} d_i^2 . \sum_{i=1}^{n} q_i^2}}$$

3- **La mesure de Jaccard :**

$$RSV(d,q) = \frac{\sum_{i=1}^{n} d_i . q_i}{\sum_{i=1}^{n} d_i^2 + \sum_{i=1}^{n} q_i^2 - \sum_{i=1}^{n} d_i . q_i}$$

4- **La mesure de Dice :**

$$RSV(d,q) = \frac{2 \times \sum_{i=1}^{n} d_i . q_i}{\sum_{i=1}^{n} d_i^2 + \sum_{i=1}^{n} q_i^2}$$

- **Les modèles probabilistes :**

[Robertston&Spack,1976], La pertinence d'un document vis-à-vis d'une requête est vue comme une probabilité de pertinence document/requête, Ces probabilités sont estimées par les probabilités qu'un terme de la requête soit dans un document pertinent et non pertinent

1- Le modèle probabiliste général : très proche du modèle probabiliste explique en dessus, pour tout document on calcule les probabilités de pertinence $P(d/q)$ et de non pertinence $P(\bar{d}/q)$ comme suit :

$$RSV(q,d) = \frac{P\left(d/q\right)}{P\left(\bar{d}/q\right)} = \sum_{i=1}^{t} \log \frac{p(q-1)}{q(p-1)}$$

Ou p = P (terme t_i présent / d pertinent), q = P (terme t_i présent / d non pertinent) et t est le nombre total de termes dans la requête Ainsi les documents sont classés par ordre de probabilité de pertinence

2- Le modèle de langue : issu des modèles probabilistes de génération de langage développé pour les systèmes de reconnaissance automatique de parole, L'objectif d'un modèle de langue est de capter les régularités linguistiques d'une langue, en observant la distribution des mots, successions de mots, dans une langue donnée

2.3- L'évaluation et les collections de test :

En RI , l'évaluation est très importante pour déterminer l'efficacité des modèles, car les résultats d'un SRI peuvent être interprété différemment par des utilisateurs multiples (un document peut

être pertinent pour certains et non-pertinent pour d'autre) , et pour ça il n y'a pas de systèmes d'évaluation stricte car on utilise des méthodes approximative basé sur des moyennes d'utilisation d'une population d'utilisateurs et surtout la seule chose qu'on peut évaluer c'est la qualités des résultats (ou leur pertinence) et non la performance du système , et cette évaluation utilise la méthodologie du **paradigme de Cranfeild** par [*Cleverdon*.1956] en se basant sur 3 éléments :

1- une collection de documents sur laquelle les recherches sont effectuées,
2- un ensemble de requêtes de test (besoins des utilisateurs)
3- la liste des documents pertinents pour chacune des requêtes établies manuellement par des experts du domaine (jugements de pertinence).

L'idée générale de ce paradigme est de créer un environnement unique et étudié afin de pouvoir comparer les systèmes équitablement, théoriquement un système efficace doit effectuer les mêmes choix que les opérateurs humains (utilisateurs ou experts). Cet environnement communément appelé **la collection de test**

2.3.1- La Collection de test :

Les collections ou Corpus de test permettent de comparer directement les résultats produit par des fonctions de classement distincte, cette méthode tient pour source les travaux de *Cyril Cleverdon* (paradigme de Cranfeild)

Chaque corpus contient une collection de documents D , et une collection de requêtes I et enfin les jugements de pertinence document/requête représenté de manière binaire ces jugements de pertinence sont produits par des opérateurs humains et ceci bien sûr dans le cas des petites collections comme celle de *Cranfeild* , voici quelque exemple de collection de test :

1- la collection TREC : TREC est une conférence très connu dans le domaine de la RI initie premièrement par le NIST et la DARPA toute les deux des agences gouvernementale américaine , la première civil et l'autre militaire dans le cadre du programme TRIPSTER , la collection TREC est née d'un besoin émis d'une collection qui reflète réellement les contenus du web qui grandissaient de manière exponentielle à l'époque (années 90) en 2004 la collection TREC.GOV2 contenaient 25 millions de documents web
2- Collection Thomson Reuters : collection assemble par l'agence de presse Thomson Reuters, elle contient 800k document en langue anglaise issu d'articles de presse, plus tard un deuxième corpus a été introduit avec 13 langues (dont le français) cette fois avec plus de 1.8 millions de documents

2.3.2- Mesures d'évaluation :

Dans cette section on fera la lumière sur les différentes métriques d'évaluation de la qualité des résultats, les plus utilisées sont la ***précision*** et le ***rappel***

2.3.2.1- Rappel & précision :

On considère une requête Q (issu d'une collection de test) et R un ensemble de documents pertinents et $|R|$ leur nombre Maintenant on a un algorithme de classement (en évaluation) qui exécute la requête Q et génère ensuite un ensemble de résultats A et $|A|$ leur nombre, et $|R \cap A|$ le nombre d'intersection entre les deux ensembles R et A comme le montre la **figure 1.3** *:*

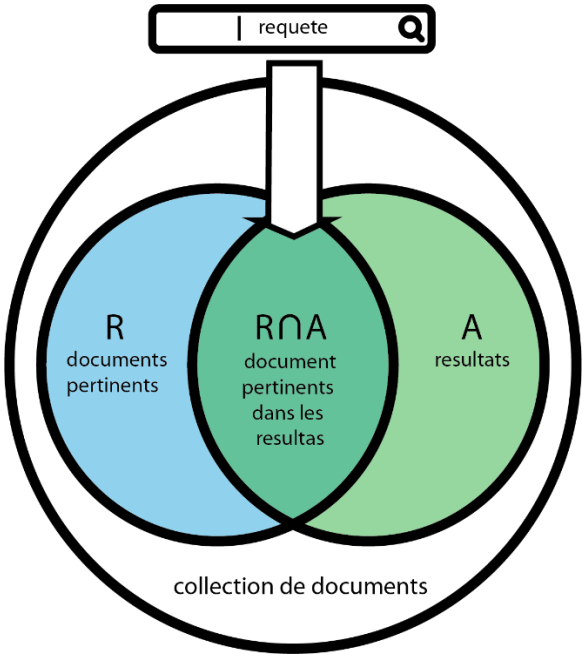


Figure 1.3 : précision & rappel

La précision : c'est la fraction des documents retrouvés (ensemble A) qui sont pertinents (ensemble R)

$$Precision = P = \frac{|R \cap A|}{|A|}$$

Le rappel : c'est la fraction des documents pertinents (ensemble R) qui ont été retrouves (ensemble A)

$$Rappel = R = \frac{|R \cap A|}{|R|}$$

En utilisant ces deux mesures on peut tracer la *courbe précision-rappel* Ainsi, si le résultat de recherche dépend d'un certain paramètre, par exemple le rang d'un document restitué, alors pour chaque valeur du paramètre les valeurs de rappel et précision peuvent être calculées
En théorie un système parfait ($Precision(P) = Rappel(R) = 1$) trouvera tout et que les résultats pertinents, en pratique ces deux taux varient en sens inverse, comme le montre la **figure 1.3**, quant à l'évaluation des algorithmes de classement on cherche à maximiser les deux mesures (précision-rappel)

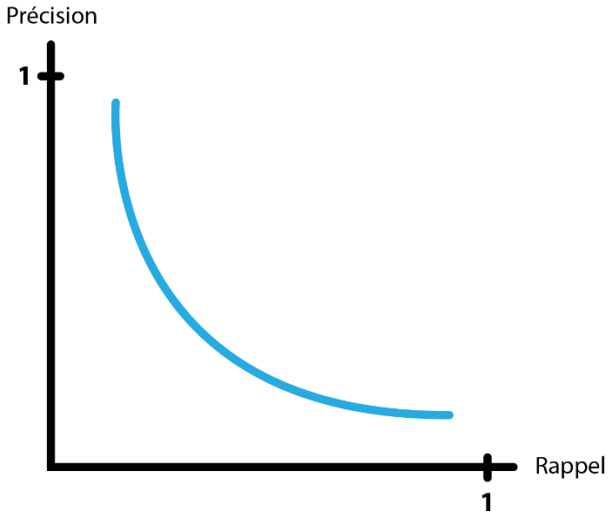


Figure 1.4 : Les mesures de précision-rappel ne sont pas statique

2.3.2.2- Les moyennes rappel-précision

La mesure de rappel et de précision utilisé seule et séparément ne sont pas de bon indicateurs de la performance d'un SRI , alors pour les évaluer on utilise des agrégations rappel-précision ou moyennes , parmi elle y'a le F-score ou F-mesure élaboré par *Van Rijsbergen* qui permet de combiner les deux paramètres rappel-précision en une mesure unique sous forme de moyenne harmonique (plus précise que la moyenne arithmétique dans le cas des algorithmes de classement) mais la F-score est une mesure qui ne permet que d'évaluer les ensembles non ordonnées (ne tient pas compte du classement qui est l'atout principale d'un algorithme de ranking efficace) alors dans ce cas on a les mesures d'évaluation ordonnées

Conclusion

La RI classique ne s'intéresse qu'aux données textuelles. Cependant, avec l'émergence du web 2.0 et l'apparition des réseaux sociaux, l'enjeux est devenu beaucoup plus important car d'autres critères ont fait leurs apparitions, comme les signaux sociaux (j'aime, partage, commentaire …), la temporalité … etc., Dans le prochain chapitre nous parlerons de la RI qui prend en considération ces paramètres : **La RI Sociale**

Partie 2

RECHERCHE D'INFORMATION SOCIALE

Introduction

La recherché d'information (RI) est un domaine qui consiste à définir des modèles et des processus dont le but de retourner, à partir d'un corpus dé documents indexés, ceux dont le contenu correspond le mieux au besoin en information exprimé par un utilisateur. Initialement développée pour des corpus de documents textuels, la RI a évolué avec l'émergence du Web et plus récemment des réseaux sociaux (RS). De nos jours, les RS représentent le moyen le plus utilisé pour la communication, le partage de connaissance et de contenus sur le Web. Avec cette dimension sociale qui vient enrichir les contenus des ressources sur le Web, les utilisateurs se retrouvent avec de nouveaux besoins en information. La RI classique ne semble pas adaptée à cette dimension, impliquant les utilisateurs et leurs interactions au sein des réseaux sociaux, d'où l'émergence de la RI Sociale (RIS), une thématique récente qui a pour objectif de prendre en compte les informations spécifiques aux RS.

1- Recherche d'information sociale (RIS)

Définition

L'idée derrière la RI Sociale est d'intégrer le contenu social généré les interactions entre utilisateurs pour améliorer la qualité des documents fournit lors des résultats de recherche, Dans ce chapitre nous présentons la recherche d'information sociale. Nous donnons tout d'abord un panorama du type d'information sociale présent dans le Web. Ensuite, nous allons définir la notion de la RI sociale, en mettant en avant les principales tâches de ce domaine. Ensuite, nous présentons un aperçu sur l'état de l'art du domaine [*Kirsch*,2005], [*Kirsch et al*,2006], [*Goh et Shubert*,2010]et [*Badache*,2016]

2.1- L'information sociale dans le web

L'information sociale est toute information générée par les internautes dans le web, elle est le fruit du web 2.0, bien avant l'utilisateur avait seulement le rôle de consommateur d'informations, ces nouvelles fonctionnalités leurs ont permis d'être aussi des producteurs de données qu'on appelle le contenu généré par l'utilisateur ou ‘’user générâtes content’’ UGC qui s'agit de :

- Les contenus publiés dans les réseaux sociaux (Facebook, Instagram, Pinterest …)
- Les relations sociales entre utilisateurs (amis, abonnées, followers …)
- Commentaires et publications de contenus multimédia …….

2.1.1- Les réseaux sociaux :

On définit les sites de réseaux sociaux comme des services Web permettant aux utilisateurs de créer un profil public ou semi-public dans un système lié, d'énoncer une liste d'autres utilisateurs avec lesquels ils partagent une connexion, et aussi consulter et parcourir leur liste de connexions et celles établies par d'autres personnes au sein du système

Ce qui rend les sites de réseaux sociaux spéciaux c'est leurs capacités à permettre d'établir une chaine sociale plus complexe que celle possible dans la réalité car un profil utilisateur sur un réseau social est théoriquement visible pour tout le monde, ce principe est expliqué dans **la figure 2.1**

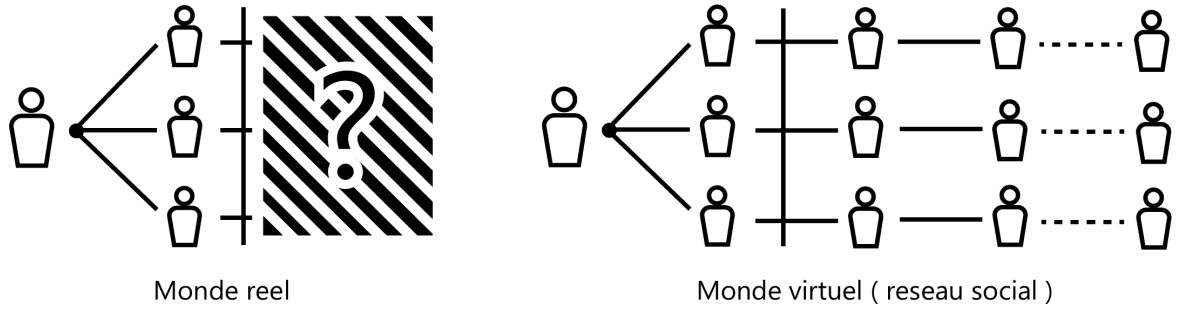


Figure 2.1 : monde réel et monde virtuel (réseau social)

Citons quelques réseaux sociaux populaires dans le monde :

- **Facebook :** avec 2 milliards de comptes actifs en 2017 c'est le plus grand réseau social du monde fondé en 2004 par *Marc Zuckerberg* et ces camarades, Facebook permet à tout utilisateurs de posséder un compte gratuitement et être en contacts avec ces amis, famille et équipe professionnel, tout en ayant la possibilité de partager avec eux des publications (informations, centre d'intérêts…) Ou encore créer des groupes ou communauté dans le but de promouvoir une idée, cause ou même faire connaitre une société, Facebook a fait l'objet de plusieurs critiques et controverses : non-respect de la vie privé, contrôle de l'information, fuite de données …
- **Instagram :** Instagram est un réseau social de partage de contenus multimédia (photos et vidéos) fonde en 2010 par *Kevin Systrom* et *Michel Mike Kreiger*, racheté en 2012 par le groupe Facebook, cette application est devenue tellement populaire qu'elle compte déjà un milliard d'utilisateurs en 2018 c'est l'un des media sociaux majeurs dans le monde qui a introduit le concept de '' l'influence sociale '' dans le domaine du web
- **Twitter :** Twitter a été créé le 21 mars 2006 par *Jack Dorsey*, *Evan Williams*, *Biz Stone* et *Noah Glass***,** c'est un service de micro blogage qui permet aux utilisateurs de s'envoyer des petits messages appelés '' tweets '', le service est rapidement devenu très populaire comptant en 2017, 313 millions d'utilisateur actifs.
- **YouTube :** c'est un site web d'hébergement de vidéos et un media social sur lequel les utilisateurs peuvent envoyer, regarder, commenter, évaluer et partager des vidéos. Il a été créé en février 2005 par *Steve Chen*, *Chad Hurley* et *Karim Jawad* puis racheté par google en 2006, YouTube a beaucoup contribué dans l'éducation, la formation en ligne et la promotion d'artistes au-devant de la scène
- **Pinterest :** Pinterest est un site de partage de centre d'intérêts orienté photographie ou dessin de toute sorte, créé en 2010 par *Paul Sciarra*, *Evan Sharp* et *Paul Silbermann*
- **Deviantart :** Deviantart est une communauté artistique online lancé en 2000 par *Angelo Sotira* et ces amis, les utilisateurs peuvent exposer leurs œuvres d'art et les noter, commenter et même se contacter pour des coopérations ou projets professionnels

Les intérêts de ces réseaux sociaux sont multiples, le plus décisif est la baisse drastique des coûts des télécommunications à grande distance (l'information peut atteindre des millions d'utilisateurs instantanément) et aussi le milieu professionnel les voit comme des outils indispensables pour améliorer la visibilité des entreprises sur le Web et aussi dans le marketing cible , en ce qui concerne notre étude sur la RIS les réseaux sociaux

suscitent beaucoup d'intérêts en termes de contenu social généré par leurs utilisateurs, qui peut être utile dans les tâches de recherche d'information.

2.1.2- Contenu généré par les utilisateurs UGC :

Voici une petite définition :

> *L'UGC ou les données générées par les utilisateurs sont toute forme de contenu textuelle ou multimédia dont la source est l'utilisateur (photo, vidéo, commentaire, texte, image …) et qui est ensuite poste sur une plateforme web (ex. Réseau social)*

Aujourd'hui, les réseaux sociaux sont la principale source des UGC's soit directement en leur permettant d'y publier du contenu sous forme d'idées et opinions avec d'autre personnes ou indirectement sous forme d'annotations (hashtag) ou toute réaction envers une ressource à travers des boutons (j'aime, partage, commentaire…), dans ce qui suit nous allons nous intéresser spécialement à ce type d'UGC

Le contenu social (UGC social) se compose de 4 types d'interactions : ***Content-Content, Content-Person, Person-Content, Person-Person*** ces interactions sont représenté par un cycle qui définit la relation entre l'utilisateur et la ressource [*Sihem Amer-Yahia et al*,2007], **figure 2.4**

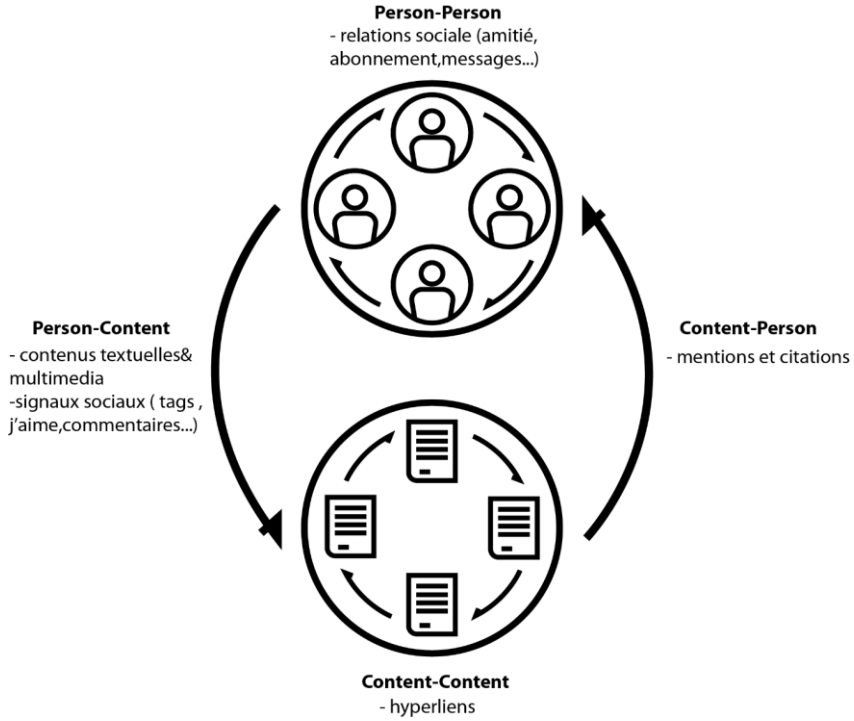


Figure 2.4 : graphe du contenu social [Sihem Amer-Yahia et al,2007],

Le contenu généré par l'utilisateur devient de plus en plus prisé par les plates formes web. De ce fait ils ont confectionné des outils afin d'inciter les utilisateurs du web à créer des contenus, et parallèlement d'autre outils pour collecter, trier et vérifier la fiabilité de ces données, l'UGC à changer l'aspect de l'information dans son format traditionnels (télévision, journaux, livres…)

2.1.2.1- Les signaux sociaux :

Aujourd'hui, les signaux sociaux représentent un des types d'UGCs les plus populaires du Web. En effet, aujourd'hui la quasi-totalité des pages Web comprennent différents boutons qui font office de balise vers les réseaux sociaux (Facebook, Twitter…), elles permettent aux utilisateurs d'exprimer s'ils soutiennent, recommandent ou n'aiment pas un contenu (texte, image, vidéo …)
Un *signal social* est donc une mesure de l'activité des médias sociaux. C'est une interaction sociale d'une personne réelle avec une ressource sur le Web à travers des fonctionnalités offertes par les réseaux sociaux, ces interactions peuvent être interprété comme une mesure de popularité d'une ressource et ainsi améliorer son classement dans les moteurs de recherche.

- **Type de signaux sociaux :** En général, chaque réseau social emploie ses propres signaux sociaux (sauf cas ou plusieurs réseaux sociaux appartenant à la même société mère ex Facebook, Instagram) dont les règles de fonctionnement diffèrent et qui n'entraînent pas toutes les mêmes significations et le même impact sur la stratégie Web

Type	Exemple	Réseaux Sociaux
Vote	J'aime +1	Facebook, YouTube
Message	Publication Tweet	Facebook, Twitter
Partage	Partage Retweet	Facebook, Twitter
Signet	Favoris Epingler **Hashtag**	Instagram, Pinterest, Twitter
Commentaire	Commentaire Répondre	Facebook, YouTube
Relation	Abonnées Amis	Facebook, Instagram

Tableau 2.1 : Liste des différents types des signaux sociaux (Wikipédia.org)

2- Notion de la RIS :

L'émergence des réseaux sociaux dans la vie quotidienne des utilisateurs, en produisant des informations qui sont rarement disponibles dans d'autres espaces d'Internet, a contesté les approches traditionnelles de RI qui classent les documents indépendamment de leur contexte social. Pour résoudre ce problème, la RI sociale prévoit une nouvelle génération de modèles de recherche qui font usage de la structure de réseau social et de données sociales afin d'améliorer le processus de RI
On définit la recherche d'information sociale par la prise en compte des données des réseaux sociaux dans le processus de recherche d'information [*Kirsch et al*,2006] :

Les systèmes de recherche d'informations sociales se distinguent des autres systèmes, par l'incorporation d'informations des réseaux sociaux et des relations dans un processus de recherche d'informations

On peut donc schématiser ces dires :

Recherche d'informations sociales = réseaux sociaux + recherche d'informations

La RI sociale combine documents et requêtes En incorporant des individus (utilisateurs des réseaux sociaux) dans le modèle, nous obtenons un meilleur aperçu de leur rôle dans le processus de recherche et de production d'informations. De nouvelles associations apparaissent entre les entités : les individus apparaissent dans leur rôle de producteurs ou de consommateurs d'informations, les requêtes portent sur les besoins en informations d'un individu ou décrivent un sujet sur lequel un individu possède des connaissances

3- Etat de l'art RIS :

Les approches de RI sociale ont étendu les modèles traditionnels avec différentes caractéristiques sociales afin de satisfaire des motivations sociales derrière les besoins d'information de l'utilisateur. En outre, de nouvelles approches sociales entrent en vue afin de répondre aux nouveaux besoins en RI. Une des différentes et plus importantes définitions proposée ces dernières années prend en considération la RIS selon elle sur 3 axes [*Teevan et al*,2012] :

- Le premier axe concerne la recherche d'information de nature sociale sous une forme basique. Il s'agit de retrouver des informations sociales qui répondent à l'utilisateur sur le Web. On distingue par exemple la recherche d'information dans les blogs, microblogs (forums), la recherche de conversations, la recherche des experts, ou encore des réponses à des questions spécifiques auprès des amis, familles, collègues, ou même des personnes inconnues, etc.
- La deuxième axe est en effet le sujet principale de notre étude, est se base sur l'exploitation des contenus sociaux pour améliorer la RI, dans laquelle l'information sociale est utilisée pour améliorer le processus de recherche d'informations, par exemple, les tags et autre signaux sociaux ont été trouvés utiles pour améliorer la recherche Web et la recherche **personnalisée, le reclassement (re-ranking) de résultats de recherche, la reformulation (expansion) de requête, la personnalisation**, etc.
- Le troisième axe concerne la recherche d'information effectuée par plusieurs personnes, ou la recherche collaborative qui est toute tâche de type résolution de problèmes, impliquant plusieurs individus interagissant, de manière synchrone ou asynchrone, lors d'une tâche commune de recherche de sites ou de pages web [*Hansen et Jarvelin*,2005], Cette catégorie est loin du cadre que nous traitons, nous ne la décrivons pas.

3.1- Exploitation de contenus sociaux pour améliorer la RI :

Cette catégorie consiste à améliorer le processus de la RI classique en utilisant l'information sociale comme une nouvelle source d'évidence qui peut intervenir à différents niveaux. Il existe principalement trois niveaux d'amélioration :

1- l'amélioration de l'index, à savoir la façon dont les documents et les requêtes sont représentés et appariés pour estimer leurs similarités.

2- la reformulation des requêtes à l'aide de connaissances supplémentaires, à savoir l'expansion de la requête de l'utilisateur.

3- le reclassement (re-Ranking) des documents retournés par un SRI (sur la base du profil utilisateur ou d'autres facteurs de pertinence sociale).
Dans cette catégorie de RI sociale, nous considérons l'exploitation des contenus sociaux dans ces trois pistes.

3.1.1- Indexation sociale :

Les pratiques d'indexation sociale reposent sur l'utilisation de systèmes permettant aux utilisateurs d'attribuer librement des mots-clés
Plusieurs travaux [*Bischoff et al*,2008] [*Dmitriev et al*,2006] ont indiqué que l'ajout des tags au contenu du document améliore la qualité de la recherche, car ils sont de bons résumés de documents. En particulier, l'information sociale peut être utile pour les documents qui contiennent quelques termes où le processus d'indexation simple ne fournit pas une bonne performance de RI. Dans tous ces travaux cités l'information sociale a été utilisée de deux manières différentes pour l'amélioration de la représentation du document :

(i) soit par l'ajout de métadonnées sociales au contenu des documents,
(ii) en personnalisant la représentation des documents,

En supposant que chaque utilisateur a sa propre vision sur un document donné Par rapport au premier point, certaines approches étudient l'utilisation des métadonnées sociales pour enrichir le contenu des documents ou aussi indexer le document à la fois avec son contenu textuel et ses contenus sociaux associés (tags et commentaires) [*Chen et al*,2009]. [*Carmel et al*,2010] [*Chelaru et al*,2012]. Cependant, chaque approche utilise une méthode différente pour pondérer les termes des métadonnées sociale, par exemple, TF-IDF. Concernant le deuxième point, étant donné un document, chaque utilisateur possède sa propre compréhension de son contenu. Chaque utilisateur emploie son propre vocabulaire pour décrire, commenter et annoter ce document. Par conséquent, la solution est de créer des indexes personnalisés.

3.1.2- La reformulation de requête :

La reformulation de requête est un processus qui consiste à transformer une requête q initiale en une autre requête q_o. Cette transformation peut être soit un raffinement ou une expansion. Le raffinement de requête réduit la requête de telle sorte que l'information inutile soit éliminée, tandis que l'expansion de requête rajoute de nouvelles informations à la requête initiale pour la rendre moins ambiguë et élargir son champ de recherche.
L'information sociale peut ainsi être utilisée pour étendre les requêtes. [*Koolen et al*,2015] et ses collègues proposent une approche d'expansion de requêtes utilisant Wikipédia comme collection externe. Ils appliquent ensuite cette approche dans la recherche de livres. D'autres pistes concernant le **"Pseudo-Relevance Feedback"** à partir de Wikipédia ont été explorées, notamment par l'approche de [*Li et al*,2007] qui traitent les requêtes dites "faibles". Ces requêtes ne permettent pas de retourner suffisamment de documents pertinents lors de la première recherche. Cette approche a montré une amélioration de qualité, en particulier sur les premiers documents renvoyés. En outre, [*Bao et al*,2007] ont trouvé que le *social-bookmarking* peut améliorer les recherches sur le web selon deux aspects : 1) les annotations représentent généralement de bons résumés pour les pages web correspondant ; 2) le nombre d'annotations indique la popularité des pages web. Ainsi, deux nouveaux

algorithmes sont proposés pour intégrer les facteurs ci-dessus dans le classement de la page :

1) *SocialSimRank* (SSR) et 2) *SocialPageRank* (SPR) :

Le *SocialSimRank* est un algorithme itératif pour évaluer quantitativement la similarité entre les annotations sociales et les requêtes. Par conséquent, une matrice $(N_A \times N_A\ ;\ N_A : annotations)$ est créée, elle stocke toutes les pondérations de similarité $S_A = (a_i, a_j)$ entre chaque paire d'annotations, le *SocialSimRank* est ensuite utilisé comme une forme d'expansion de requête, où des tags similaires sont inclus dans le calcul de similarité entre la requête et le document. D'autres travaux d'expansion de requête exploitent les tags dans l'estimation de la similarité entre la requête et le document

Le *SocialPageRank* est utiliser pour évaluer la popularité des pages web basé sur l'amélioration mutuelle entre trois ensembles distincts d'objets : (i) les pages web populaires, (ii) les utilisateurs web et (iii) les signaux sociaux récents

3.1.3- Reclassement de résultats :

En RI, le classement des résultats consiste à définir une fonction d'ordonnancement qui permet de quantifier les similarités entre les documents et les requêtes. Nous distinguons deux classes pour le classement des résultats qui diffèrent dans la manière dont elles utilisent l'information sociale. La première classe utilise l'information sociale en intégrant une pertinence sociale au processus de classement, tandis que la seconde utilise l'information sociale pour personnaliser les résultats de recherche

3.1.3.1- Classement basé sur la pertinence sociale :

Plusieurs approches ont été proposées pour améliorer le classement des documents retournés vis-à-vis d'une requête en se basant sur la pertinence sociale. La pertinence sociale se réfère à des facteurs sociaux qui caractérisent un document en termes d'intérêt social, sa popularité, sa réputation, etc. En plus de l'algorithme *SocialSimRank* présenté par [*Bao et al*,2007], on a proposé le *Social PageRank* qui est un algorithme qui calcule la qualité de la page (popularité) par le nombre d'annotations sociales. Pour chaque composant (page Web, l'annotation et l'utilisateur), un PageRank peut être calculé sur la base des liens entre eux. La popularité (PageRank) d'un utilisateur peut être dérivée de la popularité des annotations et la page Web sur laquelle l'utilisateur a effectué une annotation, de même pour la page Web et l'annotation. Ensuite, le PageRank de la page Web est utilisé dans la fonction d'ordonnancement des documents [*Bao et al*,2007]. Le *SBRank* qui indique le nombre d'utilisateurs qui ont marqué une page. Ils utilisent *SBRank* comme une fonction de pertinence dans la recherche Web [*Yanbe et al*,2007] on trouve aussi un algorithme appelé *FolkRank* élaboré par [*Hotho et al*,2006] qui est inspiré de l'algorithme *PageRank* en deux étapes : Premièrement ils transforment l'hypergraphe entre utilisateurs, tags et ressources $\mathbb{F} = (U, T, R, Y)$ en un graphe tripartite non orienté, pondéré et trié noté :

$$\mathbb{G}_{\mathbb{F}} = (V, E)$$

Avec l'ensemble V de nœuds du graphe est constitué par l'union disjointe des ensembles d'étiquettes, utilisateurs et ressources : $V = U \dot{\cup} T \dot{\cup} R$

Et toutes les cooccurrences de balises et d'utilisateurs, d'utilisateurs et de ressources, balises et ressources, deviennent des arêtes pondérées entre les nœuds respectifs :

$$E = \{\{u,t\},\{t,r\},\{u,r\} \mid (u,t,r) \in Y\}$$

Deuxièmement, ils appliquent une version de PageRank citée précédemment *FolkRank* qui prends en compte les poids des arêtes sur ce graphe. Formellement, ils ont réparti le poids comme suit :

$$\vec{w} \leftarrow \alpha\vec{w} + \beta A\vec{w} + \gamma\vec{p}$$

Où A est la version stochastique par lignes de la matrice d'adjacence de $\mathbb{G}_{\mathbb{F}}$, p est une préférence vectrice, $\alpha, \beta, \gamma \in [0,1]$ sont des constantes avec

$$\alpha + \beta + \gamma = 1.$$

[*Badache*.2016] a proposé une approche qui consiste à estimer l'importance sociale d'une ressource en exploitant ses signaux sociaux associés, soit individuellement où chaque signal représente un facteur de pertinence, soit en regroupant ces signaux en fonction du type d'importance , afin de prendre en compte ces facteurs sociaux dans l'évaluation de pertinence, il s'est appuyé sur un **modèle de langue** qui lui permet de combiner l'importance a priori de la ressource et sa pertinence vis-à-vis de la requête. La probabilité qu'une ressource D soit pertinente par rapport à une requête Q est estimée comme suit :

$$RSV(Q,D) = P(D|Q) = P(D) \cdot P(Q|D)$$

avec $P(Q|D)$ la probabilité de la pertinence textuelle en utilisant la méthode de pondération Okapi BM25 et $P(D)$ qui est une probabilité indépendante de la requête Q qui concerne le l'importance sociale de chaque document
Ajoutons à ça d'estimer la probabilité a priori est d'effectuer un simple comptage du nombre d'actions spécifiques effectuées sur une ressource. En supposant que les actions sont indépendantes les unes des autres, la formule générale est la suivante :

$$P_x(D_i) = \prod_{a_i^x \in A} P_x(a_i^x|D_i)$$

3.1.3.2- **Classement social personnalisé** :

Les intérêts des utilisateurs diffèrent et ceci est dû au fait qu'ils ont des profils différents, des domaines d'activités différents et aussi des habitudes de navigation différentes. Par conséquent, dans un système de RI, fournir les mêmes documents classés de la même manière ne convient pas peut être à tous les utilisateurs. Ainsi, une fonction personnalisée pour trier les documents différemment selon chaque utilisateur devrait améliorer les résultats de recherche. Plusieurs approches ont été proposées [*Bender et al*. 2008], [*Noll et Meinel*,2007], [*Wang et Jin*,2010], [*Bouadjenek et al*,2013] pour personnaliser le classement des résultats de recherche en utilisant l'information sociale on se réfère à ces 3 étapes :

- premièrement un classement des documents d par pertinence textuelle vis-à-vis de la requête q
- deuxièmement un processus qui détermine la probabilité que le document d intéresse l'utilisateur en ayant une vue sur ces habitudes de navigation
- on fusionne les deux classements précédents pour aboutir à un classement final

La technique de personnalisation de [*Noll et Meinel*,2007] comprend deux étapes principales, à savoir, (i) la collecte et l'agrégation de données sur les utilisateurs et les documents, et (ii) la personnalisation de la recherche Web sur la base de ces données, celle [*Bouadjenek et al*.2013] proposent de calculer la correspondance entre un document d et une requête q. En utilisant la fonction *SoPRa* élaboré pour le fait :

La fonction *SoPRa basique* se compose de deux parties la première avec (i) un score de correspondance textuelle et (ii) un score de correspondance sociale

$$score(d,q,u) = \beta \times \cos\left(\vec{q}, \overrightarrow{T_d}\right) + (1-\beta) \times sim(\vec{q},\vec{d})$$

Ils ont utilisé pour ce fait le modèle vectoriel (VSM) qui malgré son ancienneté continue à faire ces preuves, et $sim(\vec{q},\vec{d})$ est la similarité textuelle entre la requête q et le document d, $\overrightarrow{T_d}$ est le vecteur qui modélise la représentation sociale du document d

Et deuxièmement et finalement le score de classement d'un document d correspondant potentiellement à la requête q émise par un utilisateur u est calculé comme suit :

$$Rank(d,q,u) = \gamma \times \cos\left(\overrightarrow{p_u}, \overrightarrow{T_d}\right) + (1-\gamma) \times Score(q,d)$$

Ou γ est une valeur (pondération) qui satisfait $0 \leq \gamma \leq 1$, Ajoutant à ceci encore un (iii) troisième paramètre qui est l'intérêt de l'utilisateur pour les autres documents, Ils ont aussi proposé ce qu'on appelle la fonction *SoPRa étendue* qui discrimine les utilisateurs qui annotent des pages Web et qui les considèrent individuellement en tenant compte de la similitude de leurs intérêts avec l'émetteur de la requête q

$$Rank(d,q,u) = \gamma \times \sum_{U_k \in U_d} \cos\left(\overrightarrow{p_{u_k}}, \overrightarrow{p_u}\right) \times \cos\left(\overrightarrow{p_u}, \overrightarrow{T_{u_k,d}}\right)$$

$$+ (1-\gamma) \times \left[\beta \times \sum_{U_k \in U_d} cos\left(\overrightarrow{p_{u_k}}, \overrightarrow{p_u}\right) \times cos\left(\overrightarrow{p_u}, \overrightarrow{T_{u_k,d}}\right) + (1-\beta) \times sim(\vec{q},\vec{d})\right]$$

Cette fonction ajoute un autre paramètre $T_{u_k,d}$ qui est le vecteur qui modélise la représentation sociale du document d basé seulement sur les annotations faites par l'utilisateur u_k , le moteur de recherche *Lucene* a été utiliser pour évaluer cette approche

Les travaux se distinguent par l'information sociale qu'ils considèrent pour représenter le profil de l'utilisateur. Une partie dans ces travaux exploitent les tags donnés par les utilisateurs pour construire son profil, d'autres les documents qu'il mis en favoris ou les relations sociales.

La majorité de ces approches se situent dans le contexte de l'annotation sociale, d'autres critères sociaux, en particulier les signaux sociaux, sont exploités pour améliorer cette tâche. Ces approches sont présentées dans la prochaine section.

4- Evaluation de la RI Sociale :

L'évaluation de la RI se fait principalement à travers des collections de tests, construit dans le cadre de campagne d'évaluation, La RI sociale ne déroge pas de cette règle. Avec la mise en place de la tâche Microblog en 2011 (collection de Tweets mise à disposition des chercheurs en RI Sociale) dans la campagne d'évaluation TREC, ainsi que la tâche *Social Book Search* (SBS) dans la campagne d'évaluation d'INEX. Chaque tâche en RI sociale fait l'objet d'un cadre expérimental relativement particulier. Dans ce qui suit nous allons citer certaines collections standards ainsi que leur tâche correspondante.

4.1- Les tâches sociales de TREC : nous pouvons citer :

4.1.1- **Microblog Track** : est une campagne d'évaluation pour la recherche de microblog organisée chaque année depuis 2011 en collaboration avec l'atelier TREC. Le but de ce Track est de fournir à la communauté de recherche des microblogs un protocole d'évaluation des systèmes de recherche microblog. TREC Microblog comprend une tâche principale ad-hoc, connu comme *real-time ad-hoc search*, et une seconde tâche connue par **Filtering Track** introduite en 2012. Ces deux tâches sont basées sur le corpus des tweets. En 2011, la collection de test Tweets*2011* contenait environ 16 millions de tweets. L'ensemble de données est construit en utilisant l'API publique Twitter Stream qui fournit un échantillon représentatif de 1% du flux des tweets. En fin 2016 la collection contenait 210 millions de tweets,

4.2- La tâche de Social Book Search : Le SBS Lab a trois pistes.

- L'objectif de Suggestion Track est de développer des collections de tests pour évaluer l'efficacité du classement des systèmes de récupération de livres et de recommandation.
- L'objectif d'Interactive Track est de développer des interfaces utilisateur qui assistent les utilisateurs à chaque étape de la recherche lors de tâches de recherche complexes et d'explorer comment les utilisateurs exploitent les métadonnées professionnelles et le contenu généré par les utilisateurs.
- Mining Track se concentre sur la détection et la liaison des titres de livres dans les forums de discussion de livres en ligne, ainsi que sur la recherche de recherches de recherche de livres dans les messages de forum pour la recommandation automatique de livres.

La collection INEX SBS se compose de 2.8 millions de documents. Chaque document décrit un livre d'Amazon, étendu avec des métadonnées sociales de *LibraryThing*. Chaque livre est un fichier XML représenté avec des champs comme ISBN, Title, Review, Summary, Rating and Tag. La collection SBS fournie 208 requêtes ainsi que leurs jugements de pertinence,

Conclusion

La majorité des travaux de recherche cités précédemment sur la RI sociale et surtout la RI dans les micro blogs se sont concentre sur l'exploitation des données générés par les utilisateurs pour améliorer les performances de la RI classique, Dans le prochain chapitre nous allons présenter notre travail et contribution sur l'exploitation des signaux sociaux de twitter et spécialement l'hashtag afin d'améliorer les résultats de recherche, et définir notre approche ainsi que ces étapes et les outils que nous avons utilisé

Partie 3

Approche pour l'exploitation des signaux sociaux de twitter afin d'améliorer la RI

Introduction

Les SRIS exploitent principalement deux types de données :

1- La correspondance textuels ou thématique ou ce qu'on appelle le ***Natural language processing***, c'est la plus exploitée, elle est directement dépendante de la requête, elle concerne toutes les caractéristiques relatives à la distribution des termes de la requête dans le document et dans la collection.
2- Les signaux sociaux issu des réseaux sociaux ou de microblogs, ces signaux concernent des facteurs indépendants de la requête (la popularité), et mesure une sorte de qualité ou d'importance de nature sociale a priori du document.

Il est bien connu aujourd'hui que les informations sociales se crée et se partage sur les réseaux sociaux les chiffres sont si vertigineux. En 2018 les statistiques montrent qu'il y'a prêt de 4,12 milliards d'internautes, soit 54% de la population mondiale (+8% entre juillet 2017 et juillet 2018) dont 3,36 milliards d'entre eux sont inscrits sur les réseaux sociaux, soit 44% de la population mondiale (+11% entre juillet 2017 et juillet 2018). Ces utilisateurs contribuent activement sur les réseaux sociaux, par exemple, sur Facebook chaque 60 secondes environ 50000 publications sont faites et plus de 2.3 millions de j'aime sur différentes ressources, ce qui engendre une masse de données d'environ 410 Go par seconde. En effet, grâce aux outils proposés par le Web 2.0 les utilisateurs interagissent de plus en plus entre eux et/ou avec les ressources. Ces interactions (signaux sociaux), traduites par des j'aime, des *+1*, des partages, des tweets, des commentaires ou des bookmarks associés aux ressources, peuvent être considérées comme une des sources que l'on peut également exploiter pour mesurer l'intérêt à priori de la ressource en termes de popularité et de réputation, indépendamment de la requête.
Dans ce chapitre, nous décrivons notre approche pour l'exploitation des signaux sociaux et spécialement les tweets qui sont des petits messages postés périodiquement par les utilisateurs (micro bloggeurs) sur le réseau social Twitter

1- Approche proposée :

Notre approche se base sur le reclassement (Re-Ranking) des documents retournés lors de la phase classique de la phase classique de recherche en exploitant les signaux sociaux de Twitter (tweets, hashtags, retweets, likes, commentaires…) comme facteurs de pertinence, **figure 3.1**

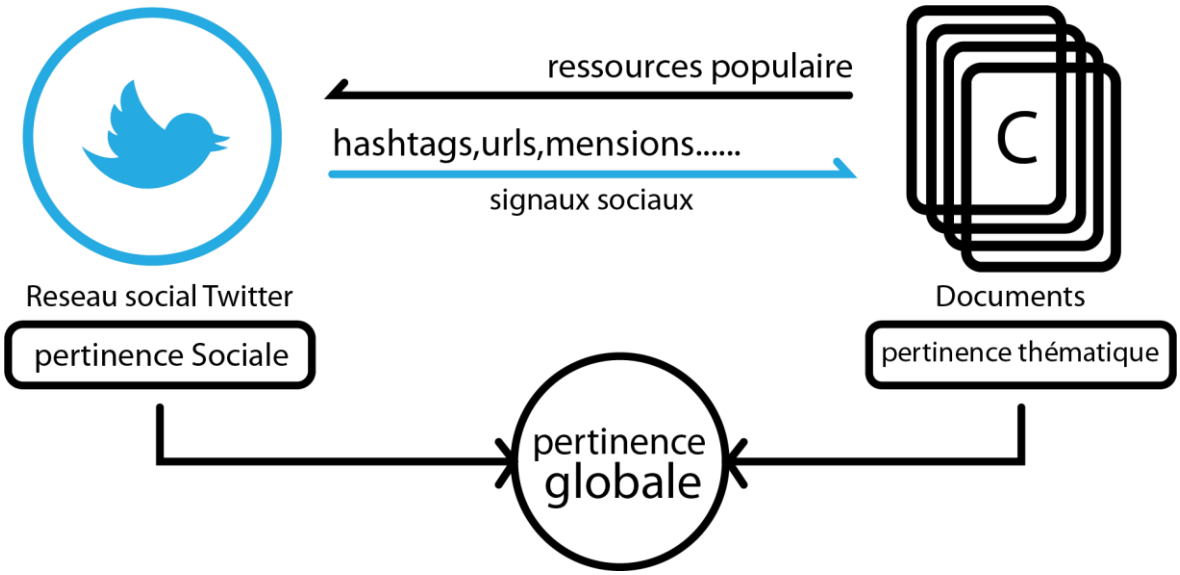


Figure 3.1 : modèle en U de la RIS

Le modèle en U que nous proposons, **figure 3.2**, classe les résultats en deux phases : la **première** se rapporte à la RI classique et se déroule comme suit :

- L'indexation du corpus de documents et la génération d'un index inversé
- Analyse et indexation de la requête saisie par l'utilisateur,
- Application du modèle vectoriel (cosinus) pour l'appariement Requête - documents
- **Affichage des résultats de la recherches sous formes de liste de documents par ordre descendant de pertinence textuelle (score thématique des documents)**

La **deuxième** phase concerne la RI Sociale en utilisant une collection de tweets, et se déroule comme suit :

- Extraction des hashtags des tweets ainsi que le nombre des signaux sociaux les contenant
- Calcul de la popularité de chaque hashtag en se basant sur les tweets le contenant puis l'élaboration d'un classement de hashtags
- Comparaison des termes d'index inversé issu des documents sélectionnés par la première phase (RI Classique) avec la liste des hashtags
- **Classement des documents issu de la phase thématique par ordre de popularité en se basant sur les scores des hashtags qui les cite (score sociale des documents)**

Au final nous allons joindre les deux resultats des deux phases (RI classique et RIS) pour élaborer un classement final

Notre recherche porte principalement sur l'exploitation du grand potentiel de l'hashtag dans la détection des opinions et des tendances, car c'est un moyen très fiable qui permet une certaine neutralité (ne prend pas en compte le statut social de l'utilisateur) et de savoir avec un grand degré d'exactitude les sujets discutés de Twitter.

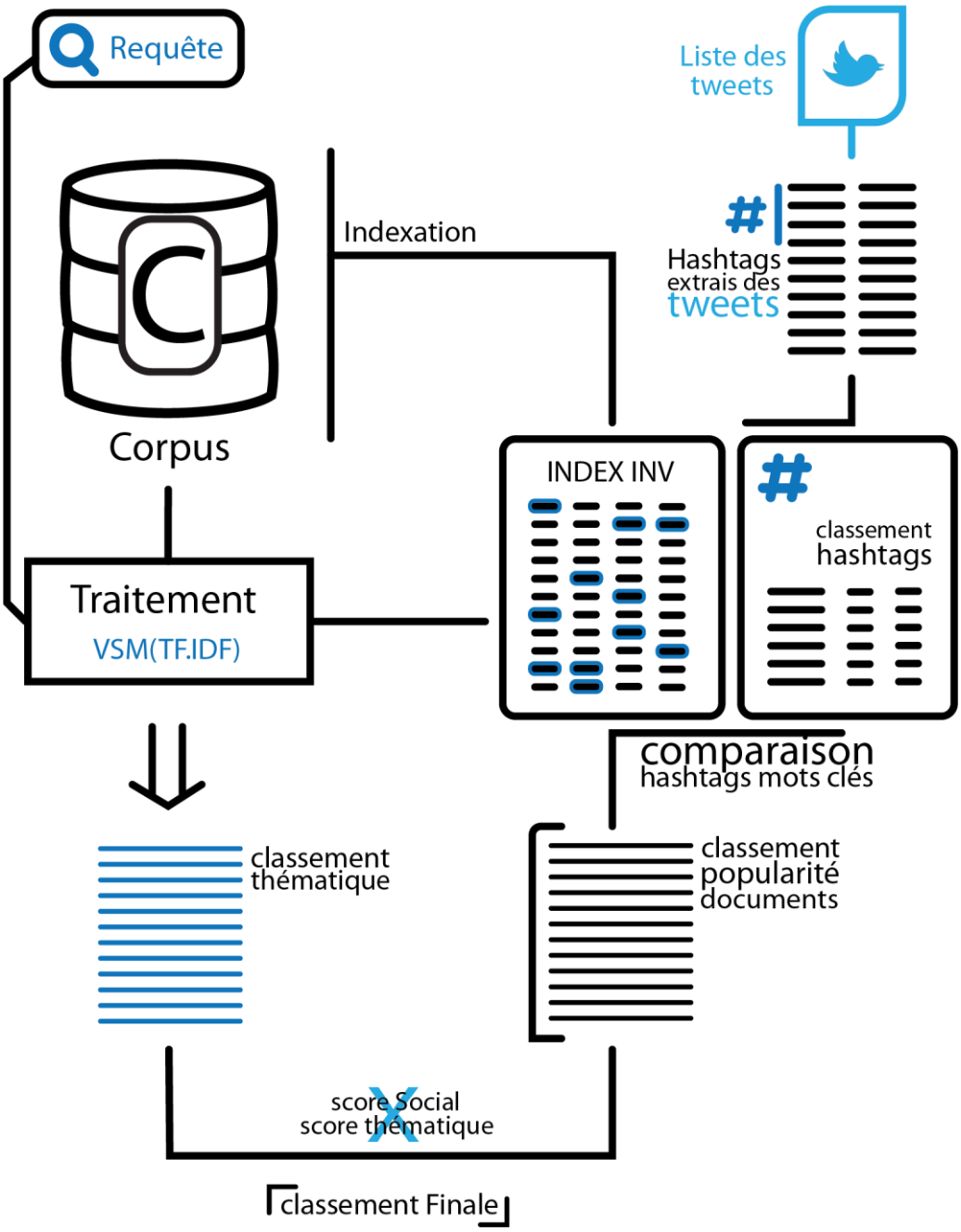


Figure 3.2 : Architecture de l'approche proposée

1.1- Annotations :

L'information sociale que nous exploitons dans notre approche peut être représentée par le quintuplé $< U, H, T, A_t, R >$ ou U, H, T, A_t, R représentent respectivement : Utilisateurs, Hashtags, Tweets, Actions sociale, Ressources

1.1.1- Utilisateurs :

Cet ensemble $U = \{U_1, U_2, \dots, U_n\}$ représente les internautes interagissant entre eux dans le réseau sociale Twitter par le biais de tweets et autres signaux sociaux (hashtag, retweet, like, commentaire), dans notre recherche nous ne nous intéressons pas à cet ensemble.

1.1.2- Hashtags :

Dans notre recherche l'Hashtag $H = \{H_1, H_2, \dots, H_n\}$ est le paramètre principal de la popularité d'une Ressource R, le score social dépendra principalement de la popularité des hashtags qui sont affiliés à un document D_i.

1.1.3- Tweets :

Les tweets $T = \{T_1, T_2, \dots, T_n\}$ sont la composante principale du réseau sociale Twitter, nous les utilisons pour déterminer la force d'un hashtag H_i en calculant le taux de popularité des tweets ou il apparait car $T_i = A_{s_i} = \{A_{re}, A_l, A_c\}$ et A_{re}, A_l, A_c sont respectivement les actions sociales réalisables sur un tweet : retweet , like , comment , puis nous avons $A_s = \{A_{s_1}, A_{s_2}, \dots, A_{s_n}\}$ l'ensemble des actions sociale des tweets i et $i \in [1, n]$.

1.1.4- Ressources :

Nous considérons un corpus $R = \{D_1, D_2, \dots, D_n\}$ composé de n documents , chaque document sera compose de m mot clés (après indexation) soit $D_{w_i} = \{w_1, w_2, \dots, w_m\}$ avec $w_i =$ poids d'un terme , et sera affilié par k hashtags qui détermineront le score social (popularité) du document D_i avec $D_{S_i} = \{H_1, H_2, \dots, H_k\}$,

1.2- Préliminaires :

Dans un premier temps nous devons indexer notre collection de document R et déterminer la pertinence thématique de ces ressources vis-à-vis d'une requête Q pour cela nous avons choisi le modèle Vectoriel (VSM) qui est le modèle le plus utilisé en RI, la corrélation entre le vecteur ressource D et le vecteur requête Q est estimée selon la **mesure cosinus, en** prenant en compte leurs 2 vecteurs :

- le vecteur document $D_i = (w_{1i}, w_{2i}, \dots, w_{ni})$
- le vecteur requête $Q = (w_{1q}, w_{2q}, \dots, w_{nq})$

Ou w_{ji} et w_{jq} sont respectivement le poids d'un terme dans un document et dans une requête, ce poids est calculé avec la formule **TF.IDF** (section 2.2.1.4 du 1er Chapitre), tel que **TF** représente la fréquence du terme dans le document

chaque terme des documents et de la requête est affilie par un nombre w qui désigne ca pondération , qui se fera avec la pondération **TF.IDF**, ceci consiste à affecter à chaque terme d'index (comme expliquer dans le chapitre 1) d'un document D_j un poids $w_{i,j}$ par rapport au document qui le contient **TF** et la collection qui le contient **IDF**, la requête Q aussi subit les mêmes traitements que ceux du document (le requêtage) , La fonction est donnée comme suit :

$$w_{i,j} = \begin{cases} \left(1 + \log f_{i,j}\right) \times \log \dfrac{N}{n_i} & si\ f_{i,j} > 0 \\ sinon\ 0 \end{cases}$$

En utilisant la fonction précédente (TFIDF) on va calculer la mesure **Cosinus** qui nous fournira un $score_{thématique}$ qui est contenus dans l'intervalle $[0; 1]$ et qui calcule l'appariement de deux vecteurs D_i, Q

$$score_{thématique} = sim(D_J, Q) = \frac{\sum_{i=1}^{t} w_{i,j} \times w_{i,q}}{\sqrt{\sum_{i=1}^{t} w_{i,j}^2} \times \sqrt{\sum_{i=1}^{t} w_{i,q}^2}}$$

A l'issu de l'appariement, les résultats de la recherche répondant a la requête seront sont retournés

1.3- Traitement des données sociales : Reclassement de résultats (Re-Ranking)

Notre principale contribution consiste en la proposition d'une fonction de calcul du score sociale d'un document, basé sur le score des hashtags qui le référencent, et en tenant compte des signaux sociaux des tweets ou les hashtags apparaissent

1.3.1- Fonction de calcul du score sociale :

Sur le réseau social Twitter, Les utilisateurs postent des tweets dans le but de créer des sujets de discussion et ceci peut être suivi avec l'ajout d'une mention (qui est un appel pour une autre personne, ou organisation), d'un lien ou le plus important de tous l'ajout d'un **Hashtag**. Chaque utilisateur peut interférer avec les tweets publiés par ces paires soit avec un retweet (équivalent de partage sur Facebook), like ou comment
Dans notre travail nous avons assigné à chaque tweet T_i un score basé sur les signaux A_S avec $S =$ Nbr de retweet / like / commentaire, nous avons donc

$$T_i = \{A_{re}, A_l, A_c\} \qquad 1.1$$

En tenant compte de ces paramètres nous proposons de calculer le score d'un tweet comme suit :

$$S_T = \gamma N_{re} + \delta N_l + \varepsilon N_c \qquad 1.2$$

Avec $\gamma + \delta + \varepsilon = 1$ et $\{\gamma; \delta; \varepsilon\}$ sont les pondérations respectivement des paramètres $\{A_{re}, A_l, A_c\}$, pour raison d'importance accrus des retweets nous avons pondéré leur nombre de manière supérieur que les deux autres valeurs (nombre de likes et de commentaires) : $\gamma > \{\delta; \varepsilon\}$

Nous calculons par la suite le score de l'Hashtag H_i ,qui est basé sur les scores des tweets qui le contiennent $T_{Hi} = \{T_{1,i}; T_{2,i}; \dots; T_{n,i}\}$, de la manière qui suit :

$$S_{H_i} = \sum_{j=1}^{n} S_{T\,j,i} \qquad 1.3$$

Après avoir calculé les scores de tous les hashtags nous les classons sous forme de tendances ou ce qu'on appelle en anglais « *The Trends* ». Nous comparons par la suite les hashtags syntaxiquement avec les termes hautement pondérés issu de **l'index inversé** des 25 documents de la phase 1, s'il y a correspondance le score de l'hashtag S_{H_i} sera immédiatement incrémenter au score social du document D_i dont est issu le terme, comme suit :

$$S_{D\,i} = (S_{H\,1,i} + S_{H\,2,i} + \cdots + S_{H\,n,i}) \qquad 1.4$$

Ensuite nous calculons le score social d'un document comme suit :

$$S_{sociale\ Di} = (1 - e^{-\lambda \log(S_{D\,i})}) \qquad 1.4$$

Nous ajoutons la fonction de normalisation : $Y = (1 - e^{-\lambda \ln(X)})$
Pour faire tendre le score social S_{Di} vers **1** ainsi :

*Plus le score social d'un document S_{D_i} (1.3) augmente plus il tendra vers la valeur maximal **1***

Nous ajoutons le logarithme népérien **ln** pour alléger le score du document initial S_{D_i} Enfin nous ajoutons une corrélation λ de type $\frac{1}{n}$ avec $n \in \mathbb{N}^+$ pour garantir un début lent pour la fonction (1.4) et pour que les hashtags impopulaires aient un faible effet sur le reclassement des résultats.

1.3.2- Calcul du score globale (sociale et thématique) :

Dans cette étape nous allons calculer le score global en combinant le score thématique (dans la première phase) avec le score social (deuxième phase) :

$$S_{thématique_{Di}} = RSV(d_i, q) = \frac{\sum_{j=1}^{t} w_{j,i} \times w_{j,q}}{\sqrt{\sum_{j=1}^{t} w_{j,i}^2} \times \sqrt{\sum_{j=1}^{t} w_{j,q}^2}}$$

$$S_{sociale_{Di}} = (1 - e^{-\lambda \ln(S_{D_i})})$$

Pour ce nous utilisons une fonction de type :

$$S_{globale_{Di}} = \alpha S_{thématique_{Di}} + (1 - \alpha) S_{sociale_{Di}} \quad 1.5$$

Dans notre recherche nous avons favorisé la pondération du score thématique, car la majorité des recherches dans le domaine clament que la RI sociale n'est qu'une manière d'améliorer les résultats de la RI classique
[*Hacid* et *Boudjenek*,2013] avec leur approche basée sur le reclassement de résultats personnalisé estime que leurs meilleurs résultats ont été obtenu avec $0.6 \leq \alpha \leq 0.8$;

2. Expérimentation de notre approche :

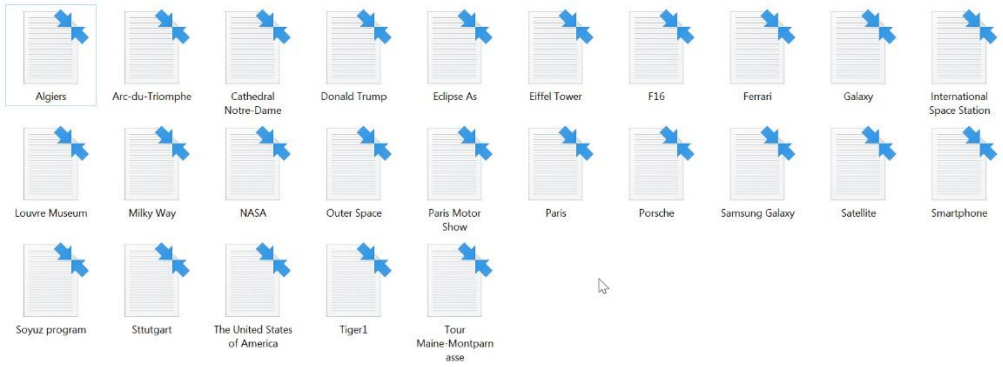
Afin d'améliorer le Processus de la RI classique (thématique) nous allons utiliser le contenu social qui est les tweets et leurs signaux sociaux pour générer les scores sociaux dans le but de reclasser des résultats initiaux en rajoutant un autre critère qui est la popularité du document D_i ou ce qu'on appelle un reclassement de résultats (Re-Ranking). Nous avons développé des programmes écrit en ***Java*** sous l'IDE *Eclipse* et exploités le moteur de recherche ***Lucene*** :

- **Le langage Java :** c'est un langage de programmation orienté objet créé par *James Gosling* et *Patrick Naughton* (de Sun Microsystems) en 1995, la particularité de ce langage est ca haute portabilité sur plusieurs systèmes d'exploitation

- **Lucene :** *Lucene* est une bibliothèque open source écrite en java développé par *Apache Fondation* qui permet d'indexer et de chercher du texte, et dispose d'une grande panoplie d'outils de personnalisation et est très populaire dans le domaine de la RI

Nous avons utilisé Pour ceci :

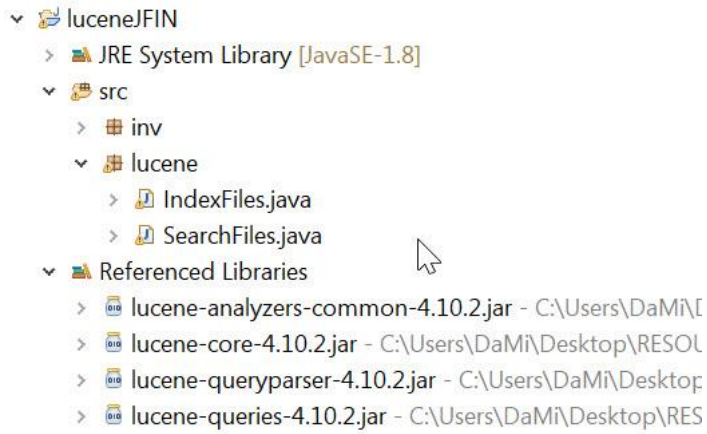
1- Une collection de 25 documents textes (.txt) écrit en anglais



2- Un exemple Excel contenant des tweets et le nombre de leurs signaux sociaux ainsi que d'autres paramètres

	A	B	E	F	K	L	M
1	Tweet Id	Text	Favorites	Retweets	comment	score_t	score_tweet
2	1049844973707902976	We're working with other government agencies, academia a	1354	294	677	503.3	6.22118641
3	1049859845329960960	There it goes! 13 photos assembled in sequence show the @Space_Station pass in front of our view of the Sun while orbiting Earth at a speed of roughly 5 miles per second on Sun, Oct 7. See more: https://t.co/o6Rmsx6JZT Learn when you can #SpotTheStation: https://t.co/IV6AZcoGh3 https://t.co/4KdzwfoMBj	9622	2200	4811	3587.7	8.18526661
4	1049862153438683136	As a @NASA astronaut, #USAF Col. Nick Hague is about to se	0	544	0	54.4	3.99636415
5	1049863565153316869	Two crew members are launching to the station Thursday. @	0	396	0	39.6	3.67882912
6	1050013804405420033	What began as a tropical disturbance in the Caribbean Sea w	1314	440	657	503.9	6.22237784
7	1050033497056968704	From space, satellites from @NASAEarth + @NOAA see #Hur	2142	785	1071	828.2	6.71925467
8	1050067290354581505	LIVE NOW: Cameras outside the International @Space_Stati	2752	1110	1376	1074.2	6.97933148
9	1050074122502463489	A view of the eye of #HurricaneMichael from @AstroSerena,	6132	2063	3066	2352.5	7.76323387
10	1050100276932341767	Just arrived in Baikonur, Kazakhstan! Tomorrow American as	0	171	0	17.1	2.83907846
11	1050112000704073728	Imagining our bold missions to the Moon and Mars just got a	1626	406	813	609.7	6.41296703
12	1050125828539990026	.@NASAEarth + @NOAA satellites examined #HurricaneMich	1192	451	596	462.3	6.13621403
13	1050148474065436672	Around the Earth 100,000 times! 🌎On Saturday, our Terra	2292	380	1146	840.2	6.73363996
14	1050164325720182784	#HurricaneMichael plowed into the Florida panhandle Wedr	692	232	346	265.4	5.58123812
15	1050183453612883969	The path through the solar system is a rocky road. Asteroids	1668	423	834	626.1	6.4395101

3- Le moteur de recherche *Lucene* dans sa version 4.10.2 :
pour indexer les documents de notre collection et ainsi extraire l'index inverse et la pondération des termes , Ensuite pour la recherche thématique via une requête saisi au préalable



4- Le programme *Extract_hashtag* que nous avons utilisé pour :
   - **extraire les hashtags des tweets ainsi que leurs scores**
   - **classer les hashtags selon leurs popularités**
   - **déceler les documents populaires parmi ceux retournés lors de la phase de recherche classique en comparant leurs termes à haute pondération avec les hashtags**
   - **calculer du score globale (score thématique + score sociale)**
5- Application de mesures d'évaluation sur les résultats

2.1- Etape 1 : les Ressources

Pour notre test nous avons utilisé :

- Premièrement nous auront besoin de l'environnement de programmation Eclipse IDE afin d'écrire notre code en ***Java***, et aussi la bibliothèque ou 'Library' Apache POI pour pouvoir lire et écrire des fichiers Microsoft (Excel dans notre cas)

- Un index inversé d'une collection de données de 25 documents, cet index contient les termes extrait après indexation avec le moteur de recherche *Lucene* des documents avec le poids de chacun de ces mots en prenant en considération son importance dans la collection et les documents ou il apparait

2.2- Etape 2 : exécution du programme Extract_hashtag

Premièrement nous avons extrait les hashtags pour ce nous avons utilisé une fonction qui parcours tous les tweets du fichier *tweets.xls*

```
int d=0;
String chaine;
for (int row=1;row<noOfRows;row++) {

          chaine = excelData[row][c];

          int i=0;
          String y=chaine;

          while (i<chaine.length()) {

            int pos = y.indexOf('#');
            if (pos !=-1) {

             String chaine3 = y.substring(pos);
             int pos2 = chaine3.indexOf(' ');
             String chaine4 = y.substring(pos2+pos+1);
             String hash=chaine3.substring(0, pos2);

             int x=pos2+pos;
             y=chaine4;
             i=x+1;

             l_chaine2.add(hash);
             score[d]=excelDatat[row][11];
             d++;

             setOrAdd_hash(hash,l_chaine);

             }else i =chaine.length();
          }
}
```

Figure 3.3 : méthode d'extraction de hashtags

Cette fonction utilise indexOf ('#') pour détecter les hashtags et ainsi les extraire des tweets et les insérer dans une liste qui sera principalement utilisée pour le fichier Excel *hashtags.xls* pour appariement des hashtags par ordre de popularité (score de l'hashtag S_{H_i})

hashtags	occurance	s_hash
#Paris	110	4138002
#hirak	111	2728846.8
#hirak_Algerie	10	340106.1
#AskNASA	5	38664
#NationalAstronomyDay,	1	32457.9
#hirak#hirak	1	24170.1
#dept	1	24170.1
#TourEiffelParis	16	10185.6
#HurricaneMichael	8	8326.2
#TourEiffel	8	6139.2
#Apollo50	5	5038.2
#Soyouz	1	1111.8
#FirstManMovie	1	1104
#APOLLO	2	1062
#AboveAndBeyond	2	1021.8
#Moon	2	858.6
#TourEiffel#TourEiffelParis#TourEif	1	767.4
#IceBridge	1	713.4
#ICESat2	1	713.4
#OppyPhoneHome.	1	695.4
#APOLLO_NASA_NABIL	1	538.8
#NASA60th	1	489.6
#APOLLONABILroute	1	352.2
#USAF	1	326.4

hash_trends

Figure 3.4 : classement des hashtags H_i (les trends)

Le score S_{H_i} de chaque hashtag est calculé par la fonction **figure 3.5** en utilisant la liste (fournit par le code **figure 3.3**) contenant tous les hashtags des tweets :

```
for (int i=0;i<l_chaine.size();i++) {
  for(int j=0;j<l_chaine2.size();j++) {
    if (l_chaine2.get(j).equals(l_chaine.get(i))) {

        occu[i]++;
        scoref[i]=scoref[i]+score[j];
      }
  }
}
```

Figure 3.5 : calcul des scores S_{H_i}

Après le calcul des scores des hashtags, nous évaluons les similarités entre ***hashtags*** et ***terme*** à haute pondération de notre index inversé avec la méthode Contains () :

```
for (int i=0;i<index.size();i++) {

    int pos = l_chaine.get(j).indexOf('#');
        String chaine1 = l_chaine.get(j).substring(pos+1);


        if (poid_List.get(i) >= 3 & chaine1.toLowerCase().contains(index.get(i).toLowerCase())) {
```

```
for (int k=0;k<6;k++) {
    for (int t=0;t<tab_Docs.length;t++) {

        int docNbr = tab_Docs[t];
        if ( tab[i][k] == docNbr  ) {
```

Figure 3.6 : similarité hashtag/termes d'index

Dans cette phase (similarité hashtagXtermesIndex), nous allons comparer chaque hashtag à tous les termes d'index inversé pointant les documents obtenus lors du premier classement thématique pour essayer de trouver une relation entre les sujets populaires de *Twitter* et ces documents,

Attribution de la valeur de λ : la valeur lambda est donnée dans le but d'assurer une ascension lente à notre fonction de score sociale ainsi un hashtag moins populaire aura une valeur qui tendra vers 0 et pour contraire (un hashtag populaire) vers 1, pour avoir des résultats optimaux nous avons testé des plages de 100 valeurs comprise dans $[0; 1]$ voici quelques aperçus de valeurs des scores sociaux $S_{sociale\ Di}$

document_	score	l=0.01	l=0.02	l=0.03	l=0.04	l=0.05	l=0.06	l=0.07	l=0.08	l=0.09	l=0.10	l=0.11	l=0.12	l=0.13	l=0.14	l=0.15	l=0.16	l=0.17
1	5038.2	0.081716	0.156754	0.22566	0.288936	0.347041	0.400397	0.449394	0.494387	0.535704	0.573644	0.608484	0.640477	0.669855	0.696833	0.721607	0.744356	0.765246
3	7248228	0.146118	0.270886	0.377423	0.468393	0.54607	0.612398	0.669034	0.717394	0.758688	0.793948	0.824056	0.849765	0.871717	0.890461	0.906467	0.920134	0.931804
5	5038.2	0.081716	0.156754	0.22566	0.288936	0.347041	0.400397	0.449394	0.494387	0.535704	0.573644	0.608484	0.640477	0.669855	0.696833	0.721607	0.744356	0.765246
8	4148955	0.141341	0.262705	0.366915	0.456396	0.53323	0.599204	0.655853	0.704495	0.746262	0.782126	0.81292	0.839362	0.862067	0.881563	0.898303	0.912677	0.925019
9	3111327	0.138866	0.258449	0.361425	0.450102	0.526464	0.592222	0.648849	0.697612	0.739604	0.775764	0.806903	0.833718	0.856809	0.876693	0.893816	0.908562	0.921259
12	3116365	0.13888	0.258473	0.361456	0.450138	0.526503	0.592262	0.648889	0.697651	0.739642	0.7758	0.806937	0.83375	0.856839	0.876721	0.893842	0.908585	0.921281
14	3111327	0.138866	0.258449	0.361425	0.450102	0.526464	0.592222	0.648849	0.697612	0.739604	0.775764	0.806903	0.833718	0.856809	0.876693	0.893816	0.908562	0.921259
15	8242.8	0.086225	0.165015	0.237012	0.302501	0.362917	0.417849	0.468045	0.513913	0.555826	0.594125	0.629121	0.6611	0.690322	0.717024	0.741424	0.763719	0.784093
22	7131	0.0849	0.162592	0.233688	0.298748	0.358285	0.412766	0.462623	0.508246	0.549996	0.588201	0.623163	0.655157	0.684434	0.711225	0.735742	0.758178	0.778709
24	7131	0.0849	0.162592	0.233688	0.298748	0.358285	0.412766	0.462623	0.508246	0.549996	0.588201	0.623163	0.655157	0.684434	0.711225	0.735742	0.758178	0.778709
34	4166047	0.141377	0.262766	0.366993	0.456486	0.533326	0.599303	0.655952	0.704592	0.746356	0.782215	0.813005	0.839442	0.862141	0.881631	0.898365	0.912734	0.925072
45	7131	0.0849	0.162592	0.233688	0.298748	0.358285	0.412766	0.462623	0.508246	0.549996	0.588201	0.623163	0.655157	0.684434	0.711225	0.735742	0.758178	0.778709
95	3093123	0.138816	0.258362	0.361313	0.449973	0.526325	0.592079	0.648705	0.69747	0.739466	0.775632	0.806778	0.8336	0.856699	0.876592	0.893723	0.908476	0.921181
	ECART	0.061218	0.108294	0.143735	0.169645	0.187786	0.199632	0.206411	0.209148	0.208692	0.205747	0.200893	0.194608	0.187283	0.179236	0.170724	0.161956	0.153095

La colonne *score* représente le score de l'hashtag H_i qui nous donne un aperçu de la popularité de l'hashtag et les autres colonnes c'est pour le score social avec des valeurs multiple de λ

l=0.82	l=0.83	l=0.84	l=0.85	l=0.86	l=0.87	l=0.88	l=0.89	l=0.90	l=0.91	l=0.92	l=0.93	l=0.94	l=0.95	l=0.96	l=0.97	l=0.98	l=0.99	l=1
0.999079	0.999155	0.999224	0.999287	0.999345	0.999399	0.999448	0.999493	0.999534	0.999573	0.999607	0.99964	0.999669	0.999696	0.999721	0.999744	0.999765	0.999784	0.999802
0.999998	0.999998	0.999998	0.999999	0.999999	0.999999	0.999999	0.999999	0.999999	0.999999	1	1	1	1	1	1	1	1	1
0.999079	0.999155	0.999224	0.999287	0.999345	0.999399	0.999448	0.999493	0.999534	0.999573	0.999607	0.99964	0.999669	0.999696	0.999721	0.999744	0.999765	0.999784	0.999802
0.999996	0.999997	0.999997	0.999998	0.999998	0.999998	0.999998	0.999999	0.999999	0.999999	0.999999	0.999999	0.999999	0.999999	1	1	1	1	1
0.999995	0.999996	0.999996	0.999997	0.999997	0.999998	0.999998	0.999998	0.999999	0.999999	0.999999	0.999999	0.999999	0.999999	0.999999	0.999999	1	1	1
0.999995	0.999996	0.999996	0.999997	0.999997	0.999998	0.999998	0.999998	0.999999	0.999999	0.999999	0.999999	0.999999	0.999999	0.999999	0.999999	1	1	1
0.999385	0.999438	0.999487	0.999531	0.999571	0.999608	0.999642	0.999673	0.999701	0.999727	0.99975	0.999772	0.999792	0.99981	0.999826	0.999841	0.999855	0.999867	0.999879
0.999308	0.999366	0.99942	0.999469	0.999514	0.999556	0.999593	0.999628	0.999659	0.999688	0.999715	0.999739	0.999761	0.999781	0.9998	0.999817	0.999833	0.999847	0.99986
0.999308	0.999366	0.99942	0.999469	0.999514	0.999556	0.999593	0.999628	0.999659	0.999688	0.999715	0.999739	0.999761	0.999781	0.9998	0.999817	0.999833	0.999847	0.99986
0.999996	0.999997	0.999997	0.999998	0.999998	0.999998	0.999999	0.999999	0.999999	0.999999	0.999999	0.999999	0.999999	0.999999	1	1	1	1	1
0.999308	0.999366	0.99942	0.999469	0.999514	0.999556	0.999593	0.999628	0.999659	0.999688	0.999715	0.999739	0.999761	0.999781	0.9998	0.999817	0.999833	0.999847	0.99986
0.999995	0.999996	0.999996	0.999997	0.999997	0.999998	0.999998	0.999998	0.999999	0.999999	0.999999	0.999999	0.999999	0.999999	0.999999	0.999999	1	1	1
0.00069	0.000632	0.000578	0.000529	0.000484	0.000443	0.000406	0.000371	0.00034	0.000311	0.000285	0.000261	0.000238	0.000218	0.0002	0.000183	0.000167	0.000153	0.00014

Comme nous pouvons le voir plus λ est proche de 1 plus la valeur des scores sociaux tend elle aussi vers **1**, nous avons pris la valeur $\lambda = 0.08$ qui assure un ecart maximal entre deux hashtags de popularité différentes

Le calcul du calcul du score social : $S_{sociale\ Di} = (1 - e^{-\lambda \ln(S_{Di})})$ se fait avec le programme suivant

```
if (poid_List.get(i) >= 3 & chaine1.toLowerCase().contains(index.get(i).toLowerCase())) {

    for (int k=0;k<6;k++) {
        for (int t=0;t<tab_Docs.length;t++) {

            int docNbr = tab_Docs[t];
            if ( tab[i][k] == docNbr  ) {

                Row row2=sheet2.createRow(t);
                Cell cellSFN = row2.createCell(5);
                Cell cellLNSD = row2.createCell(3);
                Cell cellSD = row2.createCell(1);
                Cell cellD = row2.createCell(0);
                sd[docNbr] = sd[docNbr] + score_Hash[j];

                cellSD.setCellValue(sd[docNbr]);

                double pl = 0.08 ; // PONDERATION pour ignorer les hashtags impopulaire

                double ln = Math.log(sd[docNbr]);
                       //ln = Math.log(Math.pow(sd[docNbr],pl));
                double //Socfin = 1- Math.exp(-Math.pow(ln,pl));
                       Socfin = 1- Math.exp(-(pl*ln));
                DocS[t]=docNbr;
                Score_DocS[t]=Socfin;

                cellLNSD.setCellValue(ln);;
                cellD.setCellValue(docNbr);
                cellSFN.setCellValue(Socfin);

                //System.out.println(t+" , "+sd[docNbr]);

            }
```

Figure 3.7 : calcul du score social $S_{sociale\ Di}$

Le fragment de code montré en figure 3.7 remplit la colonne *Score_fin* (formule 1.4) du fichier *ScoreD.xls* (**figure 3.8**) qui affiche le score social d'un document S_{D_i} (*sco_soc*)

documents	NumDoc	Sco_Soc	score_LN	score_FIN
Algiers	1			
Arc-du-Triomphe	2			
Cathedral Notre-Dame	3			
Donald Trump	4			
Eclipse AS	5			
Eiffel Tower	6	17092.2	9.746377	0.541461
F-16	7			
Ferrari	8			
Galaxy	9			
International Space Station	10	41526	10.63408	0.572895
Louvre Museum	11			
Milky Way	12			
NASA	13	40274.4	10.60347	0.571848
Outer Space	14			
Paris Motor Show	15			
Paris	16	17092.2	9.746377	0.541461
Porsche	17			
Samsung Galaxy	18			
Satellite	19			
Smartphone	20			
Soyuz Program	21	1251.6	7.132178	0.434799
Sttutgart	22			
The USA	23			
Tigerl	24			
Tour Maine-Montparnasse	25			

Figure 3.8 : fichier ScoreD.xls

Au final le calcul du score global qui est définit comme suit :

$$Score\ globale = \alpha.Score\ thematique + (1-\alpha).Score\ sociale$$

Là aussi nous devons choisir une valeur optimale pour α qui pondèrera le score textuel par rapport au score sociale, rappelons que la plage de valeurs choisit est [0,6; 0,8]

```
System.out.println("notre docs");
for (int i=1;i<DocS.length;i++) {

        for (int j=1;j<noOfRows2;j++) {
            Row rowH3=sheet3.getRow(j);
            Cell cellSS= rowH3.createCell(2);
            Cell cellSG= rowH3.createCell(3);

            cellSS.setCellValue(Score_DocS[excelDataD[j][0].intValue()]);

            if(excelDataD[j][0] == DocS[i]) {

                System.out.println(excelDataD[j][0]);
                cellSS.setCellValue(Score_DocS[excelDataD[j][0].intValue()]);

            }
                double A = 0.66;
                double B = 1-A;
                double scoreGLOBAL = B*Score_DocS[excelDataD[j][0].intValue()] + A*excelDataDST[j][1];
                cellSG.setCellValue(scoreGLOBAL);

    }

}
```

Figure 3.9 : fragment de code qui calcule le score social globale

Sur le fichier *ScoreGlobal.xls* les scores : thématique et sociale sont affichés

N_document	Documents_Cla	score_thema	score_social
Paris	16	0.51063	0.541461
Tour Maine-Montparnasse	25	0.26516	0
Eiffel Tower	6	0.19366	0.541461
Paris Motor Show	15	0.18751	0
Cathedral Notre-Dame	3	0.16896	0
Arc-du-Triomphe	2	0.13939	0
Tigerl	24	0.13796	0
Louvre Museum	11	0.12071	0
Samsung Galaxy	18	0.09956	0
Donald Trump	4	0.08448	0
Eclipse AS	5	0.07965	0
International Space Station	10	0.05973	0.572895
Porsche	17	0.05973	0
Sttutgart	22	0.05973	0
The USA	23	0.05973	0
F-16	7	0.05632	0
Outer Space	14	0.03982	0

Figure 3.10 : fragment de code qui calcule le score social globale

Attribution de la valeur α : comme cité antérieurement la majorité des approches dans le domaine de la RIS pondère le score thématique de manière supérieur par rapport au score social alors nous avons testé la plage **[0.6;0,8]**

document	scoreTH	a=0.60	a=0.61	a=0.62	a=0.63	a=0.64	a=0.65	a=0.66	a=0.67	a=0.68	a=0.69	a=0.70	a=0.71	a=0.72	a=0.73	a=0.74	a=0.75	a=0.76	a=0.77	a=0.78	a=0.79	a=0.80
13	0.7866	0.47196	0.47983	0.48769	0.49556	0.50342	0.51129	0.51916	0.52702	0.53489	0.54275	0.55062	0.55849	0.56635	0.57422	0.58208	0.58995	0.59781	0.60568	0.61355	0.62141	0.62928
6	0.73489	0.44094	0.44829	0.45563	0.46298	0.47033	0.47768	0.48503	0.49238	0.49973	0.50708	0.51443	0.52178	0.52912	0.53647	0.54382	0.55117	0.55852	0.56587	0.57322	0.58057	0.58792
9	0.73055	0.74863	0.74818	0.74773	0.74728	0.74682	0.74637	0.74592	0.74547	0.74502	0.74456	0.74411	0.74366	0.74321	0.74275	0.7423	0.74185	0.7414	0.74095	0.74049	0.74004	0.73959
19	0.72328	0.43397	0.4412	0.44843	0.45567	0.4629	0.47013	0.47736	0.4846	0.49183	0.49906	0.50629	0.51353	0.52076	0.52799	0.53523	0.54246	0.54969	0.55692	0.56416	0.57139	0.57862
2	0.71783	0.4307	0.43788	0.44506	0.45224	0.45941	0.46659	0.47377	0.48095	0.48813	0.49531	0.50248	0.50966	0.51684	0.52402	0.5312	0.53838	0.54555	0.55273	0.55991	0.56709	0.57427
77	0.70655	0.42393	0.43099	0.43806	0.44513	0.45219	0.45926	0.46632	0.47339	0.48045	0.48752	0.49458	0.50165	0.50871	0.51578	0.52285	0.52991	0.53698	0.54404	0.55111	0.55817	0.56524
85	0.70586	0.42351	0.43057	0.43763	0.44469	0.45175	0.45881	0.46586	0.47292	0.47998	0.48704	0.4941	0.50116	0.50822	0.51527	0.52233	0.52939	0.53645	0.54351	0.55057	0.55763	0.56468
34	0.69634	0.73069	0.72983	0.72897	0.72811	0.72726	0.7264	0.72554	0.72468	0.72382	0.72296	0.7221	0.72124	0.72039	0.71953	0.71867	0.71781	0.71695	0.71609	0.71523	0.71437	0.71352
4	0.65736	0.39442	0.40099	0.40756	0.41414	0.42071	0.42728	0.43386	0.44043	0.447	0.45358	0.46015	0.46673	0.4733	0.47987	0.48645	0.49302	0.49959	0.50617	0.51274	0.51931	0.52589
27	0.64784	0.3887	0.39518	0.40166	0.40814	0.41462	0.42109	0.42757	0.43405	0.44053	0.44701	0.45349	0.45996	0.46644	0.47292	0.4794	0.48588	0.49236	0.49883	0.50531	0.51179	0.51827
67	0.58674	0.35205	0.35791	0.36378	0.36965	0.37552	0.38138	0.38725	0.39312	0.39899	0.40485	0.41072	0.41659	0.42246	0.42832	0.43419	0.44006	0.44593	0.45179	0.45766	0.46353	0.46939
91	0.57479	0.34487	0.35062	0.35637	0.36212	0.36787	0.37361	0.37936	0.38511	0.39086	0.3966	0.40235	0.4081	0.41385	0.4196	0.42534	0.43109	0.43684	0.44259	0.44834	0.45408	0.45983
21	0.55795	0.33477	0.34035	0.34593	0.35151	0.35709	0.36267	0.36825	0.37383	0.37941	0.38498	0.39056	0.39614	0.40172	0.4073	0.41288	0.41846	0.42404	0.42962	0.4352	0.44078	0.44636
70	0.51204	0.30722	0.31234	0.31746	0.32258	0.3277	0.33283	0.33795	0.34307	0.34819	0.35331	0.35843	0.36355	0.36867	0.37379	0.37891	0.38403	0.38915	0.39427	0.39939	0.40451	0.40963
59	0.50684	0.3041	0.30917	0.31424	0.31931	0.32438	0.32944	0.33451	0.33958	0.34465	0.34972	0.35479	0.35985	0.36492	0.36999	0.37506	0.38013	0.3852	0.39026	0.39533	0.4004	0.40547
17	0.47894	0.28736	0.29215	0.29694	0.30173	0.30652	0.31131	0.3161	0.32089	0.32568	0.33047	0.33526	0.34004	0.34483	0.34962	0.35441	0.3592	0.36399	0.36878	0.37357	0.37836	0.38315
43	0.45274	0.27164	0.27617	0.2807	0.28523	0.28975	0.29428	0.29881	0.30334	0.30786	0.31239	0.31692	0.32145	0.32597	0.3305	0.33503	0.33955	0.34408	0.34861	0.35314	0.35766	0.36219
78	0.41747	0.25048	0.25466	0.25883	0.26301	0.26718	0.27136	0.27553	0.2797	0.28388	0.28805	0.29223	0.2964	0.30058	0.30475	0.30893	0.3131	0.31728	0.32145	0.32563	0.3298	0.33398
51	0.40066	0.24039	0.2444	0.24841	0.25241	0.25642	0.26043	0.26443	0.26844	0.27245	0.27645	0.28046	0.28447	0.28847	0.29248	0.29649	0.30049	0.3045	0.30851	0.31251	0.31652	0.32053
88	0.36975	0.22185	0.22555	0.22924	0.23294	0.23664	0.24033	0.24403	0.24773	0.25143	0.25512	0.25882	0.26252	0.26622	0.26991	0.27361	0.27731	0.28101	0.2847	0.2884	0.2921	0.2958
69	0.33466	0.20079	0.20414	0.20749	0.21083	0.21418	0.21753	0.22087	0.22422	0.22757	0.23091	0.23426	0.23761	0.24095	0.2443	0.24765	0.25099	0.25434	0.25769	0.26103	0.26438	0.26773
11	0.30565	0.18339	0.18644	0.1895	0.19256	0.19561	0.19867	0.20173	0.20478	0.20784	0.2109	0.21395	0.21701	0.22007	0.22312	0.22618	0.22924	0.23229	0.23535	0.2384	0.24146	0.24452
5	0.26088	0.36598	0.36286	0.37973	0.3766	0.37347	0.37035	0.36722	0.36409	0.36096	0.35784	0.35471	0.35158	0.34845	0.34533	0.3422	0.33907	0.33594	0.33281	0.32969	0.32656	0.32343

Nous avons constaté que la valeur $\alpha = 0.66$ qui nous a fourni les meilleurs résultats
Cette étape est la dernière de notre approche elle fournit le score global d'un document D_i

Apres application de la formule 1.5 nous avons le reclassement de résultats qui suit :

Score_GLOBAL	nouveau classement
0.521112582	Paris
0.311912382	Eiffel Tower
0.234206188	International Space Station
0.1750056	Tour Maine-Montparnasse
0.1237566	Paris Motor Show
0.1115136	Cathedral Notre-Dame
0.0919974	Arc-du-Triomphe
0.0910536	Tigerl
0.0796686	Louvre Museum
0.0657096	Samsung Galaxy
0.0557568	Donald Trump
0.052569	Eclipse AS
0.0394218	Porsche
0.0394218	Sttutgart
0.0394218	The USA
0.0371712	F-16
0.0262812	Outer Space

Figure 3.11 : classement de résultats finale après ajout du score social

Nous remarquons que certains documents liés à certains hashtags populaires ont gagnés des places lors du reclassement par rapport au autres résultats

C O N C L U S I O N

1- Conclusion générale :

Les travaux présentés dans ce mémoire, rentrent dans le cadre de la recherche d'information, plus précisément dans le cadre de la RI sociale. Les techniques traditionnelles de la RI se limitent à l'appariement suivant un modèle entre documents et requête, mais la croissance exponentielle de la quantité des données web et l'émergence des réseaux sociaux fait qu'il est de plus en plus difficile de trouver une information pertinente, La problématique à laquelle nous nous sommes intéressés dans ce mémoire, réside dans l'intégration et l'exploitation des signaux sociaux afin d'améliorer le processus de recherche Nous avons proposé dans ce mémoire une approche qui exploite les signaux sociaux du réseau Twitter ,
Nous avons élaboré une méthode qui extrait les hashtags des tweets , et nous avons proposé une formule qui calcule la popularité d'un hashtag ou son score afin d'affilier cette popularité à un document qui contient des termes à hautes pondération contenu dans les termes des hashtags , ainsi après un reclassement textuelle (basé sur un modèle) nous effectuons un reclassement des résultats en ajoutant un score sociale à chaque document qui présente des similarités de contenus avec le contenu présent sur twitter.

2- Perspectives :

En perspective d'amélioration de notre approche nous avons l'intention de :

- Faire des expérimentations sur notre approche et l'évaluer à l'aide de l'API public de Twitter et de l'intégrer sur un moteur plus performent (Lucene, Solr, Elasticsearch …)
- Inclure l'aspect temporel des tweets afin de favoriser les plus récents et de ne pas les pénaliser, car les Tweets les plus anciens sont susceptible d'avoir plus de signaux sociaux que les nouveaux, et aussi un tweet très ancien ne doit pas être pris en considération car la popularité d'une ressource
- Faire un appariement plus performant entre les hashtags et les termes hautement pondérés

Bibliographie

- [Badache,2016] M I.Badache . Recherche d'information sociale : exploitation des signaux sociaux pour améliorer la recherche d'information. Recherche d'information [cs.IR]. 2016
- [Chelaru,2012] M S. V. Chelaru, C. Orellana-Rodriguez, et I. S. Altingovde. Can social features help learning to rank youtube videos . 2012.
- [Bao et al,2007] M S. Bao et al. Optimizing web search using social annotations. In Proceedings of the 16th international ACM. 2007
- [Bender et al,2008] M M. Bender et al. Exploiting social relations for query expansion and result ranking. In Data engineering workshop, 2008. ICDEW 2008.IEEE,2008.
- [Bishoff et al,2008] Ms K. Bischoff, C. S. Firan, W. Nejdl, et R. Paiu. Can all tags be used for search? ,ACM. 2008
- [Bouadjenek et al,2013] M R.Bouadjenek, H.Hacid et M.Bouzeghoub. SoPRa: A New Social Personalized Ranking Function for Improving Web Search . 2013
- [Dmitriev et al,2006] M P.A. Dmitriev, N. Eiron, M. Fontoura et E. Shekita. Using annotations in enterprise search. ACM, 2006.
- [Luhn,1957] M H.P. Luhn ,A statistical approach to mechanized encoding and searching of literary information . 1957
- [Hansen et Jarvelin,2005] M P. Hansen et K. Jarvelin ,The Information Seeking and Retrieval process at the Swedish Patent and Registration Office Moving from Lab-based to real life work-task environment . 2005
- [Hotho et al,2006] M A. Hotho et al. Information Retrieval in Folksonomies: Search and Ranking. 2006
- [Howe et al,2006] M J. Howe ,The rise of crowdsourcing . 2006
- [Koolen et al,2015] M M .Koolen et al, Overview of the sbs 2015 suggestion track . 2015
- [Li et al,2007] M Y. Li et al, Improving weak ad-hoc queries using wikipedia asexternal corpus. ACM SIGIR, 2007.
- [Kirsch et al,2006] M. S.M. Kirsch, M. Gnasa, A.B. Cremers ,Beyond the web: Retrieval in social information spaces . 2006
- [Teevan et al,2012] Ms J. Teevan et al, Using related users data to enhance web search . 2012
- [Goh et Foo] Dion Goh et Shubert Foo : social information retreival systems 2010
- [Baeza et Ribeiro] Ricardo Baeza-Yates et Berthier Ribeiro-Neto : Modern Information Retreival 2nd edition 2010