République Algérienne Démocratique et Populaire
Ministère de l'enseignement Supérieur et de la Recherche Scientifique
Université Mouloud Mammeri de Tizi Ouzou

Faculté de Génie Electrique et d'Informatique
Département d'Informatique

Mémoire de fin d'études

En vue de l'obtention du diplôme de
Master en Informatique

Spécialité : Ingénierie des systèmes d'information

Présenté par :
AKLI Sylia
RAHAL Sarah

Thème :

Implémentation, évaluation et comparaison de deux approches pour la recherche d'information exploitant les signaux sociaux de Twitter

Proposé et dirigé par : Mme FELLAG Samia

Remerciements

L’achèvement de ce travail n’a pu être mené à bien qu’avec le soutien de plusieurs personnes que nous voudrions, à travers ces quelques lignes, remercier du fond du cœur.

Nous souhaitons tout d’abord adresser notre reconnaissance à la directrice de ce mémoire, Madame ***Samia FELLAG***, pour sa disponibilité, ses judicieux conseils et surtout pour sa grande patience avec nous.

Nous tenons à exprimer toute notre gratitude aux personnes suivantes :

Nabil LARBI, Salim HASSOUNA et ***Yanis CHETOUANI*** pour leur aide, orientation, soutien moral, patience et surtout leur précieux temps qu’ils nous ont accordé.

Nos remerciements s’adressent aussi aux membres du jury qui ont accepté d’évaluer notre travail.

Dédicaces

Je dédie ce modeste travail

A ma mère ♥

Pour son amour, ses encouragements et ses sacrifices.

A mon père ♥

Pour son soutien, son affection et la confiance qu'il m'a accordé.

A tous mes frères et sœurs :

Mokrane, Nabila, Faiza et Hamza.

A tous mes oncles et tantes :

Khali Mokran, Khalti Tati, Oncle Amirouche, Oncle Lekrim, Khali Brahim, Hayat, Mimi, Kahina, Sakina, Karima, Yanis, Samir, Slimane, Sidra, Ritadj et bien sur ma petite Sarah.

A mes chers amis :

Moussa, Sylia, Rafik, Kenza, Taoues et kamilia.

Sarah

Dédicaces

A mes chers parents ♥

Sylia

Résumé

Notre travail se situe dans le contexte de la recherche d'information sociale(RIS), une thématique récente qui prend en compte les informations sociales émises par les utilisateurs dans les réseaux sociaux, nous nous intéressons plus particulièrement à l'exploitation de ces signaux sociaux dans le réseau social "Twitter".

Notre objectif est de faire une comparaison entre deux approches, la première propose une fonction de reclassement (Re-ranking) basée sur la pertinence sociale qui repose sur l'utilisation des métadonnées sociales afin de déterminer le classement de popularité des hashtags.
La deuxième approche s'intéresse à l'indexation des tweets pour étendre la requête initiale.

Mots clés : recherche d'information, recherche d'information sociale, réseaux sociaux, signaux sociaux, Twitter, hashtags, re-ranking.

Abstract

Our work tackles the context of social information retrieval (SRI), a recent theme that aims to take into account information sent by users in social networks, we are more particularly interested in the exploitation of these social signals in the social network Twitter.

Our goal is to make a comparison between two approaches; the first one is based on the use of social metadata in order to determine the popularity ranking of hashtags.
The second approach is interested in indexing Tweets to use them in the expansion of the initial query.

Keywords: Information retrieval, social information retrieval, social networks, social signals, Twitter, hashtags, re-ranking.

Table des matières

Table des matières

Table des matières

Table des matières

Table des figures et tableaux

Table des figures et tableaux

Table des figures et tableaux

Introduction générale

I. Contexte:

La recherche d'information (RI) est un domaine de l'informatique qui fournit les techniques et outils permettant de représenter, stocker, organiser, rechercher et retrouver, dans une masse documentaire existante, les documents contenant l'information qui répond au mieux au besoin informationnel exprimé par l'utilisateur sous forme de requête.

L'apparition du web social[1] et l'explosion des réseaux sociaux ont 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. D'où l'émergence d'une nouvelle branche de recherche d'information que l'on nomme la recherche d'information sociale(RIS), une thématique récente qui a pour objectif de combiner entre une pertinence textuelle classique et une pertinence sociale issue des réactions des utilisateurs sur les ressources du Web.
Le but derrière l'exploitation de ces contenus, particulièrement les signaux sociaux, est d'essayer de bénéficier de ces traces provenants des actions collectives des utilisateurs afin d'améliorer la RI par rapport à un besoin informationnel.

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 le cadre de la recherche d'information sociale, plus précisément, dans l'exploitation des signaux sociaux de *Twitter* pour améliorer la RI.

II. Contribution :

Notre contribution dans le domaine de la RIS, s'intéresse à faire une implémentation, évaluation et comparaison entre deux approches déjà proposées par nos camarades (SEKOUR 2019) et (LARBI 2019)la première se base sur la reformulation de la requête et utilise *Twitter* comme collection externe afin d'étendre la requête initiale. A partir de cette dernière, un ensemble de *Tweets* jugés pertinents sera sélectionné pour étendre la requête originale et répondre aux besoins de l'utilisateur en utilisant la recherche classique dans une autre collection de documents.
La deuxième s'articule autour de l'extraction des hashtags à partir des tweets et le calcul de leur popularités en prenant en compte leurs occurrences et le nombre des signaux sociaux dans les tweets les contenant, ainsi une liste des hashtags classés par ordre de popularité sera établie pour la comparer avec l'index inverse des documents de la collection et par conséquent le document contenant un terme

[1] https://fr.ryte.com/wiki/Web_social

à haute pondération équivalent à un hashtag, tirera profit du score de popularité de ce dernier.

III. Organisation :

Notre travail est organisé selon le plan suivant :

- **Le chapitre 1 :** introduit les concepts de base de la recherche d'information (RI). Nous commençons par quelques définitions, ensuite nous allons présenter le processus en U de la RI en détaillant ses principales étapes. Enfin, nous concluons par quelques modèles de la RI et par les mesures d'évaluation des systèmes de recherche d'information (SRI).

- **Le chapitre 2 :** présente la recherche d'information sociale. Nous décrivons d'abord l'information sociale dans le Web. Ensuite, la notion de la 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 la RI. Enfin, nous présentons les principales collections de tests qui sont utilisées pour évaluer les SRI sociale.

- **Le chapitre 3 :** dans ce chapitre nous détaillons les deux approches sur lesquelles se fera notre comparaison. Nous commencerons par présenter la première approche proposée par Mohamed SEKOUR qui s'intéresse à l'indexation des *Tweets* et à la reformulation de la requête, ensuite viendra la deuxième approche donnée par Nabil LARBI qui se base sur le reclassement des résultats.

- **Le chapitre 4 :** présente notre implémentation, évaluation et comparaison des deux approches qui exploitent les signaux sociaux de *Twitter* au sein du processus de la RI, ainsi que la solution proposée pour améliorer l'approche de reclassement et nous terminons par une conclusion générale.

Chapitre 01

La recherche d'information standard

I. Introduction :

La recherche d'information est née aux Etats-Unis; très peu de temps après l'avènement des premiers ordinateurs, sa naissance remonte aux années 1950, et cela est dû à la masse volumineuse des documents qui sont stockés sous format numérique.

L'objectif fondamental de la RI consiste à mettre en œuvre une masse documentaire existante, les documents contenant l'information qui répond au besoin informationnel exprimé par l'utilisateur sous forme de requête.

Dans ce chapitre, nous allons définir les concepts de base de la RI. En décrivant les étapes d'un processus de RI, puis nous passerons voir les modèles les plus utilisés en RI, pour en terminer avec les mesures utilisées pour évaluer un SRI.

II. Historique de la RI :

- **1940 :** Apparition des SRI, focalisation de la RI sur les applications dans des bibliothèques.
- **1950** : Apparition du modèle booléen et l'élaboration de petites expérimentations sur des petites collections de documents.
- **1960 et 1970** : Apparition du système SMART, développement d'une méthodologie d'évaluation de système et conception de corpus de test(CACM).
- **1970** : Développement du système SMART. Les travaux sur ce système ont été dirigés par G. Salton. Certaines nouvelles techniques ont été implantées et expérimentées pour la première fois dans ce système (par exemple, le modèle vectoriel et la technique de relevance feedback). Du côté de modèle, il y a aussi beaucoup de développements sur le modèle probabiliste.
- **1980** : Développement de l'intelligence artificielle, ainsi on tentait d'intégrer des techniques de l'IA en RI (système expert).
- **1990 et 1995** : L'apparition d'internet, la RI a été modifiée et sa problématique plus élargie.

III. Définition de la RI :

La recherche d'information est un ensemble de méthodes et techniques pour l'acquisition, l'organisation, le stockage, la recherche et ***la sélection d'information pertinente pour un utilisateur.***

IV. Les concepts de base de la RI :

IV .1. Système de recherche d'information (SRI) :

Un système de recherche d'information (SRI) est un système qui permet de retrouver les documents pertinents à une requête d'utilisateur, à partir d'une base de documents volumineuse.
Dans cette définition, il y a trois notions clés: ***documents, requête et pertinence***.

IV.2. Document :

Un document peut être un texte, un morceau de texte, une page Web, une image, une bande vidéo, etc. On appelle document toute unité qui peut constituer une réponse à une requête d'utilisateur.

IV.3. Requête :

Une requête exprime le besoin d'information d'un utilisateur.

IV.4. Pertinence :

La pertinence est une notion centrale en RI. Elle définit le degré de correspondance entre un document et une requête.
Cette correspondance peut être considérée du point de vue utilisateur (on parle alors de pertinence utilisateur), ou du point de vue système (on parle de pertinence système) :

- ***La pertinence système***: c'est l'évaluation par le SRI, de l'adéquation entre des documents et une requête.
- ***La pertinence utilisateur***: c'est l'évaluation par l'utilisateur de la pertinence, vis-à-vis de son besoin en information, des documents retrouvés par le SRI.

V. Processus en U de la RI :

Le processus de la RI qui permet de planifier à partir d'une requête, les documents est appelé "processus en U". Il est décomposé en trois principales étapes : l'indexation, le requêtage et l'appariement.
La figure 1 illustre le processus en U de la RI :

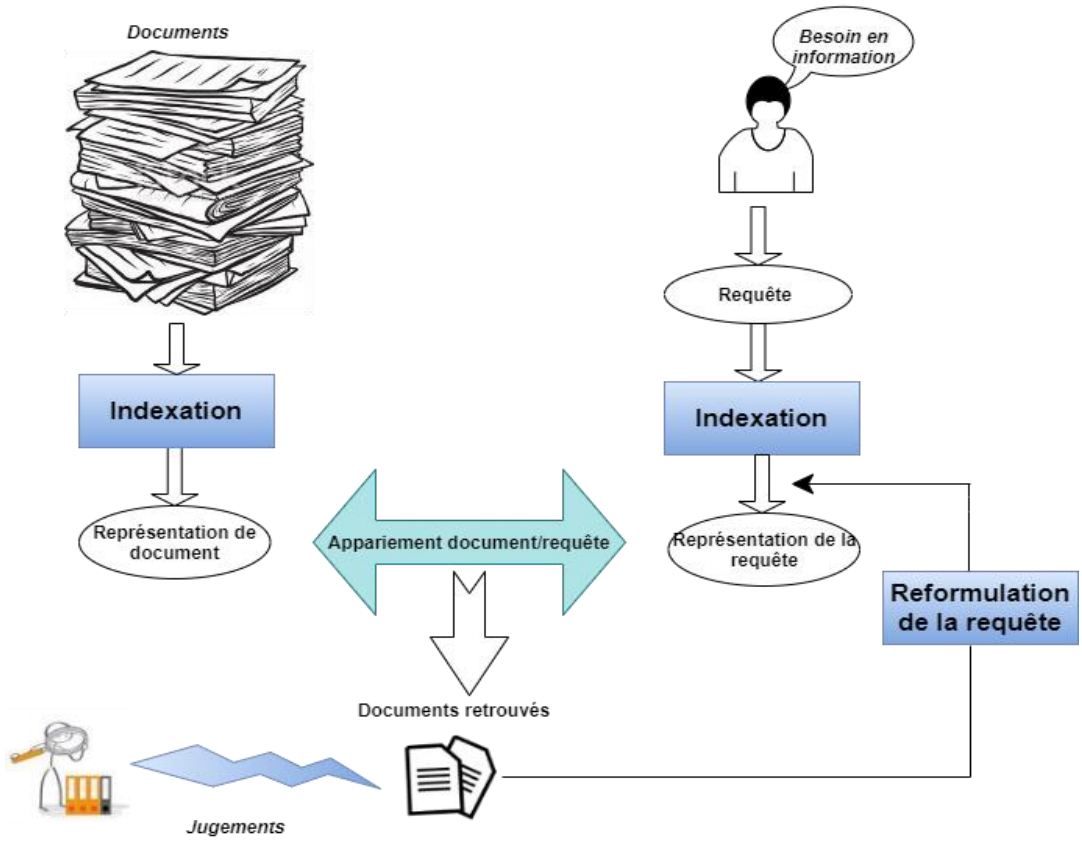

Figure 1:Processus en U de la recherche d'information.

V.1. Indexation :

Ce processus effectue certains prétraitements sur les documents et les requêtes. Cette opération vise à construire une structure d'index qui permet de retrouver très rapidement les documents incluant des mots demandés, d'autre terme elle permet d'associer à chaque document une liste des mots dite aussi descripteurs. Ces termes peuvent être extraits de trois manières.

V.1.1. Méthodes d'indexation :

a) ***Indexation manuelle :*** Chaque document est analysé par un expert ou un documentaliste, et cette indexation assure un taux élevé de correspondance entre les documents et les termes choisis par les indexeurs. En revanche cette indexation est coûteuse en matière de temps et de coût.

b) ***Indexation semi-automatique :*** Cette méthode d'indexation doit être considérée comme un compromis entre l'indexation manuelle et l'indexation automatique.

c) ***Indexation automatique :*** C'est un processus entièrement automatisé, réalisé par un programme informatique en passant par plusieurs étapes .Ce

processus est sans doute le plus utilisé dans le domaine de recherche d'information, vu son coût et son temps réduits.

V.1.2. Processus d'indexation :

Le processus d'indexation repose sur un ensemble de traitements automatisés. Nous les exposons ci-après :

a) ***L'analyse lexicale :*** L'analyse lexicale est la phase qui permet de transformer un texte d'un document en un ensemble de termes dit lexème, durant cette étape la ponctuation, les chiffres, etc, sont éliminés.

b) ***L'Elimination des mots vides :*** Afin d'obtenir un index efficace, il faut éliminer les mots non significatifs qui ne portent pas de sens pour un document tels que les articles, les conjonctions de coordination, les verbes auxiliaires…etc.

c) ***La normalisation :*** Consiste à ramener les mots de la même famille à leur forme normale (avec lemme[2] ou stemme[3]), par exemple les mots : informatique, informatiquement et informaticien, peuvent être représentés par informatique.

d) ***La pondération des termes :*** L'idée de la pondération est d'affecter à chaque terme t_i d'un document d_j ou d'une requête Q, un poids numérique w_{ij} censé le caractériser dans le document ou la requête. Le poids d'un terme dans un document est calculé comme la combinaison de ces deux facteurs:

$$w_{t,d} = tf_{t,d} * idf_t \qquad \text{Où}$$

$tf_{t,d}$ est la fréquence d'occurrence du terme t dans le document d, ses variantes sont :

- $Tf_{(t,d)} = nf_{(t,d)}$ (nombre de fois où t apparaît dans d)
- $Tf_{(t,d)} = nf_{(t,d)} / max_i (nf_{(t,d)})$ (fréquence normalisée entre 0 et 1)
- $Tf_{(t,d)} = nf_{(t,d)} / Sum_i (nf_{(t,d)})$
- $Tf_{(t,d)} = 0,5 + 0,5*(nf_{(t,d)} / max_i (nf_{(t,d)}))$ (fréquence normalisée entre 0,5 et 1)

Idf est le pouvoir de discrimination d'un terme qui est proportionnel à sa fréquence documentaire inverse (notée idf_t), ses variantes sont :

- $Idf_t = N/n$
- $Idf_t = \log (N/n)$
- $Idf_t = \log (N-n/n)$

Où « N » est le nombre de documents de la collection et « n » le nombre de documents où le terme **t** apparaît.

[2] Lemme est un mot portant des marques de flexion (par exemple: je travaille, tu travailles, il/elle travaille… le lemme est le verbe travailler.

[3]Stemme ou racine est un mot correspondant à la partie restante du mot après suppression de son suffixe ou son préfixe.

V.2. Requêtage :

C'est l'expression des besoins de l'utilisateur à travers une liste de mots-clés représentants la requête, et cette dernière peut donc être exprimée en langage naturel ou quasi naturel, dans un format structuré ou à partir d'une interface graphique.

V.3. Appariement :

Une fois les documents transformés, il est possible de rechercher ceux qui répondent le mieux à une question d'un utilisateur grâce à la relation d'appariement et cela en mesurant la valeur de pertinence entre eux. Ce processus se base sur une fonction de similarité notée $RSV\ (q, d)$ (Retrieval Status Value) entre une requête « q » et un document « d ». Le résultat obtenu se traduit par un score qui détermine la probabilité de pertinence (degré de similarité ou degré de ressemblance) du document vis-à-vis de la requête, qui permettra ensuite au SRI de renvoyer à l'utilisateur les documents susceptibles d'être pertinents.

V.4. Reformulation de la requête :

La plupart des utilisateurs sont incapables de formuler, du premier coup, des requêtes précises à des fins de recherche. De ce fait, les résultats qu'ils reçoivent du SRI en réponse à leur requête sont en partie insatisfaisants. Donc ce processus consiste à modifier la requête de l'utilisateur par ajout de termes significatifs et/ou réestimation de leurs poids. Il existe trois méthodes de reformulation de requêtes :

- ***La reformulation directe :*** Cette approche est associée aux systèmes de recherche booléens. On peut procéder à la re-formulation de requête en utilisant un vocabulaire contrôlé (thésaurus[4] ou classification[5]) pour permettre à l'utilisateur de trouver les bons termes pour compléter sa requête.
- ***La reformulation par injection de pertinence (relevance feedback) :*** L'idée est de faire participer l'utilisateur dans le processus de recherche de sorte à améliorer l'ensemble final des résultats. Une nouvelle requête sera formulée par le système à partir de la requête initiale en utilisant l'information sur la pertinence et en appliquant un algorithme spécifique de réinjection de pertinence.

Comme exemple l'algorithme de Rocchio pour la réinjection de pertinence dans le modèle vectoriel :

$$Q_1 = \propto Q_0 + \beta \frac{1}{|R|} \sum_{D_p \in R} D_p - \delta \frac{1}{|NR|} \sum_{D_{np} \in NR} D_{np}$$

Où:

[4] Thésaurus est une liste de mots organisés hiérarchiquement, ayant une valeur de termes dans un domaine de la connaissance dont le terme est la représentation verbale d'un concept.
[5] La classification de termes consiste à regrouper des termes dans des classes distinctes en fonction de leurs propriétés essentielles.

- Q_0 une requête initiale
- D_p(respectivementD_{np}) le vecteur associé à un document pertinent retrouvé (respectivement non pertinent)
- R est l'ensemble des documents pertinents de la collection,
- NR est l'ensemble des documents non pertinents de la collection,
- α, ß, γ des constantes telles que α + ß + γ =1

  - ***La reformulation par pseudo relevance feedback :*** L'idée est d'utiliser les résultats de la recherche en vue d'améliorer l'ensemble final des résultats sans l'intervention de l'utilisateur. On suppose uniquement que les tops m documents retournés sont pertinents, et on les utilise pour reformuler la requête. Le procédé de base consiste à étendre la requête initiale avec de nouveaux termes corrélés aux termes de la requête initiale. Ces termes corrélés sont obtenus à partir d'une étude statistique de co-occurrence ou de corrélation des termes dans les documents du corpus ou bien à partir de ressources externes comme un thésaurus.

VI. Modèles de RI :

Le modèle joue un rôle central dans la RI. C'est lui qui détermine le comportement clé d'un SRI. Il remplit deux fonctions : la première est de créer une représentation interne pour un document ou pour une requête basée sur des termes, la seconde est de définir une méthode de comparaison entre une représentation de document et une représentation de requête afin de déterminer leur degré de correspondance. De nombreux modèles existent, dans la suite nous présenterons certains.

VI.1. Modèle booléen :

Ce modèle est le premier modèle de RI, il se base sur la théorie des ensembles. Un document est représenté par un ensemble de termes exemple $\boldsymbol{d_1(t_1,t_2,t_5)}$, une requête est un ensemble de mots avec des opérateurs booléens : AND (∧), OR(∨), NOT (¬) exemple $\boldsymbol{q=(t_2 \wedge t_3) \vee \neg t_4}$, et l'appariement requête-document est strict et se base sur des opérations ensemblistes selon les règles suivantes :

- $\boldsymbol{RSV(d,t_i) = 1}$ **si** $\boldsymbol{t_i \in d}$**,0 sinon.**
- $\boldsymbol{RSV(d, q_1 \wedge q_2) = 1}$ **si** $\boldsymbol{RSV(d, q_1) = 1}$ **et** $\boldsymbol{RSV(d,q_2) = 1}$**, 0 sinon.**
- $\boldsymbol{RSV(d, q_1 \vee q_2) = 1}$ **si** $\boldsymbol{RSV(d,q_1) = 1}$ **ou** $\boldsymbol{RSV(d,q_2) = 1}$**,0 sinon.**
- $\boldsymbol{RSV(d, \neg q_1) = 1}$ **si** $\boldsymbol{RSV(d, q_1) = 0}$**, 0 sinon**.

Ce modèle est simple à mettre en œuvre mais il dispose de plusieurs inconvénients tels que :

- L'appariement strict et ne permet pas d'ordonner les documents par ordre de pertinence pour la requête.

- Tous les termes dans un document ou dans une requête sont pondérés à l'identique (0 ou 1).
- On ne peut pas exprimer qu'un terme est plus important qu'un autre.
- Les expressions booléennes ne sont pas accessibles à un large public et des confusions existent du fait de la différence de «sens» des opérateurs logiques AND/OR et de leurs connotations[6] respectives en langage naturel.

VI.2. Modèle vectoriel :

Ce modèle est le plus populaire en RI .Un document d_i est représenté par un vecteur de poids $\boldsymbol{w_{ij}}$ de dimension n, dans l'espace vectoriel composé de tous les termes d'indexation d_i= (w_{i1}, w_{i2}, … , w_{in}), une requête Q est aussi représentée par un vecteur de poids w_{Q_1} défini dans le même espace vectoriel que le document Q= (w_{Q_1},w_{Q_2}…,w_{Q_3}) .

Où $\boldsymbol{w_{Q_j}}$ est le poids de terme $\boldsymbol{t_j}$ dans la requête **Q**, et son poids$\boldsymbol{w_{ij}}$ dans le document $\boldsymbol{d_i}$. Ce poids peut être soit une forme de tf*idf, soit un poids attribué manuellement par l'utilisateur.

La pertinence du document **di** pour la requête Q est mesurée comme le degré de corrélation des vecteurs correspondants. Formellement, la pertinence du document $\boldsymbol{d_i}$ pour la requête **Q** est exprimée par l'une des mesures suivantes:

- ***Le produit scalaire :***

$$Sim(d_i,Q)=\sum_{j=1}^{n}(w_{Qj}*w_{ij})$$

- ***La mesure du cosinus :***

$$Sim(d_i,Q)=\frac{\sum_{j=1}^{n} w_{Qj}*w_{ij}}{(\sum_{j=1}^{n} w_{Qj}^2)*(\sum_{j=1}^{n} w_{ij}^2)}$$

- ***La mesure de Dice :***

$$Sim(d_i,Q)=\frac{2*\sum_{j=1}^{n} w_{Qj}*w_{ij}}{\sum_{j=1}^{n} w_{Qj}^2+\sum_{j=1}^{n} w_{ij}^2}$$

- ***La mesure de Jaccard :***

$$Sim(d_i,Q)=\frac{\sum_{j=1}^{n} w_{Qj}*w_{ij}}{\sum_{j=1}^{n} w_{Qj}^2+\sum_{j=1}^{n} w_{ij}^2-\sum_{j=1}^{n} w_{Qj}*w_{ij}}$$

- ***Coefficient de superposition:***

$$Sim(d_i,Q)=\frac{\sum_{j=1}^{n} w_{Qj}*w_{ij}}{min_i(\sum_{j=1}^{n} w_{Qj}^2,\sum_{j=1}^{n} w_{ij}^2)}$$

[6] Signification affective accordée à un mot ou une expression, en plus de son sens premier.

Ce qui rend ce modèle avantageux c'est sa simplicité de mise en œuvre, ainsi son langage de requête est plus simple (liste de mot clés), les performances sont meilleures grâce à la pondération des termes. Néanmoins, ce modèle ne permet pas de modéliser les associations entre les termes d'indexation. Chacun des termes est considéré comme indépendant des autres.

VI.3. Modèle probabiliste :

VI.3.1. Modèle probabiliste de base :

Dans ce modèle, on modélise la pertinence comme un événement probabiliste pour une requête Q donnée.
Estimer P(R | D) la probabilité qu'on obtienne une information pertinente par le document D dépend de Q: P(R | D)=PQ(R | D), on peut estimer de la même façon P(NR | D) la probabilité de non-pertinence de D, on retourne un document D si P(R | D) > P(NR | D) on donne au document D le poids S(D) =$\frac{P(R/D)}{p(NR/D)}$ Par bayes :

- ***Probabilité de pertinence des documents retournés :***

$$\boldsymbol{P(R/D) = \frac{P(D/R) * P(R)}{P(D)}}$$

- P(D | R): probabilité que D fasse partie de l'ensemble des documents pertinents.
- P(R): probabilité de la pertinence d'un document quelconque.
- P(D): probabilité que D soit choisi.

- ***Probabilité de pertinence pour les documents non retournés :***

$$\boldsymbol{P(NR/D) = \frac{P(D/NR) * P(NR)}{P(D)}}$$

- P(D | NR): probabilité que D fasse partie de l'ensemble des documents non pertinents.
- P(R): probabilité de la non-pertinence d'un document quelconque.
- P(D): probabilité que D soit choisi.

Comme c'est l'ordre qui est important, les modifications de score par des constantes qui ne changent pas l'ordre peuvent être ignorées, le score d'un document dépend donc seulement de P(D | R) et P(D | NR).
On suppose une indexation binaire des termes, et on utilise la présence ou l'absence d'un terme dans les documents pertinents ou non pertinents, on note :

$$\begin{cases} x_i = 1 \; si \; t_i \in D \\ x_i = 0 \; si \; t_i \notin D \end{cases}$$

On suppose l'indépendance des termes, mais on suppose que la distribution des termes dans les documents pertinents ou non pertinents est différente.

- $P(D/R) = \prod_{i=1}^{T} P_i^{x_i}(1-p_i)^{1-x_i} \; avec \; p_i = P(t_i \in D/R)$
- $P(D/NR) = \prod_{i=1}^{T} q_i^{x_i}(1-q_i)^{1-x_i} \; avec \; q_i = P(t_i \in D/NR)$

On passe en log :

- $logP(D/R) = \sum_{i=1}^{T} x_i \; log \; p_i + (1-x_i)\log(1-p_i)$

$$\boldsymbol{tel\ que}\ p_i = p(t_i \in D/R) = \frac{r_i}{n}$$

- $logP(D/NR) = \sum_{i=1}^{T} x_i \; log \; q_i + (1-x_i)\log(1-q_i)$

$$\boldsymbol{tel\ que}\ p_i = p(t_i \in D/NR) = \frac{R_i - r_i}{N-n}$$

On a donc une formule qui ressemble à un produit scalaire d'un facteur fréquentiel binaire et d'un poids dépendant du terme :

$$score(D) = \sum_{i=1}^{T} x_i * w_i$$

Avec : $\boldsymbol{w_i = log\frac{p_i(1-q_i)}{q_i(1-p_i)} = log\left(\frac{r_i}{n-r_i} / \frac{R_i - r_i}{N-R_i-n-r_i}\right)}$ *tels que :*

- R_i : Documents contenant un termet_i
- r_i : Documents ne contenant pas t_i
- N : l'ensemble des documents
- n : Documents pertinents

Pour éviter les 0, un lissage de cette formule est proposé :

$$\frac{\boldsymbol{r_i + 0.5}}{\boldsymbol{n - r_i + 0.5}} / \frac{\boldsymbol{R_i - r_i + 0.5}}{\boldsymbol{N - R_i - n - r_i + 0.5}}$$

VI.3.2. Modèle de langue:

Le principe de base du modèle de langue en RI est d'ordonner chaque document D de la collection C suivant leur capacité à générer la requête Q ainsi il s'agit d'estimer la probabilité notée$\boldsymbol{P(Q/D)}$.

Considérons la requête Q telle que $\boldsymbol{Q=m_1, m_2, .. m_n}$. La probabilité $\boldsymbol{P(Q/D)}$ peut-être calculé comme suit :

$$\boldsymbol{P(Q/D) = P(m_1, m_2, .. m_n/D) = \prod_{i=1}^{n} P(m_i/D)}$$

Selon le modèle, il faut estimer : $P(m_i/D)$ en utilisant par exemple **l'estimation par Maximum de vraisemblance** (*Maximumlikelihood*), on aura donc:

$$P(m_i/D) = \frac{freq(m)}{\sum_{m \in D} freq(m)} = \frac{TF(m,D)}{\sum_{m \in D} TF(m,D)}$$

Si un événement (ie. un mot de la séquence) n'apparait pas dans le modèle, ce dernier lui assigne une probabilité de 0 et pour remédier à ce problème, plusieurs méthodes de lissage existent comme ***le lissage de Laplace*** qui consiste à ajouter une constante 1 à toutes les fréquences du langage. Cette méthode est aussi appelée la méthode « ajouter-un ». Pour chaque fréquence α, la probabilité est estimée comme suit : (où V est l'ensemble du vocabulaire d'index)

$$P_{ajouter_{un}}(\alpha \backslash D) = \frac{|\alpha| + 1}{\sum_{\alpha_i \in V}(|\alpha_i| + 1)}$$

VII. Evaluation des systèmes de recherche d'information :

Le but d'un SRI est de retrouver l'information recherchée par l'utilisateur (ou information pertinente) et de la lui retourner dans un délai acceptable, en la lui présentant sous une forme aisément exploitable. Ceci implique la capacité du système à retrouver les documents pertinents d'une part, et la manière de présenter les résultats d'autre part. Donc l'évaluation sert à comparer les résultats retournés par le système et ceux attendus par l'utilisateur, plus on trouve une correspondance plus le système est efficace. Donc cette étape est très importante lors de la mise en œuvre des modèles.

VII.1. Collections de tests :

Une collection de tests est un ensemble de documents (ou collection de documents) à indexer, sur lesquels le système sera évalué, une liste de requêtes prédéfinies et les jugements de pertinence manuellement établis (par des assesseurs humains).

Pour qu'un corpus de test soit significatif, il faut qu'il possède un nombre de documents assez élevé. Les premiers corpus de test développés dans les années 1970 renferment quelques milliers de documents. Les corpus de test plus récents, par exemple ceux de TREC (Text REtrieval Conference) qui ont été mis en place par NIST (Institut National des Standards et Technologies en 1992) contiennent en général plus de 100.000 documents (considérés maintenant comme un corpus de taille moyenne), voir des millions de documents (corpus de grande taille).

VII.2. Mesures d'évaluation :

Comme nous l'avons déjà mentionné, le but d'un SRI est de retourner autant que possible de documents pertinents pour une requête d'utilisateur, et pour estimer si un SRI est efficace, plusieurs mesures d'évaluations sont utilisées.

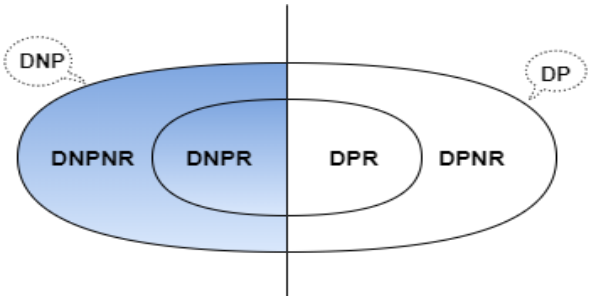

Figure 2: Ensemble des documents

- **DP** ensemble de documents pertinents,
- **DPR** ensemble de documents pertinents retrouvés,
- **DPNR** ensemble de documents pertinents non retrouvés,
- **DNP** ensemble de documents non pertinents
- **DNPR** ensemble de documents non pertinents retrouvés,
- **DNPNR** ensemble de documents non pertinents non retrouvés.

➢ **La précision** est la proportion des documents retrouvés qui sont pertinents. Une précision égale à 1 signifie que le système n'a retrouvé que des documents pertinents.

$$Précision = \frac{|Documents\ Pertinents\ Retournés|}{|Documents\ Retournés|}$$

➢ **Le rappel** est la proportion des documents pertinents qui sont retrouvés. Un rappel égal à 1 signifie que tous les documents pertinents ont été retrouvés.

$$Rappel = \frac{|Documents\ Pertinents\ Retournés|}{|Documents\ Pertinents|}$$

L'idéal serait d'avoir une précision et un rappel égaux à 1, signifiant que tous les documents pertinents sont retrouvés et qu'aucun document non pertinent n'a été retrouvé. En pratique, cet idéal n'est jamais atteint puisque ces deux quantités évoluent en sens inverse ce qui résume que le taux de précision et de rappel sont antagonistes comme le montre la figure 3.

Intuitivement, si on augmente le rappel en retrouvant plus de documents pertinents, on diminue la précision en retrouvant aussi plus de documents non pertinents. Inversement, une plus grande précision risque de rejeter des documents pertinents diminuant ainsi le rappel.

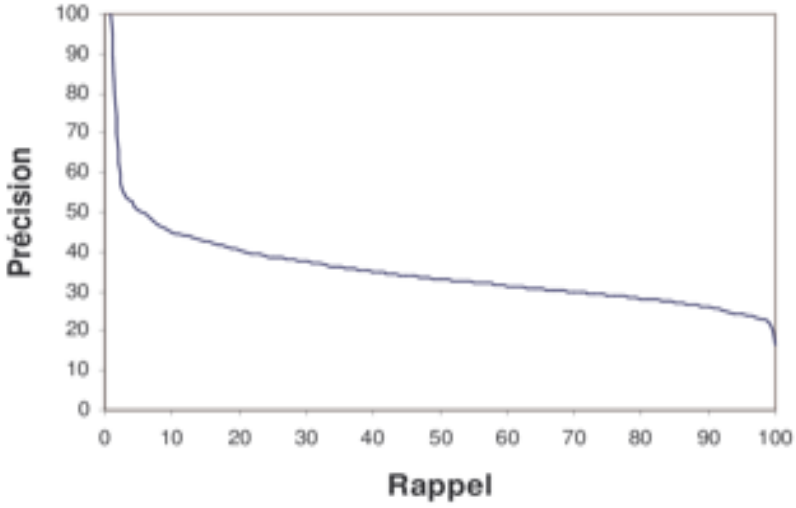

Figure 3:Courbe générale rappel/précision (LÊ THỊ LAN 2005)

Les mesures de rappel et de précision utilisées seules ne sont pas de bons indicateurs de la performance d'un SRI. Plusieurs approches proposées dont: l'agrégation du rappel et de la précision dans une seule mesure qui est le **F-Score** (ou F-mesure) (Rijsbergen 1979).

- **La F-mesure** permet d'agréger le rappel et la précision dans une mesure unique. Elle est définie comme la moyenne harmonique pondérée du rappel et de la précision, soit:

$$F = \frac{1}{\alpha * \frac{1}{Précision} + (1-\alpha) * \frac{1}{Rappel}} \quad où\ \alpha \in [0,1]$$

Les mesures de rappel, précision et F-score sont des mesures basées-ensembles. Elles permettent d'évaluer des ensembles non ordonnés de résultats. D'autres mesures d'évaluation ordonnées existent telles que :

- **La précision moyenne interpolée IAP (InterpolatedAveragePrecision)** est une mesure décrivant la précision globale du système évalué sur une requête. Elle consiste à calculer la précision des résultats sur onze points de rappel qui valent de 0, 10, 20,.., 90 et 100%. Si ces points ne sont pas atteints, les mesures sont alors interpolées. La moyenne de ces 11 précisions forme la précision moyenne interpolée.

Formellement, les mesures de précision aux 11 niveaux standards de rappel sont interpolées comme suit: Si r_j est la référence au j[ieme] niveau standard de rappel:

$$\mathbf{P}(\mathbf{r_j}) = \max_{\mathbf{r_{j+1}} > r > \mathbf{r_j}} \mathbf{P}(\mathbf{r})$$

- P(r_j) est la précision interpolée au niveau standard de rappel r_j.

Habituellement, les algorithmes de recherche sont évalués sur plusieurs requêtes. Pour chaque requête, une courbe rappel/précision interpolée est générée. Pour évaluer les performances de l'algorithme de recherche sur toutes les requêtes, on calcule la moyenne des mesures de précision à chaque niveau standard de rappel. La courbe rappel/précision du système résulte alors du moyennage des résultats des différentes requêtes. La figure 4 montre la courbe rappel /précision interpolée :

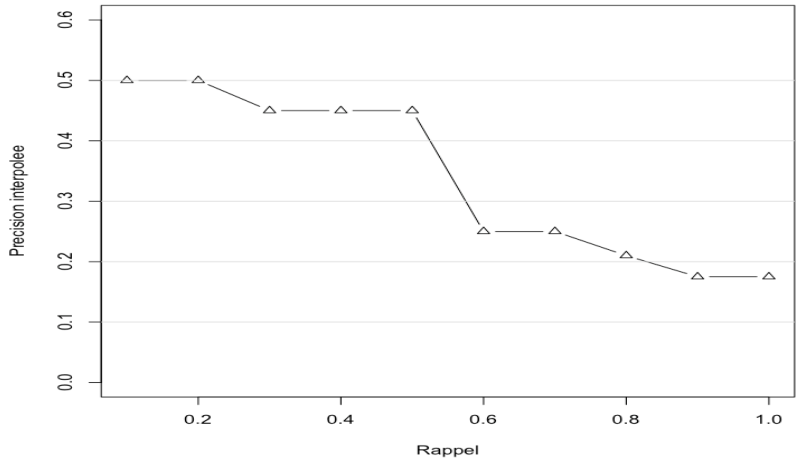

Figure 4:Courbe rappel/précision interpolée

Les courbes rappel/précision sont utilisées pour comparer les performances de différents algorithmes de recherche ce que montre la figure 5.

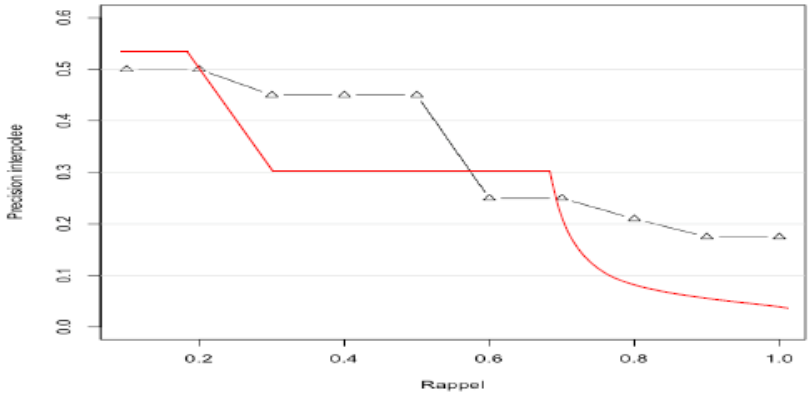

Figure 5:Compare les performances de différents algorithmes de recherche

- **La précision Moyenne :** L'idée est de générer une valeur unique de ranking en moyennant les valeurs de précision obtenues après chaque document pertinent observé.

- **MeanAveragePrecision(LaMAP) :** est la moyenne des précisions moyennes (P_{moy}) obtenues sur l'ensemble des requêtes, à chaque fois qu'un document pertinent est retrouvé :

$$MAP = \frac{\sum_{q \in Q} P_{moy}(q)}{|Q|}$$

Q étant l'ensemble des requêtes.

- **R-précision :** L'idée est de générer une valeur de ranking unique en calculant la précision au rang R, où R est le nombre de documents pertinents pour la requête courante.

$$R_{prec} = P@R = \frac{|DPR|}{R}$$

La R-précision est un bon paramètre pour observer le comportement d'un système pour chaque requête individuellement.

VIII. Conclusion :

Nous avons présenté dans ce chapitre les concepts fondamentaux de la RI classique qui ne s'intéresse qu'aux données textuelles. Cependant avec l'évolution du web et plus particulièrement les réseaux sociaux, l'enjeu est devenu beaucoup plus important, étant donné que d'autres critères, comme les signaux sociaux (ex : j'aime, commentaire,...etc.), ont fait leurs apparitions. Le chapitre suivant portera sur cette nouvelle branche de RI qui prend en considération ces critères : ***la RI Sociale.***

Chapitre 02

La recherche d'information sociale

I. Introduction :

La Recherche d'Information (RI) est le domaine consistant à trouver un objet à partir d'un corpus des documents indexés dans tout média pertinent pour **répondre à la requête d'un utilisateur**.

Initialement conçue pour des recueils de documents textuels, la RI a évolué avec l'apparition du web, où différentes catégories d'outils se sont développés pour faire face aux nouveaux challenges posés par le web et plus récemment des réseaux sociaux.

De nos jours, des millions d'utilisateurs à travers le monde ont intégré les sites des réseaux sociaux dans leurs routines quotidiennes. Ces dernières années, et particulièrement depuis 2005, les chercheurs ont pris conscience que ces réseaux sociaux peuvent être une source fructueuse pour contribuer au développement de plusieurs tâches en recherche d'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 **Recherche d'Information Sociale** (RIS), une thématique récente qui a pour objectif de prendre en compte les informations spécifiques aux *Réseaux Sociaux*.

II. L'information sociale dans le web :

L'information sociale est toute information générée par les explorateurs du web, elle est le fruit des services du web 2.0[7]. Bien avant, les internautes étaient de simples consommateurs dans le web 1.0[8], à présent ces nouvelles fonctionnalités leurs ont permis d'être aussi des producteurs de données.

L'ensemble de ces informations générées par l'utilisateur est appelé UGC « *user generated content* » qui consiste en :

- Contenus publiés dans les réseaux sociaux (Facebook, Instagram, Pinterset, etc.).
- Relations sociales entre utilisateurs (amis, followers, etc.).
- Commentaires et publications de contenus multimédia, etc.

[7]Les nouveaux outils technologiques, leur simplicité d'utilisation et les multiples possibilités d'interactivité sont les marques de ce qui a été popularisé comme le web 2.0 où l'internaute passe d'un consommateur à un co-producteur qui participe à alimenter les sites avec différents types de contenus (texte, image, son, vidéo,..). Il peut aussi agir avec d'autres internautes et partager des informations (forums, blogs, réseaux sociaux, ...).
[8] Le web 1.0 est celui des années 1990, où les internautes étaient des consommateurs d'information en read-only comme le faire dans une bibliothèque classique.

II.1. Les médias et les réseaux sociaux :

Lorsque l'on parle de **médias sociaux** et de **réseaux sociaux**, la confusion est souvent présente. Cependant ce sont deux notions complètement différentes, et il est important de pouvoir les différencier.

Les **médias sociaux**, regroupent tous les sites internet, mais également toutes les fonctionnalités sociales disponibles sur la toile. Ils permettent de publier des articles, des images, des vidéos, de partager une opinion, de parler avec d'autres utilisateurs, etc. Finalement, un média social est une fonctionnalité ou une application qui permet aux utilisateurs de se sociabiliser.

Le **réseau social**, est un site internet. Son but et de mettre en relation les personnes qui y sont inscrites.

Ainsi, on peut dire que le réseau social est un élément du média social.

Parmi les différents médias sociaux qui existent, nous listons ci-dessous les plus populaires :

- **Facebook[9]** : fondé en 2004 par *Mark Zuckerberg* et ses camarades de l'université *Harvard*. Initialement réservé aux étudiants de cette université, il s'est ensuite ouvert à d'autres universités américaines avant de devenir accessible à tous en septembre 2006. C'est un réseau social en ligne qui permet à ses utilisateurs de publier des images, des photos, des vidéos, des fichiers et documents, d'échanger des messages, joindre et créer des groupes et d'utiliser une variété d'applications. Au troisième trimestre 2019, Facebook comptait 2,45 milliards d'utilisateurs actifs chaque mois et 1,62 milliard d'utilisateurs actifs chaque jour dans le monde.
- **Youtube[10], le social media dédié aux videos :** créé en février 2005 par *Steve Chen, Chad Hurley et Jawed Karim*, trois anciens employés **de PayPal**[11], et racheté par Google en octobre 2006. Classé numéro 1 dans le partage et le visionnage de vidéos, Youtube a su fédérer une communauté de plus de 2 milliards d'utilisateurs actifs par mois en 2019.
- **Twitter[12], la plateforme de microblogage** : a été créée le 21 mars 2006 *par Jack Dorsey*, *Evan Williams, Biz Stone et Noah Glass.* Le siège social de Twitter se situe aux *États-Unis à San Francisco*. C'est un réseau social de microblogage géré par l'entreprise Twitter Inc. Il permet à un utilisateur d'envoyer gratuitement de brefs messages, appelés ***tweets***, sur internet, par messagerie instantanée ou par SMS. Ces messages sont limités à 280 caractères (140 caractères jusqu'en novembre 2017). Au premier trimestre 2019, en moyenne, sur un mois, 330 millions de personnes se sont connectées sur Twitter.

[9] https://www.facebook.com/
[10] https://www.youtube.com/
[11] PayPal est une entreprise américaine offrant un système de service de paiement en ligne dans le monde entier. La plateforme sert d'alternative au paiement par chèque ou par carte bleue.
[12] https://www.twitter.com/

- **Linkedin[13], le plus professionnel des réseaux sociaux :** est un réseau social professionnel en ligne qui a été fondé en décembre 2002 et lancé en mai 2003 par *Reid Hoffman et Allen Blue*, membres de la *PayPal Mafia[14]*, et trois autres entrepreneurs à *Mountain View (Californie).* Le 13 juin 2016, Microsoft annonce le rachat du réseau social pour un montant de 26,2 milliards de dollars américains.
- **Instagram[15], le réseau social qui monte en flèche :** fondé et lancé en octobre 2010 par *l'Américain Kevin Systrom et le Brésilien Michel Mike Krieger*. Depuis 2012, l'application appartient à Facebook, c'est une application mobile permettant le partage de photos, carrousel d'images et vidéos. Elle s'est fait connaitre notamment grâce à ses filtres et ses options de retouche de photos qui permettent à n'importe qui de rendre ses photos plus attractives avant leur partage.
- **Pinterest[16], le réseau source d'inspiration :** lancé en 2010 par *Paul Sciarra , Evan Sharp et Ben Silbermann* . Ce réseau social est aussi dédié au partage de photos et de vidéos, mais cette fois, sur un tableau thématique : les boards. Il compte aujourd'hui une communauté mondiale de plus de 250 millions d'utilisateurs.
- **Snapchat[17], l'éphémère à tout prix** : conçue et développée par des étudiants de l'*université Stanford en Californie.* L'application est lancée en septembre 2011 sur l'App Store d'Apple, puis en novembre 2012 sur Android. C'est une application mobile gratuite permettant de partager des photos, de courtes vidéos avec filtres et même de chatter sans laisser de traces.

II.2. Contenus générés par les utilisateurs :

Avec l'émergence des médias sociaux et l'évolution du web, l'interaction des utilisateurs avec le contenu du web a changé. Ils sont passés de simples consommateurs à des producteurs de l'information que l'on appelle **Contenus Générés par l'Utilisateur (UGC).**

II.2.1. Définition :

Le **contenu généré par les utilisateurs** (**CGU**, en anglais ***user-generated content***, ou **UGC**), devient populaire pendant l'année 2005, il fait référence à tous types de contenus créés par les utilisateurs sur des plateformes web. Ce contenu peut inclure : des images, des vidéos, du texte, des commentaires, des articles de blog et du son, etc. Ce dernier est généralement publié en ligne, où il peut facilement faire le buzz.

[13] https://www.linkedin.com/
[14] La Mafia PayPal est un surnom donné à un groupe d'anciens employés et fondateurs de l'entreprise PayPal.
[15] https://www.instagram.com/
[16] https://www.pinterest.com/
[17] https://www.snapchat.com/

II.2.2. Les signaux sociaux :

Les signaux sociaux sont des informations qui fournissent des renseignements sur les interactions, les émotions, les relations et les comportements sociaux d'une personne réelle avec une ressource sur le Web à travers des fonctionnalités offertes par les réseaux sociaux.
Les commentaires, les j'aime, les actions et interactions peuvent être considérés comme des signaux sociaux. Chaque signal est associé à un réseau social particulier, dont le fonctionnement diffère d'un réseau à un autre. Par exemple, un partage sur Facebook et un *Retweet* sur *Twitter* n'entrainent pas la même signification ou la même influence sur la stratégie du web.

Type	Example	Social media
Vote	Like +1	Instagram, Youtube, …etc.
Message	Post Tweet	Google+, LinkedIn, *Twitter* …etc.
Sharing	Share Retweet	Facebook , Twitter, …etc.
Marker	Favorites Pin **Haschtag**	Instagram, Pinterest , Twitter, …etc.
Comment	Comment Answer	Facebook, Twitter, …etc.
Relation	Subscribers Followers	Youtube, Instagram, …etc.

Tableau 1:Les signaux sociaux les plus utilisés (Wikipédia.org)

III. La recherche d'information sociale RIS :

III.1. Définition :

La recherche d'information sociale (RIS) est un domaine innovant datant des années 2000, après l'émergence du web 2.0 et la naissance des réseaux sociaux, qui rassemblent deux disciplines de recherche, à savoir la recherche d'information et l'analyse des réseaux sociaux.
La RIS se base sur l'identification et l'intégration des informations sociales dans un processus de recherche.

En fait, plusieurs types d'informations sociales sont utilisés dans les travaux de RIS. On peut citer des tags, relations sociales, commentaires, tweets, mentions de sympathie, conversations, hashtags, partages, etc.

III.2. Concepts de la RIS :

Les systèmes de RI Sociale se distinguent des autres types de systèmes de RI par l'exploitation des UGCs et des relations issues des réseaux sociaux dans le processus de recherche d'information.
Cependant, les modèles classiques de la RI sont aveugles face à ce contexte social qui entoure les utilisateurs et les ressources. Pour remédier à ce problème, un autre modèle de recherche qui se base sur les données sociales a été créé afin d'améliorer le processus de RI et mieux appréhender ses données, ce modèle est appelé la recherche d'information sociale (RIS) qui est illustré dans la figure 6 suivante.

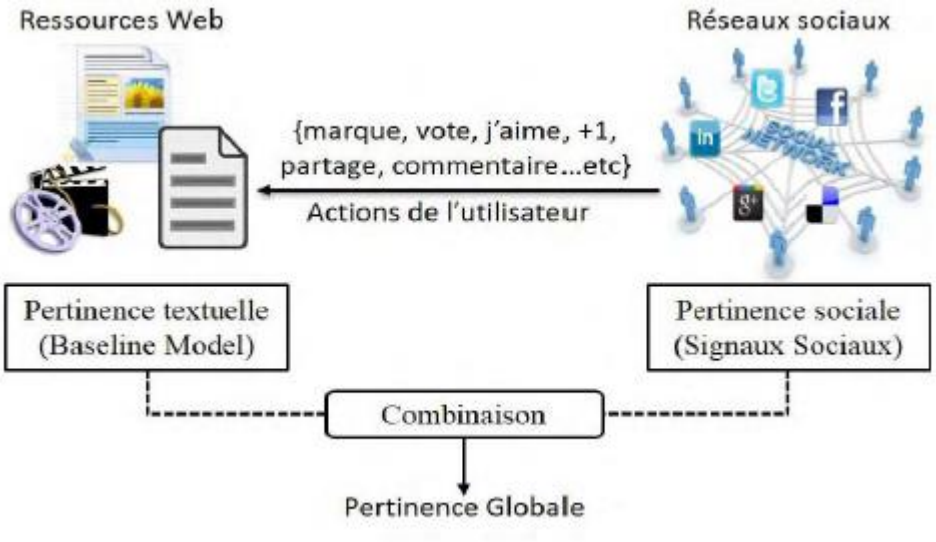

Figure 6:Modèle de recherche d'information sociale (BADACHE 2016)

IV. L'Etat de l'Art de la RIS :

A partir de la définition de Teevan[18] qui décrit la recherche d'information sociale comme étant un domaine de recherche émergeant qui explore comment les interactions sociales et les données sociales peuvent améliorer les expériences existantes en recherche d'informations. (Teevan 2012) considère la RIS sur trois axes :

- le premier axe concerne la recherche d'information de nature sociale.
- le deuxième porte sur l'exploitation des contenus sociaux pour améliorer la RI, qui est en effet le sujet de notre travail.

[18] https://www.microsoft.com/en-us/research/video/social-search-panel/

- Le troisième axe concerne la recherche d'information effectuée par plusieurs personnes, recherche collaborative d'information[19]. (Yue 2018).

IV.1. Identification et exploitation des contenus sociaux pour améliorer la RI :

L'identification des ressources sociales a pour but de retrouver les informations sociales qui répondent aux exigences de l'utilisateur, c'est-à-dire, rechercher dans les différents espaces sociales du web les informations qui seront pertinentes ou qui sont susceptibles de répondre aux exigences de l'utilisateur.

En outre, les SRI sociale doivent analyser tout contenu généré par l'utilisateur en passant en revue les blogs, les microblogs, les réseaux sociaux (j'aime, commentaires, publications, partages …)

Une fois l'information sociale identifiée et extraite, il faut trouver un moyen de comment utiliser cette ressource pour améliorer le processus de la RI, nous présentons dans ce qui suit les travaux de l'état de l'art consacrés à l'intégration de ces informations sociales en distinguant ses différents niveaux d'améliorations.

Il existe principalement trois niveaux d'amélioration : (BADACHE 2016)

- L'amélioration de l'index, à savoir la façon dont les documents et les requêtes sont représentées et appariées pour estimer leurs similarités,
- La reformulation des requêtes à l'aide de connaissances supplémentaires, à savoir l'expansion de la requête de l'utilisateur, et
- Le reclassement (*re-ranking*) des documents retournés par un SRI (sur la base du profil d'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.

IV.1.1. Indexation sociale :

Les travaux de (Bishoff 2008) (Dmitriev 2006) ont démontré que l'ajout des annotations sociales au contenu du document améliore efficacement les résultats et la qualité de la recherche.

L'information sociale peut être essentiellement utile si le document ne contient pas beaucoup de termes et que le processus d'indexation classique ne fournit pas une bonne performance de RI (BADACHE 2016).

Ces travaux proposés ont utilisé les informations sociales de deux manières différentes :

[19] La recherche collaborative d'information intègre la collaboration d'équipe avec la recherche exploratoire, de sorte que les tâches de recherche complexes peuvent être décomposées en tâches plus simples et plus petites à résoudre par les membres individuels de l'équipe.

- **Par l'ajout des données sociales au contenu du document :** dans ce cas, on utilise les métadonnées pour enrichir le document alors l'indexation du document se fait à la fois avec son contenu textuel et ses contenus sociaux associés (tags et commentaires). (Chelaru 2012)
  Cependant, chaque approche utilise une méthode différente pour pondérer les termes des métadonnées sociales. Par exemple, **TF-IDF** pour (Carmel 2010)
- **Par la représentation de documents personnalisés :** chaque utilisateur a sa manière de comprendre le contenu d'un document. Cependant pour le décrire, l'annoter ou le commenter, il va employer son propre vocabulaire. Par conséquent, on attribuera à chaque utilisateur son index personnalisé afin de mieux répondre à son besoin en informations.

Nous citons dans ce contexte le travail de Mohamed Reda Bouadjenek (Bouadjenek 2013), qui a proposé d'utiliser une représentation personnalisée des documents sociaux (PSDR) et qui se déroule en 3 phases comme l'illustre la figure7 :

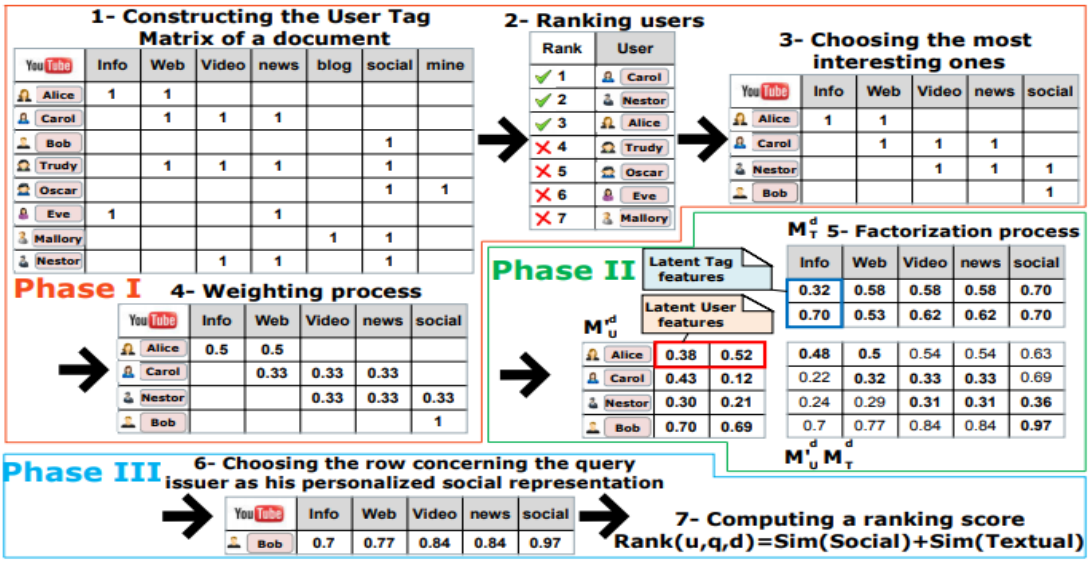

Figure 7 : Processus de création d'un PSDR pour une page web. (Bouadjenek 2013)

- **Phase I :** Représenter chaque page web qui correspond à la requête à l'aide d'une matrice $m \times n$ Users Tags matrix $M^d_{U,T}$ de m utilisateurs qui ont annoté la page web avec les n tags, et chaque entrée $w_{i,j}$ de la matrice représente le nombre de fois que l'utilisateur u_i a utilisé le terme t_j pour annoter la page web. Ils ont ensuite utilisé une fonction de classement pour classer les utilisateurs du plus pertinents au moins pertinents et d'en sélectionner uniquement les utilisateurs les plus représentatifs, ainsi ils auront une nouvelle matrice Users-Tags réduite. Le score de classement d'un utilisateur u selon un document d et une requête émise par un utilisateur u_q est donnée comme suit :

$$Rank^d_{uq}(u) = \alpha \times \log\left(\frac{|D|}{|Du|}\right) \times \frac{|Tu,d|}{|Td|} + (1-\alpha) \times \cos(u, uq)$$

Où :

- $|D|$ et $|Du|$ sont respectivement le nombre de documents et le nombre de documents étiquetés par u.
- $|Td|$ et $|Tu, d|$ sont respectivement le nombre de balises de d, et le nombre de balises utilisées par u pour annoter d.
- $Cos(u,uq)$ dénote la similitude cosinus entre un utilisateur qui annote d et l'émetteur de requête en fonction des balises qu'ils ont utilisés.
- α est un poids réglé à 0,5.

Une fois la matrice créée, ils ont procédé au calcul des poids wi, associés à chaque entrée de la matrice qui représente la mesure dans laquelle l'utilisateur u_i estime que le terme t_j est associé au document d. Ils ont proposé d'utiliser une adaptation de la mesure de $TF\text{-}IDF$ pour estimer ce poids comme suit :

$$W_{ij}=\frac{n^{d}_{ui,tj}}{|T_{ui,d}|}\times \log\left(\frac{|D_{ui}|+1}{|D_{ui,ti}|}\right)$$

Où :

- $n^{d}_{ui,tj}$ est le nombre de fois où u_i a utilisé t_j pour annoter d.
- $|D_{ui,ti}|$ est le nombre des documents marqués par u à l'aide de t_i..

- **phase II :** dans cette phase, ils ont proposé l'utilisation d'un processus de factorisation matricielle qui permet d'utiliser l'expérience et les feedbacks des autres utilisateurs afin de prédire les valeurs manquantes dans la matrice et ensuite calculer le PSDR de l'émetteur de requête en particulier.

- **phase III** : Enfin, classer les documents en fonction de leurs PSDR qui doivent être appariés avec la requête pour quantifier leurs similitudes tout en considérant également le contenu textuel des documents. En ce qui concerne ce processus, ils ont proposé de calculer le score de classement personnalisé d'un document d qui correspond potentiellement aux termes d'une requête q émise par un utilisateur u comme suit:

$$\text{Rank}(u, q, d) = \gamma \times \text{Sim}(\vec{q}, \overrightarrow{(S_{d,u})}) + (1-\gamma) \times SES(\vec{q}, \vec{d})$$

Où :

- γ est un poids compris entre 0 et 1, $SES(\vec{q}, \vec{d})$ est le Search Engine Score donné pour un document d.
- $\overrightarrow{(S_{d,u})}$ est le PSDR de document d selon l'utilisateur u.

IV.1.2. Reformulation de la requête :

La reformulation de la requête (Query Reformulation) consiste à transformer une requête initiale q_0 en une requête q. Cette transformation se fait de deux manières :

- Soit en rajoutant d'autres termes significatifs pour rendre la requête initiale moins ambigüe, on parle donc **d'expansion de requête** (Query Expansion),
- soit en éliminant des termes ou des informations inutiles de telle sorte à ce que la requête initiale soit réduite, ce que l'on appelle **raffinement de requête** (Query Reduction).

Koolen et ses collègues (Koolen 2009) ont proposé une approche qui consiste à utiliser les données de Wikipédia comme collection externe pour étendre la requête et qui sera ensuite utilisée pour la recherche de livres.
D'autres pistes concernant le "Pseudo-Relevance Feedback" à partir de Wikipedia ont été explorées, notamment par l'approche de (Li 2006) et ses collègues qui traitent les requêtes dites "faibles". Ces requêtes ne permettent pas de retourner suffisamment de documents pertinents lors de la première recherche.

(Schenkel 2008) propose de sélectionner parmi l'ensemble des termes du contexte social de l'utilisateur un sous-ensemble de termes qui peuvent être employés pour étendre la requête de l'utilisateur. Ainsi, en plus de la dimension sociale sur laquelle les auteurs se basent pour construire un contexte social de l'utilisateur, ils proposent de prendre en compte une dimension sémantique pour sélectionner des termes. Cela se fait sur la base d'une similarité sémantique entre ces termes et ceux de la requête de l'utilisateur.

En outre, (Bao 2007) et ses collègues suggèrent l'utilisation d'un graphe bipartie entre les annotations sociales et les pages Web avec des arêtes indiquant le nombre d'utilisateurs où, ils proposent deux algorithmes : le *Social Sim Rank (SSR)* et le *Social Page Rank (SPR)*.

- Le ***Social Sim Rank*** est un algorithme itératif pour évaluer quantitativement la similitude entre deux annotations.

- Le ***Social Page Rank*** est utilisé pour évaluer la popularité des pages web basées sur l'amélioration mutuelle entre trois ensembles distincts : les pages web populaires, les utilisateurs web et les signaux récents.

IV.1.3. Reclassement des résultats :

Dans la recherche d'information, le classement des résultats consiste à définir une fonction de correspondance qui permet de quantifier la similarité entre les documents et les requêtes. Deux catégories sont distinguées pour le classement des résultats sociaux, qui utilisent l'information sociale différemment.
La première catégorie utilise l'information sociale en intégrant une pertinence sociale au processus de classement, alors que la seconde procède à la personnalisation des résultats de la recherche en utilisant l'information sociale.

a. Classement basé sur la pertinence sociale :

Les diverses approches qui ont été proposées dans cette classe afin d'améliorer le classement des documents vis-à-vis d'une requête, se basent sur la pertinence sociale qui se réfère à des facteurs sociaux caractérisant un document en termes de popularité et de réputation dans les réseaux sociaux.

(Bao 2007) ont proposé le **Social PageRank** qui est un algorithme qui calcule la qualité et la popularité de la page par le nombre d'annotations sociales. Pour chaque page web, annotation et utilisateur un PageRank[20] est calculé sur la base des liens existants entre eux.

Le Social Blade RanK[21] (Yanbe 2007) qui indique le nombre d'utilisateurs qui ont marqué une page.

Nous trouvons aussi FolkRank et Adapted-Pagerank (Hotho 2006) qui sont tous une extension de l'algorithme de PageRank (Brin 1998).

La notion de base de l'algorithme Adapted-Pagerank est que si une ressource est taguée avec un nombre important de tag et par des utilisateurs importants, elle sera aussi importante.

(BADACHE 2016) quant à lui, a proposé une approche consistant à estimer l'importance sociale d'une ressource avec l'exploitation de 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 sous-jacent. Afin de prendre en compte ces facteurs sociaux dans l'évaluation de la pertinence. (BADACHE 2016) s'est appuyé sur un modèle de langue qui lui permet de combiner l'importance à priori de la ressource et sa pertinence vis-à-vis de la requête.

[20] PageRank : est un algorithme de Google qui mesure quantitativement la popularité d'une page web.
[21] Social Blade est un site Web de statistiques qui vous permet de suivre vos statistiques et de mesurer la croissance sur plusieurs plateformes de médias sociaux, y compris YouTube, Twitter et Instagram.

La probabilité qu'une ressource D soit pertinente par rapport à une requête Q est estimée comme suit :

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

Où :

- P(Q | D) représente la probabilité de la pertinence textuelle en utilisant la méthode de pondération Okapi BM25.
- P(D) qui est une probabilité indépendante de la requête Q qui concerne l'importance sociale de chaque document.

Ajouté à ça d'estimer la probabilité à 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 :

$$P(D_i) = \prod_{r_j \in R} P(r_i \mid D^{r_j})$$

Où $(rj \mid Djr)$ est lié à l'apparition de la réaction rj dans le document Di qui est calculé par le rapport entre le nombre de réaction rj dans le document Di sur le nombre total de réaction sur le même document.

b. *Classement social personnalisé :*

L'objectif de cette catégorie est d'améliorer la recherche en permettant de classer les documents différemment pour chaque utilisateur sous principe que les utilisateurs ont des profils et des besoins différents, et donc ils ne devraient pas avoir les résultats classés de la même manière. Plusieurs travaux ont été proposé dans ce contexte (Bender 2008), (Meinel 2007), (Jin 2010) et qui suivent la même idée, qu'un score de classement d'un document d récupéré lorsqu'un utilisateur soumet une requête q est déterminé par :

o Un processus de similarité entre les termes de la requête q et le document d pour générer un score de classement sans rapport avec l'utilisateur.

o Un processus de mise en correspondance d'intérêts qui calcule la similitude entre un utilisateur u et le document d pour générer un score de classement lié à l'utilisateur.

o Au final, une fusion est effectuée entre les deux classements précédents pour avoir un classement final.

o Un profil utilisateur : qui est représenté par une matrice tag-document Md avec m tags et n documents et leurs n bookmarks[22].

[22] Un bookmark d'un document D_j est un vecteur b_j avec ses composants c_{ij} qui sont mis à 1 si le tag t_i est associé à dj et à 0 sinon.

La technique de personnalisation de (Meinel 2007) comprend deux étapes principales, à savoir :

(i) la collecte et l'agrégation de données sur les utilisateurs et les documents.

(ii) la personnalisation de la recherche Web sur la base de ces données.

(Bouadjenek 2013) propose de calculer la correspondance entre un document D et une requête Q, en utilisant la fonction SoPRa élaborée pour le fait :

La fonction SoPRa basique se compose de deux parties, la première avec :

(i) Un score de correspondance textuelle et

(ii) la deuxième avec un score de correspondance sociale

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

- $sim(\vec{q},\vec{d})$ 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.

Ils ont utilisé pour ce fait le modèle vectoriel (VSM) qui malgré son ancienneté continue à faire ses preuves.

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

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

Où γ est une valeur qui satisfait $0 \leq \gamma \leq 1$.

IV.2. Exploitation de la temporalité des signaux sociaux pour améliorer la recherche :

Il existe peu de travaux qui se sont intéressés à ces questions en RI. Ceux qui s'en rapprochent correspondent à ceux réalisés par (Khodaei 2012).

(Yoshiyuki Inagaki 2010) et ses collègues ont proposé d'exploiter les caractéristiques de clic (*click through*) en RI. Parmi ces critères, un facteur appelé *ClickBuzz*, qui capte l'intérêt que suscite un document à travers le temps.
Ils ont défini le *ClickBuzz* comme une mesure pour déterminer si une page Web reçoit un niveau inhabituel d'intérêt des utilisateurs par rapport au passé. Le *ClickBuzz* est basé sur le nombre de clics sur le document au cours d'un intervalle de temps donné. Cette méthode permet d'exploiter le *feedback* des utilisateurs pour améliorer la qualité des résultats de recherche en favorisant les URL qui ont un intérêt récent pour les utilisateurs.

Les travaux de (Khodaei 2012) ne se sont pas intéressés à la tâche de recherche d'information. Ils tentent juste d'identifier l'évaluation des intérêts des utilisateurs dans le temps. Ils considèrent, en effet, que la grande masse des contenus générés par les utilisateurs dans les réseaux sociaux offre une occasion d'examiner comment les utilisateurs produisent et consomment ce type de contenu au fil du temps. Ils classent les intérêts sociaux des utilisateurs en cinq classes: "recent", "ongoing", "seasonal", "past" et "random", puis analysent Twitter ainsi que des données de Facebook sur les activités sociales des usagers. Ils discutent également trois solutions différentes où ces signaux sensibles au temps peuvent être appliqués : a) la RI personnalisée; b) la RI basée sur les amis et c) la RI collective.

(BADACHE 2016) a aussi proposé d'estimer l'importance sociale d'une ressource en exploitant le moment où l'interaction (signal) s'est produite ainsi que la date de publication de la ressource. Afin de prendre en compte cette importance dans l'évaluation de pertinence, il a repris le même modèle de langue basé sur les signaux sociaux qu'il a déjà présenté mais en prenant en compte l'aspect temporel.

- **Prise en compte de la date du signal social :** dans ce niveau, BADACHE a proposé de compter les occurrences d'un signal en le pondérant avec sa date d'apparition. La formule correspondante est la suivante :

$$\text{Count}_{\text{ta}}(t_{j,ai}, D) = \sum_{j=1}^{k} f(tj, ai, D)$$

Avec $t_j \in T_{ai}$ et T_{ai} représente l'ensemble de k moments (date) à laquelle une action a_i s'est produite.

- **Prise en compte de la date de publication de document** : l'auteur a proposé de normaliser la distribution des signaux sociaux associés à une ressource par la date de publication de la ressource. La formule correspondante est la suivante :

$$\text{Count}(a_i,) = Co(a_i, D) \cdot A(D)$$

Où

$$A(D) = \exp\left(-\frac{||t\ actuel - tD||^2}{2\delta^2}\right)$$

Avec :

- (D) représente la fonction temporelle du document, estimée en utilisant le noyau Gaussien. Cette fonction calcule la distance temporelle entre la date actuelle $t\ actuel$ et la date de la ressource tD.
- Et δ est un paramètre du noyau Gaussien.

V. Evaluation de la RI Sociale :

L'évaluation des SRI est depuis le début des travaux sur la RI un des piliers de l'évolution de ce domaine. Elle se fait principalement à travers des collections de tests[23], souvent construites dans le cadre de campagnes d'évaluation.

La RI sociale ne déroge pas à 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*** [24].

V.1. La tâche TREC Microblog :

C'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.***
Lorsqu'il a été introduit, il concernait une tâche de recherche ***ad hoc*** en temps réel (real-time Adhoc Search Task), dans laquelle l'utilisateur souhaite voir les informations les plus récentes mais pertinentes pour la requête, puis une seconde tâche connue par ***filtering track*** introduite en 2012. Ces deux dernières se basent sur le corpus des tweets.

En 2011, la collection de texte ***Tweets*** **2011** contenait 16 millions de *Tweets* (0,5 Go) exprimés dans diverses langues et publiés sur Twitter entre le 23 janvier 2011 et le 8 février 2011. L'ensemble de données est construit en utilisant l'API publique Twitter Stream qui fournit un échantillon représentatif de 1% du flux des *Tweets*.

Ces dernières années, cette collection a évolué et le corpus a été fortement enrichi par de nouveaux *Tweets*. La collection actuelle, connue sous le nom ***Tweets*** **2013**, se compose de 243 millions de *Tweets*.

V.2. La tâche Social Book Search :

La tâche ***de Social Book Search*** (SBS) étudie la recherche de livres dans des scénarios où les besoins d'information du monde réel sont généralement complexes, mais presque toutes les recherches se concentrent sur une recherche relativement simple basée sur des requêtes ou une recommandation basée sur des profils.

[23] Une collection de test est constituée d'un ensemble d'articles et un ensemble de requêtes. Ces requêtes sont évaluées par des experts afin de déterminer les réponses souhaitées (ie. les articles pertinents).
[24] *INEX* est une compagne internationale lancée en 2002 dans l'objectif de promouvoir l'évaluation de la recherche dans les documents semi-structurés en fournissant de grandes collections de test de documents XML, les mesures d'évaluation uniformes , et un forum pour les chercheurs afin de comparer leurs résultats https://inex.mmci.uni-saarland.de/

L'objectif est de rechercher et de développer des techniques pour aider les utilisateurs dans des tâches de recherche de livres complexes.

La tâche de ***Social Book Search*** se compose de trois *Tracks* :

- **Interactive Track** : une tâche interactive orientée utilisateur explorant les systèmes qui prennent en charge les utilisateurs à chacune des multiples étapes d'une tâche de recherche complexe. La piste offre aux participants une configuration de RI interactive expérimentale complète et une nouvelle interface de recherche à plusieurs niveaux passionnante pour étudier comment les utilisateurs se déplacent à travers les étapes de recherche.
- **Suggestion Track** : une tâche orientée système permettant aux systèmes de suggérer des livres basés sur des demandes de recherche combinant plusieurs signaux de pertinence d'actualité et contextuels, ainsi que des profils d'utilisateurs et des jugements de pertinence du monde réel.
- **Mining Track** : une piste NLP / Text Mining qui se concentre sur la détection et la liaison des titres de livres dans les forums de discussion de livres en ligne, ainsi que la détection de la recherche de recherches de livres dans les messages du 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 représente un fichier XML contenant des champs comme ISBN, title, review, summary, rating and tag. La collection livre 208 requêtes ainsi que leurs jugements de pertinence fournies par ***INEX***.

VI. Conclusion :

Nous avons abordé dans ce deuxième chapitre, la notion de la RI Sociale qui est une concaténation de la recherche d'information avec les réseaux sociaux qui, exploitent des données précieuses appelées UGCs afin d'améliorer les performances de la RI Classique. Par la suite, nous avons présenté les principaux travaux de l'état de l'art existant se rapportant à la RIS.

Dans le chapitre qui suit, nous présentons deux approches, qui ont été proposées par Mohamed SEKOUR et Nabil LARBI, pour la RI en exploitant les signaux sociaux de Twitter.

Chapitre 03

Approches étudiées pour la recherche

I. Introduction :

Les premiers travaux de la recherche d'information se sont focalisés sur la correspondance textuelle entre les documents et la requête, en ignorant les interactions des utilisateurs avec ces documents. Cependant, avec l'explosion des réseaux sociaux et la disponibilité de différentes informations sociales, les chercheurs ont commencé à s'intéresser davantage à l'exploitation de ces métadonnées dans le but d'améliorer le processus de recherche, ce qui a donné naissance à la RI sociale.

Dans ce chapitre, nous allons présenter et détailler deux approches qui ont été proposées par nos camarades (LARBI 2019) et (SEKOUR 2019) en vue d'améliorer la recherche d'information.

II. Approche proposée par Mohammed SEKOUR:

II.1. Architecture générale:

L'objectif de cette approche est d'améliorer la recherche d'information en exploitant les signaux sociaux de Twitter. Elle est basée sur deux étapes comme l'illustre la figure 8.

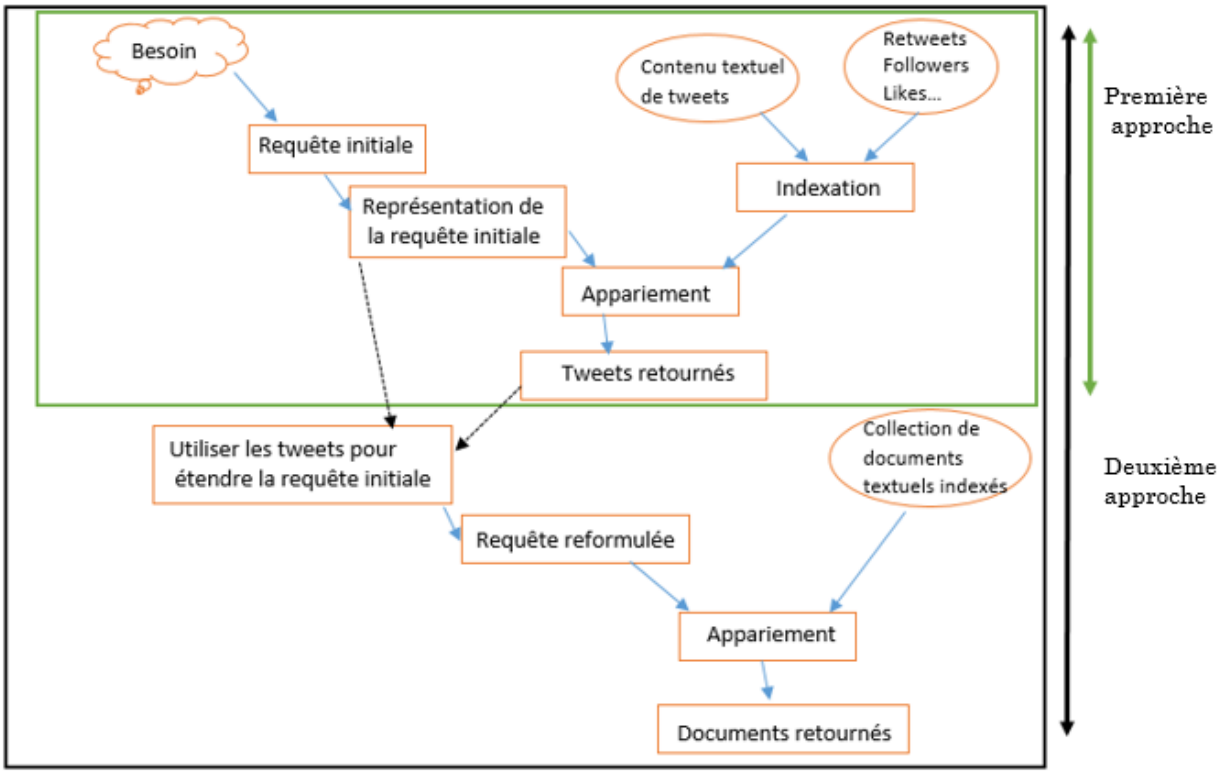

Figure 8:Architecture d'approche proposée par (SEKOUR 2019)

Dans l'architecture présentée dans la figure 8, (SEKOUR 2019) a suggéré deux étapes :

- **1ere étape** se focalise sur l'indexation classique en incluant les signaux sociaux de twitter afin de favoriser les tweets les plus pertinents.
- **2eme étape** se base sur la reformulation de la requête en effectuant la recherche dans un corpus externe, dans le but de mieux exprimer le besoin informationnel de l'utilisateur.

II.2. Notations :

L'information sociale exploitée dans la première étape est représentée par un triplé **<*U*, *Ms*, *T*>** où ***U*, *Ms*, *T***, représentent respectivement ***Utilisateurs, Métadonnées sociales et Tweets***. Concernant la deuxième étape, on rajoute un ensemble représentant des ressources pour avoir un quadruplet **<*U*, *Ms*, *T*, *R*>** expliqués comme suit :

- ***Utilisateurs U:*** ce sont les personnes qui utilisent le réseau social Twitter, ces derniers peuvent publier, réagir et aimer les *tweets* et les retweeter, etc.

- ***Métadonnées sociales Ms:*** c'est le nombre d'actions (nombre de j'aime, nombre de retweets, etc.), que les utilisateurs ont effectué sur les *Tweets ;* et on aura: $Ms = \{N_{rt}, N_j, N_f, N_c\}$ où N_r, N_j, N_f, N_c représentent respectivement le nombre de *retweets*, le nombre de j'aimes, le nombre de followers et le nombre de commentaires.

- ***Tweets T:*** c'est une collection de N tweets partagée par un utilisateur U_i, comportant des informations textuelles et un ensemble de métadonnées sociales (N_j, N_{rt}, N_f,).

- ***Ressource R:*** *c*'est une collection de *M* documents indépendante des *Tweets*. Elle peut être un texte, une page web ou toute autre information numérique.

II.3. l'indexation des Tweets :

Consiste à modifier le processus de l'indexation des *tweets,* dans le but de les reclasser par leurs popularités où il devra prendre en compte le contenu textuel et les métadonnées sociales. La figure 9 illustre le nouveau processus proposé par (SEKOUR 2019).

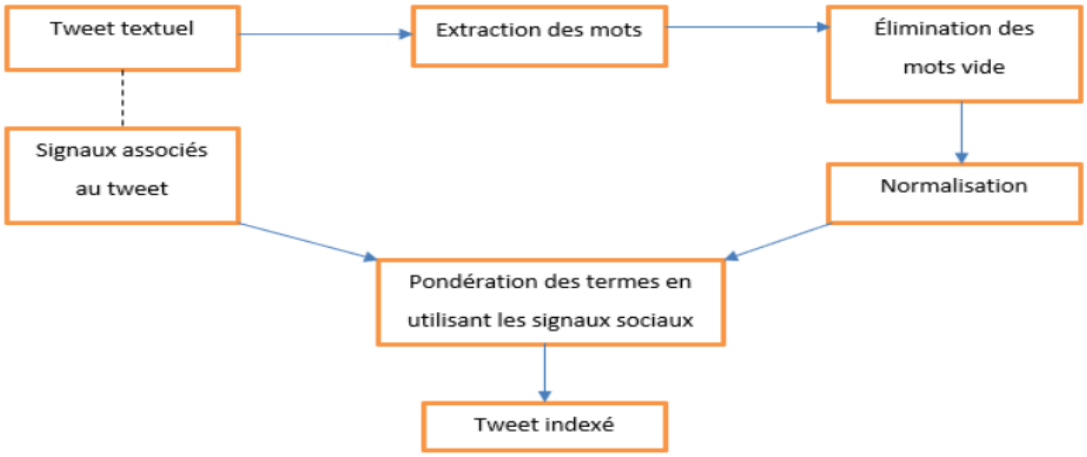

Figure 9:Processus d'indexation des tweets (SEKOUR 2019)

- ***Extraction des mots :*** consiste à extraire la liste des termes (tokens) constituant un tweet.

- ***Élimination des mots vides :*** consiste à supprimer les mots non significatifs des *Tweet*s (les pronoms, les adverbes et adjectifs, etc.) dans le but de réduire considérablement la taille de l'index.

- ***Normalisation :*** Consiste à représenter les variantes d'un mot, sous une forme unique appelée lemme ou racine.

- ***Pondération des termes :*** Consiste à donner un poids w_{ij} pour chaque terme selon le degré de sa représentativité dans le *Tweet*.

La contribution de (SEKOUR 2019) dans cette première étape, consiste à modifier cette fonction de pondération afin de prendre en compte les métadonnées relatives à chaque *Tweet*. Pour cela, il propose d'utiliser la fonction usuelle $TFIDF$, à laquelle il va intégrer le concept social de chaque *Tweet*.
Cette fonction est donnée comme suit :

$$\alpha\ (TF{*}IDF) + (1 - \alpha) \cdot (t)$$

Avec (t) représente un poids indépendant des termes relatifs aux données sociales de chaque *Tweet*. Cette mesure peut être exprimée comme suit :

$$\boldsymbol{P}(t) = \begin{cases} \left(\dfrac{N_{rt} + N_j + N_c}{N_f}\right) si\ N_f > 1 \\ 0\ sinon \end{cases}$$

Où :

- TF : fréquence du terme dans le *Tweet*(TermFrequency).
- IDF : fréquence du document inverse.
- N_{rt} : Nombre de *Retweets* associé au *Tweet*.
- N_j: Nombre de mentions j'aime.
- N_c: Nombre de commentaires.
- N_f: Nombre d'abonnés de la personne qui a partagé le *Tweet*.

α, et γ sont des poids, tel que $0.5 < \alpha < 1$ *et* $1 < \gamma < 3$. Le paramètre α est utilisé pour donner plus d'importance à la pondération textuelle, tandis que γ est utilisé pour favoriser les *retweets.* Les valeurs exactes de ces deux paramètres sont fixées de manière expérimentale.

Le principe de cette formule est de calculer le poids d'un terme en utilisant le$TFIDF$ standard au quel on rajoute une pertinence sociale. (SEKOUR 2019) propose de calculer cette pertinence de *Tweet* par le rapport entre le nombre de j'aimes, de commentaires et de *Retweets* regroupés, sur le nombre d'abonnés de la personne qui a partagé ce *Tweet*. En effet, (SEKOUR 2019) suppose qu'un *Tweet* est pertinent si la majorité des abonnés de cet utilisateur ont réagi au *Tweet* soit par un j'aime, un commentaire ou encore un *Retweet*. La structure algorithmique de cette approche peut être représentée comme suit:

ALGORITHME D'INDEXATION DES TWEETS :

```
BEGIN
    FOR EACH TWEET IN TWEETS :
        FOR EACH WORD IN TWEET :
            IF (WORD IN STOPLIST) THEN
                DELETE(WORD)
            ELSE
                PORTER_NORMALIZE(WORD)
                IF (N_f = 0) THEN
                    ω = 0.6 ( TF × IDF)
                ELSE
                    ω = 0.6 (TF × IDF) + (1 − 0.6)((γN_rt + N_j + N_c) / N_f)
                ENDIF
            ENDIF
        ENDFOR
    ENDFOR
END
```

Figure 10: Algorithme d'indexation de tweets (SEKOUR 2019)

II.4. l'expansion de la requête :

(SEKOUR 2019) propose d'utiliser Twitter comme une collection externe pour étendre la requête initiale, puis d'utiliser cette nouvelle requête basée sur la recherche classique afin de répondre aux besoins de l'utilisateur. La reformulation de la requête permet de construire une nouvelle requête, elle est souvent opérée par ajout et/ou réévaluation des poids des termes de la requête initiale.
(SEKOUR 2019) utilise l'injection de pertinence, particulièrement l'algorithme de RelevanceFeedback de Rocchio[25] (Rocchio 1971).

Une variante de cette technique a été proposée, nommée pseudo relevance feedback, qui s'intéresse à rapprocher la requête des documents pertinents sans prendre en compte les documents non pertinents. Sa formule est exprimée comme suit:

$$Q_1 = \alpha Q_0 + \beta \frac{1}{|D_p|} \sum_{D_j \in D_p} D_j$$

L'approche que [Sekour, 2019] propose consiste à faire une recherche dans la collection de *Twitter* en utilisant la première étape pour indexer les *Tweets* tenons compte de leurs métadonnées sociales. Puis, d'en sélectionner les premiers *Tweets* les plus pertinents vis-à-vis de la requête nommés D_p (documents pertinents), qu'il va utiliser dans l'algorithme de Rocchio de pseudo relevance Feedback pour rapprocher la requête le plus possible vers les *Tweet*s pertinents. Cette formule est donnée comme suit :

$$Q_1 = \alpha Q_0 + \beta \frac{1}{|T_p|} \sum_{T_j \in T_p} T_j$$

Où :

- Q_0 représente la requête initiale.
- T_p représente *les Tweet*s jugés pertinents.
- $|T_p|$ représente le nombre de *Tweets* pertinents (exemple $|T_p| = 10$).
- α, β sont des paramètres de la reformulation ($\alpha = 1$, $\beta = 0.5$).

On aura comme sortie, une nouvelle requête Q_1 beaucoup plus représentative du besoin de l'utilisateur, constituée des termes les plus pertinents extraits des *Tweets*.

[25] https://nlp.stanford.edu/IR-book/pdf/09expand.pdf

Cette nouvelle requête sera ensuite utilisée pour faire une recherche standard en utilisant la RI classique dans une autre collection hors que *Twitter*.
L'algorithme général de cette approche peut être représenté comme suit :

```
BEGIN
    FOR EACH WORD IN QUERY :
        ω = TF × IDF
    ENDFOR
    VICTOR_SPACE (QUERY)
    FOR EACH TWEET IN TWEETS :
        FOR EACH WORD IN TWEET :
            IF (WORD IN STOPLIST) THEN
                DELETE(WORD)
            ELSE
                PORTER_NORMALIZE(WORD)
                IF (N_f = 0) THEN
                    ω = 0.6 (TF × IDF)
                ELSE
                    ω = 0.6 (TF × IDF) + (1 − 0.6)(γN_rt + N_f + N_c) / N_f )
                ENDIF
            ENDIF
        ENDFOR
        VICTOR_SPACE (TWEET)
        CALCULATE_RSV (QUERY, TWEET)
        SORT_DESC_RSV (QUERY, TWEET)
    ENDFOR
    T_p := TOP_N_RSV (QUERY, TWEET)
    FOR EACH TWEET IN T_p :
        VICTOR_SPACE (Query2) = VICTOR_SPACE (query) + 0.5 × 1/N (VICTOR_SPACE (TWEET))
    ENDFOR
    FOR EACH DOCUMENT IN COLLECTION
        FOR EACH WORD IN DOCUMENT
            ω = TF × IDF
        ENDFOR
        VICTOR_SPACE (DOCUMENT)
        CALCULATE_RSV (QUERY2, DOCUMENT)
    ENDFOR
END
```

Figure 11: Algorithme général (SEKOUR 2019)

III. Approche proposée par Nabil LARBI :

Cette approche se base sur le reclassement (Re-Ranking) des documents retournés lors de la phase de la recherche classique en utilisant les signaux sociaux comme facteurs de pertinence. La figure 12, illustre l'architecture de cette approche.

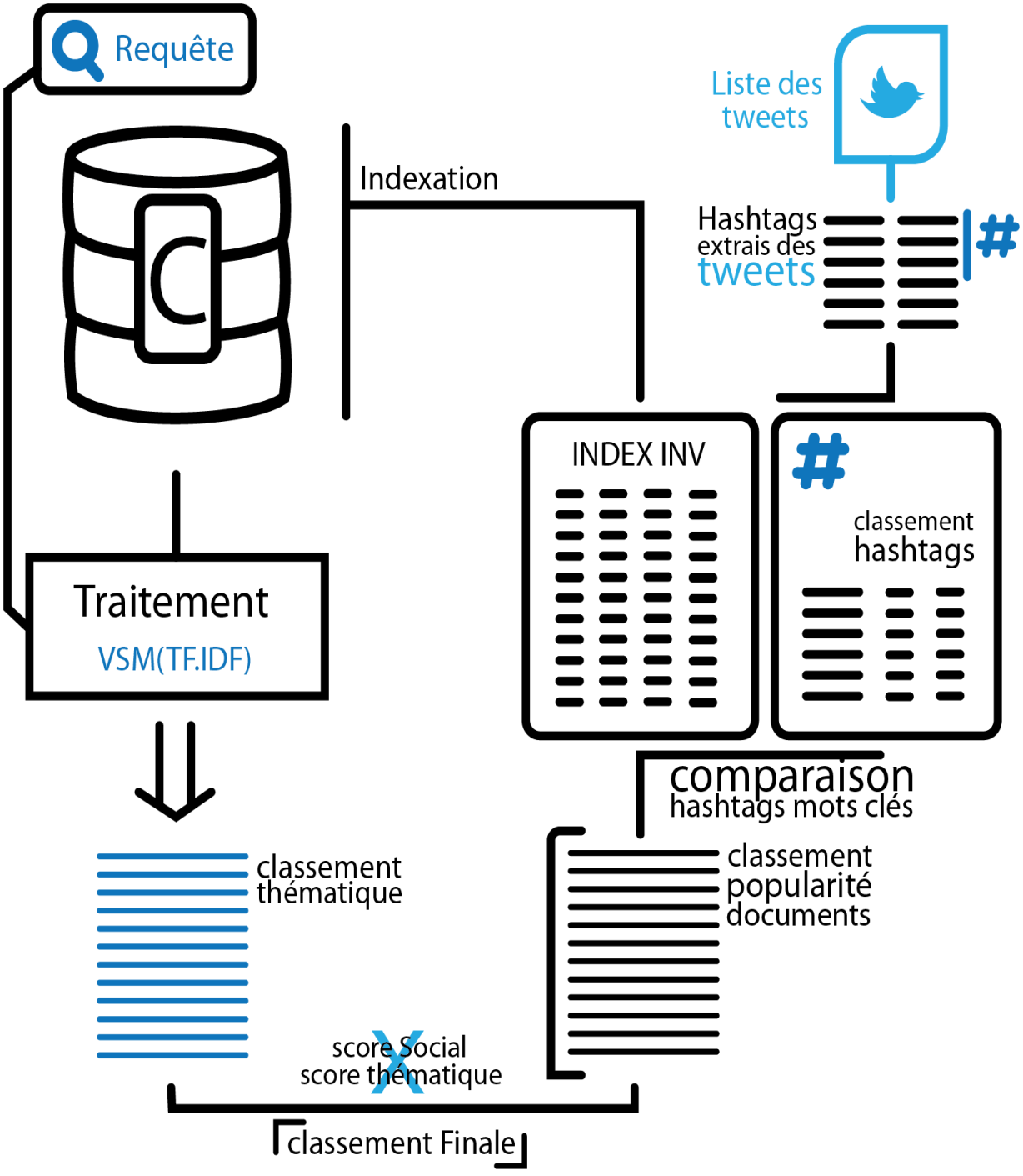

Figure 12:Architecture proposée (LARBI 2019)

L'architecture présentée dans cette démarche, classe les résultats en deux phases :

➢ ***Phase I*** : concerne la recherche d'information classique et se passe comme suit :

- Indexation du corpus de documents et la génération d'index inversé,
- Analyse et indexation de la requête saisie par l'utilisateur,
- Application du modèle vectoriel pour l'appariement requête- document,
- Affichage des résultats de recherche sous forme de liste de documents classés par ordre descendant de pertinence textuelle (score thématique des documents).

➢ ***Phase II*** : cette phase s'applique sur la RI sociale en utilisant une collection de tweets, et se passe comme suit :

- Extraction des hashtags et des signaux sociaux que contiennent les tweets,
- L'élaboration d'un classement de hashtag en calculant la popularité de chaque hashtag.
- Comparaison entre la liste des hashtags et les termes d'index inversé de la phase I.
- Classement des documents issu de la phase thématique par ordre de popularité en se basant sur les scores des hashtags qui les citent (score sociale des documents).

A l'issu de cette étape, les deux phases sont rassemblées à fin d'élaborer un classement final.

III.1. Notations :

L'information social que (LARBI 2019) a exploité dans cette approche est sous forme d'un quintuplé <U, H, T, A_t R> où U, H, T,A_t et R représentent respectivement : Utilisateurs, Hashtags, Tweets, Actions sociales et Ressources.

- ***Utilisateurs U*** : sont les acteurs interagissant entre eux dans le réseau social Twitter.
- ***Hashtags H*** : est un paramètre principale dans notre recherche, le score social dépendra principalement de la popularité des hashtags.
- ***Tweets T :*** sont les composantes principales du réseau social Twitter, ils seront utilisés pour déterminer la force d'un hashtag.
- ***Ressources R :*** c'est un corpus composé de n documents $R=\{D_1,D_2, \ldots,D_n\}$, chaque document sera composé de m mots clés $D_{wi}=\{w_1,w_{2,\ldots,}w_m\}$ avec w_i=poids d'un terme, et k hashtags qui détermineront le score social du document D_i.

III.2. Préliminaires :

Dans cette approche (LARBI 2019) indexe d'abord la collection de document R et détermine la pertinence thématique de ces ressources par rapport à une requête Q, de ce fait il a utilisé le modèle vectoriel (VSM).
La correspondance entre le vecteur ressource D et le vecteur requête Q est estimée selon la ***mesure cosinus*** où :

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

Avec 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***

Chaque terme des documents et de la requête est associé par un nombre w qui indique sa pondération, qui se calculera avec la fonction suivante :

$$w_{ij} = \begin{cases} (1 + log f_{i,j}) * log \frac{N}{n_i} \; si \; f_{i,j} > 0 \\ sinon \; 0 \end{cases}$$

Afin de calculer l'appariement de deux vecteurs D_i , Q, un score thématique sera calculé avec la mesure ***Cosinus*** en utilisant la fonction précédente de ***TFIDF***.

$$score_{thématique} = \text{sim}\,(D_j\,,\,Q) = \frac{\sum_{i=1}^{t} w_{i,j} * w_{i,q}}{\sqrt{\sum_{i=1}^{t} w_{i,j}^2} \; * \; \sqrt{\sum_{i=1}^{t} w_{i,q}^2}}$$

A la fin de cette étape, les résultats de l'appariement requête/documents seront retournés.

III.3. Traitement des données sociales: reclassement de résultats (Re-ranking)

Cette contribution consiste en la proposition d'une fonction de calcul du score sociale d'un document, basé sur le score des hashtags qui le référencient, en tenant compte des signaux sociaux des tweets où les hashtags apparaissent (LARBI 2019).

III.3.1. Fonction de calcul du score sociale :

Dans ce travail (LARBI 2019) associe à chaque tweet T_i un score basé sur les actions sociales A_s tel que S = Nbr de retweet/like/commentaire, nous avons donc :

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

En proposant des paramètres afin de calculer le score d'un tweet comme suit :

$$S_T = \gamma N_{re} + \delta N_l + \varepsilon N_c \qquad \ldots\ '1'$$

Où $\gamma, \delta \; et \; \varepsilon$ sont les pondérations respectivement des paramètres $\{A_{re},\ A_l,\ A_c\}$, pour valoriser les retweets (LARBI 2019) pondère leurs paramètres de manière supérieure que le nombre des likes et de commentaire tel que: $\gamma + \delta + \varepsilon = 1$ et $\gamma > \{\delta, \varepsilon\}$. Ainsi l'auteur pose $\gamma = 3/5$ et $\delta = \varepsilon = 1/5$.

(LARBI 2019) suggère d'ajouter la fonction de normalisation :

$$\boldsymbol{S_{T_n} = Ln\,(S_T)} \text{ ..... '2'}$$

Le calcul du score de l'Hashtag $\boldsymbol{H_i}$ qui est basé sur le score des tweets qui le contiennent $T_{Hi} = \{T_{1,i}; T_{2,i}; \dots; T_{n,i}\}$, de la manière suivante :

$$\boldsymbol{S_{H_i} = \sum_{j=1}^{n} S_{T_{j,i}}} \text{..... '3'}$$

Afin d'avoir les scores de tous les hashtags (LARBI 2019) propose un classement sous forme de tendances « ***The Trends***[26] ». Par la suite il compare les hashtags syntaxiquement avec les termes hautement pondérés résultant de ***l'index inversé*** des ***25 documents de la phase 1***, s'il y a correspondance le score de l'hashtag $\boldsymbol{S_{H_i}}$ sera immédiatement incrémenter au score social du document $\boldsymbol{D_i}$ d'où est issu le terme, comme suit :

$$\boldsymbol{S_{D_i}} = \begin{cases} \left(\boldsymbol{S_{H_{1,i}} + S_{H_{2,i}} + \cdots + S_{H_{n,i}}}\right) + \boldsymbol{S_{thématique_D}} \; \boldsymbol{Si} \text{ hashtag} \in \boldsymbol{D} \\ \boldsymbol{0 \; sinon} \end{cases} \text{ ..... '4'}$$

Où $\boldsymbol{S_{thématique_D}}$ est le score thématique d'un document D.

Le score social d'un document est calculé comme suit :

$$\boldsymbol{S_{sociale_{Di}} = (1 - e^{-\lambda \log\left(S_{D_i}\right)})} \text{.... '5'}$$

(LARBI 2019) suggère d'ajouter la fonction de normalisation :

$$\boldsymbol{Y = (1 - e^{-\lambda \ln(X)})}$$

Pour faire tendre le score social $\boldsymbol{S_{D_i}}$ vers 1 ainsi :

Plus le score social d'un document $\boldsymbol{S_{D_i}}$ ('4') augmente plus il tendra vers la valeur maximal 1.il ajoute le logarithme népérien **ln** pour alléger le score du document initial $\boldsymbol{S_{D_i}}$, puis il fait une corrélation **λ** de type $\frac{1}{n}$ avec **n** $\in \mathbb{N}^+$ pour garantir un début lent pour la fonction ('5') et pour que les hashtags impopulaires aient un faible effet sur le reclassement des résultats (LARBI 2019).

[26] Un trend sur Twitter fait référence à un sujet axé sur le hashtag qui est immédiatement populaire à un moment donné.

III.3. 2. Calcul du score global (social et thématique) :

Afin de calculer le score global (LARBI 2019) va joindre le score thématique (de la 1ere phase) avec le score social (2eme phase) :

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

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

Pour cela il utilise une fonction de type :

$$S_{global_{Di}} = \alpha S_{thématique_{Di}} * (1 - \alpha) S_{social_{Di}}$$

(LARBI 2019) a 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.

(Bouadjenek 2013) avec leur approche basée sur le reclassement de résultats personnalisé estiment que leurs meilleurs résultats ont été obtenus avec $0.6 \leq \alpha \leq 0.8$.

VI. Conclusion :

Dans ce chapitre, nous avons présenté deux modèles de recherche d'information basés sur les signaux sociaux et leurs propriétés sociales. Afin de montrer la contribution de ces signaux sociaux dans la pertinence des documents, nous avons présenté deux approches qui exploitent les signaux sociaux de Twitter. La première s'intéresse à l'indexation des *Tweet*s pour les utiliser dans la reformulation de la requête et la deuxième propose une fonction de reclassement basée sur la pertinence sociale.

Dans le chapitre suivant nous ferons une implémentation, évaluation et comparaison entre ces deux approches.

Chapitre 04

Implémentation, évaluation et comparaison des deux approches étudiées

I. Introduction :

La recherche d'information sociale combine à la fois la pertinence thématique et la pertinence sociale d'un document dans le but d'avoir un meilleur ordonnancement des résultats de recherche.

Nous nous focalisons dans ce travail, sur l'exploitation des signaux sociaux de *Twitter,* pour améliorer le processus de la RI. Nous débutons ce chapitre par présenter l'environnement de travail que nous avons utilisé pour implémenter les deux approches étudiées, nous présentons par la suite les différentes expérimentations réalisées sur ces dernières et nous finalisons par la comparaison des résultats obtenus par les deux approches.

II. Implémentations :

Afin d'améliorer le processus thématique de la RI, nous avons utilisé le contenu social de twitter pour :

- L'expansion de la requête initiale, proposée dans la première approche par (SEKOUR 2019).
- Le reclassement des résultats initiaux, proposée dans la seconde approche par (LARBI 2019).

- ***Approche de SEKOUR*** : Nous avons développé des programmes écrits en :
  - *Langage Python 3.8.3* : Python est le langage de programmation open source créé par Guido van Rossum en 1991. Ce langage s'est propulsé en tête d'analyse de données et dans le domaine du développement de logiciels.
  - *IDLE (python)* : IDLE est un environnement de développement intégré pour le langage Python.
- ***Approche de LARBI*** : dans cette dernière nous avons utilisé :
  - *Le langage Java* : Java est un langage de programmation orienté objet créé par James Gosling et Patrick Naughton, en 1995.
  - *Eclipse IDE* : Eclipse est un environnement de développement (IDE) historiquement destiné au langage Java.

III. Expérimentations :

Dans les expérimentations réalisées nous avons utilisé une collection de test composée :

- d'un corpus documentaire composé de 100 documents textuels, extraits de Google, écrits en Anglais et qui se rapportent aux thèmes suivants : marriage, Irak, petrole, covid, donald trump, donal english, nature,

animaux, virus, election, coronavirus,... et d'autres pris au hasard, comme l'illustre la figure 13.

- ❖ d'un ensemble de 317 tweets avec leurs signaux sociaux, extraits de twitter en utilisant Twitter Archiver[27], comme le montre la figure 14.
- ❖ de dix requêtes, ce que montre la figure 15.
- ❖ Pour les jugements de pertinence, nous avons fait examiner le corpus documentaires à 10 camarades, et ces derniers ont jugé tous les documents qui répondent aux 10 requêtes. Au final, nous avons élaboré les jugements finaux, un extrait de ces jugements est illustré dans la figure 16.

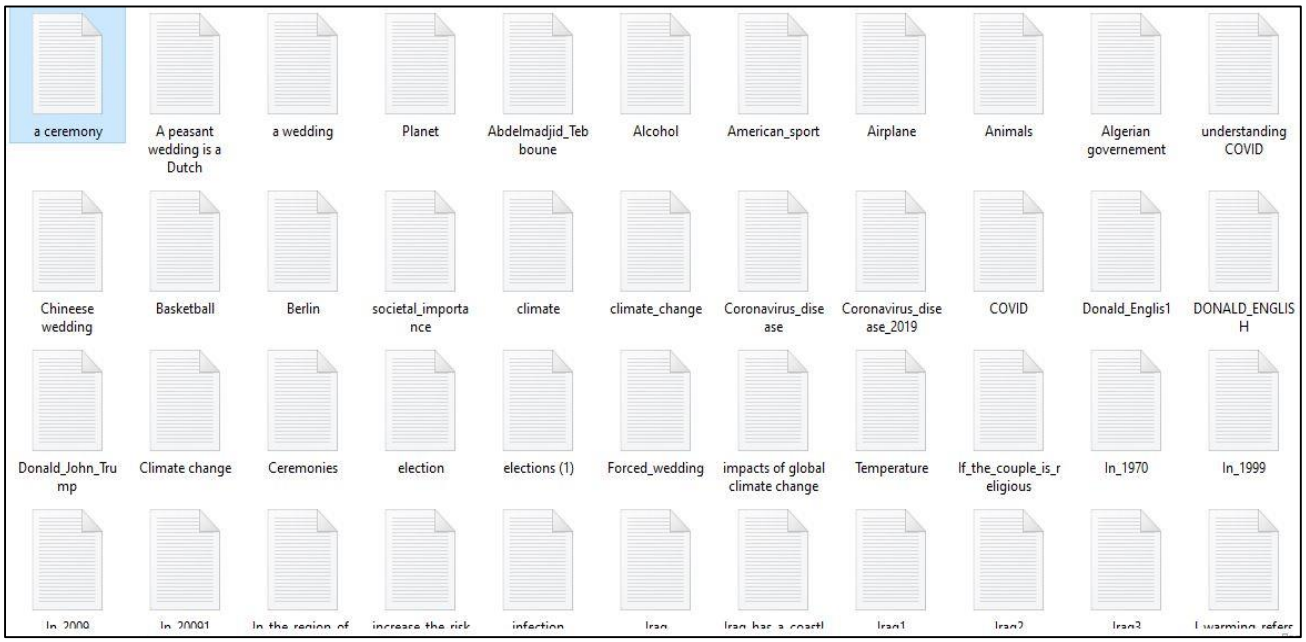

Figure 13: Extrait de la collection de documents.

Date	User_name	Text	ID	Retweets	Favorites	Followers	Follows	Comments	Hashtgs
27/09/2020 15:11	@Frederick987	All that #Trump has done over	1310205345600602113	33	4	13022	13164	51	1
27/09/2020 15:11	@scotnews_edits	#Covid has reduced Scottish i	1310205343570563072	5	0	446	81	4	1
27/09/2020 15:11	@LondonDairy	Hello. Given the current #CO\	1310205341750243328	3	0	2105	9	4	1
27/09/2020 15:11	@DavidDo07294382	can make you want to throw i	1310205340710043648	1	0	2	40	0	0
27/09/2020 15:11	@Buckwheatie1	#President needs to say THIS	1310205340370305031	2	0	56	294	0	2
27/09/2020 15:11	@ohiomary	Older poll workers dropped ou	1310205340105965569	0	0	1424	597	1307	0
27/09/2020 15:11	@HenshinJosh	Bring back Street Pass as the	1310205338260512768	0	0	101	383	0	0
27/09/2020 15:11	@sandrabarbour2	The Democrats released this	1310205337929224193	0	0	62	128	0	3
27/09/2020 15:11	@dad1337	Hey Kentucky how come #mo	1310205337895604225	32	0	9	2	0	1
27/09/2020 15:11	@Fo13549948	#covid in locker room?	1310205337547415552	0	0	5	154	0	1
27/09/2020 15:11	@MatthewCharle3	The only lie going on right now	1310205337128050688	15	0	136	1096	0	1
27/09/2020 15:11	@DrEricDing	3) Even before Tulsa MAGA r	1310205336066969600	123	56	266966	5872	3956	0
27/09/2020 15:11	@AmpinityNews	#Coronavirus update: Immunity	1310205335710445568	4	0	87	211	0	3
27/09/2020 15:11	@RemaNagarajan	Please keep that in mind	1310205335701827586	0	1	7800	40	91	0
27/09/2020 15:11	@Lisbeth2026	It is being discovered that fata	1310205334766510083	65	0	26	171	0	0
27/09/2020 15:11	@EdwardGerwer	An angry woman wrecked th	1310205334498275329	0	0	172	190	16	1
27/09/2020 15:11	@Walshton_21	Sad that people will risk contr	1310205334422720515	2	11	8703	6748	66	0
27/09/2020 15:11	@disgust_me	#Covid wasn't enough for Wr	1310205333986476032	3	0	108	157	2	1
27/09/2020 15:11	@amBackoff	I guess you didn't actually rea	1310205333751697410	6	0	163	916	1	1
27/09/2020 15:11	@lisavdsluijss	This bitch doesn't have Coron	1310205332317188096	0	0	14	77	0	0
27/09/2020 15:11	@SteenagekidDad	you cant atop the #virus... all t	1310205332166189057	0	0	369	1190	0	1
27/09/2020 15:11	@crykidkrow	but i'm pretty sure if they make	1310205331188994053	1	0	710	635	8	0
27/09/2020 15:11	@kairoscanada	#CreativityofLove: COVID-19	1310205329356075008	0	2	7264	1371	227	1

Figure 14:Extrait de la collection de tweets.

[27] **Twitter Archiver** est une extension disponible pour Google Spreadsheets. L'application enregistre le contenu Twitter dans une feuille de calcul Google. Pour cela, il doit avoir accès à un compte Google et à un compte Twitter.

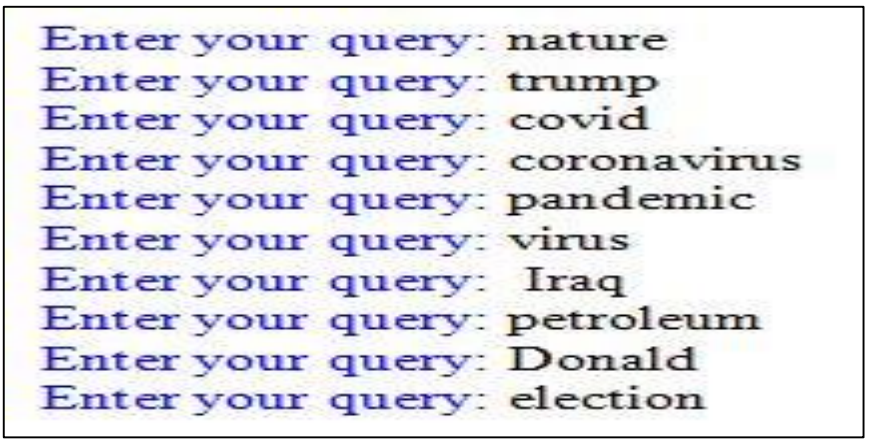

Figure 15: Ensemble de requêtes.

Jugements de pertinence finaux pour les requêtes:

id_docs	Requêtes/ Corpus documentaire	virus	nature	trump	covid	coronavirus	pandemic	iraq	election	donald	petroleum
0	Wedding traditions vary widely. Often there										
1	The two families prepare a piece of entertair										
2	It is forbidden in Islam for parents or anyone										
3	A planet is an astronomical body orbiting a st										
4	On 12 December 2019, Tebboune was electec										
5	Reproducing contemporary mass loss accurat										
6	American sport, culture and, many would say										
7	An **airplane** or **aeroplane** is a powered										
8	**Animals** are multicellular that form the biolo										
9	The Algerian government has started to ea				*						
10	As our understanding of COVID-19 has develc				*						
11	In the early 1970, Saddam nationalized oil an										
12	the tea ceremony is the equivalent of an excl										
13	**Basketball**, colloquially referred to as **hoops**,										
14	Beauty in nature has historically been a preva		*								
15	**Coronavirus disease** is an infectious disease					*					
16	You can be infected by breathing in the virus	*			*						
17	Crude oil and crude oil products are mixtures										
18	Democratic challengers from Arizona to Soutl										
19	**Donald English**, 1930-1998. After a degree in l									*	
20	DONALD ENGLISH was a Methodist minister o									*	

Figure 16: Extrait des jugements de pertinence finaux.

➢ Métriques d'évaluation utilisées :

Afin d'évaluer les requêtes, nous avons utilisé les métriques suivantes :

❖ **La précision :**

$$Précision = \frac{|DPR|}{|DR|}$$

- |DPR| : représente le nombre de documents pertinents retournés par le système.
- |DR| : représente le nombre de documents retournés par le système.

❖ **La précision moyenne AVG:**

AVG(P) représente la précision moyenne et se calcule en moyennant les valeurs de précision obtenues après chaque document pertinent observé.

❖ **La moyenne des précisions moyennes MAP :**

$$MAP = \frac{\sum_{q \in Q} P_{moy}(q)}{|Q|}$$

- Q étant l'ensemble des requêtes.
- (P_{moy}) est la moyenne des précisions.

❖ **La précision au rang :**

$$P@R = \frac{|DPR|}{R}$$

- R étant le rang.

III. 1. Approche d'expansion de requête proposée par (SEKOUR 2019):

III. 1. 1. Indexation des tweets :

Les tweets sont indexés en tenant compte de leur contenu textuel et les métadonnées sociales (j'aime, commentaires, retweets).

Le processus d'indexation du contenu des tweets a suivi les étapes classiques (extraction des tokens, élimination des mots vides et la normalisation).

Pour la pondération, une fonction TF.IDF a été utilisée, à laquelle a été intégrée la pertinence sociale (retweets, commentaires, followers, j'aime) relative à chaque tweet, comme l'illustre le script de la figure 17 :

```
# Pondération des termes des tweets
def computeTFIDFPoids(tfBow, idfs, Nrt, Nj, Nc, Nf, y, alpha):
    w={}
    for word, val in tfBow.items():
        tfidf=val*idfs[word]
        if Nf==0:
            w[word]=0.6*tfidf
        else:
            w[word]=alpha*tfidf+(1-0.6)*((y*Nrt+Nj+Nc)/float(Nf))

    return w
```

Figure 17: Script de pondération des tweets.

- α et Y sont des paramètres tel que : 0,5<α<1 et 1<Y<3. Les valeurs exactes de ces deux paramètres ont été fixées de manière expérimentale (SEKOUR 2019).

III. 1. 2. Expansion de la requête :

A l'issue de la phase d'indexation des tweets, nous avons sélectionné les 100 premiers tweets pertinents vis-à-vis de la requête initiale, en utilisant le script de la figure18, afin d'étendre la requête initiale avec l'algorithme de Rocchio de pseudo relevance feedback, montré dans la figure 19.

```
#affiche la sim entre la requête initiale et les tweets par ordre top N
sim=pd.DataFrame(similarite,columns= ['Tweet','Similarity'])
sim.sort_values(by=['Similarity'], inplace=True, ascending=False)
b=sim.nlargest(100, 'Similarity')['Tweet']
```

Figure 18: Script de sélection des tweets pertinents.

```
#Expansion de la requete
for val in newDataFrame:
    #print(newDataFrame)
    somme=0
    Q0=newDataFrame[val][0]
    data_top = newDataFrame.head()
    for val2 in data_top.index:
        somme=somme+newDataFrame[val][val2]
    resultat=(val,(alpha*Q0)+(beta*somme/(len(TweetPertinent))))
    res.append(resultat)
```

Figure 19: Script de l'expansion de la requête.

Nous sélectionnons par la suite les trois premiers termes ayant les valeurs de poids les plus élevées, pour former la requête étendue.

- Pour fixer le nombre de tweets pertinents à utiliser pour l'expansion de la requête, nous avons réalisé des tests préliminaires sur l'ensemble de la collection de tweets, en variant le nombre de tweets (25, 50, 100, 150 et 200). Les meilleurs résultats sont obtenus lorsque |TweetPertinent|=100.
- Pour définir le nombre de termes pertinents à utiliser pour étendre la requête, nous avons réalisé des tests en prenant entre 2 à 10 termes. Nous avons constaté que plus nous élargissons le nombre, plus les termes ajoutés s'éloignent de la requête initiale et ne sont pas vraiment significatifs.
  Les meilleurs résultats sont obtenus lorsque |termesPertinents|=3.
- Les paramètres α et ß sont des paramètres de la reformulation, choisis en fonction de l'importance que l'on souhaite donner à la requête initiale, Rocchio a proposé α=1 et (SEKOUR 2019) a utilisé ß=0,5.

Au final, le système retourne les 20 premiers documents pertinents par rapport à la requête étendue, comme l'illustre le script de la figure 20.

```
#affiche la sim entre la requête etendue et les documents
simf=pd.DataFrame(similaritef,columns= ['Doc','Similarity'])
simf.sort_values(by=['Similarity'], inplace=True, ascending=False)
bf=simf.nlargest(20, 'Similarity')['Doc']
```

Figure 20: Script du calcul de similarité entre requête étendue et les documents.

III. 1. 3. Résultats expérimentaux de la première approche :

Nous allons présenter les résultats des expérimentations requête par requête.

- Pour la requête initiale « nature » le déroulement a donné les résultats suivants :

```
Enter your query:nature
```

Figure 21: Requête initiale.

Après la phase d'indexation, un extrait de la pondération des termes dans les tweets est présenté dans la figure 22:

	odd	tbrc	turn	...	thursday	mennie	forward
0	0.002703	0.002703	0.002703	...	0.002703	0.002703	0.002703
1	0.008072	0.008072	0.008072	...	0.008072	0.008072	0.008072
2	0.001330	0.001330	0.001330	...	0.001330	0.001330	0.001330
3	0.200000	0.200000	0.200000	...	0.200000	0.200000	0.200000
4	0.014286	0.014286	0.014286	...	0.014286	0.014286	0.014286
..	...	...	...	...	...	...	...
312	0.255351	0.255351	0.529417	...	0.255351	0.255351	0.255351
313	0.136471	0.136471	0.319182	...	0.136471	0.136471	0.136471
314	0.138214	0.138214	0.335541	...	0.138214	0.138214	0.138214
315	0.710622	0.710622	1.697258	...	0.710622	0.710622	0.710622
316	0.123405	0.123405	0.299590	...	0.123405	0.123405	0.123405

[317 rows x 2898 columns]

Figure 22: Extrait de la pondération des termes dans les tweets.

Un extrait des résultats de la similarité entre la requête initiale et les tweets de la collection, est présenté comme le montre la figure 23 :

	Tweet	Similarity
0	0	0.002703
1	1	0.008072
2	2	0.001330
3	3	0.200000
4	4	0.014286
..	...	...
312	312	2.893113
313	313	1.894979
314	314	2.037402
315	315	10.206564
316	316	1.819109

[317 rows x 2 columns]

Figure 23: Extrait des résultats de similarité entre tweets et requête initiale.

A partir de ces résultats, nous avons récupéré l'index des 100 premiers tweets ayant le plus grand score afin de les utiliser pour extraire les trois termes de plus grands poids, qui seront les termes candidats pour l'expansion.

	Tweet	Similarity
255	255	6.086912
315	315	5.753275
242	242	4.793787
278	278	4.268838
299	299	4.217482
..	...	...
275	275	0.457917
215	215	0.453028
208	208	0.452776
218	218	0.449526
231	231	0.447947

[100 rows x 2 columns]

Figure 25: Extrait des tweets pertinents.

	terme	weight
1730	nature	1.125601
1950	covid	0.364908
1454	virus	0.273624

Figure 24: Termes pertinents.

La nouvelle requête est :

```
New query is: nature covid virus
```

Figure 26: Nouvelle requête.

Nous avons calculé la similarité entre cette nouvelle requête et le corpus documentaire (100 documents).

L'index des 20 premiers documents (représentés par leurs identificateurs) retournés par score de similarité décroissant sont :

Documents	Similarity
82	11.321667
95	10.369401
69	8.506966
83	8.491250
81	8.491250
78	8.324069
79	8.324069
92	7.880860
97	6.912934
98	6.912934
84	6.793000
93	6.682567
91	6.452200
73	5.622538
75	5.622538
77	5.117414
76	5.117414
71	5.104180
67	5.104180
68	5.104180

Figure 27: Résultats retournés après expansion de la requête.

❖ Evaluation de la requête « nature » :

Nous avons établi l'évaluation par rapport à la requête initiale « nature » et par rapport à la requête étendue « nature, covid, virus ». Les résultats obtenus sont présentés dans le tableau 2 :

Docs_Init	Precision	Docs_Eten	Precision
82	0	82	0
81	0	95	0,5
83	0	69	0,33333333
95	0	83	0,25
79	0	81	0,2
78	0,16666667	78	0,33333333
69	0,14285714	79	0,28571429
84	0,125	92	0,375
92	0,11111111	97	0,33333333
91	0,1	98	0,3
93	0,09090909	84	0,27272727
97	0,08333333	93	0,33333333
98	0,07692308	91	0,38461538
72	0,14285714	73	0,35714286
75	0,13333333	75	0,33333333
77	0,125	77	0,3125
76	0,17647059	76	0,35294118
71	0,16666667	71	0,33333333
68	0,15789474	67	0,31578947
67	0,15	68	0,3

Tableau 2: Précision de la requête « nature ».

- Docs_Init : sont les documents retournés par le système par rapport à la requête initiale « nature ».
- Docs_Eten : sont les documents retournés par le système par rapport à la requête étendue « nature, covid, virus ».
- Les documents retournés en couleur noir sont les documents jugés non pertinents.
- Les documents retournés en couleur rouge sont les documents jugés pertinents.

Le tableau 3, résume les précisions de la requête « nature » aux rangs 5, 10 et 20 ainsi que l'AVG(P).

NATURE	AVG	P@5	P@10	p@20
Résultats_thématique	0,16333333	0	0,1	0,15
Résultats_étendue	0,37833333	0,2	0,3	0,3

Tableau 3: Evaluation de la requête « nature ».

Nous remarquons que la précision au rang 5 de la requête thématique est nulle, alors que celle de la requête étendue est plus élevée. Aux rangs 10 et 20 la précision de la requête thématique remonte mais elle reste en dessous de la précision de la requête étendue. La précision moyenne de la requête étendue est nettement supérieure à celle de la requête thématique.

❖ Evaluation de la requête « virus » :

Le tableau 4 présente les résultats de précision de la requête « virus » :

Docs_Init	Precision	Docs_Etendue	Precision
95	1	94	1
98	0,5	97	0,5
97	0,33333333	96	0,33333333
93	0,5	91	0,5
92	0,6	92	0,6
94	0,66666667	81	0,5
96	0,57142857	90	0,57142857
91	0,625	93	0,5
99	0,55555556	68	0,44444444
90	0,6	95	0,5
69	0,54545455	80	0,45454545
82	0,5	82	0,41666667
18	0,46153846	77	0,38461538
23	0,42857143	78	0,35714286
19	0,4	83	0,33333333
78	0,375	89	0,3125
79	0,35294118	99	0,35294118
81	0,33333333	61	0,33333333
83	0,31578947	75	0,31578947
37	0,3	73	0,3

Tableau 4: Précision de la requête « virus ».

Le tableau 5 résume les précisions de la requête « virus » aux rangs 5, 10 et 20 ainsi que l'AVG(P).

VIRUS	AVG	P@5	P@10	p@20
Résultats_thématique	0,66666667	0,6	0,6	0,3
Résultats_étendue	0,58666667	0,6	0,5	0,3

Tableau 5: Evaluation de la requête « virus ».

Nous remarquons que la valeur de la précision au rang 5 est la même pour la requête thématique et la requête étendue, elle reste constante pour la thématique au rang 10 puis diminue au rang 20. Alors qu'elle diminue au rang 10 pour l'étendue puis encore au rang 20, pour être identique à la valeur de la thématique à ce même rang. La précision moyenne de la requête thématique est plus élevée que celle de la requête étendue.

❖ Evaluation de la requête « coronavirus » :

Le tableau 6 montre les résultats de précision de la requête « coronavirus »:

Docs_Initial	Precision	Docs_Eten	Precision
82	0	95	1
81	0	82	0,5
95	0	92	0,66666667
83	0	98	0,5
92	0	97	0,4
84	0	81	0,33333333
98	0	83	0,28571429
97	0	69	0,25
69	0	93	0,33333333
93	0	91	0,4
91	0	84	0,36363636
19	0	94	0,41666667
18	0	79	0,38461538
18	0	78	0,35714286
80	0,06666667	96	0,33333333
23	0,0625	71	0,3125
94	0,05882353	75	0,29411765
90	0,05555556	67	0,27777778
71	0,05263158	73	0,26315789
56	0,05	68	0,25

Tableau 6: Précision de la requête « coronavirus ».

Les résultats de précisions de la requête « coronavirus » aux rangs 5, 10 et 20 ainsi que l'AVG(P) sont présentés dans le tableau 7.

Coronavirus	AVG	P@5	P@10	p@20
Résultats_thématique	0,07	0	0	0,05
Résultats_étendue	0,564	0,4	0,4	0,25

Tableau 7: Evaluation de la requête « coronavirus ».

Nous constatons que les valeurs de précision de la requête initiale sont très faibles voir nulles pour les rangs 5 et 10 alors que celles de la requête étendue sont très importantes.
Le même constat est observé pour l'AVG.

❖ Evaluation de la requête « covid » :

Le tableau 8 illustre les résultats de précision de la requête « covid » :

Docs_Ini	Precision	Docs_Eten	Precision
95	0	95	1
82	0	82	0,5
69	0	92	0,66666667
92	0	69	0,5
80	0,2	98	0,4
83	0,16666667	97	0,33333333
78	0,14285714	93	0,42857143
91	0,125	91	0,5
93	0,11111111	83	0,44444444
97	0,1	81	0,4
98	0,09090909	79	0,36363636
79	0,08333333	78	0,33333333
84	0,07692308	94	0,38461538
75	0,07142857	84	0,35714286
68	0,06666667	68	0,33333333
60	0,125	67	0,3125
73	0,11764706	73	0,29411765
67	0,11111111	64	0,27777778
71	0,10526316	71	0,26315789
64	0,1	75	0,25

Tableau 8: Précision de la requête « covid ».

Le tableau 9 résume les précisions de la requête « covid » aux rangs 5, 10 et 20 ainsi que l'AVG(P).

covid	AVG	P@5	P@10	p@20
Résultats_thématique	0,165	0,2	0,1	0,1
Résultats_étendue	0,596	0,4	0,4	0,25

Tableau 9: Evaluation de a requête « covid ».

A partir du tableau 9, nous constatons que les valeurs de précision à tous les rangs ainsi que la valeur de la précision moyenne de la requête étendue sont plus élevées que celles de la requête initiale.

❖ Evaluation de la requête « donald » :

Le tableau 10 montre les résultats de précision de la requête « donald »:

Docs_Ini	Precision	Docs_Eten	Precision
82	0	95	1
69	0	82	0,5
97	0	69	0,33333333
79	0	92	0,5
81	0	97	0,4
83	0	98	0,33333333
78	0	83	0,28571429
92	0	81	0,25
45	0,11111111	93	0,33333333
75	0,1	78	0,3
73	0,09090909	79	0,27272727
50	0,08333333	91	0,33333333
48	0,07692308	84	0,30769231
56	0,07142857	94	0,35714286
71	0,06666667	75	0,33333333
61	0,0625	73	0,3125
64	0,05882353	67	0,29411765
67	0,05555556	68	0,27777778
49	0,05263158	71	0,26315789
68	0,05	61	0,25

Tableau 10: Précision de la requête « donald ».

Le tableau 11 résume les précisions de la requête « donald » aux rangs 5, 10 et 20 ainsi que l'AVG(P).

Donald	AVG	P@5	P@10	p@20
Résultats_thématique	0,11	0	0,1	0,05
Résultats_étendue	0,504	0,4	0,3	0,25

Tableau 11: Evaluation de la requête « donald ».

A partir du tableau 11, nous remarquons que la précision au rang 5 de la requête thématique est nulle, alors que celle de la requête étendue est nettement plus élevée. Au rang 10 la précision de la requête thématique remonte mais elle reste en dessous de la précision de la requête étendue. Au rang 20, la précision de la requête thématique est de nouveau faible comparativement à celle de la requête étendue. La précision moyenne de la requête étendue est nettement supérieure à celle de la requête thématique.

❖ Evaluation de la requête « election » :

Le tableau 12 montre les résultats de précision de la requête « election » :

Docs_Init	Precision	Docs_Eten	Precision
82	0	95	1
69	0	82	0,5
83	0	69	0,33333333
79	0	92	0,5
78	0	83	0,4
95	0	81	0,33333333
81	0	78	0,28571429
75	0	79	0,25
73	0	98	0,22222222
71	0	97	0,2
50	0	93	0,27272727
67	0	91	0,33333333
45	0,07692308	84	0,30769231
48	0,07142857	61	0,28571429
49	0,06666667	64	0,26666667
56	0,0625	67	0,25
64	0,05882353	75	0,23529412
68	0,05555556	68	0,22222222
61	0,05263158	73	0,21052632
92	0,05	71	0,2

Tableau 12: Précision de la requête « election ».

Le tableau 13 récapitule les précisions de la requête « election » aux rangs 5, 10 et 20 ainsi que l'AVG(P).

Election	AVG	P@5	P@10	p@20
Résultats_thématique	0,05	0	0	0,05
Résultats_étendue	0,525	0,4	0,4	0,2

Tableau 13: Evaluation de la requête « election ».

A partir du tableau 13, nous constatons que la précision aux rangs 5 et 10 de la requête thématique est nulle, alors que celle de la requête étendue est nettement plus élevée. Au rang 20 la précision de la requête thématique remonte mais elle reste faible comparativement à celle de la requête étendue. La précision moyenne de la requête étendue est nettement supérieure à celle de la requête thématique.

❖ Evaluation de la requête « iraq » :

Le tableau 14 illustre les résultats de précision de la requête « iraq »:

Docs_Init	Precision	Docs_Eten	Precision
82	0	82	0
81	0,5	95	0
83	0,33333333	81	0,33333333
95	0,25	83	0,25
79	0,2	79	0,2
78	0,16666667	78	0,16666667
84	0,14285714	92	0,14285714
92	0,125	84	0,125
69	0,11111111	69	0,11111111
91	0,1	91	0,1
93	0,09090909	93	0,09090909
97	0,08333333	97	0,08333333
98	0,07692308	98	0,07692308
75	0,07142857	75	0,07142857
73	0,13333333	73	0,13333333
76	0,125	76	0,125
77	0,11764706	77	0,11764706
74	0,16666667	74	0,16666667
71	0,21052632	71	0,21052632
94	0,2	94	0,2

Tableau 14: Précision de la requête « iraq ».

Le tableau 15 résume les précisions de la requête « iraq » aux rangs 5, 10 et 20 ainsi que l'AVG(P).

Iraq	AVG	P@5	P@10	p@20
Résultats_thématique	0,2525	0,2	0,1	0,2
Résultats_étendue	0,21	0,2	0,1	0,2

Tableau 15: Evaluation de la requête « iraq ».

A partir du tableau 15, nous constatons que les résultats de la requête initiale et la requête étendue sont égaux en ce qui concerne les mesures de P@R.

Concernant l'AVG, la valeur de précision moyenne de la requête initiale est plus élevée que celle de la requête étendue, car comme nous le voyons dans le tableau 14, les documents pertinents retournés par le système sont tous au même rang à l'exception du document 81 qui montre un écart d'une position qui fait cette différence.

❖ Evaluation de la requête « pandemic » :

Le tableau 16 montre les résultats de précision de la requête « pandemic »:

Docs_Init	Precision	Docs_Eten	Precision
82	0	95	1
81	0	82	0,5
95	0	92	0,66666667
83	0	98	0,5
84	0	97	0,4
92	0	83	0,33333333
98	0	81	0,28571429
97	0	93	0,375
93	0	91	0,44444444
91	0	69	0,4
94	0	84	0,36363636
69	0	94	0,41666667
90	0	79	0,38461538
87	0	78	0,35714286
96	0	96	0,33333333
79	0	90	0,375
78	0	68	0,35294118
89	0	71	0,33333333
86	0	67	0,31578947
80	0,05	73	0,3

Tableau 16: Précision de la requête « pandemic ».

Le tableau 17 résume les précisions de la requête « pandemic » aux rangs 5, 10 et 20 ainsi que l'AVG(P).

Pandemic	AVG	P@5	P@10	p@20
Résultats_thématique	0,05	0	0	0,05
Résultats_étendue	0,54833333	0,4	0,4	0,3

Tableau 17: Evaluation de la requête « pandemic ».

A partir du tableau 17, nous constatons que la précision aux rangs 5 et 10 de la requête thématique est nulle, alors que celle de la requête étendue est nettement plus élevée. Au rang 20 la précision de la requête thématique remonte mais elle reste faible comparativement à celle de la requête étendue. La précision moyenne de la requête étendue est nettement supérieure à celle de la requête thématique.

❖ Evaluation de la requête «trump » :

Le tableau 18 nous montre les résultats de précision de la requête « trump » :

Docs_Init	Precision	Docs_Eten	Precision
95	0	95	1
92	0	92	1
98	0	98	0,66666667
97	0	97	0,5
93	0	82	0,4
91	0	93	0,5
69	0	69	0,42857143
82	0	91	0,5
94	0	94	0,55555556
78	0	83	0,5
84	0,09090909	81	0,45454545
79	0,08333333	79	0,41666667
83	0,07692308	78	0,38461538
81	0,07142857	84	0,42857143
90	0,06666667	96	0,4
96	0,0625	90	0,4375
87	0,11764706	61	0,41176471
75	0,11111111	64	0,38888889
64	0,10526316	75	0,36842105
67	0,1	67	0,35

Tableau 18: Précision de la requête « trump ».

Le tableau 19 résume les précisions de la requête « trump » aux rangs 5, 10 et 20 ainsi que l'AVG(P).

TRUMP	AVG	P@5	P@10	p@20
Résultats_thématique	0,105	0	0	0,1
Résultats_étendue	0,63285714	0,4	0,5	0,35

Tableau 19: Evaluation de la requête « trump ».

A partir du tableau 19, nous constatons que la précision aux rangs 5 et 10 de la requête étendue est nettement plus élevée en comparaison avec la requête thématique qui est nulle. Au rang 20 la précision de la requête thématique remonte mais elle reste en dessous de la requête étendue. La précision moyenne de la requête étendue est nettement supérieure à celle de la requête thématique.

❖ Evaluation de la requête « Petroleum » :

Le tableau 20 illustre les résultats de précision de la requête « petroleum »:

Docs_Init	Precision	Docs_Eten	Precision
82	0	82	0
69	0,5	69	0,5
81	0,33333333	95	0,66666667
95	0,25	81	0,5
78	0,2	83	0,4
79	0,16666667	78	0,33333333
83	0,14285714	79	0,28571429
84	0,125	92	0,375
71	0,22222222	84	0,33333333
73	0,2	71	0,4
75	0,18181818	73	0,36363636
92	0,16666667	75	0,33333333
76	0,15384615	97	0,30769231
98	0,14285714	98	0,28571429
97	0,13333333	93	0,33333333
72	0,125	91	0,375
91	0,11764706	72	0,35294118
74	0,11111111	74	0,33333333
93	0,10526316	77	0,31578947
78	0,1	76	0,3

Tableau 20: Précision de la requête « petroleum ».

Le tableau 21 résume les précisions de la requête « petroleum » aux rangs 5, 10 et 20 ainsi que l'AVG(P).

Petroleum	AVG	P@5	P@10	p@20
Résultats_thématique	0,36	0,2	0,2	0,1
Résultats_étendue	0,44333333	0,4	0,4	0,3

Tableau 21: Evaluation de la requête « petroleum ».

A partir du tableau 21, nous remarquons que les valeurs de précisions aux rangs 5 et 10 sont constantes aussi bien pour la requête initiale que pour la requête étendue, bien que celles de la requête étendue soient plus élevées. Au rang 20 la valeur de la précision a baissé pour les deux requêtes mais celle de la requête étendue reste plus élevée. La valeur de la précision moyenne de la requête étendue est plus élevée que celle de la requête thématique.

➢ Résultats récapitulatifs des deux approches pour toutes les requêtes :

Les résultats récapitulatifs de l'approche d'expansion et la thématique sur l'ensemble des requêtes sont illustrés dans le tableau 22.

Expansion (SEKOUR)	VIRUS	NATURE	TRUMP	COVID	CORONAVIRU	PANDEMIC	IRAQ	ELECTION	DONALD	PETROLEUM
P@5	0,6	0,2	0,4	0,4	0,4	0,4	0,2	0,4	0,4	0,4
P@10	0,5	0,3	0,5	0,4	0,4	0,4	0,1	0,2	0,3	0,4
P@20	0,3	0,3	0,35	0,25	0,25	0,3	0,2	0,2	0,25	0,3
AVG(P)	0,5867	0,3783	0,6329	0,596	0,564	0,5483	0,21	0,525	0,504	0,4433
MAP	0,498852381									
THEMATIQUE	VIRUS	NATURE	TRUMP	COVID	CORONAVIRU	PANDEMIC	IRAK	ELECTION	DONALD	PETROLEUM
P@5	0,6	0	0	0,2	0	0	0,2	0	0	0,2
P@10	0,6	0,1	0	0,1	0	0	0,1	0	0,1	0,2
P@20	0,3	0,15	0,1	0,1	0,05	0,05	0,2	0,05	0,05	0,1
AVG(P)	0,6667	0,1633	0,105	0,165	0,07	0,05	0,2525	0,05	0,11	0,36
MAP	0,19925									

Tableau 22: Tableau récapitulatif des résultats de la thématique et de l'expansion.

La figure 28, illustre les résultats de la précision moyenne obtenus pour toutes les requêtes :

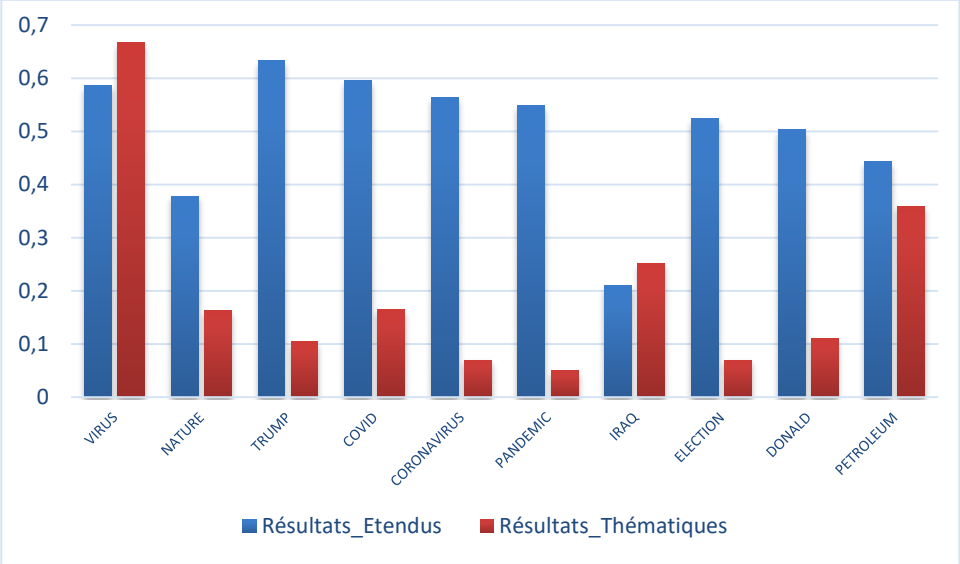

Figure 28: Histogramme comparatif des AVG(P) de la thématique et de l'expansion.

A partir des résultats de l'histogramme présenté dans la figure 28, nous visualisons que :

- Pour les requêtes « virus » et « iraq » les résultats initiaux (de la thématique) sont supérieurs aux résultats de l'approche d'expansion et donc meilleurs, car pour ces deux requêtes, pour les deux approches (thématique et expansion) le nombre de documents pertinents retournés est identique, et leurs positions dans les résultats thématiques étaient supérieures comparativement à l'expansion cependant l'AVG de la thématique est plus élevée.

- Pour le reste des requêtes « nature », « trump », « covid », « coronavirus », « pandemic », « election », « donald » et « petroleum » les résultats de l'approche d'expansion sont meilleurs, car le nombre de documents pertinents retournés par le SRI de l'expansion est plus élevé en comparaison avec la thématique.

Le résultat du taux de variation entre la thématique et l'expansion est illustré dans le tableau 23.

	Thématique (A)	Expansion (B)
MAP	0,20125	0,498852381
Taux	**59,65740414**	

Tableau 23: Taux de variation entre la thématique et l'expansion.

A partir du tableau 23, nous constatons qu'entre l'approche de départ (la thématique) et l'approche d'arrivée (expansion) la valeur de la MAP a augmenté de 59,65%, ce qui signifie que l'approche d'expansion est meilleure.

❖ Conclusion :

En se basant sur les résultats obtenus de la comparaison faite entre l'approche d'expansion proposée par (SEKOUR 2019) et les résultats de la recherche classique, nous concluons que les meilleurs résultats sont ceux de l'approche de (SEKOUR 2019), car l'aspect social et l'ajout des termes (expansion) permettent d'élargir la recherche et augmentent les chances d'observer plus de documents pertinents.

III. 2. Approche de reclassement des résultats (re-ranking) proposée par (LARBI 2019) :

Afin d'améliorer le processus de la RI classique (thématique), l'auteur a pris en compte le contenu social en exploitant les tweets avec leurs signaux sociaux (likes, comments et retweets) pour générer les scores sociaux (les scores des tweets et les scores des hashtags) dans l'objectif de reclasser les résultats initiaux et pour ce faire nous avons développé des programmes écrit en Java sous l'IDE Eclipse.

III .2.1. Génération de l'index inverse :

Afin de générer l'index inverse et l'appariement requête/documents nous avons développé un programme écrit en Python, pour comparer les termes de cet index avec les hashtags que l'on va extraire par la suite :

```
def computeTFIDFDocs(tfBagOfWords, idfs):
    tfidf = {}
    for word, val in tfBagOfWords.items():
        tfidf[word] = val * idfs[word]
    return tfidf
#calcul des poids des docs
poidsTabDocs=[]
for val in range(0,len(tfTabDocs)):
    tfidfDocsPoid=computeTFIDFDocs(tfTabDocs[val],idfsDocs)
    poidsTabDocs.append(tfidfDocsPoid)
```

Figure 29: Script de pondération des termes de documents.

```
#calcul de la similarité entre requete et document:
similarite=[]
for vals in range (0,len(affichageDocs)):
    g=0
    for val in affichageDocs.keys():
        g=g+(affichageDocs.loc[vals,:][val])

    similarite.append([vals,g])

#affiche la sim entre req et document
sim=pd.DataFrame(similarite,columns= ['Docs','Similarity'])
sim.sort_values(by=['Similarity'], inplace=True, ascending=False)
```

Figure 30: script de calcul de similarité Requêtes\Documents.

Un extrait de l'index inverse est présenté dans la figure 31.

Terme	ID_documents
package	43
heritage	40
animal	8
military	59
contraction	41
married	27
business	9
treatment	60 , 86
mariage	97
pharmaceutical	10
wedding	19,24,40
trump	21,26,45,59,84,85,86,87,88
family	1
nature	14,33,34,48,49,50,62,72,76,78
respiratory	70,51
protection	48,68,78
vaccine	60,89

Figure 31: Extrait de l'index inverse.

III.2.2. L'Extraction des hashtags :

- le programme Extract_Hashtag que nous avons utilisé permet de :
  - Extraire les hashtags des tweets et de calculer leurs scores.
  - Classer les hashtags selon leurs popularités.
  - Récupérer les documents populaires parmi ceux retournés lors de la phase de recherche classique en comparant leurs termes avec les hashtags afin de calculer le score social.
  - Calculer le score global (score thématique + score social).

Nous commençons par l'extraction des hashtags, pour ce faire nous commençons par parcourir tous les tweets de la collection afin de détecter les hashtags, le script permettant de le réaliser est illustré dans la figure 32.

```
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(' ');
                  if (pos2==-1)pos2=chaine3.length();
                  String chaine4 = y.substring(pos2+pos);

                  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 32: Script d'extraction de hashtags (LARBI,2019).

La fonction indexOf('# ') a pour but de détecter les hachtags, les extraire et les insérer dans un fichier excel nommé hashtags.xls qui sera utilisé pour l'appariement des hashtags et les termes d'index inverse.

#NOM?	score_Hashtag
#COVID19	22,47576929
#election	13,94261981
#coronavirus	8,064528998
#CoronavirusOutbreak	7,977625099
#SputnikUpdates	7,19923015
#Covid_19	6,806068009
#Lizzo	6,449838311
#DonaldTrump	5,356586275
#America	5,356586275
#USElection2020	5,356586275
#NewJersey	5,267858159
#DataVisualization	5,204006687
#DataScience	5,204006687
#DataJournalism	5,204006687
#Charts	5,204006687
#Data	5,204006687
#housewares	5,148656592
#bowl	5,148656592
#silver	5,148656592
#basket	5,148656592
#housewarming	5,148656592
#giftforhome	5,148656592
#entrywaydecor	5,148656592
#kitchendecor	5,148656592
#electionsecurity	5,056245805
#Protect2020	5,056245805
#virus	4,915591745
#Elections2020:	4,865994804
#BlueSweep	4,865994804
#ElectionDay	4,865994804
#USElection	4,865994804
#USA	4,865994804

Figure 33: Classement des hashtags.

Le score S_{H_i} de chaque hashtag est calculé avec la fonction suivante :

```
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 34: Script de calcul des scores S_{H_i}.

Après le calcul des scores des hashtags nous évaluons les similarités entre les hashtags et les termes à haute pondération de l'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) > 3.0 & chaine1.toLowerCase().contains(index.get
            System.out.println(chaine1+" for doc ");
            listhp.add(chaine1);
        for (int t=1;t<tab_Docs.length;t++) {
                for (int k=0;k<20;k++) {
                    int docNbr = tab_Docs_num[t];
                    String docName = tab_Docs[t-1];
                try {
                    if ( tab[i][k] == docNbr  ) {
```

Figure 35: Similarité hashtags/termes d'index.

❖ *Attribution des valeurs lambda et alpha :*

- Selon (LARBI 2019) la valeur de lambda λ a été testée sur un intervalle de [0,1].
- Il a pu constater que λ =0.08 assurait un écart maximal entre deux hashtags de popularité différentes.
- Ainsi pour la valeur de α, la plage de valeurs choisie est [0.6 ,0.8], l'auteur a déduit que la valeur de α=0.66 lui a fourni de meilleurs résultats.

III.2.3. Calcul du score social :

Le score social est généré par le programme suivant :

```
if (poid_List.get(i) > 3.0 & chaine1.toLowerCase().contains(index.get(i).toLowerCase())) {
    System.out.println(chaine1+" for doc ");
    listhp.add(chaine1);
for (int t=1;t<tab_Docs.length;t++) {
        for (int k=0;k<20;k++) {
            int docNbr = tab_Docs_num[t];
            String docName = tab_Docs[t-1];
        try {
            if ( tab[i][k] == docNbr  ) {
                Row row2=sheet2.createRow(t);
                Cell cellSFN = row2.createCell(6);
                Cell cellLNSD = row2.createCell(4);
                Cell cellSD = row2.createCell(2);
                Cell cellD = row2.createCell(1);
                Cell cellND = row2.createCell(0);
                sd[docNbr] = sd[docNbr] + score_Hash[j];
                cellSD.setCellValue(sd[docNbr]);
                cellND.setCellValue(docName);
                double pl = 0.08 ; // PONDERATION pour ignorer les hashtags impopulaire
                double ln = Math.log(sd[docNbr]);
                double Socfin = 1- Math.exp(-(pl*ln));
                DocS[t]=docNbr;
                Score_DocS[t]=Socfin;
                cellLNSD.setCellValue(ln);;
                cellD.setCellValue(docNbr);
                cellSFN.setCellValue(Socfin);
            }}
            catch ( ArrayIndexOutOfBoundsException E ) {
                k++;
            }

        }
```

Figure 36: Fragment de code qui calcule le score_FIN.

Ce fragment de code dans la figure 36 remplit la colonne Score_FIN $(\boldsymbol{Y = 1 - e^{-\lambda LN(X)}})$ du fichier nommé ScoreD.xls qui affiche le score social d'un document S_{D_i} , montré dans la figure 37.

	A	B	C	D	E	F	G	H	I
1	nom_Doc	Docs	Sco_Soc	Scor_Them	score_LN		score_FIN		
2	conseque	82		3,99431					
3	iraq presi	81	4,9255128	2,995732	0,053889696		-0,263221595		
4	parasites	95	7,9113237	2,995732	0,069338758		-0,238003622		
5	food	83		2,995732					
6	computer v	92	7,3121777	2,396586	0,066788802		-0,24172009		
7	trump surn	84	0,7871481	2,396586	-0,008350154		0		
8	climaate	98		1,997155					
9	wedding ce	97		1,997155					
10	The extrac	69	3,683554	1,997155	0,044290577		-0,283202193		
11	computer v	93	6,9127467	1,997155	0,064965687		-0,244472423		
12	virus comp	91		1,997155					
13	Donald_Er	19	4,6333602	1,497866	0,051877613		-0,267072896		
14	arizona	18		1,497866					
15	arizona	18		1,497866					
16	covid iraq	80	17,402263	1,497866	0,094482303				
17	climate	23		1,497866					
18	computer v	94	6,4134577	1,497866	0,062527054		-0,248287331		
19	Virus	90	6,2470287	1,331437	0,061670282		-0,249665917		
20	Petroleum	71	4,8144727	1,198293	0,053139874		-0,264638382		
21	Petroleum	56	2,884692	1,198293	0,036138778		-0,304253836		
22									
23									

Figure 37: Résultats de score_fin.

III.2.4. Calcul du score global :

Le script qui permet de calculer le score global est décrit dans la figure 38 :

```
System.out.println("DOCUMENTS POPULAIRE SUR TWITTER :");
 for (int i=1;i<DocS.length;i++) {

    for (int j=1;j<noOfRows2;j++) {
            Row rowH3=sheet3.getRow(j);
            Cell cellSS= rowH3.createCell(5);
            Cell cellSG= rowH3.createCell(7);
            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][2];
                cellSG.setCellValue(scoreGLOBAL);   }}
```

Figure 38: Génération du score global.

Après avoir exécuté ce fragment de code nous obtenons le reclassement de résultats comme suit :

M11

	A	B	C	D	E	F	G	H	I
1	nom_Docs	Docs		Scor_Them		score_FIN		Score_GLOBAL	
2	consequences	82		3,99431		0		2,396586	
3	food	83		2,995732		0		1,797439	
4	parasites	95		2,995732		-0,238003622		1,702238	
5	iraq president	81		2,995732		-0,263221595		1,692151	
6	trump surname	84		2,396586		0,067534959		1,464966	
7	computer virus	92		2,396586		-0,24172009		1,341264	
8	climaate	98		1,997155		0		1,198293	
9	wedding ceremony	97		1,997155		0		1,198293	
10	virus computer	91		1,997155				1,198293	
11	computer virus2	93		1,997155		-0,244472423		1,100504	
12	The extraction and proce	69		1,997155		-0,283202193		1,085012	
13	covid iraq	80		1,497866		0,094482303		0,936513	
14	arizona	18		1,497866		0		0,89872	
15	arizona	18		1,497866		0		0,89872	
16	computer viruses	94		1,497866		0		0,90	
17	Donald_Englis1	19		1,497866		-0,267072896		0,79189	
18	Virus	90		1,331437		-0,249665917		0,698996	
19	Petroleum Company	71		1,198293		-0,264638382		0,61312	
20	Petroleum	56		1,198293		-0,304253836		0,597274	
21	climate	23		1,497866		0		0,021212	
22									
23									
24									

Figure 39: Résultats du reclassement.

III.2.5. Résultats expérimentaux de la deuxième approche :

Afin d'évaluer cette approche, nous avons utilisé les mesures d'évaluations de précision et de MAP pour chaque requête, pour qu'on puisse comparer entre le SRI initial (recherche thématique) et SRI après le reclassement (sociale).

Durant ces expérimentations, nous avons constaté que l'approche de (LARBI 2019) ne nous a pas retournés de résultats satisfaisants, à cet effet nous avons proposé une solution qui nous a retourné des résultats plus intéressants.

Cette solution consiste :

- A annuler la fonction de normalisation du score social $\mathbf{Y= (1 - e^{-\lambda LN(X)})}$ car les valeurs de score social sont faibles et il est inutile de les normaliser.

- Pour fixer la valeur de lambda (de la formule 5, page 58), nous avons réalisé un ensemble de tests en considérant différentes valeurs. Au final la valeur λ=0 ,8 nous a donné de meilleurs résultats comparativement à la valeur λ=0 ,08 considérée par (LARBI 2019).

❖ Evaluation de la requête « nature » :

Dans ce qui suit :

- Docs_Init : sont les documents retournés par le système pour la requête initiale (recherche classique).
- Docs_Recl : sont les documents retournés par le système selon le reclassement proposé par (LARBI 2019).
- Recl_Propos : sont les documents retournés par le système selon le reclassement que nous avons proposé.

Le tableau 24, montre les résultats de précision de la requête « nature »:

Docs_Init	Precision	Docs_Recl	Precision	Recl_Propos	Precision
82	0	82	0	82	0
81	0	83	0	95	0
83	0	79	0	81	0
95	0	84	0	78	0,25
79	0	95	0	83	0,2
78	0,16666667	81	0	79	0,16666667
69	0,14285714	78	0,14285714	69	0,14285714
84	0,125	69	0,125	92	0,125
92	0,11111111	97	0,11111111	84	0,11111111
91	0,1	98	0,1	91	0,1
93	0,09090909	75	0,09090909	93	0,09090909
97	0,08333333	77	0,08333333	72	0,16666667
98	0,07692308	92	0,07692308	97	0,15384615
72	0,14285714	68	0,07142857	98	0,14285714
75	0,13333333	67	0,06666667	75	0,13333333
77	0,125	91	0,0625	76	0,1875
76	0,17647059	93	0,05882353	71	0,17647059
71	0,16666667	71	0,05555556	77	0,16666667
68	0,15789474	72	0,10526316	68	0,15789474
67	0,15	76	0,15	67	0,15

Tableau 24: Précision de la requête « nature ».

Le tableau 25, résume les précisions de la requête « nature » aux rangs 5, 10 et 20 ainsi que l'AVG.

Nature	AVG	P@5	P@10	p@20
Résultats_thématique	0,16333333	0	0,1	0,15
Résultats_RECLASSEMENT	0,13	0	0,1	0,15
Résultats_reclassement_proposé	0,20333333	0,2	0,1	0,15

Tableau 25: Evaluation de la requête « nature ».

Les valeurs de précisions aux rangs 5, 10 et 20 sont identiques pour le thématique et le reclassement proposé par (LARBI 2019), identiques aussi au rang 10 et 20 que le reclassement proposé mais plus élevé pour ce dernier au rang 5 que les deux autres. Cependant la précision moyenne nous permet d'ordonnancer ces solutions par valeur décroissante comme suit: reclassement proposé, thématique puis reclassement.

❖ Evaluation de la requête « virus » :

Le tableau 26, illustre les résultats de précision par rapport à la requête « virus »:

Docs_Init	Precision	Docs_Recl	Precision	Recl_Propos	Precision
95	1	95	1	99	0
98	0,5	98	0,5	92	0,5
97	0,33333333	97	0,33333333	95	0,66666667
93	0,5	93	0,5	98	0,5
92	0,6	92	0,6	97	0,4
94	0,66666667	94	0,66666667	93	0,5
96	0,57142857	96	0,57142857	94	0,57142857
91	0,625	91	0,625	96	0,5
99	0,55555556	99	0,55555556	91	0,55555556
90	0,6	37	0,5	90	0,6
69	0,54545455	90	0,54545455	69	0,54545455
82	0,5	82	0,5	19	0,5
18	0,46153846	18	0,46153846	37	0,46153846
23	0,42857143	23	0,42857143	81	0,42857143
19	0,4	78	0,4	82	0,4
78	0,375	79	0,375	18	0,375
79	0,35294118	83	0,35294118	23	0,35294118
81	0,33333333	69	0,33333333	78	0,33333333
83	0,31578947	19	0,31578947	79	0,31578947
37	0,3	81	0,3	83	0,3

Tableau 26: Précision de la requête « virus ».

Le tableau 27, résume les précisions de la requête « virus » aux rangs 5, 10 et 20 ainsi que l'AVG.

VIRUS	AVG	P@5	P@10	p@20
Résultats_thématique	0,66666667	0,6	0,6	0,3
Résultats_RECLASSEMENT	0,65833333	0,6	0,5	0,3
Résultats_reclassement_proposé	0,56666667	0,4	0,6	0,3

Tableau 27: Evaluation de la requête « virus ».

A partir du tableau 27, nous distinguons qu'au rang 5, les valeurs de précision de la thématique et le reclassement proposé par (LARBI 2019) sont identiques est supérieures en comparaison à celle du reclassement proposé. Au rang 10, la précision du reclassement proposé augmente pour devenir égale à celle de la thématique qui reste stagnée, et nous constatons une baisse dans la précision du reclassement proposé par (LARBI 2019). Au rang 20, nous remarquons que les valeurs de précisions des trois approches marquent une baisse et deviennent identiques. Pour la précision moyenne, nous constatons une légère différence entre le reclassement et la thématique, bien que ce dernier soit légèrement plus élevé.

❖ Evaluation de la requête « coronavirus » :

Le tableau 28 nous montre les résultats de précision pour la requête « coronavirus » :

Docs_Initial	Precision	Docs_Recl	Precision	Recl_Proposé	Precision
82	0	82	0	82	0
81	0	83	0	95	0
95	0	95	0	81	0
83	0	81	0	83	0
92	0	84	0	92	0
84	0	92	0	84	0
98	0	98	0	93	0
97	0	97	0	69	0
69	0	91	0	98	0
93	0	93	0	97	0
91	0	69	0	91	0
19	0	80	0,08333333	80	0,083333333
18	0	18	0,07692308	94	0,076923077
18	0	18	0,07142857	19	0,071428571
80	0,06666667	94	0,06666667	18	0,066666667
23	0,0625	19	0,0625	18	0,0625
94	0,05882353	90	0,05882353	71	0,058823529
90	0,05555556	71	0,05555556	56	0,055555556
71	0,05263158	56	0,05263158	90	0,052631579
56	0,05	23	0,05	23	0,05

Tableau 28: Précision de la requête « coronavirus ».

Le tableau 29 nous illustre les précisions de la requête « coronavirus » aux rangs 5, 10 et 20 ainsi que l'AVG.

coronavirus	AVG	P@5	P@10	p@20
Résultats_thématique	0,07	0	0	0,05
Résultats_RECLASSEMENT	0,08	0	0	0,05
Résultats_reclassement_proposé	0,08	0	0	0,05

Tableau 29: Evaluation de la requête « coronavirus ».

A partir du tableau 29, nous remarquons que les valeurs de précisons des trois approches aux différents rangs sont identiques et très faibles, voir nulles aux rangs 5 et 10, ce qui signifie qu'on observe aucun document pertinent à ce niveau. Les valeurs de précisions moyennes sont égales pour les approches de reclassement et légèrement supérieures comparativement à celle de la thématique.

❖ Evaluation de la requête « covid » :

Le tableau 30 nous montre les résultats de précision pour la requête « covid » :

Docs_Ini	Precision	Docs_Recl	Precision	Recl_Propos	Precision
95	0	95	0	95	0
82	0	82	0	69	0
69	0	69	0	92	0
92	0	92	0	80	0,25
80	0,2	83	0	83	0,2
83	0,16666667	80	0,16666667	91	0,1666667
78	0,14285714	97	0,14285714	93	0,1428571
91	0,125	98	0,125	78	0,125
93	0,11111111	91	0,11111111	97	0,1111111
97	0,1	93	0,1	98	0,1
98	0,09090909	78	0,09090909	84	0,0909091
79	0,08333333	79	0,08333333	79	0,0833333
84	0,07692308	84	0,07692308	71	0,0769231
75	0,07142857	75	0,07142857	73	0,0714286
68	0,06666667	68	0,06666667	60	0,1333333
60	0,125	67	0,0625	75	0,125
73	0,11764706	64	0,05882353	68	0,1176471
67	0,11111111	71	0,05555556	67	0,1111111
71	0,10526316	73	0,05263158	64	0,1052632
64	0,1	60	0,1	82	0,1

Tableau 30: Précision de la requête « covid ».

Le tableau 31 nous résume les précisions de la requête « covid » aux rangs 5, 10 et 20 ainsi que l'AVG.

covid	AVG	P@5	P@10	p@20
Résultats_thématique	0,165	0,2	0,1	0,1
Résultats_RECLASSEMENT	0,135	0	0,1	0,1
Résultats_reclassement_proposé	0,19	0,2	0,1	0,1

Tableau 31: Evaluation de la requête « covid ».

A partir du tableau 31, nous distinguons qu'au rang 5 la valeur de précision de la thématique et le reclassement proposé est identique et supérieure à celle du reclassement proposé par (LARBI 2019) qui est nulle. Aux rangs 10 et 20 la thématique et le reclassement proposé marquent une baisse qui a été compensée par une hausse dans l'approche de reclassement et ces trois approches deviennent identiques avec une précision de 0,1. Concernant la précision moyenne, la valeur de l'approche de reclassement proposée est supérieure aux autres.

❖ Evaluation de la requête « Donald » :

Le tableau 32, illustre les résultats de précision pour la requête « Donald » :

Docs_Ini	Precision	Docs_Recl	Precision	Recl_Propos	Precision
82	0	69	0	82	0
69	0	82	0	69	0
97	0	97	0	97	0
79	0	81	0	81	0
81	0	78	0	78	0
83	0	79	0	79	0
78	0	83	0	83	0
92	0	92	0	45	0,125
45	0,11111111	45	0,11111111	92	0,11111111
75	0,1	71	0,1	71	0,1
73	0,09090909	73	0,09090909	73	0,09090909
50	0,08333333	56	0,08333333	56	0,08333333
48	0,07692308	50	0,07692308	50	0,07692308
56	0,07142857	48	0,07142857	48	0,07142857
71	0,06666667	49	0,06666667	49	0,06666667
61	0,0625	75	0,0625	75	0,0625
64	0,05882353	61	0,05882353	61	0,05882353
67	0,05555556	64	0,05555556	64	0,05555556
49	0,05263158	67	0,05263158	67	0,05263158
68	0,05	68	0,05	68	0,05

Tableau 32: Précision de la requête « donald ».

Le tableau 33 nous montre les précisions de la requête « donald » aux rangs 5, 10 et 20 ainsi que l'AVG.

donald	AVG	P@5	P@10	p@20
Résultats_thématique	0,11	0	0,1	0,05
Résultats_RECLASSEMENT	0,11	0	0,1	0,05
Résultats_reclassement_proposé	0,13	0	0,1	0,05

Tableau 33: Evaluation de la requête « donald ».

A partir du tableau 33, nous constatons que les valeurs de précision sont nulles pour les trois solutions au rang 5. Au rang 10 la précision remonte tout en restant identique pour les trois approches, puis au rang 20 les valeurs sont de nouveau faibles et égales. Les valeurs de la précision moyenne sont identiques pour la thématique et le reclassement et en dessous de celle du reclassement proposé.

❖ Evaluation de la requête « Election » :

Le tableau 34 nous montre les résultats de précision de la requête « Election »:

Docs_Init	Precision		Docs_Recl	Precision		Recl_Propos	Precision
82	0		82	0		69	0
69	0		69	0		82	0
83	0		83	0		95	0
79	0		79	0		81	0
78	0		78	0		45	0,2
95	0		95	0		83	0,16666667
81	0		81	0		79	0,14285714
75	0		75	0		78	0,125
73	0		73	0		71	0,11111111
71	0		67	0		73	0,1
50	0		48	0		56	0,09090909
67	0		49	0		50	0,08333333
45	0,07692308		64	0		75	0,07692308
48	0,07142857		68	0		67	0,07142857
49	0,06666667		61	0		64	0,06666667
56	0,0625		92	0		68	0,0625
64	0,05882353		45	0,05882353		61	0,05882353
68	0,05555556		71	0,05555556		92	0,05555556
61	0,05263158		56	0,05263158		48	0,05263158
92	0,05		50	0,05		49	0,05

Tableau 34: Précision de la requête « Election ».

Le tableau 35 nous résume les précisions de la requête « election » aux rangs 5, 10 et 20 ainsi que l'AVG.

election	AVG	P@5	P@10	p@20
Résultats_thématique	0,07	0	0	0,05
Résultats_RECLASSEMENT	0,06	0	0	0,05
Résultats_reclassement_proposé	0,2	0,2	0,1	0,05

Tableau 35: Evaluation de la requête « Election ».

A partir du tableau 35, nous remarquons qu'aux rangs 5 et 10 les valeurs de précision de la thématique et le reclassement proposé par (LARBI 2019) sont nulles face aux valeurs du reclassement proposé qui sont plus élevées. Au rang 20, les trois approches sont identiques. La valeur de la précision moyenne est nettement plus élevée pour le reclassement proposée que pour les deux autres.

❖ Evaluation de la requête « Iraq »:

Le tableau 36, montre les résultats de précision de la requête « iraq »:

Docs_Init	Precision	Docs_Recl	Precision	Recl_Proposé	Precision
82	0	82	0	82	0
81	0,5	83	0	81	0,5
83	0,33333333	95	0	83	0,33333333
95	0,25	81	0,25	95	0,25
79	0,2	79	0,2	92	0,2
78	0,16666667	92	0,16666667	78	0,16666667
84	0,14285714	78	0,14285714	84	0,14285714
92	0,125	84	0,125	79	0,125
69	0,11111111	69	0,11111111	69	0,11111111
91	0,1	91	0,1	91	0,1
93	0,09090909	97	0,09090909	93	0,09090909
97	0,08333333	98	0,08333333	97	0,08333333
98	0,07692308	75	0,07692308	98	0,07692308
75	0,07142857	93	0,07142857	75	0,07142857
73	0,13333333	73	0,13333333	73	0,13333333
76	0,125	77	0,125	74	0,1875
77	0,11764706	74	0,17647059	76	0,17647059
74	0,16666667	76	0,16666667	71	0,22222222
71	0,21052632	71	0,21052632	94	0,21052632
94	0,2	94	0,2	77	0,2

Tableau 36: Précision de la requête « iraq ».

Le tableau 37, résume les précisions de la requête « iraq » aux rangs 5, 10 et 20 ainsi que l'AVG.

irak	AVG	P@5	P@10	p@20
Résultats_thématique	0,2525	0,2	0,1	0,2
Résultats_RECLASSEMENT	0,1925	0,2	0,1	0,2
Résultats_reclassement_proposé	0,26	0,2	0,1	0,2

Tableau 37: Evaluation de la requête « iraq ».

A partir du tableau 37, nous remarquons que les valeurs de la précision des trois approches sont identiques aux rangs 5, 10, et 20 bien qu'au rang 10 les valeurs marquent une baisse de 50% qui est compensée par une hausse de 50% au rang 20. Pour les valeurs de la précision moyenne, nous constatons une légère différence entre la thématique et le reclassement proposé, quoique la valeur supérieure soit atteinte par le reclassement proposé.

❖ Evaluation de la requête « Pandemic » :

Le tableau 38, présente les résultats de précision pour la requête « Pandemic » :

Docs_Init	Precision		Docs_Recl	Precision		Recl_Propos	Precision
82	0		82	0		82	0
81	0		83	0		95	0
95	0		95	0		81	0
83	0		81	0		83	0
84	0		84	0		92	0
92	0		92	0		93	0
98	0		98	0		91	0
97	0		97	0		84	0
93	0		93	0		98	0
91	0		91	0		97	0
94	0		94	0		94	0
69	0		87	0		90	0
90	0		96	0		69	0
87	0		90	0		80	0,07142857
96	0		69	0		87	0,06666667
79	0		79	0		96	0,0625
78	0		78	0		89	0,05882353
89	0		89	0		86	0,05555556
86	0		80	0,05263158		79	0,05263158
80	0,05		86	0,05		78	0,05

Tableau 38: Précision de la requête « pandemic ».

Le tableau 39, résume les précisions de la requête « pandemic » aux rangs 5, 10 et 20 ainsi que l'AVG.

pandemic	AVG	P@5	P@10	p@20
Résultats_thématique	0,05	0	0	0,05
Résultats_RECLASSEMENT	0,05	0	0	0,05
Résultats_reclassement_proposé	0,07	0	0	0,05

Tableau 39: Evaluation de la requête « pandemic ».

A partir du tableau 39, nous distinguons que les valeurs sont très faibles aussi bien pour la précision aux différents rangs que pour la précision moyenne, une valeur légèrement plus élevée que les deux autres de la précision moyenne est observée pour le reclassement proposé.

❖ Evaluation de la requête « Trump » :

Le tableau 40, montre les résultats de précision pour la requête « Trump » :

Docs_Init	Precision	Docs_Recl	Precision	Recl_Propos	Precision
95	0	95	0	95	0
92	0	92	0	92	0
98	0	98	0	93	0
97	0	97	0	91	0
93	0	93	0	98	0
91	0	91	0	97	0
69	0	82	0	69	0
82	0	69	0	82	0
94	0	94	0	94	0
78	0	84	0,1	90	0
84	0,09090909	79	0,09090909	81	0
79	0,08333333	83	0,08333333	78	0
83	0,07692308	90	0,07692308	79	0
81	0,07142857	81	0,07142857	83	0
90	0,06666667	96	0,06666667	84	0,06666667
96	0,0625	78	0,0625	96	0,0625
87	0,11764706	87	0,11764706	75	0,05882353
75	0,11111111	75	0,11111111	64	0,05555556
64	0,10526316	64	0,10526316	67	0,05263158
67	0,1	67	0,1	87	0,05

Tableau 40: Précision de la requête « trump ».

Le tableau 41, résume les précisions de la requête « trump » aux rangs 5, 10 et 20 ainsi que l'AVG.

trump	AVG	P@5	P@10	p@20
Résultats_thématique	0,105	0	0	0,1
Résultats_RECLASSEMENT	0,11	0	0,1	0,1
Résultats_reclassement_proposé	0,06	0	0	0,1

Tableau 41: Evaluation de la requête « trump ».

A partir du tableau 41, nous constatons qu'au rang 5 les valeurs de précision sont identiques et nulles pour les trois approches, une augmentation à 0,1 est observée pour le reclassement proposé par (LARBI 2019) au rang 10. Au rang 20, une stagnation est observée pour ce dernier, et les deux autres approches augmentent pour l'atteindre et devenir égales. Concernant la précision moyenne, la valeur du reclassement proposé est très faible comparativement aux deux autres, et c'est le reclassement de (LARBI 2019) qui atteint la valeur la plus élevée.

❖ Evaluation de la requête « Petroleum » :

Le tableau 42, montre les résultats de précision de la requête « Petroleum »:

Docs_Init	Precision	Docs_Recl	Precision	Recl_Propos	Precision
82	0	82	0	82	0
69	0,5	69	0,5	69	0,5
81	0,33333333	79	0,33333333	95	0,33333333
95	0,25	83	0,25	81	0,25
78	0,2	95	0,2	78	0,2
79	0,16666667	81	0,16666667	79	0,16666667
83	0,14285714	78	0,14285714	83	0,14285714
84	0,125	75	0,125	92	0,125
71	0,22222222	92	0,11111111	71	0,22222222
73	0,2	71	0,2	73	0,2
75	0,18181818	73	0,18181818	84	0,18181818
92	0,16666667	84	0,16666667	75	0,16666667
76	0,15384615	98	0,15384615	91	0,15384615
98	0,14285714	97	0,14285714	93	0,14285714
97	0,13333333	91	0,13333333	74	0,13333333
72	0,125	93	0,125	76	0,125
91	0,11764706	74	0,11764706	72	0,11764706
74	0,11111111	76	0,11111111	78	0,11111111
93	0,10526316	72	0,10526316	98	0,10526316
78	0,1	78	0,1	97	0,1

Tableau 42: Précision de la requête « petroleum ».

Le tableau 43, illustre les précisions de la requête « petroleum » aux rangs 5, 10 et 20 ainsi que l'AVG.

petroleum	AVG	P@5	P@10	p@20
Résultats_thématique	0,36	0,2	0,2	0,1
Résultats_RECLASSEMENT	0,35	0,2	0,2	0,1
Résultats_reclassement_proposé	0,35	0,2	0,2	0,1

Tableau 43: Evaluation de la requête « petroleum ».

A partir du tableau 43, nous remarquons que les valeurs de précision sont égales aux rangs 5 et 10 pour les trois approches. Au rang 20, elles restent toutes les trois égales tout en marquant une baisse de 50%.

La valeur de la précision moyenne des deux approches de reclassement sont égales, et en dessous de la valeur de la thématique qui est légèrement supérieure.

➢ Résultats récapitulatifs des trois approches pour toutes les requêtes :

Le tableau 44 nous récapitule les résultats retournés par le SRI classique (thématique), SRI de reclassement ainsi que le SRI du reclassement proposé.

THEMATIQUE	VIRUS	NATURE	TRUMP	COVID	CORONAVIRU	PANDEMIC	IRAK	ELECTION	DONALD	PETROLEUM
P@5	0,6	0	0	0,2	0	0	0,2	0	0	0,2
P@10	0,6	0,1	0	0,1	0	0	0,1	0	0,1	0,2
P@20	0,3	0,15	0,1	0,1	0,05	0,05	0,2	0,05	0,05	0,1
AVG(P)	0,6667	0,1633	0,105	0,165	0,07	0,05	0,2525	0,07	0,11	0,36
MAP	0,20125									
Reclassement (LARBI)	VIRUS	NATURE	TRUMP	COVID	CORONAVIRU	PANDEMIC	IRAK	ELECTION	DONALD	PETROLEUM
P@5	0,6	0	0	0	0	0	0,2	0	0	0,2
P@10	0,5	0,1	0,1	0,1	0	0	0,1	0	0,1	0,2
P@20	0,3	0,15	0,1	0,1	0,05	0,05	0,2	0,05	0,05	0,1
AVG(P)	0,6583	0,13	0,11	0,135	0,08	0,05	0,1925	0,06	0,11	0,35
MAP	0,187583333									
Reclassement Proposé	VIRUS	NATURE	TRUMP	COVID	CORONAVIRU	PANDEMIC	IRAK	ELECTION	DONALD	PETROLEUM
P@5	0,4	0,2	0	0,2	0	0	0,2	0,2	0	0,2
P@10	0,6	0,1	0	0,1	0	0	0,1	0,1	0,1	0,2
P@20	0,3	0,15	0,1	0,1	0,05	0,05	0,2	0,05	0,05	0,1
AVG(P)	0,5667	0,2033	0,06	0,19	0,08	0,07	0,26	0,2	0,13	0,35
MAP	0,211									

Tableau 44: Tableau récapitulatif des résultats de l'approche de reclassement.

La figure 40, illustre les résultats de l'AVG(P) sous forme d'un histogramme.

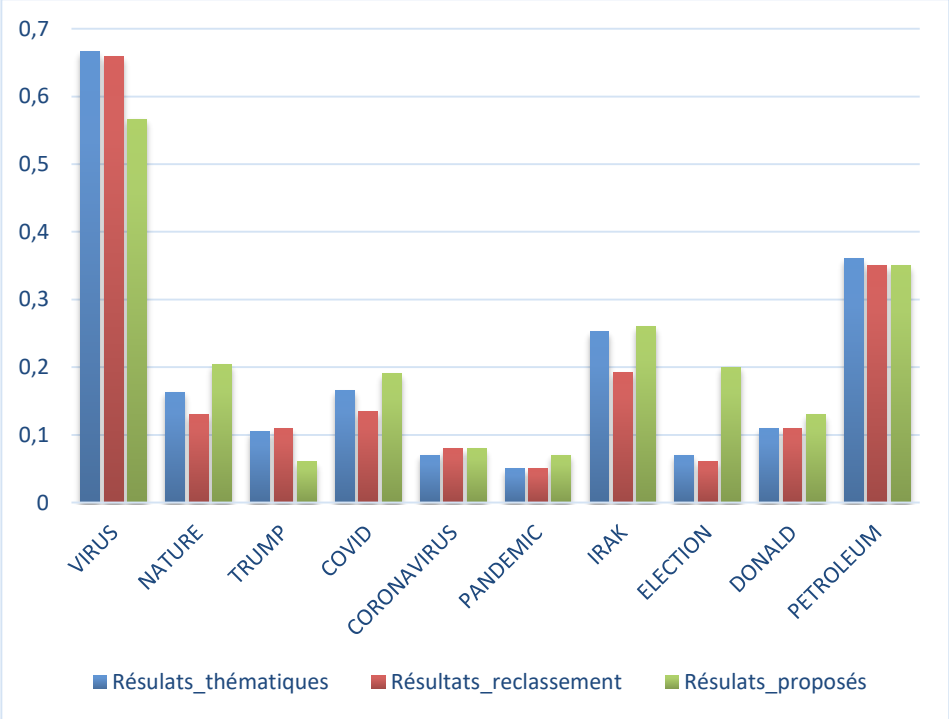

Figure 40: Histogramme récapitulatif de l'approche de reclassement.

A partir de l'histogramme de la figure 40, nous constatons que :

- Pour les requêtes « virus » et « petroleum » les résultats de la recherche classique (thématique) sont plus élevés donc meilleurs comparativement aux résultats des approches de reclassement.
- Pour les requêtes « nature », « covid », « pandemic », « election », « iraq » et « donald » les résultats du reclassement proposé sont meilleurs.
- Pour la requête « trump » les résultats du reclassement proposé par (LARBI 2019) sont meilleurs.

Les résultats du taux de variation entre la thématique et les deux approches de reclassement sont illustrés dans le tableau 45.

	Thématique (A)	Reclassement LARBI (B)
MAP	0,20125	0,187583333
Taux	-7,285651013	

	Thématique (A)	Reclassement Proposé (B)
MAP	0,20125	0,211
Taux	4,620853081	

Tableau 45: Taux de variation entre la thématique et les approches de reclassement.

A partir du tableau 45, nous constatons que :

Entre la thématique et l'approche de reclassement proposée par (LARBI 2019) la valeur de la MAP a diminué d'un taux de 7,28%, ce qui insinue que les résultats de la thématique sont plus performants.

Entre la thématique et l'approche de reclassement proposée la valeur de la MAP a augmenté d'un taux de 4,62%, ce qui signifie que le reclassement proposé apporte un gain et donne de meilleurs résultats.

❖ Conclusion :

En se basant sur les résultats obtenus de la comparaison faite entre les résultats de la recherche classique et l'approche de reclassement proposée par (LARBI 2019), nous avons conclu que cette dernière retourne des résultats peu appréciables, comparativement à la thématique. Ce résultat est dû en grande partie à la formule de normalisation utilisée dans cette approche.
La solution de reclassement que nous avons proposé, a prouvé son efficacité en apportant un gain par rapport à la thématique et au reclassement proposé par (LARBI 2019).

IV. Comparaison des approches étudiées, (SEKOUR 2019) (LARBI 2019) et notre proposition :

➢ Comparaison de la requête « nature » :

Le tableau 46 résume les précisions aux rangs 5, 10 et 20 ainsi que l'AVG.

Nature	AVG	P@5	P@10	p@20
Résultats_étendue	0,37833333	0,2	0,3	0,3
Résultats_RECLASSEMENT	0,13	0	0,1	0,15
Résultats_reclassement_proposé	0,20333333	0,2	0,1	0,15

Tableau 46: Comparaison de la requête « nature ».

A partir du tableau 46, nous remarquons que la précision au rang 5 du reclassement proposé par (LARBI 2019) est nulle, alors que celle de la requête étendue et le reclassement proposé est nettement plus élevée. Au rang 10 la précision du reclassement proposé par (LARBI 2019) remonte pour être égale à la précision du reclassement proposé qui a baissé, cependant, les valeurs de précision à ce rang pour ces deux approches restent en dessous de la précision de la requête étendue. Au rang 20, la précision de la requête étendue garde sa valeur est reste supérieure comparativement aux approches de reclassement. La précision moyenne de la requête étendue est nettement supérieure à celle des approches de reclassement.

➢ Comparaison de la requête « virus » :

Le tableau 47 résume les précisions aux rangs 5, 10 et 20 ainsi que l'AVG.

VIRUS	AVG	P@5	P@10	p@20
Résultats_étendue	0,58666667	0,6	0,5	0,3
Résultats_RECLASSEMENT	0,65833333	0,6	0,5	0,3
Résultats_reclassement_proposé	0,56666667	0,4	0,6	0,3

Tableau 47: Comparaison de la requête « virus ».

A partir du tableau 47, nous constatons que la précision au rang 5 du reclassement proposé par (LARBI 2019) est celle de la requête étendue sont identiques et plus élevée en comparaison avec le reclassement proposé. Au rang 10 la précision du reclassement proposé par (LARBI 2019) et l'approche d'expansion proposée par (SEKOUR 2019) baissent tout en restant égales et la précision du reclassement proposé remonte et dépasse les deux approches. Au rang 20, la précision des trois approches marque une baisse tout en restant égales. La précision moyenne du reclassement de (LARBI 2019) est supérieure à celle des autres approches qui indique une légère différence.

➢ Comparaison de la requête « coronavirus » :

Le tableau 48 montre les résultats de précision de la requête « virus » aux rangs 5,10 et 20 ainsi que l'AVG :

coronavirus	AVG	P@5	P@10	p@20
Résultats_étendue	0,564	0,4	0,4	0,25
Résultats_RECLASSEMENT	0,08	0	0	0,05
Résultats_reclassement_proposé	0,08	0	0	0,05

Tableau 48: Comparaison de la requête « coronavirus ».

A partir du tableau 48, nous constatons qu'aux rangs 5 et 10 les valeurs de la précision de l'approche d'expansion proposée par (SEKOUR 2019) sont nettement plus élevées comparativement aux valeurs de précision des approches de reclassement qui sont nulles. Au rang 20, la valeur de précision des approches de reclassement remonte légèrement et reste toujours en dessous de la précision de l'approche d'expansion qui montre une baisse. La valeur de précision moyenne de l'approche de (SEKOUR 2019) est nettement plus élevée en comparaison avec les valeurs des approches de reclassement qui sont identiques et très faibles.

➢ Comparaison de la requête « covid » :

Le tableau 49 résume les précisions aux rangs 5, 10 et 20 ainsi que l'AVG.

covid	AVG	P@5	P@10	p@20
Résultats_étendue	0,596	0,4	0,4	0,25
Résultats_RECLASSEMENT	0,135	0	0,1	0,1
Résultats_reclassement_proposé	0,19	0,2	0,1	0,1

Tableau 49: Comparaison de la requête « covid ».

A partir du tableau 49, nous remarquons que la précision au rang 5 du reclassement proposé par (LARBI 2019) est nulle, et la valeur de l'approche d'expansion et supérieure en comparaison avec le reclassement proposé. Au rang 10, le reclassement remonte et le reclassement proposé baisse et les deux valeurs deviennent identiques et restent en dessous de la précision de l'approche d'expansion. Au rang 20, la valeur de précision des approches de reclassement ne montre aucune évolution, et reste toujours en dessous de la valeur de l'expansion malgré sa baisse. La valeur de précision moyenne de l'approche de (SEKOUR 2019) est nettement plus élevée en comparaison avec les valeurs des approches de reclassement.

➢ Comparaison de la requête « donald »:

Le tableau 50 illustre les précisions aux rangs 5, 10 et 20 ainsi que l'AVG.

donald	AVG	P@5	P@10	p@20
Résultats_étendue	0,504	0,4	0,3	0,25
Résultats_RECLASSEMENT	0,11	0	0,1	0,05
Résultats_reclassement_proposé	0,13	0	0,1	0,05

Tableau 50: Comparaison de la requête « donald ».

A partir du tableau 50, nous constatons qu'au rang 5 la précision de l'approche d'expansion est nettement plus élevée comparativement aux deux approches de reclassement qui sont nulles. Au rang 10, la précision des approches de reclassement remonte tout en restant en dessous de la valeur de précision de la requête étendue même si sa valeur a baissé. Au rang 20, la précision des approches de reclassement est de nouveau faible et reste en dessous de la précision de l'approche de (SEKOUR 2019). La valeur de la précision moyenne de l'approche d'expansion est supérieure à la valeur des approches de reclassement.

➢ Comparaison de la requête « election » :

Le tableau 51 résume les précisions aux rangs 5, 10 et 20 ainsi que l'AVG.

election	AVG	P@5	P@10	p@20
Résultats_étendue	0,525	0,4	0,4	0,2
Résultats_RECLASSEMENT	0,06	0	0	0,05
Résultats_reclassement_proposé	0,2	0,2	0,1	0,05

Tableau 51: Comparaison de la requête « election ».

A partir du tableau 51, nous remarquons qu'au rang 5 la précision de l'approche d'expansion est nettement plus élevée comparativement à l'approche de reclassement proposé et le reclassement de (LARBI 2019) qui est nulle. Au rang 10, l'approche d'expansion et le reclassement de (LARBI 2019) gardent leurs valeurs, et le reclassement proposé indique une baisse. Au rang 20, la précision de l'approche de (SEKOUR 2019) baisse mais reste supérieure aux approches de reclassement qui sont très faibles. La valeur de la précision moyenne de l'approche d'expansion est hautement supérieure à la valeur des approches de reclassement.

➢ Comparaison de la requête « iraq » :

Le tableau 52, résume les précisions aux rangs 5, 10 et 20 ainsi que l'AVG.

iraq	AVG	P@5	P@10	p@20
Résultats_étendue	0,21	0,2	0,1	0,2
Résultats_RECLASSEMENT	0,1925	0,2	0,1	0,2
Résultats_reclassement_proposé	0,26	0,2	0,1	0,2

Tableau 52: Comparaison de la requête « iraq ».

A partir du tableau 52, nous constatons que les valeurs de précision de l'approche d'expansion proposée par (SEKOUR 2019) et les approches de reclassement aux différents rangs sont identiques. Bien qu'au niveau de la précision moyenne, c'est la valeur du reclassement proposé qui est la plus élevée.

➢ Comparaison de la requête « pandemic »:

Le tableau 53 résume les précisions aux rangs 5, 10 et 20 ainsi que l'AVG.

pandemic	AVG	P@5	P@10	p@20
Résultats_étendue	0,54833333	0,4	0,4	0,3
Résultats_RECLASSEMENT	0,05	0	0	0,05
Résultats_reclassement_proposé	0,07	0	0	0,05

Tableau 53: Comparaison de la requête « pandemic ».

A partir du tableau 53, nous constatons que pour les approches de reclassement les valeurs sont très faibles aussi bien pour la précision aux différents rangs que pour la précision moyenne, alors que les valeurs de précision et précision moyenne de l'approche d'expansion proposée par (SEKOUR 2019) sont nettement plus élevées.

➢ Comparaison de la requête « trump »:

Le tableau 54 résume les précisions aux rangs 5, 10 et 20 ainsi que l'AVG.

trump	AVG	P@5	P@10	p@20
Résultats_étendue	0,63285714	0,4	0,5	0,35
Résultats_RECLASSEMENT	0,11	0	0,1	0,1
Résultats_reclassement_proposé	0,06	0	0	0,1

Tableau 54: Comparaison de la requête « trump ».

A partir du tableau 54, nous constatons qu'au rang 5 la précision de la requête étendue est nettement plus élevée comparativement aux valeurs des deux approches de reclassement qui sont nulles. Au rang 10, la précision du reclassement proposé garde sa valeur nulle, le reclassement remonte mais reste en dessous de la valeur de l'approche d'expansion. Au rang 20, la précision de l'approche de (SEKOUR 2019) baisse mais reste supérieure aux valeurs des approches de reclassement. La valeur de la précision moyenne de l'approche d'expansion proposée par (SEKOUR 2019) est hautement supérieure aux valeurs des approches de reclassements qui sont faibles.

➢ Comparaison de la requête « petroleum »:

Le tableau 55 résume les précisions aux rangs 5, 10 et 20 ainsi que l'AVG.

pandemic	AVG	P@5	P@10	p@20
Résultats_étendue	0,44333333	0,4	0,4	0,3
Résultats_RECLASSEMENT	0,35	0,2	0,2	0,1
Résultats_reclassement_proposé	0,35	0,2	0,2	0,1

Tableau 55: Comparaison de la requête « petroleum ».

A partir du tableau 55, nous constatons que pour les deux approches de reclassement les valeurs sont identiques et en dessous des valeurs de l'approche d'expansion proposée par (SEKOUR 2019), aussi bien pour la précision aux différents rangs que pour la précision moyenne.

➢ Résultats récapitulatifs de toutes les approches pour toutes les requêtes :

Les résultats récapitulatifs des approches étudiées sur l'ensemble des dix requêtes sont illustrés dans le tableau 56.

Expansion (SEKOUR)	VIRUS	NATURE	TRUMP	COVID	CORONAVI	PANDEM	IRAQ	ELECTIO	DONALI	PETROLEUM
P@5	0,6	0,2	0,4	0,4	0,4	0,4	0,2	0,4	0,4	0,4
P@10	0,5	0,3	0,5	0,4	0,4	0,4	0,1	0,2	0,3	0,4
P@20	0,3	0,3	0,35	0,25	0,25	0,3	0,2	0,2	0,25	0,3
AVG(P)	0,58667	0,37833	0,63286	0,596	0,564	0,54833	0,21	0,525	0,504	0,44333
MAP	0,498852381									
THEMATIQUE	VIRUS	NATURE	TRUMP	COVID	CORONAVI	PANDEM	IRAK	ELECTIO	DONALI	PETROLEUM
P@5	0,6	0	0	0,2	0	0	0,2	0	0	0,2
P@10	0,6	0,1	0	0,1	0	0	0,1	0	0,1	0,2
P@20	0,3	0,15	0,1	0,1	0,05	0,05	0,2	0,05	0,05	0,1
AVG(P)	0,66667	0,16333	0,105	0,165	0,07	0,05	0,2525	0,07	0,11	0,36
MAP	0,20125									
Reclassement (LARBI)	VIRUS	NATURE	TRUMP	COVID	CORONAVI	PANDEM	IRAK	ELECTIO	DONALI	PETROLEUM
P@5	0,6	0	0	0	0	0	0,2	0	0	0,2
P@10	0,5	0,1	0,1	0,1	0	0	0,1	0	0,1	0,2
P@20	0,3	0,15	0,1	0,1	0,05	0,05	0,2	0,05	0,05	0,1
AVG(P)	0,65833	0,13	0,11	0,135	0,08	0,05	0,1925	0,06	0,11	0,35
MAP	0,187583333									
Reclassement Proposé	VIRUS	NATURE	TRUMP	COVID	CORONAVI	PANDEM	IRAK	ELECTIO	DONALI	PETROLEUM
P@5	0,4	0,2	0	0,2	0	0	0,2	0,2	0	0,2
P@10	0,6	0,1	0	0,1	0	0	0,1	0,1	0,1	0,2
P@20	0,3	0,15	0,1	0,1	0,05	0,05	0,2	0,05	0,05	0,1
AVG(P)	0,56667	0,20333	0,06	0,19	0,08	0,07	0,26	0,2	0,13	0,35
MAP	0,211									

Tableau 56: Tableau récapitulatif des résultats des approches étudiées.

La figure 41, illustre les résultats de la précision moyenne obtenus pour toutes les requêtes :

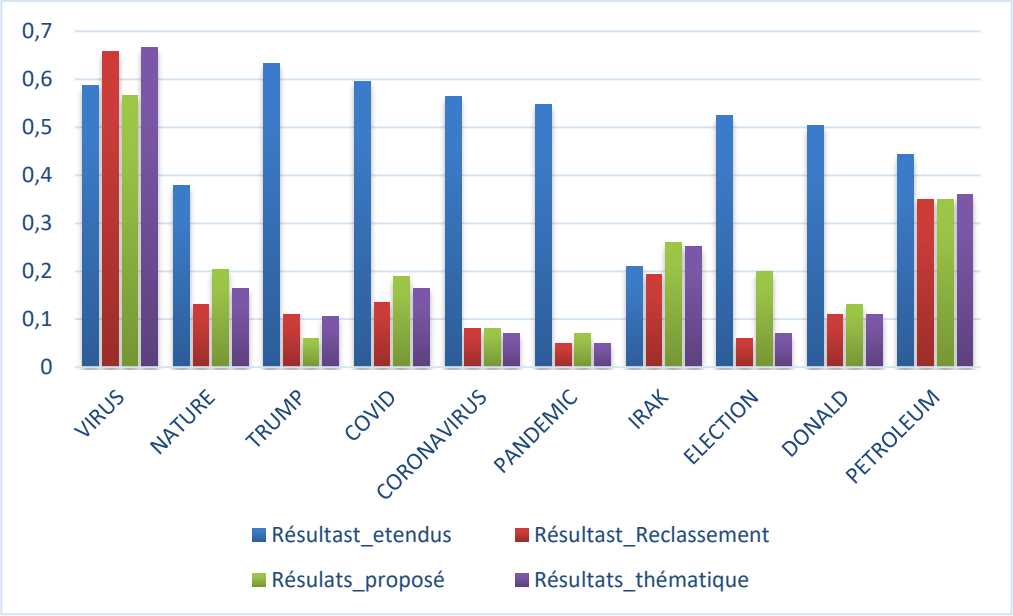

Figure 41: Histogramme des résultats des approches étudiées.

A partir des résultats de l'histogramme illustré dans la figure 41, nous constatons que globalement les résultats de l'approche d'expansion de (SEKOUR 2019) sont plus élevés comparativement aux deux approches de reclassement et à la thématique, ce qui insinue que cette approche est meilleure.

Le tableau 57, illustre les résultats de calcul du taux de variation entre les approches étudiées.

	Reclassement LARBI (A)	Reclassement Proposé (B)
MAP	0,187583333	0,211
Taux	11,09794645	
	Reclassement LARBI (A)	Expansion (B)
MAP	0,187583333	0,498852381
Taux	62,39702562	
	Reclassement Proposé (A)	Expansion (B)
MAP	0,211	0,498852381
Taux	57,70291813	

Tableau 57: Taux de variation entre les approches étudiées.

A partir des résultats des taux de variations montrés dans le tableau 57 nous distinguons que :

- Entre l'approche du reclassement proposé et l'approche proposée par (LARBI 2019) la valeur de la MAP a augmenté d'un taux de 11%, cette évolution positive indique que l'approche du reclassement proposé donne de meilleurs résultats.
- Entre l'approche de reclassement de (LARBI 2019) et l'approche de (SEKOUR 2019), la valeur de la MAP de l'approche d'expansion proposée par (SEKOUR 2019) a augmenté d'un taux de 62,4%, cette évolution signifie que les résultats de l'approche d'expansion proposée par (SEKOUR 2019) sont plus performants comparativement aux résultats de l'approche de reclassement proposée par (LARBI 2019).
- Entre l'approche du reclassement proposé et l'approche de (SEKOUR 2019), la valeur de la MAP de l'approche d'expansion proposée par (SEKOUR 2019) a augmenté d'un taux de 57,7%, cependant les résultats de l'approche d'expansion proposée par (SEKOUR 2019) sont plus performants comparativement aux résultats de l'approche de reclassement proposée.

V. Conclusion :

Dans ce chapitre, nous avons présenté l'implémentation, l'évaluation et la comparaison des deux approches étudiées proposées par (SEKOUR 2019) et (LARBI 2019). Nous avons également proposé une solution afin d'améliorer les résultats de l'approche de reclassement proposée par (LARBI 2019).

Les résultats de l'approche de reclassement proposée par (LARBI 2019) sont peu appréciables à cause de la fonction de normalisation du score social $\mathbf{Y= (1 - e^{-\lambda LN(X)})}$ qu'il a proposé, la solution que nous avons apporté et qui consiste principalement à éliminer cette fonction a bien prouvé son efficacité.

Les comparaisons qui ont été faites entre les deux approches de reclassement et l'approche d'expansion de la requête proposée par (SEKOUR 2019), ont montré que cette dernière donne des résultats plus performants, ce qui nous semble logique du fait que cette approche étend la requête initiale et ajoute plus de termes pertinents au SRI, ce qui augmente ses chances de retourner plus de documents pertinents en comparaison avec les SRI des approches de reclassement

Conclusion générale

I. Conclusion générale :

Les travaux présentés dans ce mémoire, rentrent dans le contexte de la recherche d'information, plus précisément dans la RI sociale. La RI classique se limite à l'appariement entre requête et documents sans prendre en compte l'aspect utilisateur ou sociale d'une ressource. Cependant, avec la croissance exponentielle des données web et l'émergence des réseaux sociaux, il devient de plus en plus difficile de retrouver une information pertinente.

La problématique à laquelle nous nous sommes intéressées dans ce travail, réside en l'intégration et l'exploitation des informations sociales en vue d'améliorer le processus de la recherche.

Nous avons débuté ce travail, par présenter les concepts fondamentaux de la recherche d'information classique, ensuite nous avons introduit la recherche d'information sociale et ses concepts de base.

Par la suite, nous avons présenté deux approches proposées dans l'objectif d'améliorer la RI. La première proposée par (SEKOUR 2019) se base sur l'expansion de la requête et la seconde proposée par (LARBI 2019) et se base sur le reclassement des résultats.

Ensuite nous avons implémenté ces deux dernières, dans le but de les évaluer et de les comparer. A l'issue de ces évaluations, nous avons constaté que les résultats de l'approche de reclassement proposée par (LARBI 2019) étaient peu appréciables. Convaincues du potentiel de cette approche, nous avons pensé à revoir ses formules et réévaluer ses paramètres pour apporter notre contribution et proposer une solution. Notre proposition s'est avérée intéressante puisqu'elle a retourné des résultats très appréciables.

Au final, les évaluations réalisées sur les deux approches de reclassement et l'approche d'expansion ont montré que cette dernière retournait de meilleurs résultats, ce qui nous parait logique du fait que cette approche se base sur l'ajout de termes pertinents et par conséquent, augmente les probabilités de restituer plus de documents pertinents.

II. Perspectives :

Dans les travaux futurs et en perspectives, nous avons l'intention :

- De réaliser les évaluations des approches proposées sur une collection de test utilisée dans les compagnes d'évaluation de la RI, afin de positionner ces approches.

- D'utiliser une ontologie dans l'approche d'expansion proposée par (SEKOUR 2019), afin que les termes pertinents utilisés pour l'expansion de la requête soient dans le même contexte sémantique que la requête, ce qui nous pensons permettra de retourner des résultats plus significatifs.

- De revoir la formule de score social proposée par (LARBI 2019), car nous avons constaté que cette dernière n'apporte pas l'amélioration souhaitée aux résultats. Cependant, nous avant apporté des modifications sur trois requêtes (nature, virus et trump) afin d'apercevoir le comportement des résultats. Ces changements consistent à:

  - Mettre en valeur absolue la formule de score de tweets comme suit: $\mathbf{score\_tweets = |Ln(\gamma N_{re} + \delta N_l + \varepsilon N_c)|}$ car les valeurs retournées dans la formule(2) (section III.3.1 chapitre 3) sont négatives et cela est dû au contenu social infime de la collection de tweets.

  - Pondérer le score des hashtags dans la formule de score social de documents $\mathbf{Score\_social\_Document = score\_thématique + \alpha * score\_social}$ tel que $\alpha = 2$.

Bibliographie :

BADACHE, Ismail. «Recherche d'Information Sociale Exploitation des Signaux Sociaux pour Améliorer la Recherche d'information.» Toulouse, 05 02 2016.

Bao, S. *Optimizing Web Search Using Social Annotations.* Canada, 2007.

Bender, Bender M M. *Exploiting social relations for query expansion and result ranking in data engineering workshop.* 2008.

Bishoff, Firan Nejdl et Paiu. *Can all tags be used for search.* 2008.

Bouadjenek, R.Bouadjenek, H.Hacid et M.Bouzeghoub. *SoPPa: a new social personalized ranking function for improving web search .* 2013.

Brin, S. & Page, L. *The anatomy of a large-scale hypertextual Web search engine.* California, 1998.

Carmel. *Social bookmark weighting for search and recommendation.* 2010.

Chelaru, Orellana-Rodriguez et Altingovde. *Can social features help learning to rankyoutube videos .* 2012.

Dmitriev, Eiron Fontoura Shekita. *Using annotations in entreprise search.* 2006.

FELLAG, Samia. «Recherche d'information dans les documents XML: modèle basé sur une propagation sélective des termes.» Tizi-Ouzou, s.d.

HAMMACHE, Arezki. «Recherche d'information: un modèle de langue combinant mots simples et mots composés.» s.d.

Hotho, M A.Hotho. *Information retrieval in folksonomies: Search and ranking.* 2006.

Jin, Wang &. *Exploring Online Social Activities for Adaptive Search Personalization.* California, 2010.

Khodaei, Ali & Omar Alonso. *Temporally-aware signals for social search.* 2012.

Kirch, Sebastian Marius. «Social information retrieval.» November 2005.

Koolen, M., Kazai, G., & Craswell, N. *Wikipedia Pages as Entry Points for Book Search.* Barcelona, 2009.

Kuramoto, Helio. «Les systèmes de recheche d'information en langage naturel.» France, Avril 1995.

LARBI, Nabil. «Exploitation des signaux sociaux de twitter afin d'améliorer la recherche d'information .» 2019.

Li, M Y.Li. *improving weak ad-hoc queries using wikipedia asexternal corpus.* 2006.

Meinel, Noll M. & G. *Web Search Personalization via Social Bookmarking and Tagging.* Potsdam, 2007.

REMADNA, Ahmed. «Classification des posts sur les réseaux sociaux .» Lyon, s.d.

Rijsbergen, Van. *Information retrieval.* London:Butterworths., 1979.

Rocchio. *Relevence feedback in information retrieval. .* 1971.

Schenkel, R.,. *Efficient Top-k Querying over Social-Tagging Networks.* Singapore, 2008.

SEKOUR, Mohamed. «Exploitation des signaux sociaux de Twitter pour améliorer la recherche d'information.» 2019.

Tamine, Lamjed Ben Jabeur et Lynda. «Vers un modèle de recherche d'information sociale pour l'accès aux ressources bibliographiques.» Université Paul Sabatter, Toulouse, s.d.

Teevan. *Using related users data to enhance web search.* 2012.

Yanbe. *Towards Improving Web Search by Utilizing Social Bookmarks.* Kyoto, 2007.

Yoshiyuki Inagaki, Narayanan Sadagopan, Georges Dupret, Anlei Dong, Ciya Liao, Yi Chang, and Zhaohui Zheng. *Session based click features for recency.* 2010.

https://www.researchgate.net/publication/271521876

https://hal.archives-ouvertes.fr/hal-01444570v2

http://www.guillaumedesbieys.com

http://social-book-search.humanities.uva.nl/#/overview

https://www.microsoft.com/en-us/research

https://www.cairn.info

https://scholar.google.com

https://medium.com/@datamonsters/text-preprocessing-in-python-steps-tools-and-examples-bf025f872908

https://medium.com/towards-artificial-intelligence

http://nltk.org/book

https://www.lelivrescolaire.fr

https://ieeexplore.ieee.org