République **A**lgérienne **D**émocratique et **P**opulaire
Ministère de l'**E**nseignement **S**upérieur et de la **R**echerche **S**cientifique

UNIVERSITE **M**OULOUD **M**AMMERI DE **T**IZI-**O**UZOU

FACULTE DU **G**ENIE **E**LECTRIQUE ET D'**I**NFORMATIQUE
DEPARTEMENT D'INFORMATIQUE

Mémoire de Fin d'Etudes de MASTER ACADEMIQUE

Domaine : **Mathématiques et Informatique**
Filière : **Informatique**
Spécialité : **Systèmes informatiques**

Réalisé par :
Feriel HANNACHI
Samah LADAOUI

Thème

Recherche d'information dans Twitter : proposition d'une approche de recherche d'influenceurs

Mémoire soutenu publiquement le 17/07/ 2019 devant le jury composé de :

Président : Mr S.SADI

Examinateur : M L.BELKACEMI

Encadré par : M Fatiha AMIROUCHE

Résumé

Notre travail s'inscrit dans le domaine de la recherche d'information (RI) dans les microblogs, plus particulièrement dans la recherche d'influence dans la plateforme Twitter. En effet, étant donné Twitter une plateforme de prédilection, elle est convoitée par plusieurs bloggeurs où chacun essaye d'impacter et d'influencer sur la société par le biais de ses posts. Nous avons proposé une approche pour la mesure de l'influence basée sur les métriques abonnements/abonnés et nous avons défini un t-index qui repose sur la mesure bibliométrique g-index. Dans cette approche nous avons défini un ratio d'influence qui combine un ratio d'abonnement et le t-index. L'évaluation de notre approche repose sur un modèle de recherche qui intègre la mesure d'influence dans le calcul de pertinence des tweets.

Mots clés : recherche d'information, influenceur, influence, Twitter, microblogs.

Abstract

Our work is a part of the field of social information retrieval (SIR) , more particularly in the search for influencers in the Twitter platform. Indeed, given Twitter a very popular platform, it is coveted by several bloggers where everyone tries to impact and influence the society through his posts. In this context we have contributed to this problem by proposing an approach for the measurement of the influence which is based on followers/followee metrics and we have defined a t-index based on the g-index (a bibliometric measure). In this approach we have defined an influence ratio that combines a followers ratio and the t-index. The evaluation of our approach is based on a retrieval model that integrates the measure of influence into the calculation of tweets relevance .

Keywords : information retrieval, influencer, influence, Twitter, microblogs, social information retrieval.

Remerciements

Au terme de la rédaction de ce mémoire, nous estimons qu'il est important d'accorder quelques lignes de reconnaissances à toute personne ayant contribué de prêt ou de loin.

Tout d'abord, nous adressons notre plus profonde gratitude à notre promotrice Mme AMIROUCHE Fatiha, qui a toujours su orienté nos recherches, et pour tout le temps qu'elle nous a accordée.

Nous tenons à remercier également les membres de jury d'avoir accepté d'évaluer notre travail.

Nous tenons à exprimer notre sincère reconnaissance envers monsieur FERROUK Massinissa et Mlle Wassila AZZOUG pour leurs précieuses aides. Nos remerciements vont également à nos amis(es) qui nous ont toujours encouragées que ce soit d'un point de vue moral ou intellectuel, et plus particulièrement à monsieur M'hmed TAYEB.

Dédicaces

A nos chers parents,

A nos frères et sœurs,

A nos familles,

A nos amis(es).

Table des matières

Table des figures

Liste des tableaux

Introduction

Cadre général et objectifs

Depuis sa création, internet est devenu un outil incontournable du 21ème siècle , et grâce au web 2.0 de nombreux réseaux sociaux ont vu le jour. Parmi ces différentes platformes, nous avons choisi de porter un vif interêt à Twitter.

Twitter est une platforme de microblogings très en vogue dans notre temps, elle constitu à elle seule une véritable mine d'or en information, cela est du à l'immensité du volume d'informations qu'elle produit et manipule, générées par un important nombre d'utilisateur.

Au premier trimestre de cette année 2019, Twitter compte plus de 330 millions d'utilisateurs actifs avec plus de 500 millions de tweets publiés chaque jours. Ces chiffres nous permettent de remettre en cause, la qualité des tweet publié sur ce réseau. La qualité du tweet est associé au message publié ainsi qu'à l'importance de son auteur.

Dans le cadre de la recherche d'infomation(RI), l'importance du bloggeur est assimilé à son influence, où cette dernière représente l'impacte d'un utilisateur sur les autres. Plusieurs experts de la RI ont tenté de mesurer l'influence, proposant ainsi plusieurs approches différentes qui reposent sur les caractéristiques de Twitter.

Contribution

Dans notre travail, nous proposons une approche qui permet de mesurer l'influence dans Twitter. Nous nous sommes basé sur les métriques abonnées/abonnements qui définissent notre ratio d'abonnement, et nous avons défini le t-index, ce dernier est inspiré d'une des mesures qui permet de mesurer l'impcat d'un scientifique : le g-index. Par la suite nous avons combiné ces deux mesure en un seul ratio, dit ratio d'influence.

Organisation de la thèse

Notre travail est réparti sur quatres chapitres :

Chapitre 1 : Génaralité sur la recherche d'information, dans ce chapitre nous présentons les concepts généraux liés à la racherche d'infomation, puis nous présentons la platefrome de prédilection Twitter.

Chapitre 2 : Mesure de l'influence dans Twitter, ce chapitre est un état de lart sur les différentes approches déjà proposées dans le cadre de la mesure de l'influence.

Chapitre 3 : Contribution à la recherche d'influenceur, dans ce chapitre, nous expliquont l'approche que nous avons proposé.

Chapitre 4 : Implémentation de l'approche proposée, ce dernier chapitre se base sur les détails d'implémentation et de mise en oeuvre de notre approche, ainsi qu'a l'évaluation de notre approche.

Nous terminons notre mémoire sur une conclusion générale.

Chapitre 1

Généralités sur la recherche d'information :

1.1 Introduction

La recherche d'information (*information retrieval* en anglais) a pour but de retrouver, parmi une collection de documents préalablement stockée, les documents qui répondent au besoin informationnel d'un utilisateur , formellement exprimé par une requête. Le présent chapitre est consacré à l'introduction de la recherche d'information (RI). Il s'articule sur deux parties : La première présente les concepts de base de la RI en général, la seconde introduit la RI dans twitter en particulier.

1.2 Définitions :

La RI est une discipline de l'informatique qui regroupe un ensemble d'outils permettant de retrouver des éléments(dits documents) pertinents (ie. répondant à un besoin informationnel) à partir d'un ensemble (ou collection) de documents préalablement enregistrés. La RI combine des techniques de stockage, d'organisation, de représentation et de recherche de l'information à partir de données massives, dans l'unique but de fournir aux utilisateurs un accès facile à des informations pertinentes pouvant susciter leur intérêt.[Ricardo Baeza-yates 99]

La RI est mise en œuvre à travers un ensemble de programmes informatiques spécifiques composant le système de recherche d'information (SRI). Un SRI est un système informatique qui permet de retrouver à partir d'une collection documentaire préalablement stockée, l'ensemble des documents pertinents pour un besoin en information d'un utilisateur, formulé à l'aide d'une requête.

1.2.1 Le besoin d'information :

Cette notion est souvent liée aux besoins des utilisateurs dont trois types ont été définis :

Besoin vérificatif :l'utilisateur cherche à vérifier les informations qu'il possède déjà, c'est à dire qu'il recherche une donnée particulière et sait la plupart du temps comment y accéder.

Besoin thématique connu : dans ce cas, l'utilisateur est en quête de nouvelles informations dans le but de clarifier ou de trouver de nouveaux concepts liés à un sujet ou à un domaine connu.

Besoin thématique inconnu : ici l'utilisateur est en dehors de sa zone de confort, c'est à dire qu'il recherche des informations qui ne sont pas en relation avec les domaines ou sujets qui lui sont familiers.

1.2.2 La requête :

La requête représente l'expression du besoin informationnel de l'utilisateur, elle est exprimée par un ensemble de mots spécifiques faisant référence au besoin formulé.

1.2.3 La pertinence

La pertinence est une notion très importante en RI. Elle représente le degré de concordance d'un document avec la requête utilisateur. On distingue deux types de pertinence :

- **Pertinence système :**La pertinence système est définie par un score attribué par le SRI à chaque document, dans le but d'évaluer la correspondance de son contenu avec celui de la requête. Cette pertinence est objective et déterministe.
- **Pertinence utilisateur :** Elle est définie par les jugements de pertinence de l'utilisateur vis à vis des documents que le SRI lui retourne en réponse à sa requête. Ce type de pertinence est subjectif, car un même document retourné en réponse à une même requête, peut être jugé différemment par deux utilisateurs différents. La pertinence utilisateur est dite évolutive car, si à un instant t donné un document est jugé non pertinent, à l'instant t+1, il pourrait être jugé pertinent car la connaissance de l'utilisateur sur le sujet aura évolué.

La qualité d'un SRI réside dans sa capacité à calculer une pertinence système qui approche au mieux la pertinence utilisateur.

1.2.4 Document /collection de documents

- **Document :**représente l'information de base manipulable et accessible par le SRI. Il peut s'agir d'un texte, un passage de texte, une page web, une image....
- **Collection de documents :**aussi appelé fond documentaire ou corpus (documentaire), est un ensemble de documents préalablement stockés et organisés en vue de recherche.

1.3 Le processus de la recherche d'information

Le processus de la RI est représenté schématiquement par le processus en U de la figure 1.1 [Boubekeur. 08]suivante.

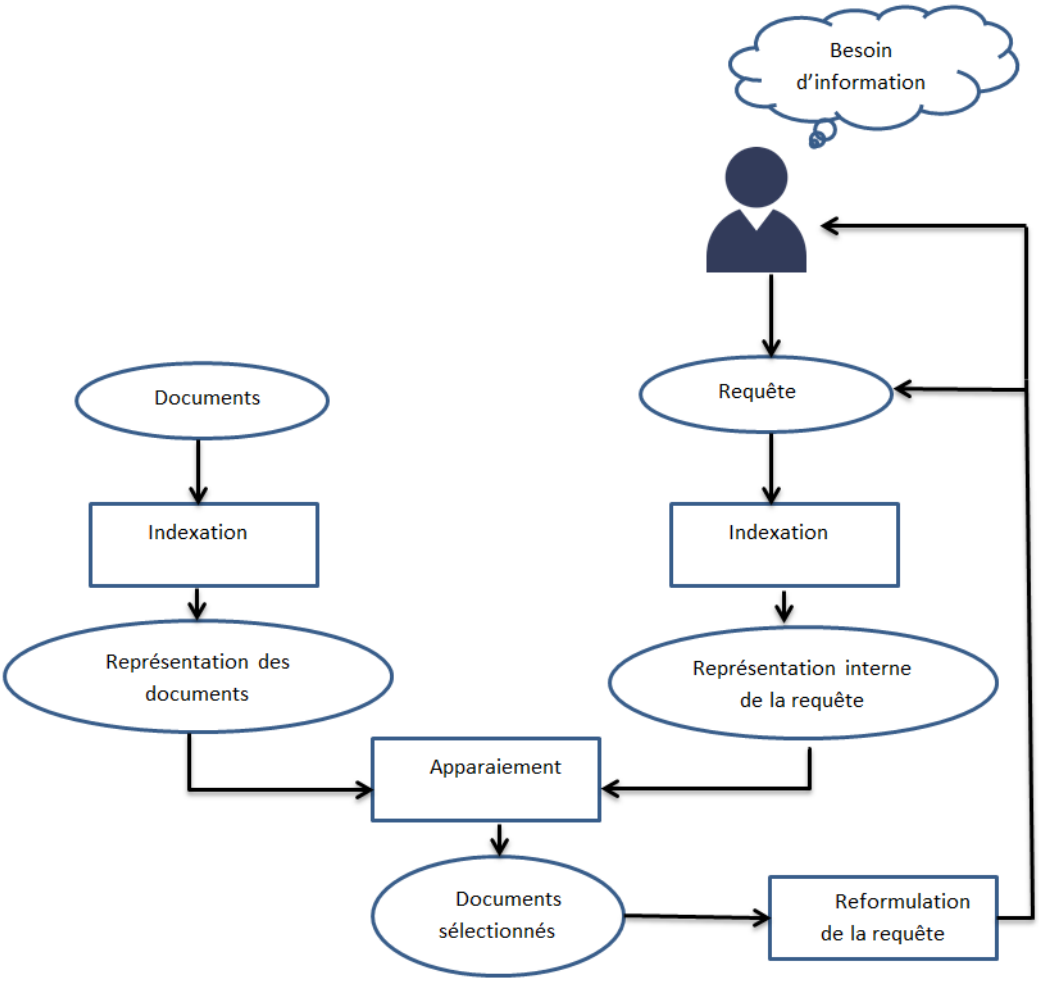


FIGURE 1.3.1 – Processus de recherche d'information[Nicholas J. Belkinl. 92]

Cette figure fait ressortir trois processus de base mis en œuvre dans la recherche d'information : le processus d'indexation, le processus d'appariement, et le processus de reformulation de la requête

1.3.1 Le processus d'indexation :

L'indexation[Badache. 16] consiste à sélectionner un ensemble de termes représentatifs du contenu d'un document ou d'une requête. Le résultat de l'indexation est une représentation interne dite *descripteur* de l'unité textuelle correspondante, composée de l'ensemble des termes sélectionnés. Les descripteurs des documents sont rangés dans une structure de données appelée *index*. L'ensemble des termes d'index constitue le langage d'indexation.

L'indexation peut être manuelle, automatique ou semi-automatique :

— L'indexation manuelle est réalisée par un humain : chaque document est alors analysé par un spécialiste du domaine correspondant ou par un documentaliste.
— En Indexation automatique, chaque document est analysé à l'aide d'un processus entièrement automatisé.
— Dans l'indexation semi-automatique, chaque document est d'abord analysé par un processus automatique qui construit son descripteur initial, puis le documentaliste intervient pour le choix final des termes à partir du descripteur initial.

L'indexation est une étape indispensable dans le processus de RI. De sa qualité dépend en grande partie la qualité de la recherche. Dans la RI moderne, l'indexation automatique est l'approche la plus utilisée. Nous détaillerons ce processus dans la section 1.4.

1.3.2 Le processus d'appariement requête-document :

Le processus d'appariement (ou processus de recherche) permet de calculer le degré (ou score) de pertinence de chaque document de la collection par rapport à la requête utilisateur, puis retourne l'ensemble des documents les plus pertinents à l'utilisateur. On distingue deux types d'appariement :

— **Appariement exact :** Le résultat retourné est une liste de documents respectant exactement la requête spécifiée. Ces documents ne sont pas ordonnés.
— **Appariement approché :** Le résultat retourné est une liste de documents censés être pertinents pour la requête. Les documents sont triés selon leur degré de pertinence pour la requête.

Les différents types d'appariement dépendent du modèle de recherche utilisé. Nous détaillerons les modèles de recherche en section 1.5.

1.3.3 Le processus de reformulation de la requête :

Dans les SRI, on constate souvent que la requête initiale est insuffisante pour répondre au besoin de l'utilisateur. La raison en est que, l'utilisateur ne sait pas ce qu'il recherche, ou alors il exprime mal son besoin informationnel. L'objectif de la reformulation de requête est de permettre d'affiner l'expression de la requête en se basant sur le corpus documentaire ou sur les résultats d'une première recherche initiée par la requête initiale.

Il existe deux approches principales pour la reformulation de requête :

— L'expansion automatique des requêtes.
— La réinjection de pertinence (*Relevance Feedback*).

1.3.3.1 Expansion automatique de requête :

L'expansion de requête consiste à construire une nouvelle requête q1, qui se rapproche du besoin informationnel de l'utilisateur, en étendant la requête initiale q0 avec des mots du corpus qui leur sont sémantiquement liés, et/ou en repondérant les termes de la requête.

1.3.3.2 Réinjection de pertinence :

La réinjection de pertinence consiste à construire une nouvelle requête q1, en étendant la requête initiale q0 à partir des résultats d'une première recherche initiée par q0. Le processus de réinjection de pertinence contient trois étapes essentielles :

- L'échantillonnage : Dans cette étape, un échantillon de document est construit et ce à partir d'éléments jugés pertinents par l'utilisateur.
- L'extraction des évidences : cette étape consiste à sélectionner les termes qui seront ajoutés à la nouvelle requête. Ces termes sont issus des documents jugés pertinents ou non par l'utilisateur.
- La réécriture de la requête : cette étape consiste à créer une nouvelle requête tout en utilisant la requête initiale avec les informations extraites dans l'étape précédente. Il existe plusieurs formules qui sont utilisées, la plus connue est celle de Rocchio[J. 71] qui est adaptée au modèle vectoriel, décrite comme suit :

$$\mathrm{Q}_N = \alpha.\mathrm{Q}_0 + \beta.\frac{1}{|R|}\sum_{r\epsilon R}\mathrm{r} \; -\gamma.\frac{1}{R'}\sum_{r'\epsilon R'}\mathrm{r'}$$

- QN représente le vecteur de la nouvelle requête (requête reformulée) ;
- Q0 est le vecteur de la requête initiale ;
- R est l'ensemble des vecteurs r des documents jugés pertinents par l'utilisateur ;
- R' est l'ensemble des vecteurs r' des documents jugés non pertinents par l'utilisateur ;
- α, β,γ sont de paramètres de reformulation. Tel que : $\alpha+\beta+\gamma= 1$.

1.4 L'indexation automatique

L'indexation automatique[Boubekeur. 08] consiste à analyser chaque document à l'aide d'un processus entièrement automatisé.

1.4.1 L'analyse lexicale :

L'analyse lexicale consiste à identifier et extraire les unités lexicales du texte du document. Une unité lexicale représente une suite de caractères délimitée par des séparateurs (blancs,virgules....). Chaque unité lexicale définit un mot (ou un groupe de mots) du document.

1.4.2 Élimination des mots vides :

Cette étape consiste à éliminer les mots non porteurs d'information pour la recherche, tel que les conjonctions, les prépositions et les articles, dans le but de réduire la taille de l'index et donc améliorer le temps de réponse du système. Deux techniques sont principalement utilisées lors de l'élimination des mots vides :

— L'utilisation d'une liste prédéfinie des mots vides, appelée également anti-dictionnaire ou stoplist ;
— L'utilisation de mesures statistiques sur la collection de documents pour déterminer les mots les plus fréquents ou les mots rares qui sont alors considérés comme des mots vides.

1.4.3 La normalisation des termes :

Le but de la normalisation est de ramener les mots de la même famille à leur forme normale(mots ayant la même variante morphologique), pour cela on utilise deux procédures : la lemmatisation et la troncature. La lemmatisation prend en compte la forme canonique du mot, pour un verbe par exemple, on va considérer sa forme à l'infinitif, pour un mot, adjectif ... on va prendre en compte sa forme au masculin. La troncature quant à elle, consiste à éliminer les suffixes des mots retenus à l'insu de l'étape précédente.

1.4.4 La pondération des termes :

Permet d'attribuer à chaque terme d'index une valeur numérique qui mesure son importance dans le document auquel il appartient et/ou son importance dans la collection documentaire. Plusieurs fonctions de pondération ont été proposées, dont :

— **la fréquence d'occurrence TF (ou *term frequency*)** : mesure la fréquence d'apparition du terme dans le document. Plus un terme est fréquent dans un document, plus on le considère important dans la description de ce dernier. TF est généralement exprimée par l'une des déclinaisons suivantes :

1. TF : utilisation brute.

2. $0.5+0.5\ (\frac{tf}{max(tf)})$.

— **IDF (***inverse of document frequency***) :** mesure l'importance d'un terme dans toute la collection sur la base de sa fréquence documentaire (df) inverse. La fréquence documentaire (df) d'un terme étant le nombre de documents de la collection où il apparaît. L'idée est que les termes qui apparaissent dans peu de documents sont plus spécifiques au (donc plus représentatifs du) contenu de ces documents que ceux qui apparaissent dans tous les documents de la collection. IDF est exprimée par l'une des formulations suivantes :

1. IDF=$\log(\frac{N}{n})$.
2. IDF=$\log(\frac{N-n}{n})$.

Où :

— N est le nombre de documents dans la collection.
— n est le nombre de document dans la collection où le terme apparait.
— **TF*IDF :** est une combinaison des deux premières mesures qui concrétise l'idée qu'un terme est important dans un document d'une collection si d'une part, il est très fréquent dans ce document (TF élevé), et peu fréquent dans les autres documents de la collection (IDF élevé). C'est cette mesure qui est utilisée le plus souvent en RI. Elle s'exprime comme suit :

$$w_{ij} = tf_{ij} * idf_j$$

On distingue un autre facteur qui est la taille du document car plus ce dernier est long plus le terme apparait plus fréquemment. Son but est de normaliser les fréquences en fonction de la taille des documents.

1.5 Modèles de recherche :

Si c'est l'indexation qui construit le descripteur du document et l'organise dans une structure d'index, c'est le modèle de recherche qui interprète cet index en une structure logique adéquate en vue de son appariement avec la requête interne.

Il existe plusieurs modèles de recherche dont les modèles booléen, vectoriel, probabiliste... Nous les explicitons dans ce qui suit.

1.5.1 Le modèle booléen :

Le modèle booléen [Boubekeur. 08]est l'un des premiers modèles implémentés dans les SRI. Il est très simple et très rapide à mettre en œuvre. Ce modèle est basé sur la théorie des ensemble et l'algèbre de bool. Dans ce modèle, un document

d est représenté par un ensemble de mots clés, une requête q est représentée par une expression logique composée de mots-clés reliés par des opérateurs logiques (AND, OR, NOT).

La pertinence ou RSV (*Retrieval status values*) du document d pour la requête q est récursivement définie comme suit :

— RSV(d,t)=1 si le terme t ϵd, sinon 0 ;
— RSV(d, NOT q_i)=1-RSV(d, q_i) ;
— RSV(d, q_i AND q_j)=RSV(d,q_i) AND RSV(d, q_j) ;
— RSV(d, q_i ORq_j)=RSV(d,q_i) OR RSV(d, q_j).

Ce modèle présente deux inconvénients majeurs :
— Difficulté de formulation de bonnes requêtes par les utilisateurs (en effet le langage logique n'est pas à la portée de tout le monde) ;
— Les documents qui sont retournés aux utilisateur ne sont pas ordonnés par ordre de pertinence (en effet, la RSV est binaire : un document est soit pertinent, soit non pertinent).

1.5.2 Le modèle vectoriel :

Ce modèle introduit par Gérard Salton[Salton. 83]. est basé sur la théorie de l'algèbre et plus précisément sur le calcul vectoriel[Hammache. 13].

Ici, les documents et les requêtes sont représentés sous forme de vecteurs de poids dans un espace à n-dimensions où chaque dimension représente un terme d'index. Le vecteur associé à un document est représenté comme suit :

d_j = (w_{1j},w_{2j}, ...,w_{nj}) et celui associé à la requête est défini par :q = (w_{1q},w_{2q}, ...,w_{nq}) tel que : wij (respect. wiq) correspond au poids d'un terme dans le document dj (respect. dans la requête q).

La correspondance d'un document avec la requête utilisateur est définie par la similarité de leurs vecteurs associés calculée à partir d'une mesure de similarité vectorielle parmi les suivantes :

— **Le produit scalaire :**

$$\mathrm{RSV(q,d}_j\mathrm{)}=\sum_{i=1}^{n} w_{iq} * w_{ij}$$

— **La mesure de Jaccard :**

$$\mathrm{RSV(q,d}_j\mathrm{)}=\frac{\sum_{i=1}^{n} w_{iq*W_{ij}}}{\sum_{i=1}^{n} w_{iq}^2+\sum_{i=1}^{n} w_{ij}^2-\sum_{i=1}^{n} w_{iq*W_{ij}}}$$

— **La mesure de cosinus :**

$$\text{RSV(q,d}_j\text{)}=\frac{\overrightarrow{q}*\overrightarrow{d_i}}{||\overrightarrow{q}||*||\overrightarrow{d_i}||}=\frac{\sum_{i=1}^{n} w_{iq*}W_{ij}}{\sqrt{\sum_{i=1}^{n} w_{iq}^2*\sum_{i=1}^{n} w_{ij}^2}}$$

— **La mesure de Dice :**

$$\text{RSV(q,d}_j\text{)}=\frac{2*\sum_{i=1}^{n} w_{iq*}W_{ij}}{\sum_{i=1}^{n} w_{iq}^2+\sum_{i=1}^{n} w_{ij}^2}$$

1.5.3 Le modèle probabiliste :

Ce modèle est basé sur des calculs probabilistes pour estimer la pertinence d'un document pour une requête. Dans ce modèle, documents et requête sont représentés par des vecteurs de poids dans l'espace vectoriel des termes d'index[Hammache. 13]. La pertinence d'un document pour une requête est calculée comme suit :

$$\text{RSV(d}_i\text{,q)}=\frac{P(per/q,di)}{P(Nper/q,d_i)}$$

Tel que :
— P(per/q,di) : est la probabilité qu'un document di soit pertinent(per) pour la requête q ;
— P(Nper/q,di) :est la probabilité qu'un document di soit non pertinent(per) pour la requête q.

En appliquant la formule de Bayes pour les deux probabilités ci-dessus, on obtient :

$$\text{P(per/q,d}i\text{)}=\frac{P(per/q).P(d_i/per,q)}{P(d_i)}$$

$$\text{P(Nper/q,d}_i\text{)}=\frac{P(Nper/q).P(d_i/Nper,q)}{P(d_i)}$$

Où :
— P(d$_i$)est la probabilité de choisir le document di, elle est considéré comme étant constante ;
— P(d$_i$/per, q) défini la probabilité que d$_i$fasse parti des documents pertinents en réponse à la requête q ;
— P(d$_i$/$Nper, q$) défini la probabilité que d$_i$fasse parti des documents non pertinents en réponse à la requête q ;
— P(per/q) et P(Nper/q) : indiquent respectivement la probabilité de pertinence et de non pertinence d'un document quelconque.

1.5.4 Le modèle de langue :

Le modèle de langue utilise une approche différente par rapport aux modèles décris jusqu'ici. Contrairement aux autres modèles qui évaluent le degrés de similarité des documents et requêtes, le modèle de langue estime que la pertinence d'un document pour une requête est en rapport avec la probabilité que la requête puisse être générée par le document. Formellement, Le principe de ce modèle consiste à construire un modèle de langue pour chaque document d, soit Md , puis de calculer la probabilité qu'une requête Q puisse être générée par Md , soit P(Q/Md). La pertinence associé à ce modèle est mesurée comme suit :

$$RSV(d,Q) = P(Q/M_d) = P(Q = (q_1,\ q_2,\ q_3...q_n)/M_d) = \\ \Pi_{i=1}^{n} P(q_i = t_i/M_d) = \Pi_{i=1}^{n} P(q_i/M_d)$$

Sachant que :

$$\Pi_{i=1}^{n} P(q_i/M_d) = \Pi_{i=1}^{n} \frac{tf(q_i,d)}{|d|}$$

Et tf(t/d) représente la fréquence du terme ti dans le document d. Dans ce type d'estimation, quand un terme ne figure pas dans le document, la pertinence vaut systématiquement 0. Pour résoudre ce problème, on a recours à des techniques de lissage. Le lissage consiste à attribuer des valeurs de probabilités non nulles aux termes qui n'apparaissent pas dans le document.

1.6 Évaluation des SRI

Évaluer un SRI sert à vérifier si les modèles mis en œuvre sont efficaces en particulier en termes de qualité des résultats fournis. Plusieurs mesures d'évaluation ont été proposées et mises en œuvre à travers de nombreuses campagnes d'évaluation depuis les années 90. Nous les introduisons en sections suivantes.

1.6.1 Mesures de l'évaluation :

Les mesures d'évaluation permettent d'estimer l'efficacité d'un SRI. Le but étant de mesurer, pour chaque requête la capacité du système à retourner des documents pertinents. Dans ce qui suit nous définissons deux mesures d'évaluations : le rappel et la précision.

Rappel et Précision :

Le rappel et la précision sont deux mesures de bases essentielles pour évaluer l'efficacité des SRI.

Le rappel : représente le ratio de documents pertinents retournés par rapport à l'ensemble des documents pertinents de la collection documentaire.[Hammache. 13]

Où :

$$\text{R}=\frac{nombres\ de\ documents\ pertinents\ retournés}{nombres\ de\ documents\ pertinents\ total}$$

La précision :représente le ratio de documents pertinents retournés par rapport au nombre total de documents retournés par le SRI[Hammache. 13].

Où :

$$\text{P}=\frac{nombres\ de\ documents\ pertinents\ retournés}{nombres\ total\ de\ documents\ retrournés\ par\ le\ système}$$

On dit qu'un SRI est idéal lorsqu'il retourne tous les documents pertinents (rappel=1) et tous les documents pertinents total(précision=1), mais en pratique ce n'est pas le cas parce qu'il y a une relation inverse entre ces deux quantités,lorsque l'une augmente l'autre diminue. On peut le voir dans ce qu'on appelle la courbe rappel/précision [Boubekeur. 08].

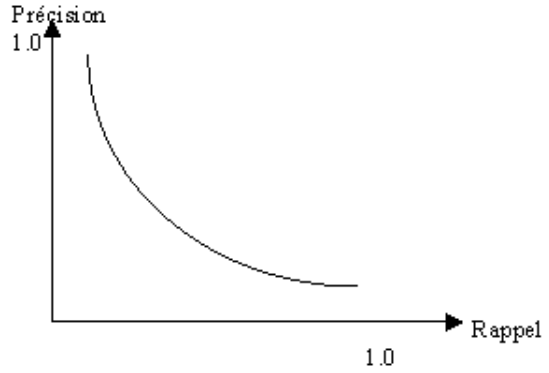


FIGURE 1.6.1 – Courbe rappel/précision

Des mesures duales au rappel et précision ont été proposées comme suit :

Bruit :Le Bruit représente la proportion de documents non pertinents retournés par les SRI en réponse à une requête de l'utilisateur, elle est définie comme suit :

$$\text{Bruit}=1\text{-R}$$

Silence : Le Silence est la proportion de documents pertinents non sélectionnés par le SRI au cours de la recherche :

$$\text{Silence}=1\text{-P}$$

De là on déduit qu'un bon SRI est celui qui revoie le maximum de documents pertinents (réduisant le silence de système), tout en rejetant les documents non pertinents (faisant le moins de bruit possible).

1.6.2 Campagnes d'évaluation

De nombreuses compagnes d'évaluation sont apparues, parmi elles, on trouve la campagne TREC que nous décrivons dans ce qui suit.

TREC (Text REtrieval Conference)

TREC est créée en 1992 avec 25 participants issus du monde académique et industriel, la compagne TREC[1] a pour but de soutenir la recherche au sein de la communauté de la RI en fournissant divers moyens nécessaires à l'évaluation à grande échelle des méthodologies et modèles de recherche. Parmi ces moyens, TREC fournit des collections de test et un framework d'évaluation.

Les collections de tests :elles sont composées d'un ensemble de documents, d'un ensemble de requêtes et de jugements de pertinence.

Voici les différents élément qui constitue un projet TREC :

Les tâches : les tâches dépendent de l'intérêt des chercheur, par exemple la RI ad-hoc, la RI dans le web, la RI dans les microblogs ;

Les participants :ce sont ceux qui participent au projet TREC. Ils sont issus de différents pays.

Structure et principe de construction de la collection : le format d'un document TREC est le SGML, il est identifié par un numéro,de plus, il est décrit par un auteur, une date de production et un contenu textuel. Concernant la requête TREC, celle ci est également identifiée par un numéro mais elle est décrite par une description sur les caractéristiques des documents pertinent qui lui sont associés.

1.7 La recherche d'information dans Twitter :

La recherche d'information sociale (*Social information seeking*) représente un nouveau domaine d'étude qui concerne l'extraction, l'acquisition et la recherche d'information à partir des espaces sociaux sur internet[shah. 17].

1. https ://trec.nist.gov/

L'un de ces espaces sociaux les plus populaires sur le net est Twitter.

1.7.1 Présentation de Twitter :

Twitter est une plateforme de microblogging fonctionnant en temps réel, qui permet d'envoyer et de consulter gratuitement des messages courts appelés tweets (gazouillis) - le logo de Twitter est d'ailleurs un oiseau (figure 1.3) -. La taille d'un tweet ne dépasse pas 140 caractères. Twitter a été créé en mars 2006 par Jack Dorsey à San Francisco . Il a été lancé par la suite au mois de juillet de la même année. Depuis lors, Twitter a évolué exponentiellement[Damak. 14]. En janvier 2019, twitter comptait plus de 326 millions d'utilisateurs actifs, et une moyenne de 500 millions de messages sont tweetés chaque jour.

La figure suivante illustre l'évolution du logo de Twitter :

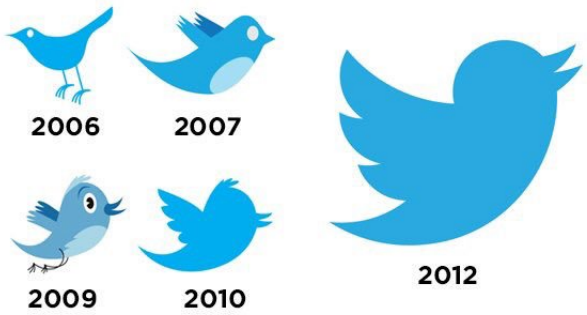


FIGURE 1.7.1 – Évolution du logo de Twitter

1.7.1.1 Fonctionnement de twitter :

Pour obtenir un compte Twitter il suffit de se rendre sur le site "https ://twitter.com/"- la figure 1.7.2 montre l'interface d'accueil du site-, et de s'inscrire.

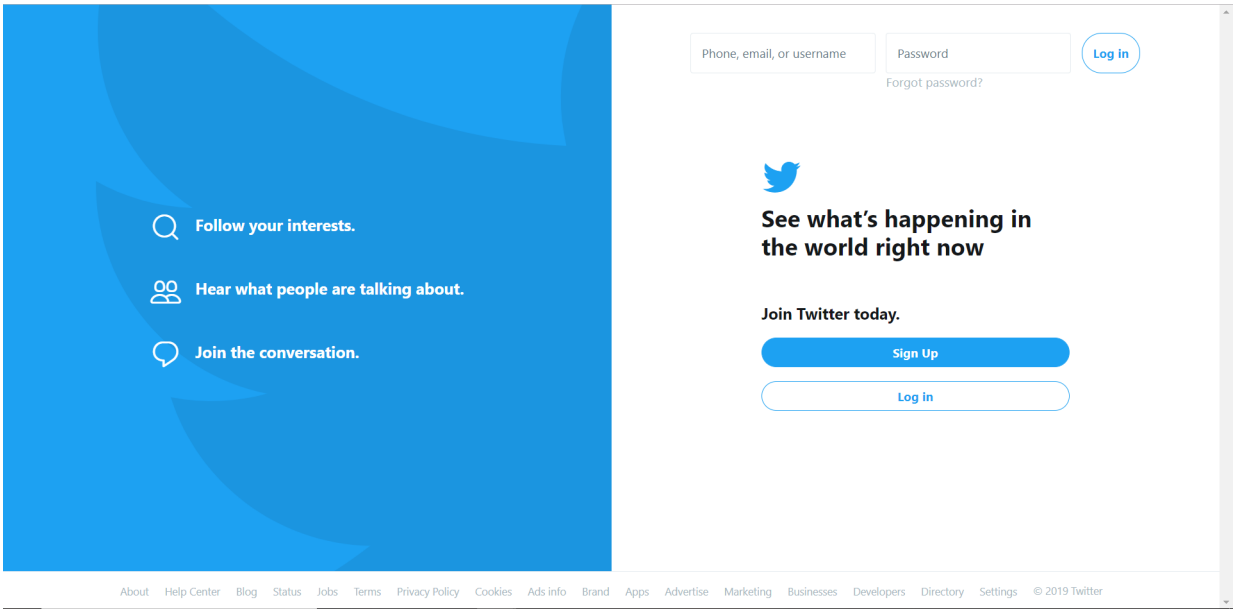


FIGURE 1.7.2 – Accueil du site Twitter

Une fois le compte crée, vous accédez à votre profil comme le montre la figure 1.5.

Vocabulaire associé à Twitter :

Dans la figure suivante voici l'exemple d'un profil Twitter qui explicite tous les termes qui lui sont associés.

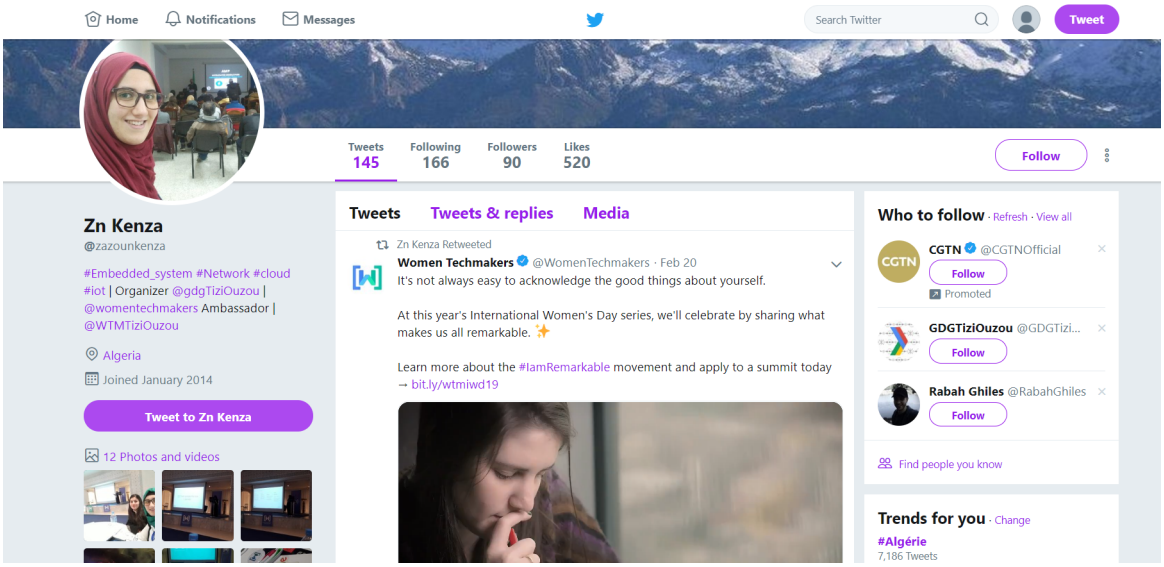


FIGURE 1.7.3 – Interface d'un profil sur Twitter

— **Twitto :** c'est l'utilisateur de Twitter.
— **Twitt ou Tweet :**sont les messages, postés sur twitter.
— **Timeline :** c'est l'ensemble des messages qui apparaissent dans votre fil d'actualité et qui sont soit postés par vous(timeline personnel) ou bien par vos abonnements. Les message sont affichés du plus récent au plus ancien.
— **Follower :** c'est une personne qui vous suit, c'est à dire qui s'est abonnée à vos publications , cette personne verra donc vos tweets dans son timeline.
— **Following :** terme utilisé pour désigner les comptes Twitter que vous suivez.
— **RT :** qui veut dire retweet , représente l'action qui consiste à rediffuser ou bien partager un message déjà publié par un autre twitto pour qu'il soit visible à vos abonnés.
— **MP :**abréviation de message privé, ou aussi DM (Direct Message en anglais),c'est une fonction qui permet d'envoyer un message de manière privé, on peut envoyer un message à une personne qu'on suit. Si cette dernière souhaite nous répondre elle doit nous suivre aussi.
— **Mention :** permet de mentionner une personne sur l'un de nos tweet, la procédure est simple il suffit de précéder le nom de la personne par le symbole "@".
— **Hashtag :**permet d'identifier et de faciliter la recherche de tweet ou twitto qui parle d'un sujet qui nous interesse, il est constitué du symbole "#" et est suivi d'un mot.
— **Tendances :** représentent les sujets en vogue sur Twitter, elles diffèrent d'un twitto à un autre, selon sa localisation et ses abonnements.

1.7.1.2 Les différentes relations sur Twitter

On distingue plusieurs relations publiques sur Twitter : user-to-tweet, tweet-to-user, user-to-user. Elles sont résumées dans le tableau suivant :

	User	Tweet
User	Follows (suit)/Is followed by (suivi par). Mention. Replies to (répond à). Posted by (posté par).	Posts (poste). Retweets. Likes (aime). Replies (répond).
Tweet	Retweeted by (retweeté par). Liked by (aimé par). Replied by (répondu par).	Replies/is replied from. Retweet /is retweeted from.

TABLE 1.1 – Les différentes relations sur twitter

1.7.2 La RI dans Twitter :

Dans cette section, nous nous intéressons à la façon dont on accède à l'information sur Twitter, puis nous nous étalons sur les facteurs de pertinence qui sont liés à ce réseau social.[Damak. 14].

1.7.2.1 Accès à l'information dans les microblogs :

- **La recherche en temps réel :**
  L'utilisateur ici cherche à obtenir une information pertinente et récente, donc la date de publication du document est un facteur de pertinence très important dans la RI en temps réel.
- **La détection d'opinion :**
  Son objectif majeur est de retrouver les documents qui expriment des opinions concernant un sujet ou une requête donnée. Par exemple Shamma et al ont constaté que les tweets peuvent être utilisés pour annoter les débats politiques avec les opinions des téléspectateurs,autrement dit, les tweets peuvent participer à la prédiction et au suivi d'un sujet médiatique important.
- **La détection de tendance :**
  Elle vise à identifier les sujets qui sont au top de l'actualité et ce toujours en temps réel, elle est d'une grande utilité pour les journalistes et les analystes

parce qu'elle leur permet de réagir rapidement sur un sujet.

— **La recherche de microbloggers :**
c'est le fait de rechercher les microbloggers les plus populaires et les plus influents, ils peuvent être des leaders ,influenceurs ou encore des débatteurs.

— **La recherche thématique :**
Sert à classer les utilisateurs en fonctions de leur centre d'intérêt, les sujets discutés sont donc identifiés et classés selon les thématiques abordées. L'une des solutions à ce problème est de regrouper les tweets selon les hashtags qu'il contiennent.

1.7.2.2 Facteurs de pertinence dans les microblogs :

Il existe plusieurs facteurs de pertinence à prendre en considération lors de la conception des approches de recherche de microblogs[Damak. 14], nous distinguons ainsi :

Facteurs de pertinence relatif au contenu :

Nous avons considéré deux facteurs de pertinence relatifs à certaines spécificités de contenu :

— **La popularité du tweet** : Si on trouve plusieurs tweets ayant le même contenu qu'un autre tweet, alors ce dernier est considéré comme étant populaire.

— **La longeur d'un tweet :** on se base sur la longueur du tweet, autrement dit ce facteur comptabilise le nombre de termes dans un tweet, car plus un tweet est long plus il contient de l'information.

Facteurs de pertinence basé sur l'hypertextualité :

ce facteur est basé sur les URLs , on citera :

— **La présence d'URLs dans le tweet :** ce facteur est liés aux URLs. On estime que la présence d'URL dans un tweet a un but informationnel enrichissant .

— **Le nombre d'URLs dans le tweet :** Comptabilise le nombre d'URLs postées dans un tweet.

— **La fréquence de l'URL dans le corpus :**il calcule le nombre de fois que l'URL apparait dans le corpus.

Facteurs de pertinence basés sur les hashtag :

On distingue :

— **La présence de hashtag :** permet de baser la recherche sur le hashtag, si un tweet contient un hashtag on retourne la valeur binaire 1, sinon 0.

— **La fréquence du hashtag du tweet :** il permet de calculer la fréquence du hashtag dans un corpus.

Facteurs de pertinence liés à la qualité du tweet :

Ce facteur prend en considération les critères spécifiques au tweets. On a :

— **Le retweet** : en se base sur le RT d'un tweet, si un tweet a été retweeté cela veut dire que son contenu est potentiellement informatif et intéressant,dans ce cas le message sera précédé par RT.
— **La fraicheur :**Ce facteur n'est rien d'autre que la différence entre date de publication d'un tweet donnée et la date de la soumission de la requête, il est mesuré en secondes.

Facteurs de pertinence reposant sur la popularité des auteurs :

Nous allons tenir compte de deux facteurs spécifiques dans cette section :
— **Le nombre de tweets de l'auteur :**Le but de ce facteur est de mettre en évidence les tweets postés par des auteurs actifs par rapport aux tweets des tweets postés par des auteurs moins actifs.
— **Le nombre de citations de l'auteur :** se base sur les mentions d'un auteur, plus il est mentionné plus il est populaire.

1.7.2.3 Évaluation de la RI dans les microblogs :

Dans cette partie nous allons aborder l'évaluation de la RI selon la tâche TREC microblog [2] ainsi que les mesures d'évaluations les plus utilisées.

TREC microblog :

TREC microblog est une tâche de la compagne TREC qui est consacrée à la RI dans les microblogs. Elle est organisée annuellement depuis sa création en 2011, décrite également comme étant une tâche ad hoc temps réel.

Tweets2011 corpus :

Dans le cadre de TREC2011, Twitter a fourni des identificateurs pour environ 16 millions de tweets échantillonnés entre le 23 janvier et le 8 février 2011. Le corpus est conçu pour être un échantillon réutilisable et représentatif de la twittosphère c-à-d les tweets importants et les tweets spam sont inclus.

Mesures dévaluation :

1. F-mesure :c'est une mesure qui combine le rappel et la précision, elle est défini comme suit : Mesure F = 2*RP/(R+P).
2. La précision moyenne MAP : est utilisée comme une mesure supplémentaire pour évaluer l'efficacité de recherche, tout en tenant compte de la précision, du rappel et du rang des documents.
3. La précision p@30 : est la mesure officielle pour l'évaluation de la tâche de recherche en temps réel dans TREC microblog 2011. Cette mesure évalue la capacité d'un système à retourner les tweets pertinents, parmi les 30 premiers de la liste des résultats.

2. https ://trec.nist.gov/data/microblog.html

1.8 Conclusion

Au cours de ce chapitre, nous avons constaté que la notion de la recherche d'information a considérablement évolué, passant de la RI classique à la RI dans les microblogs.

Nous avons exposé dans un premier temps les notions fondamentales de la RI, à travers les concepts de base de la RI, le processus de recherche , et les modèles de recherche d'information.

Dans un second temps, nous avons défini la recherche d'information dans les microblogs en se focalisant sur Twitter, nous avons également introduit les facteurs de pertinence dans les microblogs les différents aspects de la recherche d'information dans Twitter.

Dans le prochain chapitre, nous nous intéressons en particulier à la recherche d'influenceurs et à la mesure de l'influence des bloggeurs sur Twitter.

Chapitre 2

Mesure de l'influence sociale dans Twitter

2.1 Introduction

De nombreux travaux de recherche ont été entrepris pour tenter de mesurer l'influence d'un bloggueur sur Twitter. Dans ce chapitre, nous définissons les notions d'influence en général, et d'influence sociale, puis nous exposons les travaux de mesure de l'influence dans Twitter.

2.2 L'influence sociale

Le dictionnaire anglais Oxford définit l'influence comme "la capacité de causer un effet sur le caractère, le développement ou le comportement de quelqu'un ou de quelque chose. . . " [1].

L'influence sociale est une relation établie entre une entité A dite "influenceur", et une autre entité B "l'influencé", qui a pour effet de provoquer chez B, une réaction, ou un comportement visés en réponse à une action de A . Dans (D.Cercel et S.trausan-Matu, 2014), les auteurs définissent l'influence sociale comme suit : soit A et B deux utilisateurs d'un réseau social donné, A a un pouvoir sur B, c'est-à-dire que A a la capacité de modifier l'opinion de B de manière directe ou indirecte.

L'influence sociale est une partie intégrante des réseaux sociaux. L'étude de l'influence sociale et l'analyse de sa propagation sur les réseaux sociaux offre de nombreux avantages dont :

1. D'un point de vue social, cela nous amène à mieux comprendre les besoins des utilisateurs ;
2. D'un point de vue services publiques, cela aide à fournir une base théorique pour la prise de décisions ainsi que l'orientation de l'opinion publique ;
3. D'un point de vue veille économique et sécuritaire, Cela aide à promouvoir et instaurer la sécurité nationale, la stabilité économique, le progrès économique, etc.

2.2.1 Propriétés de l'influence

Les propriétés [Peng. 18] associés à l'influence sont les suivantes :

— **Dynamique :**L'influence peut augmenter comme elle peut diminuer .
— **Propagative :** L'influence se transmet d'une personne à une autre sur le réseaux social, créant ainsi une chaîne d'influence ;

1. Oxford Dictionaries, http ://oxforddictionaries.com/definition/influence ?q=influence

— **Transitive :**l'influence est transitive ; si A influence B et que B influence C alors A influence C indirectement ;
— **Mesurable :** l'influence peut être mesuré par un nombre réel continu, appelé valeur d'influence.
— **Subjective :**l'influence est subjective. Par exemple, si A parle d'un produit et si B trouve que les opinions de A sont toujours bonnes, alors B prendra en considération l'avis de A, et si C n'est pas d'accord avec les opinions de A alors il y a de forte chances pour qu'il n'essaye pas le produit ;
— **Asymétrique :**Ainsi, si A influence B , il n'est pas certain que B influence A à son tour ;
— **Sensible aux évènements :**L'influence peut tarder à se construire mais il suffit d'un seul évènement pour la faire basculer et la détruire.

2.3 L'influence sur Twitter

L'influence sur Twitter est devenue un sujet de recherche important, ce qui a suscité l'intérêt de plusieurs experts qui ont proposé plusieurs approches différentes afin de mesurer cette influence, Dans ce qui suit nous définissons qu'est-ce-qu'un influenceur sur twitter, et introduisons une revue de la littérature des approches et mesures proposées.

2.3.1 Définition d'un influenceur sur Twitter

Un twitto est dit influent s'il a la capacité de s'imposer sur le réseau social par ses idées , ses opinions, ses informations véhiculées par ses tweets.

Pour mesurer l'influence d'un Twitto, différentes approches ont été proposées. Dans ce qui suit, nous introduisons une revue de la littérature sur ce sujet.

2.3.2 Approches de mesure de l'influence sur twitter

2.3.2.1 L'approche de Cha et al. :

En 2010 cha et al [Cha 10] ont proposé 3 mesures d'influence :

— **Indegree influence :** désigne le nombre d'abonnés d'un utilisateur et indique directement la taille de l'audience pour cet utilisateur u.

— **Retweeted Influence :**indique le nombre de fois que d'autres utilisateurs ont rediffusé les tweets publiés par un utilisateur.

— **Mention Influence :** désignant le nombre de mentions contenant un nom d'utilisateur et indique la capacité de cet utilisateur à engager d'autres personnes dans une conversation.

Les auteurs ont découvert que bien que les retweets et les mentions soient bien corrélés les uns aux autres, le nombre d'abonnés ne correspondaient pas bien aux deux autres mesures. Basé sur cela, ils ont émis l'hypothèse que le nombre d'abonnés peut ne pas être une bonne mesure d'influence Cette conclusion est également validée par une étude récente (Cataldi & Aufaure 2015).

2.3.2.2 Mesure traditionelle

Les mesures traditionnelles [Fabian Riquelem 16] sont utilisées sur twitter et ont été adoptés par certains chercheurs, comme le cas de la *closeness centrality* (C c) et de la *betweeness centrality* (Cb).

La closeness centrality [Boccaletti S]d'un utilisateur est basée sur le concept de distance minimale, c'est-à-dire le nombre minimum d'arêtes traversées pour aller d'un noeud i à un noeud j. Elle est définie selon l'équation suivante :

$$\mathrm{C}_c = \frac{n-1}{\sum_{i \neq j} (D)_{ij}}$$

La betweenness centrality[G] d'un nœud est basée sur le concept des plus courts qui doivent traverser i pour connecter tous les autres nœuds du réseau, mesurant ainsi sa capacité de communication au sein de ce twitto. Soit bjik le nombre des plus courts chemins du nœud j au nœud k passant par i et bjk le nombre des plus courts chemins passant du nœud j au nœud k. selon l'équation suivante :

$$\mathrm{C}_b = \frac{1}{(n-1)(n-2)} \Sigma_{j \neq i} \Sigma_{k \neq j \text{ et } k \neq i} \frac{b_{jik}}{jk}$$

2.3.2.3 L'approche weng et al.

Dans un ensemble de données issues de Twitter, Weng et al. [Weng J. 10] ont observé que 72.4% des utilisateurs suivent plus de 80% de leurs abonnés, et 80.5% des utilisateurs ont suivi 80% de leurs amis. Ainsi les auteurs ont déduit que deux raisons contradictoires peuvent expliquer une telle réciprocité. Soit la relation est juste due au hasard, car un utilisateur suit quelqu'un par souci de courtoisie. Ou bien, cette relation peut être une preuve de similitude, autrement dit, un twitto

suit un ami parce qu'il s'intéresse à ses publications et vice-versa. Ce phénomène est connu sous le nom de "homophilie".

En se basant sur cette observation, les auteurs ont proposé une approche pour mesurer l'influence sur twitter, connue sous le nom de twitterRank, une extension de PageRank [2] qui exploite la structure du réseau social basé sur les relations d'abonnements. Cet algorithme se distingue de PageRank dans le sens que l'utilisateur fait une recherche spécifique à un sujet c'est-à-dire, la probabilité de transition d'un twitto à un autre est spécifique à un sujet, formellement :

$$P_t(i,j) = \frac{|T_j|}{\sum_{\alpha:\, i\ follows\ j} * T_\alpha} * sim_t(i,j)$$

Où :
$|T_j|$:est le nombre de tweets publiés par j ;
$\sum_{\alpha:\, i\ follows\ j} * T_\alpha$:Le nombre de tweets publiés par tous les amis de i ;
$sim_t(i,j)$:La similarité entre i et j sur un sujet t.

2.3.2.4 L'approche Romero et al.

Les auteurs ont proposé un algorithme inspiré de l'algorithme HITS [3] (Kleinberg, 1999), exploitant la structure du réseau social basé sur la relation de retweet, et la notion de passivité, où les arcs sont pondérés par le ratio de l'influence exercée sur un utilisateur sur sa passivité, un utilisateur passif est un utilisateur qui retweet pas ou rarement les tweets des autres. Ainsi la passivité d'un utilisateur mesure son pouvoir à rejeter l'influence des autres utilisateurs, partant de l'hypothèse qu'un twitto est fortement influent si :

— Il est lui-même passif pour les utilisateurs qui l'influencent.
— Il a une grande influence sur les utilisateurs passifs.

Les auteurs ont proposé un algorithme appelé IP dans lequel un score d'influence et un score de passivité est attribué à chaque utilisateur. L'algorithme IP est similaire à l'algorithme HITS dans la recherche d'autorité et des Hubs qui les lient, tel que : Le score de passivité correspond au score d'autorité et l'influence correspond au score de Hub.[Romero D. M. 11]

2. L'algorithme PageRank développé en 1998 est à l'origine du moteur de recherche Google. Cet algorithme assigne un score à toutes les pages du web indépendamment de toute requête. Quant à la mesure PageRank, c'est une distribution de probabilité sur les pages. Elle mesure la probabilité pour un utilisateur navigant au hasard, d'atteindre une page donnée. Elle repose sur le concept qu'un lien émis par une page A vers une page B est assimilé à un vote de A pour B. Plus une page reçoit de votes, plus elle est considérée importante.

3. Le modèle de Kleinberg à la différence de PageRank distingue les « autorités » (pages recevant beaucoup de liens) des « Hubs » (pages contenant beaucoup de liens vers de bonnes pages). Le moteur de recherche ask.com est basé sur l'algorithme HITS.

2.3.2.5 Approche Anger et kittl.

Dans leur approche[Anger 11] les auteurs se sont basés sur les relations d'abonnement, retweet et de mention. Se basant sur ces métriques, les auteurs ont défini trois paramètres de mesure de l'influence :

- **Le ratio d'abonnement :** définit comme le nombre d'abonnés divisé par le nombre d'abonnements.
- **Le ratio de retweet :** définit comme la somme des nombres de retweets et de mentions divisé par le nombre de tweets publiés.
- **Le ratio d'interaction :** définit par le nombre d'utilisateurs qui retweetent ou mentionnent divisé par le nombre d'abonnés.

Dans leur approche, les auteurs proposent une mesure de l'influence dite SNP (*social networking potential*), calculée comme étant la moyenne du ratio de retweet et d'interaction. Les auteurs ne prennent pas en considération le ratio d'abonnement dans le calcul du SNP car ils ont déduit qu'il n'était pas un facteur déterminant pour le calcul d'influence.

2.3.2.6 L'approche de Ben Jabeur et al.

Dans leur approche[Benjabeur L. 12] les auteurs ont construit le réseau social twitter en se basant sur les relations de retweet et de mention, et à partir de ces deux métriques les auteurs ont proposé un algorithme similaire à PageRank : InfRank .

Dans le calcul de InfRank le poids d'un arc allant d'un utilisateur i à un utilisateur j est calculé par la proportion du nombre de fois que i retweet j par rapport au nombre total de tweets publiés par i, les auteurs ont ensuite attribué à chaque utilisateur du réseau un score d'influence initiale basé sur sa popularité (nombre d'abonnés), puis calculé InfRank qui est la probabilité de transition de i à j qui se propage par le biais de retweet.

2.3.2.7 Ding et al.

Les auteurs ont implémenté l'algorithme spreadRank [Ding 13] inspiré de PageRank qui exploite les relations de retweet dans le réseau social d'influence. Les auteurs ont mesuré la propagation des utilisateurs sur une cascade d'information dont les arcs sont pondérés par le rapport retweets/nombre total de tweets publiés par l'utilisateur retweeté. La figure suivante illustre un exemple d'une cascade d'information de profondeur quatre, tout en tenant compte que la propagation des nœuds supérieurs est plus élevée en comparaison à celle des nœuds inférieurs :

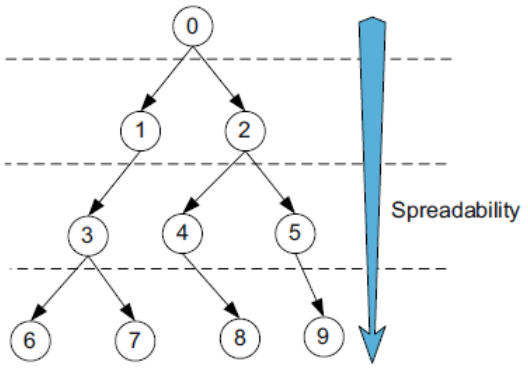


FIGURE 2.3.1 – exemple de cascade d'information

Les auteurs ont suggéré qu'un utilisateur est plus influent si ses tweets sont retweetés très tôt après leurs diffusions. La figure suivante compare deux intervalles de temps de rediffusion :

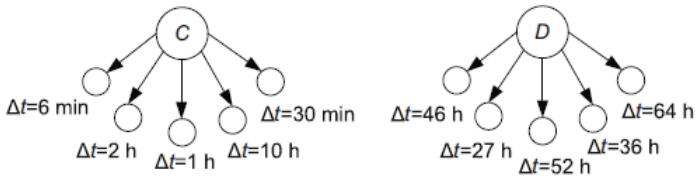


FIGURE 2.3.2 – Un exemple de l'intervalle de temps-ainsi on déduit que l'utilisateur C diffuse l'information beaucoup plus rapidement que l'utilisateur D

L'algorithme spreadRank tient en compte la position de l'utilisateur dans la cascade d'information ainsi que l'intervalle de temps entre les retweets. Les auteurs ont combiné ces deux mesures pour calculer la probabilité de transition de chaque utilisateur. Une grande propagation est équivalente à un minimum d'intervalle de temps et une pondération élevée.

2.3.2.8 L'apporoche de Sung et al.

Dans leurs approches [Sung J. 13] les auteurs ont proposé un algorithme similaire à PageRank nommé InterRank, qui exploite les relations d'abonnements dans le réseau social twitter. Les auteurs ont confirmé que les gens préféraient interagir avec des personnes similaires (même sujet d'intérêt). Ainsi cette similarité est liée à la diffusion d'influence dans le réseau social de twitter. Formellement :

$$InterRank(p_i) = \frac{1-d}{N} + d \sum_{pj \epsilon M(pi)} CS(p_i, p_j) * \frac{InterRank(p_j)}{L(p_j)}$$

Tel que :
— Pi : représente un utilisateur de twitter ;
— M(pi) : représente l'ensemble des suiveurs qui suivent un utilisateur pi ;
— L(pi) : le nombre d'amis d'un utilisateur twitter pj ;
— CS (pi, pj) : représente le *cosine similarity* entre les utilisateurs pi et pj ;
— d est un paramètre généralement fixé à 0.85 ;
— N nombre de nœuds du réseau.

2.3.2.9 L'approche Azaza et al.

Les auteurs ont introduit une mesure multicritères de l'influence qui consiste à combiner sept métriques : la réponse, les favoris, les retweets, le nombre de followers, le hashtag, la mention et le lien URL. [Azaza L. 15].

2.3.2.10 F.Boubekeur, M.Ferrouk et L.Belkacemi

Dans cette approche, une nouvelle mesure d'influence a été proposée : le ratio d'influence. Le ratio d'influence est définit comme étant le rapport entre la capacité d'un blogueur à influencer (ie. son influence imposée) et son exposition à l'influence (ie. influence subie) [F. Boubekeur. 17], prenant en considération la relation de retweet dans leur réseau d'influence.

Le ratio d'influence est ainsi défini :

$$\tau_{inf}(u_i) = \frac{influence\ imposée\ de\ u_i + 1}{influence\ subie\ de\ u_i + 1} + \mu$$

Où :
— $\tau_{inf}(ui)$:ratio d'influence du blogueur ui dans le réseau social d'influence ;
— $\mu = 0.05$:constante utilisée pour assurer la convergence du calcul récursif du ratio d'influence ;
— le " + 1 "dans la formule évite la division par zéro ;
— L'influence imposée de u_i : comme la capacité du blogueur u_i à faire propager ses tweets à travers le réseau d'influence par d'autres blogueurs (retweet). Formellement :

$$influence\ imposée\ de\ u_i = \sum_{u_j : u_j retweet\ u_i} w(u_i, u_j) * \tau_{inf}(u_j)$$

Où :

$$w(u_i, u_j) = \frac{nombre\ de\ tweets\ publiés\ par\ u_i\ et\ rediffusés par\ u_j}{nombre\ de\ tweets\ publiés\ par\ u_i}$$

— L'influence subie u_i : comme étant l'influence que d'autres blogueurs font subir à un blogueur u_i qui a rediffusé au moins une fois l'un de leurs tweets. Formellement :

$$influence\ subie\ de\ u_i = \sum_{u_k:u_i retweet\ u_k} w(u_k, u_i) * \tau_{inf}(u_k)$$

Où :

$$w(u_k, u_i) = \frac{nombre\ de\ tweets\ publiés\ par\ u_k\ et\ rediffusés\ par\ u_i}{nombre\ de\ tweets\ publiés\ par\ u_k}$$

2.3.2.11 L'approche Kwak et al.

Ici, les auteurs ont comparé trois mesures d'influence différentes : le nombre d'abonnés, pageRank et le nombre de retweets, ils ont finit par déduire que le classement des utilisateurs les plus influents, différait selon la mesure [Haewoon Kwak 10]. Pour ce fait, ils ont considéré 20 célébrités de différents domaine (acteurs, musiciens, politiciens...), ils ont déduit que :

— Le nombre de followers favorise plus les acteurs, musiciens et les animateurs TV ;
— PageRank favorise : les acteurs, présidents, et news ;
— Le nombre de retweets favorise les news.

2.4 Conclusion

Dans ce chapitre nous avons pu voir l'influence sociale sur les plateformes de microbloggings, par la suite nous nous sommes basées sur l'influence sur Twitter en relatant plusieurs études phares en relation avec cette thématique.

Le prochain chapitre portera sur l'approche que nous proposons.

Chapitre 3

L'approche proposée

3.1 Introduction

Dans le chapitre précédent nous avons passé en revue de nombreuses approches de mesure de l'influence sur Twitter. Le principe fédérateur de ces approches est principalement de combiner plusieurs signaux sociaux afin de tenter de mesurer l'influence d'un blogueur sur Twitter.

Dans ce qui suit nous présentons notre contribution à la définition d'une mesure de l'influence que nous utilisons pour définir un modèle de RI social. Notre approche combine 3 métriques basées sur les abonnés/abonnements, et sur les retweets.

3.2 Description de l'approche proposée :

3.2.1 Modélisation du réseau social Twitter :

Le réseau social d'influence est construit de par les utilisateurs qui sont liés les uns aux autres par des relations (retweet, abonnement, mention, ...). Dans notre cas, nous nous sommes intéressées à une instance de ce réseau dite le réseau social de l'influence.

Le réseau social de l'influence dans twitter est modélisé par un graphe G=(U,E) où U et E=U x U représentent respectivement l'ensemble des twittos et les relations d'influence entre eux. Les relations d'influence considérées sont les relations de retweets et d'abonnements.

3.2.2 Mesure de l'influence d'un Twitto :

L'influence d'un Twitto permet de rendre compte de la notoriété et de la popularité d'un blogueur sur le réseau social. L'objectif de la mesure de l'influence est de pouvoir quantifier cette notoriété. Plusieurs approches ont été passées en revue en chapitre 2, qui proposent de mesurer l'influence d'un Twitto en se basant sur différents signaux sociaux issus des relations sociales sur Twitter. Ainsi, le nombre de mentions , le nombre d'abonnés et le nombre de retweets (Cha et al. 2010) sont autant de signaux sociaux utilisés.

Pour mesurer l'influence sur Twitter, nous proposons pour notre part d'utiliser et de combiner les relations d'abonnement et de retweet dans un score d'influence unique. L'intuition derrière l'utilisation de ces deux signaux vient de l'observation que :

— Si un twitto est influent alors le nombre de ses abonnées est relativement élevé par rapport à ses abonnements. Ainsi, un Twitto influent est plus «

suivi » que « suiveur ». De plus, si un Twitto est plus « suivi » que « suiveur » alors il est très probablement un blogueur influent.
— Si un twitto est influent alors le nombre de retweets liés à ses tweets est probablement considérablement élevé. De même, si les tweets d'un blogueur sont fortement retweetés, ce blogueur devient par ce fait influent.

De ces observations nous est venue l'idée de proposer les mesures suivantes :

3.2.2.1 Le ratio d'abonnement :

C'est le ratio des abonnements aux abonnés d'un blogueur donné u. Formellement :

$$Ra(u) = \frac{nombre\ d'abonnées(u)+0.5}{nombre\ d'abonnements(u)+0.5}$$

Où :
— 0.5 est une valeur de lissage qui évite une division par zéro.

Un ratio supérieur à 1 signifie que le blogueur est plus « suivi» que suiveur ». Le ratio d'abonnement ainsi défini est proportionnel à l'influence du blogueur.

3.2.2.2 Le taux de retweets :

Pour le ratio de retweet, nous nous sommes inspirées d'un indicateur bibliométrique connu, le g-index [1] , que nous avons adapté aux tweets et retweets. Le g-index sert à quantifier la productivité d'un chercheur, il a été suggéré en 2006 par Leo Egghe.

L'idée du g-index est de donner un poids plus important aux articles(ou publications) qui sont le plus citées, et ce même si le chercheur n'a pas un grand nombre de publications. Son fonctionnement est simple, étant donné un ensemble d'articles classés par ordre décroissant du nombre de citations qu'ils ont reçues, l'indice g est le plus grand nombre unique, tel que les premiers articles g reçoivent ensemble au moins g^2 citations, formellement :

$$g^2 \leq \Sigma_{i \leq g}\, c_i$$

tel que :

— C_i représente le nombre total de citations.

1. https ://jennydelasalle.wordpress.com/2016/05/25/explaining-the-g-index-trying-to-keep-it-simple/

Par exemple un indice g de 20 signifie qu'un chercheur a publié au moins 20 articles qui, au total, ont reçu au moins 400 citations.

Par analogie, et en considérant les tweets comme des publications d'un chercheurs, et les retweets comme des citations (ou des références) à des publications (ou postes) d'un twitto, nous définissons un nouvel indicateur d'influence, le t-index, qui mesure la productivité d'un twitto donné. Comme pour le g-index, le t-index favorise les tweets qui ont reçu un plus grand nombre de retweets.
Formellement :

$$t - index(u) = max(t) |\, t^2 \leq \sum_{i \leq t} Retweet_i$$

Où :
— u représente un twitto ;
— t représente le nombre maximum de tweets publiés tel que ensemble ces tweets ont été retweetés au minimum t^2fois ;
— i représente le numéro du tweet.

3.2.2.3 Mesure d'influence combinée :

Nous proposons de combiner les deux ratios précédents en un score additif pondéré comme suit :

$$Infl(u) = \delta Ra(u)/max(Ra(U)) + (1 - \delta)(t - index(u)/max(t - index(U))$$

Où :
— *Infl(u)* : représente la mesure de l'influence du twitto u en fonction de son ratio d'abonnement et de son ratio de retweets ;
— δ :Paramètre permettant d'ajuster l'importance du ratio d'abonnement par rapport au ratio de retweets. Ce paramètre est compris entre 0 et 1. Il peut être déterminé expérimentalement ;
— $max(Ra(U)$ et $max(t - index(U)$ sont utilisés pour la normalisation. Ils représentent respectivement le nombre maximal des ratios d'abonnements et le nombre maximal des t-index.

3.2.3 Exemple explicatif :

Pour illustrer le pertinence de notre score d'influence, nous considérons deux twittos ayant le même nombre d'abonnés et d'abonnements, soit 1200 abonnées et 100 abonnements chacun. Pour appliquer notre ratio d'influence, nous avons

supposé les données citées dans les tableaux 3.1 et 3.2, et nous avons calculé manuellement les valeurs correspondantes.

Remarques :

Pour l'expérimentation nous avons fixé δ=0.5 ;

Les tweets sont classés par ordre décroissant par rapport au nombres de leur retweets.

Twitto 1 :

Tweets	Retweet
t_1	20
t_2	18
t_3	11
t_4	1
t_5	1
t_6	0

TABLE 3.1 – Nombres de retweets des tweet du twitto1

Twitto 2 :

Tweets	Retweet
t_1	10
t_2	5
t_3	5
t_4	1
t_5	1
t_6	0
t_7	0
t_8	0

TABLE 3.2 – Nombres de retweets des tweet du twitto 2

Observation :

Avant d'appliquer notre ratio d'influence, on constate rien qu'on observant que le twitto 1 devrait être plus influent que le twitto2, et ce, malgré le fait que le twitto2 comptabilise plus de tweets que le twitto1, mais ce qui fait la force de ce dernier, c'est que ses tweets sont fortement retweetés, et donc il a plus d'impact que le twitto2.

Application du ration d'influence :

Dans cette partie, nous allons utiliser le ratio d'abonnement et le t-index définis plus haut, afin de pouvoir déterminer qui des twittos sont les plus influents.

Twitto 1 :

Tweets	Retweet	t^2	somme	t-index	Ra(u)	Ratio d'influence
t_1	20	1	20	5	1200.5/100.5= 11.94 (Formule 1)	11.94*0.5+0.5*5= 8.47 (Formule 3)
t_2	18	4	38			
t_3	11	9	49			
t_4	1	16	50			
t_5	1	25	51			
t_6	0	36	51			

TABLE 3.3 – Application de la lois de calcule au twitto 1

Twitto 2 :

Tweets	Retweet	t^2	somme	t-index	Ra(u)	Ratio d'influence
t_1	10	1	10	4	1200.5/100.5= 11.94 (Formule 1)	11.94*0.5+0.5*4= 7.97 (Formule 3)
t_2	5	4	15			
t_3	5	9	20			
t_4	1	16	21			
t_5	1	25	22			
t_6	0	36	22			
t_7	0	49	22			
t_8	0	64	22			

TABLE 3.4 – Application de la lois de calcule au twitto 2

Comparaison des résultats obtenus :

D'après les tableaux ci-dessus, le nombre de tweets du Twitto 2 dépasse celui du twitto 1, mais d'après les calculs, le ratio d'influence du twitto 1 est supérieur à celui du twitto 2, donc notre ratio d'influence favorise l'utilisateur qui a eu le plus d'impact en retweet, en considérant toujours le même nombre d'abonnés et d'abonnements.

3.3 Vers un nouveau modèle de recherche d'information social : Utilisation du ratio d'abonnement dans le modèle de recherche

Nous proposons d'utiliser notre mesure d'influence dans un contexte de recherche d'information sociale dans Twitter. La problématique consiste à retrouver, parmi une collection de posts (ou tweets), l'ensemble des tweets qui répondent à une requête donnée, la requête portant sur un sujet quelconque, est donnée sous forme textuelle, ou comme une suite de mots clés. La recherche sociale permet de tenir compte, en plus de la pertinence thématique du tweet pour la requête, de sa pertinence sociale.

— La pertinence thématique d'un tweet est mesurée comme étant le degré de similarité du tweet avec la requête, elle peut être calculée par tout modèle classique de la RI adapté à la recherche de texte court.
— Dans notre approche, la pertinence sociale d'un tweet est liée l'influence de son auteur mesurée par notre ratio d'influence comme défini en formule (3).

Nous avons choisi de définir un modèle de recherche social qui associe la pertinence thématique d'un tweet à sa pertinence sociale dans un score linéaire comme suit :

$$Rel(Q,t) = \alpha RSV(Q,t) + (1-\alpha)Infl(u)$$

Avec :

— RSV(Q,t) représente la pertinence thématique du tweet t pour la requête Q ;
— Elle est calculée comme le degré de correspondance du tweet pour la requête. α est un paramètre compris entre 0 et 1.

3.4 Conception d'une solution de RI basé sur le facteur d'influence

Afin de mettre en œuvre de notre approche précédemment introduite , nous tâchons d'explorer les différents aspects à considérer en vue de cette finalité.

Dans le domaine qui nous intéresse, à savoir celui de la RI, il nous faut nous attarder sur les deux phases primordiales de toutes application de ce type respectivement celle de l'indexation des documents et celle de recherche.

Dans le but de synthétiser les données nécessaires à la réalisation de notre projet, et afin de simplifier la modélisation de chacun des aspect de notre application, nous avons opté pour l'emploi du langage de modélisation unifié UML. Il s'agit d'un langage de modélisation graphique basé sur l'emploi de pictogrammes et fournissant une méthode normalisé pour la visualisation de la conception d'un système donné.

Notre application ne se distinguant pas de part les cas d'utilisations (perçus par un utilisateur classique) des système de recherches d'informations mais s'illustrant davantage par le domaine d'application qu'il traite (les tweets et utilisateurs réseau sociale Twitter) et la méthode de recherche et de classement de résultats qu'il offre ; le diagramme des cas d'utilisation qui en résulte est une simple redite de ceux déjà existant pour ce type de projet.

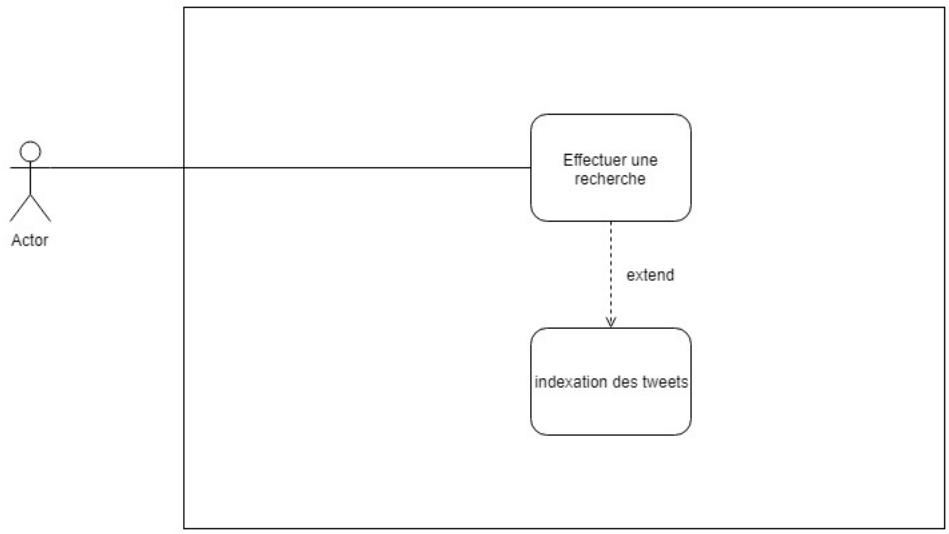


FIGURE 3.4.1 – Diagramme de cas d'utilisation

Étant donné la nature de notre application il nous a semblé opportun de subdiviser celle-ci en deux modules un premier consacré à l'indexation des documents, et un second chargé d'exécuter les recherches, afin d'illustrer cette décision architecturale nous avons opté pour l'emploi de l'un des diagrammes offerts par UML ,le diagramme de packages (il est à noter que les packages offerts par les bibliothèques et la sdk java ne sont pas représentés pour des raisons de lisibilité).

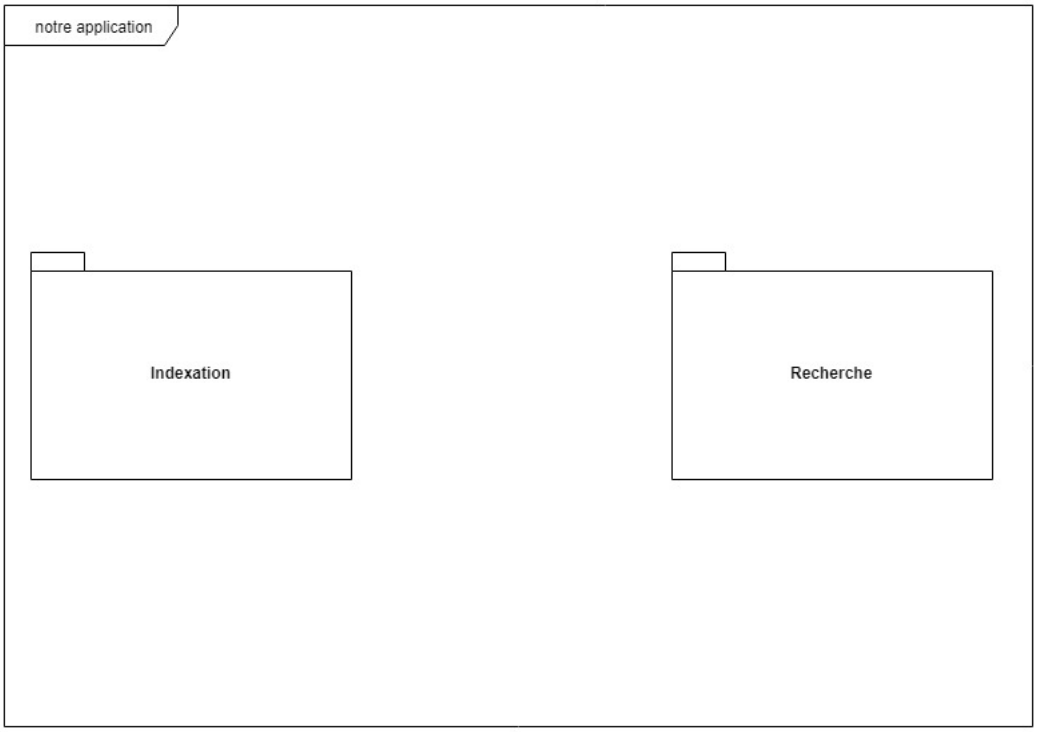


FIGURE 3.4.2 – Diagramme de package de l'implémentation

Afin de mieux nous représenter les traitements effectués par notre système dans le temps nous avons représenter le déroulement d'une séquence de recherche complète et d'une séquence d'indexation exécutée au préalable via l'emploi de diagrammes de séquence permettant de représenter le déroulement des traitements et les interactions entre les éléments de notre système.

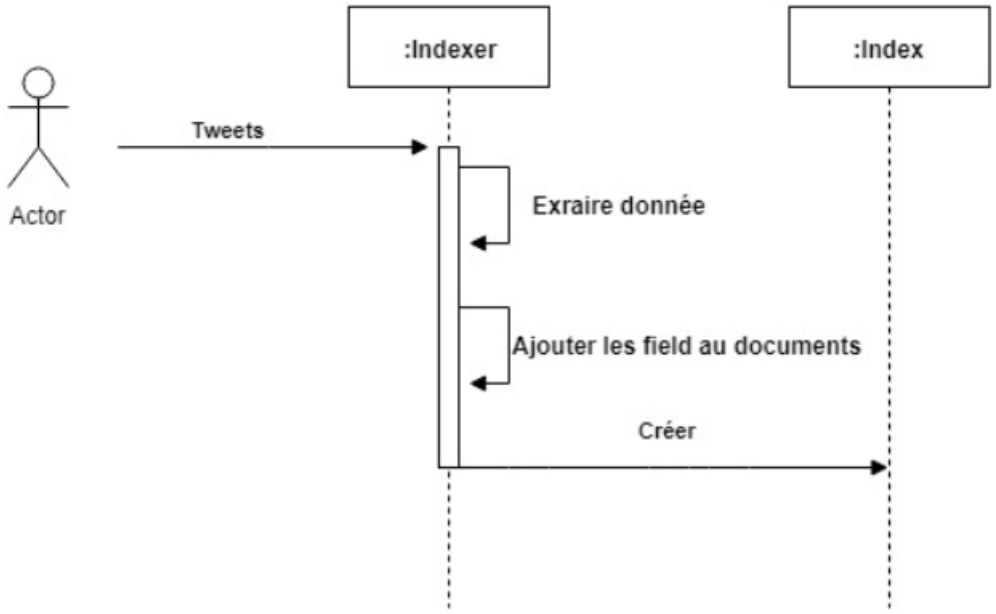


FIGURE 3.4.3 – Diagramme de séquence de l'indexation

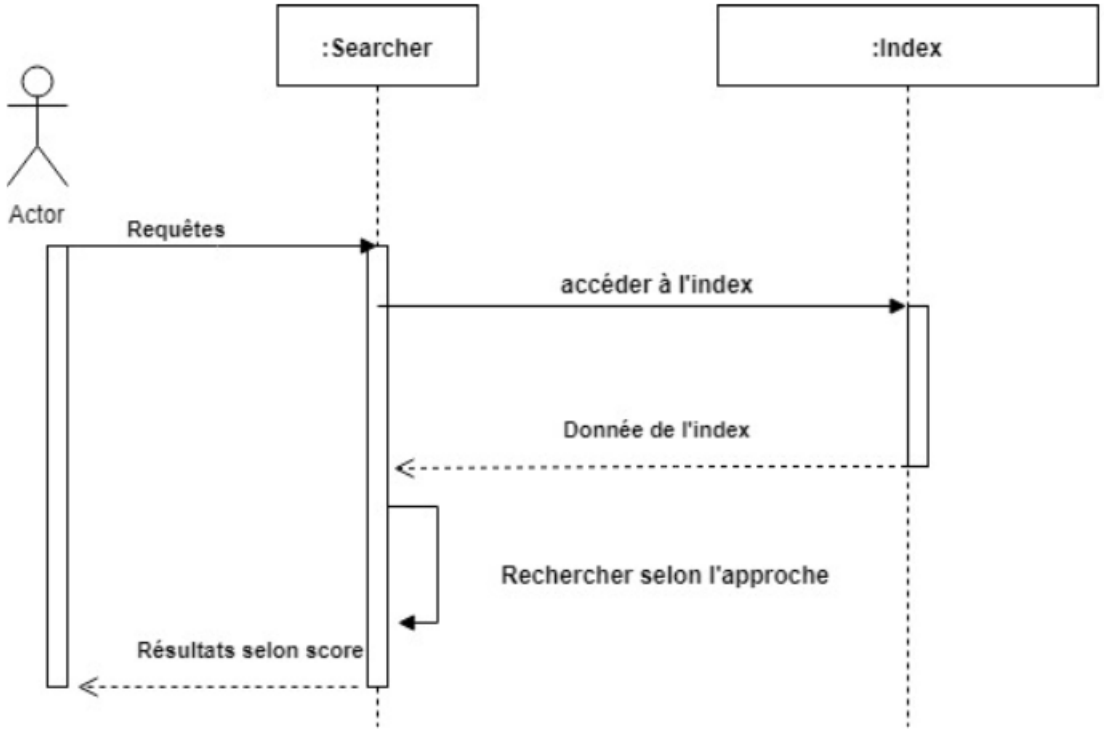


FIGURE 3.4.4 – Diagramme de séquence de la recherche

3.5 Conclusion

Au long de ce chapitre, nous avons décrit notre approche. Nous avons défini dans un premier temps le ratio d'abonnement qui est le rapport entre le nombre d'abonnés et d'abonnements, par la suite nous nous sommes inspirées du g-index pour définir le t-index qui se base sur le nombre de retweets.

Chapitre 4

Implémention et évaluation

4.1 Introduction

Dans ce chapitre, nous présentons les différents outils utilisés pour l'implémentation de notre approche. Nous exposerons également quelques résultats de tests.

4.2 Outils de développement

Pour implémenter notre approche, nous avons été amenées à étendre certaines classes java de la plateforme lucene. Pour réaliser nos tests nous avons utilisés la collection trec microblog 2011 dont le corpus est composé d'indentifiants de tweets. Pour extraire ces tweets de Twitter, nous avons utilisé l'api twitter 4j, les tweets étant fournis en format json, nous avons été amenées à utiliser l'api jackson pour les indexer sous java (sous lucene). L'évaluation des résultats est réalisée sous trec_eval. Notre implémentation s'articule ainsi autour des outils suivants :

4.2.1 Eclipse IDE

Eclipse est un environnement de développement (IDE) historiquement destiné au langage Java, mais grâce à un système de plugin il peut être utilisé avec d'autres langages de programmation comme le PHP et le C/C++.

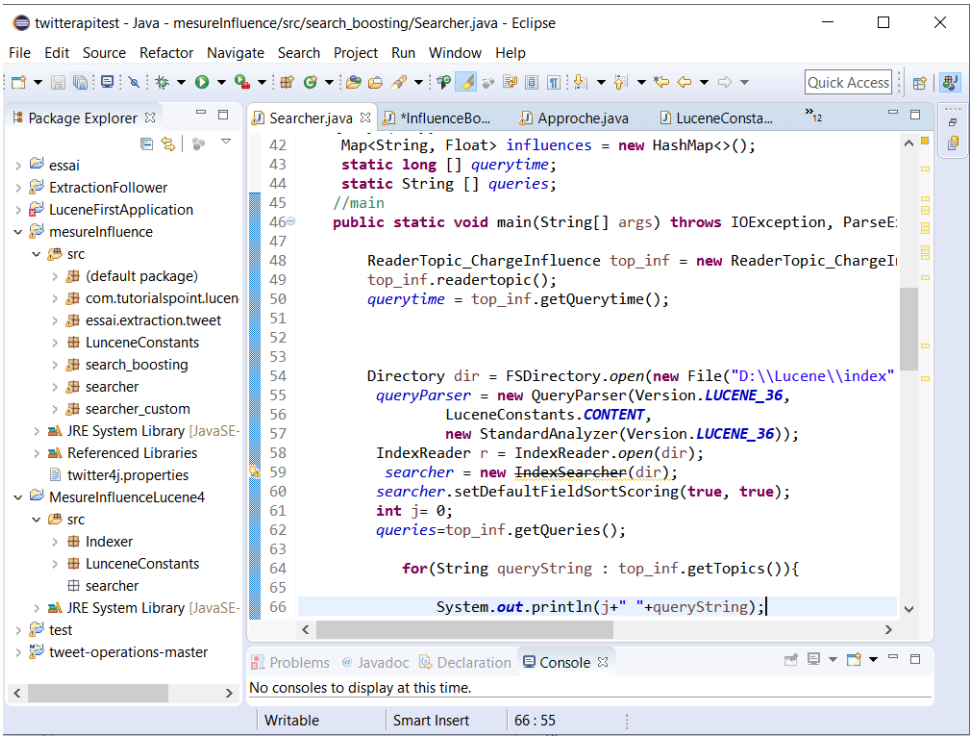


FIGURE 4.2.1 – Interface de l'IDE eclipse

4.2.2 Langage java

Le langage java est un langage de programmation informatique orienté objet créé par James Gosling et Patrick Naughton, employés de Sun Microsystems. La particularité de java est qu'il est facilement portable sur d'autres systèmes d'exploitation tels que linux, Windows, Mac... avec peux ou aucune modification.

4.2.3 L'API jackson

Jackson est une bibliothèque très populaire et efficace basée sur Java, elle permet de sérialiser ou de mapper des objets Java sur JSON et inversement. Notre choix s'est porté sur cette API car il présente les avantages suivant :

- Elle est facile à utiliser ;
- Elle fournit un mappage par défaut pour la plupart des objets à sérialiser ;
- Elle est très performante ;
- Elle crée des résultats JSON propres et compacts, faciles à lire.
- Elle ne nécessite aucune autre bibliothèque que jdk ;
- Elle est open source.

4.2.4 L'API twitter 4j

Twitter propose plusieurs APIs permettant d'accéder à ses services, cela va des opérations de consultation de comptes (tweets, listes d'amis et de followers, etc) aux opérations de modification (supprimer des amis, poster des tweets, etc). Parmi ces APIs, l'API twitter 4j, cette dernière est une librairie facilitant l'utilisation des API Twitter (autant REST que Streaming), elle est ecrite en Java5 et peut donc peut fonctionner sur une JVM classique ainsi que sur Android.

4.2.5 Lucene 3.6

Lucene est un moteur de recherche [1] textuelle développé sous java par la fondation apache se concentrant sur l'indexation et la recherche de textes.

4.2.5.1 Architecture de Lucene :

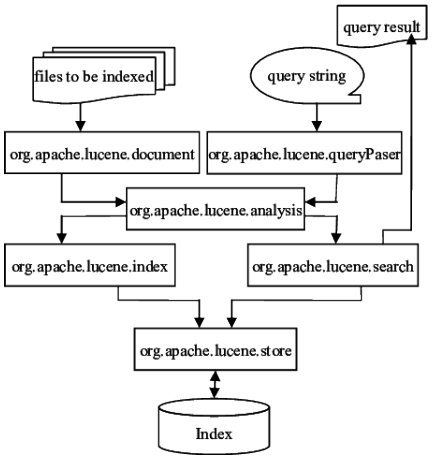


FIGURE 4.2.2 – Architecture de Lucene

Lucene se découpe en 7 paquetages principaux :

— Org.apache.lucene.analysis : il contient du code afin de convertir du texte en élément indexable. Il contient la classe Analyzer qui permet d'extraire les mots importants pour l'index et supprimer le reste.

— Org.apache.lucene.document : contient des classes relatives aux documents, tel que la classe : Document. Cette classe représente un rassemblement de champs(Fields), ainsi les métadonnées sont indexées et stockées séparément comme des champs d'un document.

1. https ://lucene.apache.org/

— Org.apache.lucene.index : il contient le code pour accéder aux index. On y trouve la classe IndexWriter qui permet de créer un index et d'ajouter des documents dans un index existant.
— Org.apache.lucene.queryparser : son rôle est d'analyser les requêtes afin de générer la requête sous forme d'objet query qui pourront ensuite être réutilisé par le parseur. On y trouve la classe la classe QueryParser qui est utilisé pour générer un décompositeur analytique qui peut chercher à travers l'index.
— Org.apache.lucene.search : il se charge de fournir les objets pour chercher dans les indexes. Il fourni les classes suivantes :
  — IndexSearcher : la classe IndexSearcher est la classe qui se charge de l'ouverture de l'index en lecture seulement.
  — Query : c'est la méthode la plus basique d'interrogation de lucene, elle est utilisée pour égaliser les documents qui contiennent des champs avec des valeurs spécifiques.
  — Hits : la classe Hits est un conteneur d'index pour classer les résultats de recherche de document. Pour des raisons de pertinences, les exemples de classements ne chargent pas tous les documents de l'index pour une requête donnée, mais seulement une partie d'entre eux.
— Org.apache.lucene.store : représente une couche d'abstraction d'entrée sortie. On y trouve les classes :
  — Directory : Les fichiers peuvent être écrits une fois, lorsqu'ils sont créés. Une fois qu'un fichier est créé, il ne peut être ouvert qu'en lecture ou supprimé. L'accès aléatoire est autorisé à la fois en lecture et en écriture.
  — FSDirectory : c'est une classe qui étend de la classe Directory, elle sert à stocker des fichiers d'index dans le système de fichiers.
— Org.apache.lucene.util : les classes sont utilisées dans les autres paquetages. Par exemple on y trouve la classe Version qui permet de préciser la version de lucene utilisée.

4.2.5.2 La recherche sous lucene :

Avant de soumettre une requête à lucene, il faut d'abord passer par la phase d'indexation. Le schéma suivant illustre le processus d'indexation et les classes utilisées.

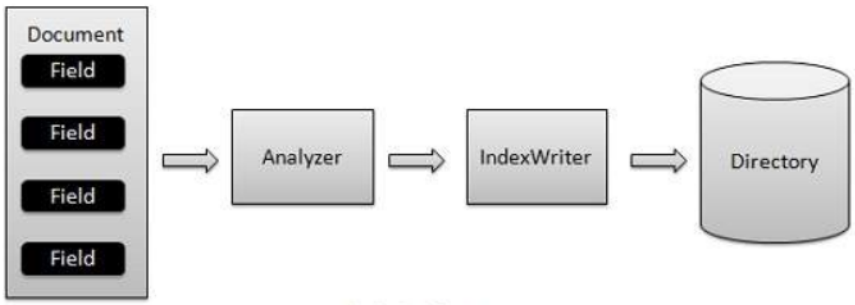


FIGURE 4.2.3 – Processus d'indexation

La figure 1.3 met en oeuvre deux principales classes : Analyzer et IndexWriter. Le fonctionnement est le suivant :

Nous ajoutons le ou les documents contenant le ou les champs à IndexWriter qui analyse le ou les documents à l'aide de l' Analyzer , puis on crée, ouvre ou édite les index selon les besoins et on les stockeou met à jour dans un répertoire (Directory). IndexWriter est utilisé pour mettre à jour ou créer des index et c'est la classe la plus importante du processus d'indexation.

Une fois l'index crée, nous pouvons effectuer une recherche sur une requête.

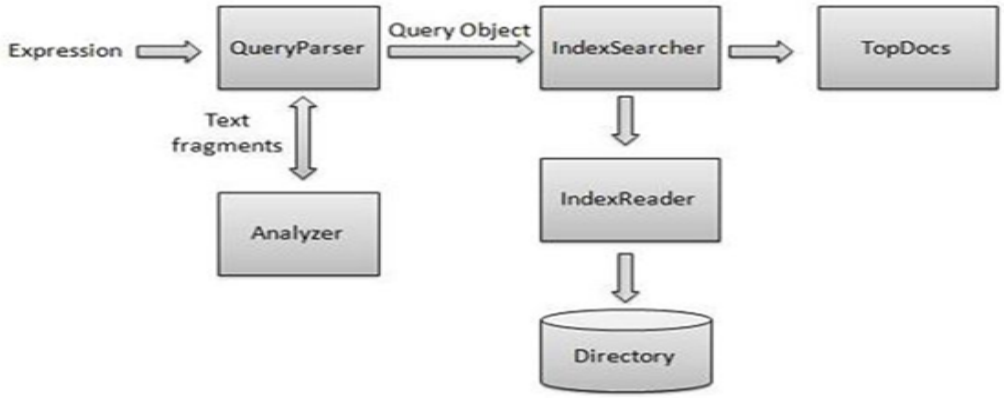


FIGURE 4.2.4 – Processus de recherche

Le processus de recherche met en œuvre les classes des packages org.apache.lucene.search et org.apache.lucene.queryparser :

— IndexSearcher : c'est la classe qui donne accès aux indexes en recherche.
— Analyzer : fait partie du processus de recherche pour normaliser les critères de recherche.
— QueryParser : analyseur de requêtes.

— Query : représente la requête de l'utilisateur et elle est utilisée par un index-Searcher.
— Hits : une collection d'éléments résultats de la recherche.
— Hit : un élément de la collection des résultats. Document : c'est l'unité contenant l'information.

4.2.6 La collection TREC microblogs2011

Dans ce cadre de la recherche d'information, TREC microblog2011 [2] fourni un corpus de tweets du 23 au 8 février 2011 classés par ordre chronologique. Ces tweets peuvent être récupérés à partir de twitter tool [3]. TREC microblog2011 fourni également :
— 50 requêtes
— Les jugements de pértinence associés à ces requêtes.

4.2.7 Trec eval

Trec_eval est un outil utilisé pour évaluer les classements de document triés par pertinence. L'évaluation repose sur deux fichiers :
— Qrels (pertinence des requêtes) : répertorie les jugements de pertinence pour chaque requête ;
— Résultats : contient le classement des documents renvoyés par votre SRI.
Trec_eval est téléchargeable à partir de :"https ://trec.nist.gov/trec_eval/".

Utilisation :

Pour pouvoir executer Trec_eval, il vous suffit de taper la commande suivante :

$ ./trec_eval [-q] [-m measure] qrel_file results_file

Où :
trec_eval : est le nom du programme exécutable ;
-q : c'est pour pouvoir matcher les qrels et les résultats ;
qrel_file : est le chemin du fichier avec la liste des documents pertinents pour chaque requête ;
-m : affiche uniquement une mesure spécifique ("-m all_trec" affiche toutes les mesures, " -m official "est le paramètre par défaut qui affiche uniquement les mesures principales) ;
results_file : chemin du fichier avec la liste des documents récupérés par votre application.

2. https ://trec.nist.gov/data/microblog2011.html
3. https ://github.com/lintool/twitter-tools

4.3 Aperçu de notre implémentation

4.3.1 Notre collection de tests

Afin de réaliser notre implémentation, nous avons utilisé une collection réduite de la collection TREC microblogs 2011. Cette collection contient :

— 40603 tweets ;
— 49 requêtes ;
— Des jugements de pertinence associés à ces requêtes.

De cette collection nous avons extrait une sous collection composée de :

— 5 requêtes ;
— Les tweets associés à ces requêtes, pour plus de réalité nous avons ajouté des tweets aléatoires à la collection.
— Les jugements de pertinences associés à ces requêtes.

4.3.2 Les classes implémentées

Pour implémenter notre approche nous avons etendu Lucene avec les deux classes suivantes : Approche et InfluenceBoosting.

4.3.2.1 La classe Approche :

Dans cette classe, nous avons implémenté une fonction qui retourne le t-index. Son fonctionnement est simple, on lui soumet l'ID de utilisateur et à partir de cet ID, la fonction calcule le t-index et le retourne. Cette fonction est appelée à partir de l'indexer puis stocké dans un field lors de la phase d'indexation.Voici un aperçu ce cette classe.

```
public static int t_index( Long idau) throws JsonParseException, JsonMappingException, IOException {

    //Map <Long,ArrayList> t= new  HashMap<Long, ArrayList>();

      File f = new File("D:\\Lucene\\data1\\");
      String line;
      String line1;
       ObjectMapper objectMapper = new ObjectMapper();
        objectMapper.configure(DeserializationConfig.Feature.FAIL_ON_UNKNOWN_PROPERTIES, false);


        int i=0;
        int i2=0;
        Long c=(long) 0;
        Long cent=(long) 0;
        ArrayList values = new ArrayList<Long>();
        ArrayList value = new ArrayList<Long>();
        long idtt=idau;
        for(File file : f.listFiles())
        {
               System.out.println(file.getName());
                BufferedReader reader = new BufferedReader( new FileReader(file));
                while((line = reader.readLine())!=null){
                    //System.out.println("je suis dans la boucle 1");
                    Object object;
                     object=    objectMapper.readValue(line, Object.class);
                    Long id = object.getUser().getId();
                    Long count= object.getRetweet_count();
                    if(id==idtt){
                        //System.out.println("je suis dans la boucle 2");
                        i=i+1;
                        c= count+c;
                        values.add(count);}
```

FIGURE 4.3.1 – Récupération des des valeurs de retweets de chaque tweet d'un twitto

Une fois les tweets récupérés, on les ajoute dans un ArrayList, pour pouvoir déterminer le t-index par la suite. Voici la boucle qui s'en charge.

```
int t_index=0;
Long val= (long)0;
Collections.sort(values);
 Collections.reverse(values);

 Long index = (Long) values.get(0);

 for(int i5 = 1; i5 < values.size(); i5++){

       index = (Long)((Long)(values.get(i5))+index);
    if(values.get(i5)== val ){
         break;
      }
      if(i5*i5 <=index){
          t_index=i5+1;

      }
 }
```

FIGURE 4.3.2 – Boucle qui calcule le t-index

4.3.2.2 La classe InfluenceBoosting :

Cette classe étend de la classe CustomScoreQuery de lucene, c'est dans cette dernière que l'ajout du score social au score thématique se fait. Pour cela, on commence par récupérer les fields de l'index que nous utilisons dans l'approche,

tel que : l'ID de l'auteur, le nom de l'auteur, le t-index, puis à partir de ces valeurs, nous ajoutons le score social de chaque tweet à son score thématique et nous le retournons. Il faut savoir que cette classe est appelé au moment de la phase de recherche. Voici un aperçu de cette classe :

```
public RecencyBooster(IndexReader r) throws IOException  {
        super(r);

        values = FieldCache.DEFAULT.getStrings(r, LuceneConstants.AUTH);
       // iddocs = FieldCache.DEFAULT.getLongs(r, LuceneConstants.ID_TWEET);
        Ra=FieldCache.DEFAULT.getFloats(r, LuceneConstants.RATIO_A);
        Tindex=FieldCache.DEFAULT.getInts(r, LuceneConstants.T_INDEX);
        id=FieldCache.DEFAULT.getLongs(r, LuceneConstants.ID_AUTHEUR);
    }
```

FIGURE 4.3.3 – Fonction qui récupère les valeurs des fields

```
public float customScore(int doc, float subQueryScore,float valSrcScore) throws JsonParseException, JsonMappingException, IOException {

        Double score;
        double max=0;


        //float ra = Ra[doc];
        String auth= values[doc];
        int t= Tindex[doc];
        Long idau=id[doc];


        double ra;
        double abonnée;
        double  abonnement;

        System.out.println(auth);
    if(auth!=null){


            System.out.println(t);

            return   (float) (0.5f*subQueryScore+0.5f*(t/15));

        }

    else return subQueryScore;
```

FIGURE 4.3.4 – Fonction qui retourne le nouveau score

4.4 Tests et résultats

Pour réaliser nos tests, nous avons utilisé la collection citée plus haut. Dans un premier temps nous avons réalisé une recherche thématique, par la suite une recherche thématique en lui ajoutant le score sociale. Dans ce qui suit, nous montrerons les résultats obtenus.

4.4.1 Résultats avec le score thématique :

Voici quelques résultats obtenus suite à la recherche thématique effectuée sur les requêtes de notre collection :

```
Q0 32415024995631105   1 1.8508724
Q0 30581416408391680   2 1.3486786
Q0 30269207937548288   3 0.9417196
Q0 32229379287289857   4 0.7341952
Q0 29756063234392064   5 0.67832106
Q0 33634329997348864   6 0.61548567
Q0 30261845583466496   7 0.48854634
Q0 34809237582389248   8 0.39272198
Q0 29580472631689218   9 0.39272198
Q0 32839394880651264   10 0.3652281
```

FIGURE 4.4.1 – Aperçu des résultats de la recherche thématique

4.4.2 Résultats avec le t-index

Voici quelques résultats obtenus suite à la recherche effectuée avec le t-index sur les requêtes de notre collection :

```
Q0 32415024995631105   1 0.9254362
Q0 30581416408391680   2 0.6743393
Q0 28970140049608704   3 0.54244745
Q0 30269207937548288   4 0.4708598
Q0 32229379287289857   5 0.3670976
Q0 29756063234392064   6 0.33916053
Q0 33634329997348864   7 0.30774283
Q0 30261845583466496   8 0.24427317
Q0 34809237582389248   9 0.19636099
Q0 29580472631689218   10 0.19636099
```

FIGURE 4.4.2 – Aperçu des résultats de la recherche thématique

Nous constatons que le t-index à eu un impact sur la recherche, pour cela il suffit de voir l'ordre des tweets retournés.

4.4.3 Résultats avec le ratio d'abonnement

Nous n'avons pas effectué des recherches sur notre ratio d'abonnement car la collection ne dispose des informations nécessaires.

4.4.4 Évaluation de l'approche

Nous avons fait une évaluation sur la collection de test choisie mais les résultats n'étaient pas probants. Nous pensons avoir fait un mauvais choix de collection (échantillon choisi). D'autres échantillonnage sont en cours pour construire une collection plus adéquate.

4.5 Conclusion

Dans ce dernier chapitre, nous avons proposé le cadre expérimental de notre approche que nous avons présenté dans le chapitre précédent, par la suite nous avons présenté un aperçu de notre implémentation, pour terminer nous avons montrer quelques résultats de l'approche.

Conclusion et perspectives

Dans notre travail nous nous sommes intéressées à la recherche d'influenceurs dans le réseau social Twitter, l'objectif étant de retrouver les personnes les plus influentes sur ce réseau social. Nous avons tout d'abord commencé par introduire les généralités liées la RI classique puis la RI sociale, par la suite nous avons relaté un ensemble d'apporches qui ont été proposées par d'autres chercheurs, nous avons par la suite proposé notre approche.

Dans notre approche nous avons exploité des métriques liées à Twitter : abonné, abonnement, ainsi que d'une mesure bibliométrique : le g-index que nous avons adapté à Twitter.

Apports, limites et perspectives

Ce projet nous a permi de développer nos propres capacités que ce soit sur le plan pratique ou même personnel.

Au cours de ce projet, nous avons réussi à proposer une approche dans le cadre de la recherche d'influenceurs dans Twitter, par la suite nous avons implémenté notre approche et nous avons montré quelques résultats de tests liés à la recherche thématique et au t-index. Mais par contraintes dues au dataset, nous n'avons pas pu montrer les résultats qui concerne le ratio d'abonnement.

Nous n'avons pas pu évaluer notre approche, faute de la collection choisie, mais de nouveaux échantillonnage sont en cours.

Annexes

Annexe A

Le g-index

Le g-index est une des mesures proposées dans la mesure de l'impacte d'un scientifique. Il est présenté comme une amélioration de l'indice h de Hirsch [1] pour mesurer la performance d'un ensemble d'articles. Si cet ensemble est classé par ordre décroissant du nombre de citations qu'ils ont reçues, le g-index est le plus grand nombre (unique) tel que les premiers articles g reçu (ensemble) au moins g^2 citations.

Le g-index a fait l'objet de plusieurs études dans le but de determiner sa perfomance. Si on le compare à l'index-h, on en déduit deux avanatages :

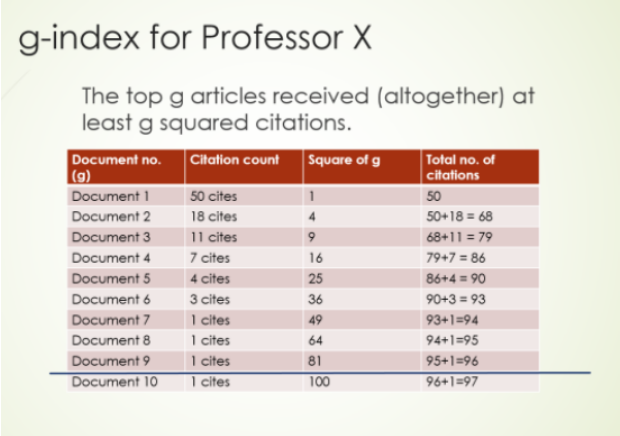
Premièrement, la pondération des citations reçues par les documents est prise en compte dans le calcul de l'indice g ;

Deuxièmement, l'indice g pour un scientifique donné n'est pas limité par son nombre total de publications. Selon ces caractéristiques, l'indice g pourrait être plus approprié que l'indice h pour évaluer des scientifiques sélectifs, qui ont moins de chances d'obtenir des valeurs élevées de l'indice h.

Calcul du g-index :

Pour mieux expliquer le calcul du g-index, nous considérons l'exemple ci-dessous :

1. e h-index (ou facteur h), créé par le physicien Jorge Hirsch en 2005, est un indicateur d'impact des publications d'un chercheur. Il prend en compte le nombre de publications d'un chercheur et le nombre de leurs citations.



Document no. (g)	Citation count	Square of g	Total no. of citations
Document 1	50 cites	1	50
Document 2	18 cites	4	50+18 = 68
Document 3	11 cites	9	68+11 = 79
Document 4	7 cites	16	79+7 = 86
Document 5	4 cites	25	86+4 = 90
Document 6	3 cites	36	90+3 = 93
Document 7	1 cites	49	93+1=94
Document 8	1 cites	64	94+1=95
Document 9	1 cites	81	95+1=96
Document 10	1 cites	100	96+1=97

FIGURE A.1 – g-index du scientifique x.

Comme nous pouvons le voir dans le tableau ci-dessus, nous calculons le rang^2 des documents (g^2) ainsi que la somme des citations. Pour determiner le g-index, nous commençons par vérifier la condition rang$^2 \leq$total nombre des citations, dès que la condition n'est plus satisfaite, nous nous arrêtons, dans l'exemple cité ci-dessus, la condition n'est plus satisfaite au document 10 (100 $\succ$97), on en déduit que g-index=9.

Remarque :

Les documents dont le nombre de citations vaut 0, ne sont pas prit en considération.

Bibliographie

[Anger 11] Kittl C. Anger I. *« Measuring influence on twitter. In S. N. Lindstaedt, M. Granitzer (Eds.), ».* 2011.

[Azaza L. 15] Savonnet M. Frame A. Azaza L. Kirgizov S. *« Evaluation de l'influence sur Twitter : Application au projet « Twitter aux Elections Européennes 2014 ».* Mai 2015.

[Badache. 16] Ismail Badache. *« Recherche d'information sociale : exploitation des signaux sociaux pour améliorer la recherche d'information, ».* 2016.

[Benjabeur L. 12] Boughanem M. Benjabeur L. Tamine L. *« Un modèle de recherche d'information sociale dans les microblogs : cas de Twitter ».* 2012.

[Boccaletti S] MorenoY Chavez M Boccaletti S Latora V & Hwang. *«2006 Phys. Rep.424 175., ».*

[Boubekeur. 08] F. Boubekeur. *«Contribution à la définition de modèles de recherche d'information flexibles basés sur les CP-Nets, ».* 2008.

[Cha 10] Haddadi H. Benevenuto F. Gummadi P. K. Cha M. *Measuring user influence in twitter : The million follower fallacy. In.* 2010.

[Damak. 14] Firas Damak. *«Etude des facteurs de pertinence dans la recherche de microblogs., ».* 2014.

[Ding 13] Jia Y. Zhou B. Han Y. He L. Zhang J. Ding Z. *Measuring the spreadability of users in microblogs.* 2013.

[F. Boubekeur. 17] M. Ferrouk et L. Belkacem F. Boubekeur. *« Nouvelle mesure de l'influence sur twitter. ».* 2017.

[Fabian Riquelem 16] Pablo Ganzalez-Cantergiani Fabian Riquelem. *« Measuring user influence on Twitter : A survery ».* 2016.

[G] Sabidussi G. *« 1996 Psychometrika ,31 558 ».*

[Haewoon Kwak 10] Hosung Park Haewoon Kwak Changhyun Lee & Sue Moon. *«What is Twitter, a Social Network or a News Media ? ».* 2010.

[Hammache. 13] Arezki Hammache. *«Recherche d'Information : un modèle de langue combinant mots simples et mots composés, ».* 2013.

[J. 71] Rocchio J. *« Relevance Feedback in Information Retrieval. ».* 1971.

[Nicholas J. Belkinl. 92] et al Nicholas J. Belkinl. *« Proceedings of the 15th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval. ».* 1992.

[Peng. 18] S. Peng. *« Influence analysis in social networks : A survery ».* 2018.

[Ricardo Baeza-yates 99] Berthier Ribeiro-Neto. Ricardo Baeza-yates. *«Modern Information Retrieval., ».* 1999.

[Romero D. M. 11] Asur S. Huberman B. A. Romero D. M. Galuba W. *« Influence and Passivity in Social Media ».* 2011.

[Salton. 83] et al Salton. *« Introduction to Modern Information Retrieval. ».* 1983.

[shah. 17] Shirag shah. *«Social Information Seeking., ».* 2017.

[Sung J. 13] Moon S. Sung J. & Lee JG. *« The influence in Twitter : Are they really influenced ? ».* 2013.

[Weng J. 10] Jiang J. He Q. Weng J. Lim E.-P. *« TwitterRank : finding topic-sensitive influential Twitterers ».* 2010.