REPUPLIQUE ALGERIENNE DEMOCRATIQUE ET POPULAIRE

MINISTERE DE L'ENSEIGNEMENT SUPERIEUR ET DE LA RECHERCHE SCIENTIFIQUE

Université Saad Dahleb Blida 1

FACULTE DES SCIENCES

DEPARTEMENT D'INFORMATIQUE

Mémoire de fin d'étude en vue de l'obtention du diplôme de Master académique en informatique

Option : Ingénierie des logiciels

Thème :

Utilisation de l'expansion de la requête pour améliorer la recherche d'information dans les microblogs

Présenter par :

KOUIDER AKI Oussama **&** **GHELLAL Mebarek**

Soutenu le : 17/08/2023

Devant le jury composé de :

Promotrice Mme BOUCETTA Zouhel

Président Mme. Ouahrani L **Examinateur M.FERFFARA S**

Année Universitaire 2022/2023

REMERCIEMENTS

Avant tous, nous tenons à remercier DIEU le tout puissant de nous avoir donné le courage, la force et la volonté d'accomplir ce travail.

Nous tenons tout d'abord à exprimer nos gratitudes et nos reconnaissances les plus sincères à notre enseignante promotrice ***Mme BOUCETTA Zouhel,*** *qu'elle soit ici remerciée, pour ses encouragements et ses précieux conseille, ainsi que pour sa simplicité, sa gentillesse, ses qualités humaines, sa sympathie et surtout d'avoir accepté de diriger ce travail.*

Nos Remerciements vont également à ***Mme OUAHRANI LEILA****, pour l'honneur qu'elle nous fait en acceptant de présider le jury qui va juger ce travail.*

Nous adressons nos remerciements à ***M FERFERRA Soufiane,*** *pour avoir accepté d'examiner ce travail.*

Nos hommages également à tous ***nos Enseignants*** *du Département de l'informatique pour avoir fortement contribué à enrichir nos connaissances.*

Nos remercions enfin, tous ceux qui ont contribué de près ou de loin et qui ont pris une part active à la réalisation de ce travail et dont les noms ne sont pas cités

DEDICACE

Je remercie, tout d'abord, Dieu pour m'avoir donné la force, la patience et la résilience nécessaires pour accomplir ce travail. Allhamdoulah.

Je dédie affectueusement ce modeste travail,

À mon cher père et ma chère mère

En reconnaissance de tous ce que vous avez faits pour me permettre d'atteindre cette étape de ma vie. Aucune dédicace ne saurait exprimer mon respect, mon amour éternel et ma considération pour les sacrifices que vous avez consentis pour mon instruction et mon bien être. Que le tout puissant vous accorde une longue vie ! J'espère que vous trouverez dans ce travail toute ma reconnaissance et tout mon amour.

***À mes professeurs**, pour leur sagesse et leur guidance tout au long de mon parcours universitaire. Vous avez allumé la flamme de la connaissance en moi et m'avez montré la voie à suivre.*

À la fleur de notre maison, ma petite Soeur « Nour elhouda ».

À mes très chère Sœur 'Asma' et 'khadidja'

À mes frères 'Abdellah', 'Abdelnour', 'Rachid'

***À ma petite nièce, Meriem et Taline** C'est avec un grand amour et une profonde affection que je dédie une partie de ce travail. Tu es une source constante de joie et d'inspiration dans ma vie.*

Je vous souhaite tous, plein de succès et de bonheur que Dieu les garde.

A toute ma famille et mes amis

***A mon chère amie et mon binôme Oussama** Ta contribution a été inestimable dans la réalisation de ce projet. Ta disponibilité, ta patience et ton encouragement ont été une source d'inspiration constante*

À toute personne qui m'aime et me souhaite la réussite dans la vie et dans les études.

Merci à tous et à toutes

Mebareck

DEDICACE

Je remercie, tout d'abord, ***Dieu*** *de m'avoir donné le courage, et la chance d'étudier, grâce à Dieu que je suis là.*

Je dédie ce modeste travail :

A ma chère ***Maman*** *et à* ***l'âme de mon cher Père***

Pour tous les sacrifices qu'ils ont consenti à mon égard, pour leur amour, leur soutien, leur compréhension, et pour leurs prières tout au long de ma vie. Merci beaucoup pour la bonne éducation que vous m'as donnée, grâce à Dieu, puis grâce à vous je suis devenu ce que je suis aujourd'hui j'espère vous rendre fier de moi, je vous aime énormément, Que Dieu vous garde ***maman*** *et vous procure santé et longue vie,* ***papa*** *que Dieu ait pitié de vous et vous accorde le paradis.*

A ma chère ***sœur*** *que dieu te protège et guider vos pas vers un avenir inchaallah prometteur. Je te souhaite que de bonheur, de santé et de réussite dans ta vie.*

A mon cher ***grand père*** *et a* ***l'âme de ma grande mère,*** *qui mon toujours aimé et comblé par ses bénédictions et ses prières, grand père que Dieu bénisse votre santé et vous protège. A mes tantes et oncles qui ont toujours été à nos côtés, je vous respecte et vous apprécie, que Dieu vous protège.*

A mon chère amie et mon binôme ***Mebareck*** *qui a partagé avec moi ce modeste travail, merci pour ta patience, ton encouragement et pour ton aide, j'espère qu'on gardant que des bons souvenirs.*

A mes amies ***Hamza*** *et* ***Noureldin*** *qui m'a encouragé, Je te souhaite de bonheur, et de réussite. Aux personnes qui m'ont toujours aidé et encouragé.*

Oussama

Résumé

Dans l'ère actuelle des réseaux sociaux, les plateformes de microblogging comme Twitter, qui compte 238 millions d'utilisateurs actifs par mois et plus de 850 millions de tweets envoyés par jour, constituent une source d'information massive. Cependant, le volume considérable de ces publications complique l'accès à l'information pertinente. Les tweets sont des documents courts, souvent rédigés dans un langage mal orthographié et contenant des abréviations et des argots, ce qui pose un défi particulier pour les modèles de recherche d'information actuels.

La recherche d'informations dans le corpus des tweets est complexe, en raison à la fois du volume du corpus et des caractéristiques des tweets. Les défis comprennent l'absence fréquente des termes de la requête dans le tweet et le fait que chaque terme n'apparaît généralement qu'une seule fois dans le texte. Par conséquent, la sélection des meilleurs tweets repose sur un appariement lexical entre la requête et les tweets, ce qui peut entraîner un nombre élevé de tweets non pertinents dans le haut de la liste de résultats.

Pour améliorer le classement des tweets pertinents, nous avons proposé un système basé sur une nouvelle approche de l'expansion de la requête via le Pseudo relevant feedback. Notre modèle exploite à la fois l'aspect thématique et temporel des tweets. En utilisant le corpus TREC 2011, nous avons détecté les grandes concentrations de tweets, identifié les sujets principaux parmi ces concentrations à l'aide de l'approche du Biterm et utilisé leurs termes les plus fréquents pour l'expansion de la requête.

Cette approche permet d'améliorer la qualité du classement des tweets pertinents, fournissant ainsi une méthode plus efficace pour extraire des informations pertinentes de la masse de données générées par Twitter

Mots clés : Twitter, expansion de la requête, recherche temporel, burst, modèle de topic.

Abstract

In today's social media era, microblogging platforms like Twitter, which has 238 million monthly active users and more than 850 million tweets sent daily, are a massive source of information. However, the sheer volume of these publications complicates access to relevant information. Tweets are short documents, often written in misspelled language and containing abbreviations and slang, which poses a particular challenge for current information retrieval models.

Finding information in the corpus of tweets is complex, due to both the volume of the corpus and the characteristics of the tweets. Challenges include the frequent absence of query terms in the tweet and the fact that each term usually only appears once in the text. Therefore, the selection of the best tweets relies on a lexical match between the query and the tweets, which can lead to a high number of irrelevant tweets at the top of the results list.

To improve the ranking of relevant tweets, we proposed a system based on a new approach of query expansion via Pseudo relevant feedback. Our model exploits both the thematic and temporal aspects of tweets. Using the TREC 2011 corpus, we detected large concentrations of tweets, identified top topics among these concentrations using the Biterm approach, and used their most frequent terms for query expansion.

This approach improves the ranking quality of relevant tweets, providing a more efficient method to extract relevant information from the mass of data generated by Twitter.

Keywords:

Twitter, Burst, Query expansion, Temporal search, Topical model.

الملخص

في عصر وسائل التواصل الاجتماعي اليوم، تعد منصات المدونات الصغيرة مثل Twitter، التي لديها 238 مليون مستخدم نشط شهريًا وأكثر من 850 مليون تغريدة يتم إرسالها يوميًا، مصدرًا هائلاً للمعلومات. ومع ذلك، فإن الحجم الهائل لهذه المنشورات يعقد الوصول إلى المعلومات ذات الصلة. التغريدات عبارة عن مستندات قصيرة، غالبًا ما تكتب بلغة بها أخطاء إملائية وتحتوي على الاختصارات واللغة العامية، مما يشكل تحديًا خاصًا لنماذج استرجاع المعلومات الحالية.

يعد العثور على المعلومات في مجموعة التغريدات أمرًا معقدًا، نظرًا لكل من حجم المجموعة وخصائص التغريدات. تشمل التحديات الغياب المتكرر لمصطلحات الاستعلام في التغريدة وحقيقة أن كل مصطلح يظهر عادة مرة واحدة فقط في النص. لذلك، يعتمد اختيار أفضل التغريدات على تطابق معجمي بين الاستعلام والتغريدات، مما قد يؤدي إلى عدد كبير من التغريدات غير ذات الصلة في أعلى قائمة النتائج.

لتحسين ترتيب التغريدات ذات الصلة، اقترحنا نظامًا يعتمد على نهج جديد لتوسيع الاستعلام عبر التعليقات الزائفة ذات الصلة. يستغل نموذجنا كلا الجانبين الموضوعي والزمني للتغريدات. باستخدام مجموعة TREC 2011، اكتشفنا تركيزات كبيرة من التغريدات، وحددنا الموضوعات الرئيسية بين هذه التركيزات باستخدام نهج Biterm، واستخدمنا مصطلحاتهم الأكثر شيوعًا لتوسيع الاستعلام.

يعمل هذا النهج على تحسين جودة ترتيب التغريدات ذات الصلة، مما يوفر طريقة أكثر فاعلية لاستخراج المعلومات ذات الصلة من كتلة البيانات التي تم إنشاؤها بواسطة Twitter.

.

الكلمات المفتاحية تويتر محركات البحث التدوينات البحث الوقتي

Table des matières

Liste des figures

Listes des tableaux

Listes des abreviations

RI	La recherche d'information
SRI	Systéme de la recherche d'information
RIS	Recherche information social
Tf	Term Frequency
Idf	Inverse of Document frequency
MAP	Mean Average Precision
TREC	Text Retrieval Conference
UGC	User Generated Content
LDA	Latent Dirichlet Allocation
NLTK	Natural Language Toolkit
KDE	Kernel density estimation

INTRODUCTION GENERALE

1. Contexte

À l'heure actuelle, le monde connaît des avancées technologiques importantes dans divers secteurs, grâce au domaine de l'informatique. L'informatique implique l'étude des techniques de traitement automatique de l'information et joue un rôle crucial dans la société actuelle axée sur l'information. La recherche d'informations se concentre sur la structure, l'analyse, l'organisation, la recherche et la classification des informations. Le principal défi consiste à trouver efficacement les documents les plus pertinents parmi le vaste volume d'informations disponibles, répondant aux attentes de l'utilisateur. Cette opérationnalisation de la recherche d'informations est facilitée par des outils informatiques appelés Information Retrieval Systems (IRS).

Le périmètre de cette thèse s'articule autour du domaine et des techniques de recherche d'information dans les tweets. Twitter, étant l'une des plates-formes de microblogging et de réseaux sociaux Web 2.0 les plus récentes et les plus largement utilisées, sert de principal moyen de recherche d'informations dans les tweets. En raison de son immense popularité, Twitter héberge une grande quantité d'informations, car il permet aux utilisateurs non seulement de consommer, mais aussi de contribuer à la production de contenu.

Avec 238 millions d'utilisateurs actifs par mois et plus de 850 millions de tweets envoyés par jour, Twitter est devenu le service de microblogging le plus populaire. Cependant, ce grand volume de publications pose des défis aux microblogueurs lorsqu'il s'agit d'accéder à l'information. Un tweet est un document concis limité à 140 caractères, souvent écrit dans un langage abrégé, contenant des mots d'argot et des mots mal orthographiés pour transmettre des informations en utilisant un nombre minimal de caractères. Récupérer des informations à partir d'un corpus de tweets est particulièrement difficile pour les modèles de recherche d'informations existants en raison du volume du corpus et des caractéristiques spécifiques des tweets, comme discuté par Choi en 2012.

Le modèle de recherche rencontre deux problèmes principaux lorsqu'un utilisateur soumet une requête : l'absence de termes de requête dans le tweet et la limitation de chaque terme apparaissant au plus une fois dans le texte. Les modèles traditionnels sélectionnent les tweets en fonction d'une correspondance lexicale entre la requête et les tweets, ce qui entraîne une forte probabilité que des tweets non-pertinents apparaissent en

haut des résultats de recherche. Pour améliorer le classement des tweets pertinents, de nombreuses études ont incorporé des preuves temporelles pour reclasser les tweets obtenus à partir de la recherche initiale.

Dans ce contexte, notre travail vise à améliorer l'efficacité de la recherche dans le corpus des tweets sur une base temporelle.

2. Problématique

Les modèles classiques de recherche d'informations sont confrontés à des défis face au volume croissant de tweets dans le corpus. La courte durée des tweets et la qualité du langage utilisé aggravent encore ces difficultés. Pour résoudre ces problèmes et améliorer le classement des tweets pertinents dans les résultats de recherche, des recherches récentes ont introduit des aspects temporels. Ces facteurs supplémentaires permettent d'estimer la pertinence d'un tweet par rapport à une requête et permettent de réorganiser les tweets en fonction de leurs scores de pertinence. Notre travail s'inscrit dans cette démarche.

3. Objectif

L’objectif est de développer un outil de recherche d'informations spécifiquement conçu pour les microblogs, tels que les tweets. Cet outil vise à améliorer le classement des tweets pertinents pour une requête donnée.

Pour y parvenir nous proposons de faire une expansion des requetés par les termes les plus pertinents en utilisons une approche temporelle

Notre approche consiste à employé une technique temporelle et thématique pour calculer la pertinence des termes de topics par rapport à une requête en se basant sur la concentration des tweets.

Pour évaluer notre approche, nous avons mené des expériences en utilisant la collection de microblogs TREC2011, en particulier l'ensemble de données de test.

Notre mémoire est organisée de la manière suivante :

Le premier chapitre : résume les concepts de base de la RI, nous commençons par donner une définition de la RI, puis nous décrivons les différents modèles servants comme cadre théorique pour la modélisation du processus de RI. Nous illustrons également les systèmes de

la RI en présentant leur définition et les étapes d'un processus de recherche (D'indexation… Etc.), enfin, nous terminons par présenter les mesures d'évaluation des systèmes de recherche d'information.

Le deuxième chapitre : présent un flash sur la recherche d'informations dans les microblogs. Nous commençons par la présentation des spécificités des plateformes de microblogging : cas de Twitter. Par la suite, nous présentons quelques aspects de la recherche d'information temporelle dans les tweets. Enfin, nous terminons par détailler quelques travaux voisins.
Le troisième chapitre : présente notre contribution, il s'agit d'une nouvelle approche pour la recherche des Microblogs pertinents, dont l'objectif principal est d'améliorer les performances de la recherche dans le corpus de tweets via l'expansion de la requête, via l'usage de l'aspect thématique et l'aspect temporel.

Le quatrième chapitre : englobe le détaillé concerne les outils d'implémentation utiliser dans notre travail, la collection de tests (TREC2011 Microblogs Track), les résultats de nos expérimentations et enfin l'évaluation des résultats.

Chapitre 1

La recherche d’information

Introduction :

La recherche d'information (RI) remonte aux années 1950, qui correspondent aux débuts de l'informatique. À l'origine, la RI était associée aux applications en bibliothèque. Cependant, avec l'avènement du web et la croissance de la quantité d'informations disponibles, ce domaine a étendu son champ d'application au-delà des bibliothèques et s'applique désormais à l'ensemble du web.

La RI consiste à élaborer des modèles et des processus permettant de récupérer, à partir d'un ensemble de documents indexés, ceux qui répondent le mieux aux besoins en information d'un utilisateur.

L'évolution de la recherche d'information a été profondément marquée par l'avènement du Web et plus récemment les réseaux sociaux. Ces plateformes sont devenues l'outil de communication, de partage de connaissances et de contenus le plus utilisé sur le Web. Avec l'intégration de cette dimension sociale, de nouveaux besoins d'information émergent, donnant naissance à la recherche d'information sociale (RIS).

Dans ce chapitre, nous présentons les concepts de base de la recherche d'information (RI) et les différents modèles qui ont été proposés pour fournir un cadre théorique pour la modélisation du processus de recherche d'information (RI).

1. La Recherche d'information

1.1. Définition

La recherche d'informations est une branche de l'informatique liée à l'acquisition, organisation, stockage, récupération et distribution d'informations. Les informations de recherche sont conçues pour répondre aux besoins de l'utilisateur en mettant en évidence un mécanisme pour établir une correspondance entre les exigences de la bibliothèque de documents et les documents.

1.2. Processus de la recherche d'information

Pour la recherche d'information, les documents doivent être stockés, recherchés et explorés. Cette tâche implique l'utilisation de plusieurs concepts :

- **Document :** un document représente l'unité d'information de base accessible et utilisée par le SRI et peut constituer une réponse à la demande utilisateur. Les

documents peuvent être du texte, des pages Web, des cartes, des articles de blog photos, et des vidéos.

- **Collection des documents :** une collection de documents (un corpus) est une collection d'informations disponibles et accessibles aux utilisateurs, ou tout simplement, c'est l'ensemble Documents dans lesquels les utilisateurs recherchent des informations.
- **Requetés :** une requête est une expression des besoins d'information d'un utilisateur, qui lance le processus de recherche. Elle peut être exprimée en langage naturel, booléen ou graphique, etc. Les requêtes représentent l'interface entre l'utilisateur et le système de recherche d'informations (SRI).

Dans le cadre de trouver les documents pertinents qui répondent de manière plus précise à la requête, le SRI utilise un processus qui comprend deux étapes principales : l'indexation et l'appariement requête /document.

La figure (1.1) représente le schéma d'un SRI [Belkin, 92] :

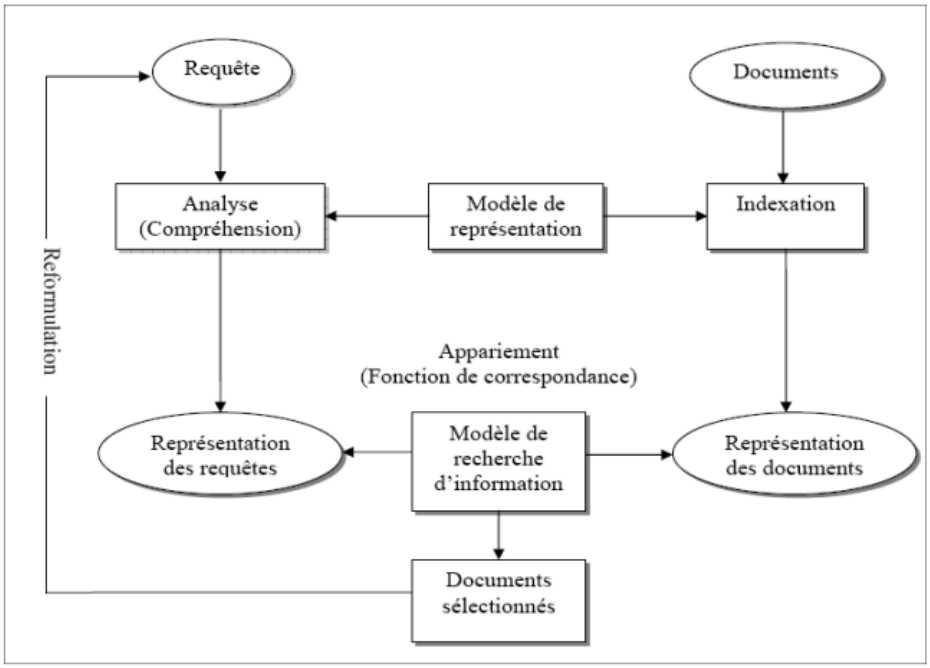


FIGURE 1. 1 PROCESSUS EN U DE RECHERCHE D'INFORMATION

Le schéma ci-dessus montre clairement que le processus de recherche d'information se décompose en deux processus comme suit :

- **Modèle de représentation.**
- **Modèle de recherche d'information.**

1.3. L'indexation

Afin de faciliter l'exploitation des documents bruts qui peuvent être coûteux et difficiles à traiter, les SRI ont recours à l'indexation. Cela implique la création d'un descriptif comprenant des mots-clés représentant au mieux le document ou la requête, avec un poids associé, qui est ensuite stocké dans une structure appelée index, permettant une interrogation facile. Toutefois, cette opération peut être relativement longue, en fonction du nombre et de la taille des documents dans la collection, ainsi que du type d'indexation utilisé (Manuelle, semi-automatique ou automatique). [Furnas et *al.*, 1987].

a) **Indexation manuelle :** elle est réalisée par un documentaliste qui lit chaque document et fournis une terminologie particulière pour classer chaque document.
b) **Indexation automatique :** le processus d'indexation est entièrement automatisé. La plupart des SRI suivent cet indice en raison de sa rapidité et des économies de coûts par rapport à l'indexation manuelle. [Rijsbergen, 1979].
c) **Indexation semi-automatique :** c'est une combinaison entre l'indexation manuelle et automatique. Elle se base sur l'indexation automatique suivie par une intervention humaine pour sélectionner les termes pertinents et approuver la représentation finale. [Rijsbergen, 1979].

L'indexation automatique est composée de plusieurs étapes qui sont applicables sur chaque document et requête.

1) **L'analyse lexicale :** c'est la première étape du processus d'indexation. Lors de cette étape, les documents textuels sont transformés à des ensembles de termes. La ponctuation, et la mise en page va être supprimée.
2) **Elimination des mots vides :** la procédure consiste à éliminer les mots qui ne portent pas de sens (mots vides) tels que les pronoms personnels, les articles, les conjonctions et les prépositions. De plus, les mots qui apparaissent un nombre excessif de fois dans le document sont également retirés.
3) **Lemmatisation (radicalisation) :** dans cette tâche, chaque mot est remplacé par sa racine. Il existe plusieurs techniques utilisées pour la lemmatisation, nous citons :
    - ✓ La troncature.
    - ✓ Les variétés de successeurs (n grammes).
    - ✓ L'élimination des affixes (Porter).
    - ✓ La table de consultation (dictionnaire).

4) **Pondération :** cette étape consiste à effectuer un poids pour chaque terme.

Ce poids est la valeur numérique, représente l'importance du terme dans le document. Cette dernière est calculée en utilisant des approches basées sur des aspects statistiques. La majorité des techniques de pondération sont fondée sur la combinaison entre deux facteurs :

- **Tf (Term Frequency) :** C'est une mesure utilisée en analyse de texte pour évaluer l'importance d'un terme dans un document. Elle correspond simplement au nombre de fois que le terme apparaît dans le document.
- **Idf (Inverse of Document frequency) :** c'est une mesure utilisée pour évaluer l'importance d'un terme dans une collection dans le but d'identifier les termes qui discriminent le plus un document par rapport aux autres documents de la collection. [Damak, 2014]

La formule qui représente cette mesure est la suivante :

$$Idf_t = \log\left(\frac{N}{1+n}\right) \qquad (1.1)$$

N : est le nombre de documents dans la collection.

n : est le nombre de documents où t'apparaît.

t : est le terme

- **Tf Idf *:*** combine la pondération locale et globale d'un terme pour évaluer son importance dans une collection de documents. Elle fournit une estimation précise de l'importance du terme dans la collection. Exprime par cette formule :

$$\boldsymbol{TfIdf_{t,d} = Tf_{t,d} * I\,df_t} \qquad (1.2)$$

1.4. L'appariement

Cette étape survient après que les documents ont été indexes et la requête a été analysée. En utilisant la mise en correspondance de la requête et de l'index, le SRI prédit les documents pertinents pour l'utilisateur et calcule un score de pertinence qui évalue la similarité entre la requête et le document. Pour ce faire, il utilise une valeur appelée RSV (Retrieval Status Value) qui prend en compte la pondération du terme calculée en fonction de la requête (q) et du document (d). [Rijsbergen, 1979].

2. Expansion de la requête

L'expansion de requête est une technique utilisée dans la recherche d'informations pour améliorer la précision et le rappel des résultats de recherche. Cette méthode consiste à modifier la requête initiale en ajoutant des termes supplémentaires pertinents pour le sujet de recherche [Jinxi Xu et W. Bruce Croft,1996].

Le feedback est la principale méthode utilisée pour identifier ces termes supplémentaires, et elle peut être classée en trois types :

2.1. Feedback Explicites

Le feedback explicite implique une interaction directe entre l'utilisateur et le système de recherche, où l'utilisateur évalue activement la pertinence des documents renvoyés. Cette approche est connue pour sa précision car elle s'appuie sur des informations spécifiques fournies par l'utilisateur, offrant une compréhension claire de ses préférences et contribuant à l'amélioration des résultats de recherche.

Cependant, le feedback explicite présente certains défis. Un inconvénient majeur est qu'il nécessite une participation active des utilisateurs, ce qui implique d'investir du temps et des efforts pour évaluer les résultats de la recherche. Cette contrainte peut être peu pratique ou décourageante pour certains utilisateurs. [Jinxi Xu et W. Bruce Croft,1996].

2.2. Feedback Implicites

À l'inverse, feedback implicite est une méthode qui déduit les préférences de l'utilisateur à partir de son comportement lors de l'utilisation du système de recherche. Cela peut inclure des facteurs tels que les documents sur lesquels l'utilisateur a cliqué ou le temps passé à lire chaque document.

elle offre plusieurs avantages, notamment une collecte facile et l'absence de participation active des utilisateurs, ce qui la rend plus pratique dans de nombreux contextes.

Cependant, feedback implicite a ses limites. Il peut être moins précis que le feedback explicite car il repose sur des hypothèses sur les intentions de l'utilisateur, ce qui peut parfois conduire à des interprétations erronées. [Jinxi Xu a W. Bruce Croft,1996].

2.3. Pseudo Relevent Feedback

La technique de "Pseudo Relevant Feedback" (retour pseudo pertinent) est une approche utilisée dans la recherche d'informations pour améliorer la précision des résultats de recherche en ajustant la requête initiale du chercheur.

Lorsqu'un utilisateur soumet une requête de recherche, le système récupère une liste de documents qui semblent pertinents en fonction de cette requête. Cependant, ces résultats initiaux peuvent être imparfaits ou ne pas couvrir tous les aspects souhaités par l'utilisateur. C'est là que le Pseudo Relevant Feedback intervient.

La technique du Pseudo Relevant Feedback consiste à sélectionner certains documents initiaux, souvent les premiers de la liste des résultats, et à les utiliser pour générer une nouvelle requête améliorée. Cela se fait généralement en identifiant les termes ou les concepts les plus fréquents dans ces documents sélectionnés et en les incorporant dans la nouvelle requête. Ainsi, la nouvelle requête modifiée est plus raffinée et plus susceptible de donner des résultats pertinents.

3. Les modèles de recherche d'information

Un modèle de RI a pour rôle de fournir une formalisation du processus de recherche d'information. Ce modèle doit remplir plusieurs fonctions, la plus importante étant de fournir un cadre théorique pour la modélisation de la mesure de pertinence. Cette mesure, appelée RSV (Retrieval Status Value), est utilisée pour évaluer le degré de similitude entre les représentations des documents et les requêtes lors du processus de mise en correspondance. Ces modèles sont divisés en deux types :

1. **Les modelés exacts :** les documents retournent par ces modèles répondant exactement à la requête **(modèle booléen).**
2. **Les modèles partiels :** les documents retournent par ces modèles répondant à tout ou partie de la requête.

3.1. Le modèle booléen

Le modèle booléen est basé sur la théorie des ensembles et l'algèbre booléenne [Salton,1975]. Les documents dans ce modèle son représentent par une conjonction des termes (non pondérés), un document est représenté comme suit : **D=t1 et t2 et t3 … tn.**

La requête est une expression booléenne composée par des mots clés connecter entre eux par les opérateurs logiques **et (∧), ou (∨) et non (¬).** Une requête est représentée comme suit **(t1 ou t2) et (t3 …tn).**

La mesure de pertinence document requête est calculée à travers une fonction de correspondance qui est basé sur l'absence et la présence des termes de requête dans le document et si l'expression logique de la requête **q** est impliquée par l'index de chaque document **Di.**

Le résultat de cette fonction est binaire est représenté comme suit :

$$RSV(d, ti) = \begin{cases} 1 & si\ t_i \in d \\ 0 & sinon \end{cases} \quad (1.3)$$

Ce modèle est simple pour la mise en œuvre, il est très utilisé mais possède deux inconvénients :

- Les requetés sont difficiles à représenter pour les utilisateurs.
- Tous les documents on la même pertinence et il n'est pas possible de les ordonner selon leur proximité par rapport aux besoins d'informations.

3.2. Le modèle vectoriel

Le modèle vectoriel [Salton et *al*, 1968] est un modèle basé sur l'algèbre. Chaque document $\mathbf{d_i}$ et requêtes **q** sont représentés par des vecteur $\mathbf{w_{ij}}$ de dimension **n** dans un espace vectoriel composé de tous les termes d'indexation T = {t1, t2, …,Tn},

Ce model prend en considération le poids de terme dans chaque document .il est soit :

- Une forme **Tf*idf.**
- Un poids attribué par l'utilisateur.

Le degré de corrélation entre les vecteurs correspondants des documents **di** et la requête **q**, est utilisé pour évaluer la pertinence des documents. Plusieurs mesures de corrélation peuvent être utilisées pour exprimer cette corrélation. Ses mesures de similarité utilisées sont :

- **La mesure du cosinus**

$$RSV(dj, Q) = \frac{\sum_{j=1}^{n} qj * dj}{\sqrt{\sum_{j=1}^{n} q_j^2} * \sqrt{\sum_{j=1}^{n} d_j^2}} \quad (1.4)$$

- **Le produit scalaire**

$$\boldsymbol{RSV} = (\boldsymbol{d_j}, \boldsymbol{Q}) = \sum_{\boldsymbol{j=1}}^{\boldsymbol{n}} \boldsymbol{q_j} \times \boldsymbol{d_{ij}} \qquad (1.5)$$

- **La mesure de Jacard**

$$\text{RSV (dj , Q)} = \frac{\sum_{j=1}^{n} qj * dij}{\sum_{j=1}^{n} q_j{}^2 + \sum_{j=1}^{n} d_{ij}^2 - \sum_{j=1}^{n} q_j * d_{ij}} \qquad (1.6)$$

Le modèle vectoriel de base est largement employé en RI en raison de sa capacité à classer les résultats de recherche en fonction de leur pertinence pour une requête utilisateur. Contrairement au modèle booléen, il offre l'avantage de pouvoir ordonner les résultats. Cependant, ce modèle repose sur l'hypothèse d'indépendance entre les termes d'index et ne prend pas en compte les relations sémantiques pouvant exister entre ces termes dans un même document ou une même requête [Azzoug, 2013].

3.3. Le modèle probabiliste

Ce modèle est proposé par Robertson et Sparck Jones au début des années 1960. Il est fondé sur une théorie mathématique de probabilité, cette approche calcule la probabilité de pertinence d'un document pour une requête. Il vise à identifier les documents ayant une forte probabilité d'être pertinents et une faible probabilité d'être non pertinents.

La probabilité qu'un document soit pertinent pour une requête **q** est représentée par **P(q|d_i)**, tandis que la probabilité qu'il ne soit pas pertinent est notée **P(np|d_i).** La mesure du rapport entre ces probabilités détermine la similarité entre le document **di** et la requête **q** dans ce modèle.

On définit **P (r|d)** comme la probabilité que le document **d** appartienne à l'ensemble des documents pertinents, et **P (nr|d)** comme la probabilité que le document appartienne à l'ensemble des documents non-pertinents [Rijsbergen, 1979]

Le score d'appariement **RSV (d, q)** entre le document **d** et la requête **q** est calculée par la formule (Robertson et *al.*, 1994) :

$$RSV(q,d\ ) = \frac{P\ (\mathrm{r}|\mathrm{d})}{\mathrm{P}\ (\mathrm{nr}|\mathrm{d})} \qquad (1.7)$$

Ce qui donne après l'application de la règle de Bayes et quelques transformations :

$$RSV(q,d\ ) = \frac{P\ (\mathrm{d}|\mathrm{r})}{\mathrm{P}\ (\mathrm{d}|\mathrm{nr})} \qquad (1.8)$$

Tel que :

- **P (d/r)** est la probabilité que le document appartienne à l'ensemble **r** des documents pertinents
- **P (d/nr))** est la probabilité que le document appartienne à l'ensemble **nr** des documents non pertinents).

3.4. Modelé de langue

Le modèle de langue fait référence à un modèle statistique utilisé pour estimer la probabilité d'une requête de recherche étant donné un document ou d'une séquence de mots étant donné un contexte plus large.

Un modèle de langue est utilisé pour mesurer à quel point une requête de recherche est cohérente avec un document ou pour évaluer la pertinence d'un document par rapport à une requête. Il est utilisé pour classer et ordonner les résultats de recherche afin de fournir les documents les plus pertinents en premier.

Les modèles de langue dans la recherche d'informations peuvent être basés sur différents algorithmes, tels que les modèles de Markov, les modèles n-grammes, les modèles de séquences conditionnelles, les réseaux de neurones récurrents (RNN).

4. Évaluation d'un SRI

Pour atteindre son objectif de localisation des documents pertinents, un système de recherche d'information doit satisfaire les besoins d'information de l'utilisateur. Cela est dû à la réalité que toutes les évaluations tournent autour du concept central de pertinence, qui se rapporte à une correspondance entre une demande et un document. L'excellence d'un système

doit être mesurée en confrontant les réponses du système aux réponses demandées par l'utilisateur. Si les réponses du système sont plus conformes aux réponses idéales, alors la qualité du système est plus élevée.

4.1. Collection de test

Pour arriver à une telle évaluation, on doit connaître d'abord les réponses idéales de l'utilisateur. Ainsi, l'évaluation d'un système se fait à l'aide d'un corpus de test.

Dans un corpus de test, il y a : un ensemble de documents, un ensemble de requêtes, la liste de documents pertinents pour chaque requête.

Pour qu'un corpus de test soit significatif, il faut qu'il possède un nombre de documents assez élevé. Pour la construction d'un corpus de test, les jugements de pertinence constituent la tâche la plus difficile. [Jian-Yun,2001]

En RI, les collections de références peuvent être trouvées grâce à des campagnes d'évaluation, ce sont les principales sources.

Le projet TREC est un programme international initié au début des années 90 par le NIST (National Institute of Standards and Technology) et du DARPA (Defense Advanced Reserach Projet Agency). Ce programme offre des moyens homogènes d'évaluation des systèmes de recherche d'information. Il est devenu la référence en recherche d'information pour diverses raisons. En effet, il a permis de définir les tâches en recherche d'information et de construire de larges collections de tests.

Dans ce qui suit, nous allons définir les différents éléments qui constituent le projet TREC:

- **Tâches** : L'objectif est de permettre l'évaluation d'approches spécifiques en recherche d'information concernant le filtrage, le croisement de langues, la recherche dans de très large corpus (100 giga octet et plus), les modèles d'interactions.
- **Les participants** : 25 groupes ont participé à la première édition de TREC en 1992 et 66 groupes de 16 pays différents ont également participé à TREC8.
- **Source d'information** : les documents de la collection sont issus de la presse écrite en 1999 (Financial Time, Résumés de publication USDOE, SAN Jose Mercury news, etc.).
- **Structure et principe de construction de la collection** : un document TREC est généralement présenté sous le format SGML. Il est identifié par un numéro et décrit par un auteur, une date de production et un contenu textuel. Une requête TREC est

également identifiée par un numéro. Elle est décrite par un sujet générique, une description brève et une description étendue sur les caractéristiques des documents pertinents associés à la requête.

4.2. Les mesures d'évaluation d'un SRI

Pour faire l'évaluation de la performance d'un SRI les deux métriques de base les plus utilisées sont la précision et le rappel. Celles-ci sont définies pour le cas simple où un système renvoie un ensemble de documents vis-à-vis d'une requête [Voorhees, 2006] :

- **La précision :** détermine l'aptitude d'un SRI à rejeter les documents non pertinents pour une requête, il est exprimé par :

$$\mathrm{P} = \frac{|\text{Documents pertinents trouvés (RA)}|}{|\text{Documents trouvés (A)}|} \tag{1.9}$$

- **Le rappel :** la capacité du système à restituer le maximum de documents pertinents pour une requête. Il mesure la proportion de documents pertinents restitués par le système relativement à l'ensemble des documents pertinents contenus dans la base documentaire, il est exprimé par :

$$R\mathrm{p} = \frac{|\text{Documents pertinents trouvés (RA)}|}{|\text{Documents pertinents (R)}|} \tag{1.10}$$

L'obtention d'une précision d'environ 30 % et d'un taux de rappel 60 % est courante. Cependant, il convient de noter que ces deux métriques ne sont pas indépendantes ; ils sont liés les uns aux autres de manière significative. Au fur et à mesure que l'un monte, l'autre diminue. Il est vain d'évaluer l'excellence d'un système basé sur une seule métrique. Au lieu de cela, nous devons considérer la courbe précision-rappel du système qui a généralement un modèle spécifique [Jian-Yun, 2001].

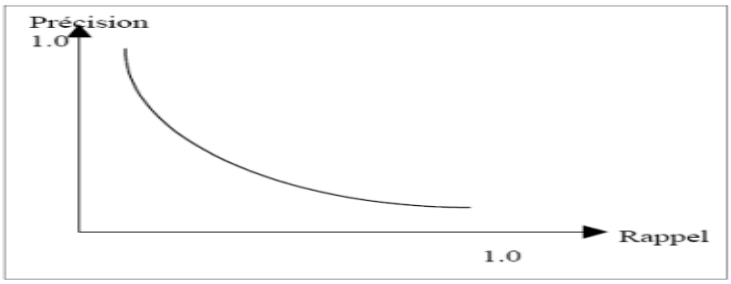


FIGURE 1. 2 : COURBE GENERALE DE PRECISION/RAPPEL

- **F-mesure :** La F-mesure, est une mesure qui combine la précision et le rappel, nommée F-mesure ou F-score introduite par [Rijsbergen, 1979] est définie par :

$$\text{F} - \text{mesure} = \frac{2*\text{P}*\text{Rp}}{\text{P}+\text{Rp}} \qquad (1.11)$$

Tel que P : précision et RP : rappel.

- **La Précision Moyenne (Average precision-AP) :** C'est la moyenne des valeurs de précisions après chaque document pertinent, elle se calcule comme suit

$$AP_q = \frac{1}{R}\Sigma_{i=1}^{N}(p(i) \times R(i)) \qquad (1.12)$$

Où :

R(i) = 1, si le i^{ème} document est pertinent.

R(i)=0, si le document i ^{ème} est non pertinent.

p(i) : la précision à i documents restitués.

R : le nombre de documents pertinents pour la requête q.

N : le nombre de documents restitué par le système.

- **MAP (Mean Average Precision) :** C'est la moyenne des précisions moyennes (Average precision-AP) obtenues sur l'ensemble des requêtes à chaque fois qu'un document pertinent est retrouvé.

$$\boldsymbol{MAP = \frac{\sum_{q \in Q} AP}{|Q|}} \qquad (1.13)$$

Avec :

***AP* :** est la précision moyenne d'une requête q,

Q : est l'ensemble des requêtes.

|Q| : est le nombre de requêtes.

Conclusion

Dans ce chapitre, nous avons présenté les notions de base de la RI telle que les modèles de recherche et le processus de RI (indexation, appariement) et nous avons fini par présenter les étapes d'évaluation d'un SRI et les mesures utilisées pour évaluer les modèles et les systèmes de recherche.

La recherche d'information classique se base sur un processus simple qui permet de retourner les documents qu'il faut selon les besoins qui sont exprimés sous forme de requêtes et qui est très efficace pour ce qui est des documents classiques, mais la RI classique deviens vite obsolète dans un contexte de microblogging.

La particularité des microblogs et l'apparition des réseaux sociaux ont mis au défi la RI. Et pour répondre aux spécificités des réseaux sociaux un nouveau type d'approches fait son apparition et ces approches sont recensées dans ce qu'on appelle la recherche d'informations sociale et c'est là l'objet de notre prochain chapitre.

Chapitre 2 Recherche d’information adhoc dans les microblog

Introduction

Les microblogs sont une version condensée des blogs traditionnels et constituent une source d'information en temps réel. Les utilisateurs exploitent des plateformes de microblogging pour partager et consulter des microblogs. Ces plateformes, sous forme de réseaux sociaux, se caractérisent par des interactions sociales intenses et une diversité de sujets abordés, ce qui les différencie des autres sources d'information.

Il existe plusieurs plateformes de microblogging, y compris Twitter, Friend Feed, Tumblr et Posterous, parmi lesquelles Twitter domine incontestablement. Avec plus de 238 millions d'utilisateurs publiant en moyenne 850 millions de tweets chaque jour[1], Twitter sert également de source d'information majeure, avec environ 2,1 milliards de requêtes soumises chaque jour à son moteur de recherche.

La recherche d'informations (RI) dans les microblogs présente des différences notables par rapport à la recherche sur le web. Ces différences sont attribuées à la nature unique des microblogs par rapport aux documents web - notamment leur contenu concis et souvent mal orthographié - ainsi qu'aux motivations de recherche spécifiques, comme la recherche d'informations récentes. En conséquence, la RI dans les microblogs pose des défis considérables pour les modèles de recherche d'informations actuels.

Dans ce chapitre, nous nous concentrerons sur Twitter et explorerons en détail la recherche d'informations dans l'ensemble de données des tweets.

1. La recherche d'information dans les réseaux sociaux

L'information sociale dans le web est basée sur l'internet de plus en plus influencé par des services intelligents présentés par la suite, qui permettent à l'utilisateur de contribuer au développement, d'annoter et de collaborer dans la production du contenu. Les utilisateurs sont passés de simples consommateurs à des producteurs d'information. Leurs contributions peuvent être de différentes natures : les contenus publiés dans les plateformes sociales telles que les blogs et les wikis, les réactions, les informations publiées par les autres utilisateurs tels que les annotations et les commentaires, etc. L'ensemble de ces informations est appelé contenu généré par des utilisateurs. [Damak et *al* ,2013].

[1] URL : https://www.proinfluent.com/nombre-utilisateurs-twitter

1.1 Les réseaux sociaux

Les médias sociaux se réfèrent à des sites et applications conçus pour faciliter la communication entre les individus du monde entier en interagissant à travers des publications, des discussions ou des appels vocaux et vidéo. Les médias sociaux visent à construire et faciliter la communication entre les communautés du monde entier en permettant aux personnes de partager leurs intérêts, activités et opinions à travers ces applications.

La capacité de partager du contenu, qu'il s'agisse de photos, de publications ou d'événements, a changé notre façon de vivre et a ajouté de nouvelles méthodes qui facilitent de nombreuses tâches. La plupart des grandes entreprises possèdent des comptes sur différents médias sociaux. Nous présentons dans le tableau (2.1) , le nombre d'utilisateurs actifs de chacun des principaux réseaux sociaux[2] :

Réseau social	Nombre d'utilisateur 2023
Facebook	2.963 Md
YouTube	2.527 Md
WhatsApp	2 Md
WeChat	2 Md
Tiktak	1.313 md
Messenger	1.92

TABLEAU 2. 1 : LES RESEAUX SOCIAUX LES PLUS POPULAIRES

1.2. Les types de réseaux sociaux

Les réseaux sociaux sont classés selon le contenu d'information existée dans ses plateformes pour cela on les distingue on deux catégories :

1.2.1. Le contenu généré par l'utilisateur (User Generated Content)

Le terme "User Generated Content" (UGC) désigne le contenu généré par les utilisateurs. Il fait référence à tout type de contenu, qu'il s'agisse de textes, d'images, de vidéos, de commentaires, de publications sur les réseaux sociaux, de blogs, etc., créé et partagé par des utilisateurs sur différentes plateformes en ligne. Parmi les on a :

[2] URL: https://www.blogdumoderateur.com/chiffres-reseaux-sociaux/

- **Wiki :** est un site web dynamique où l'information est construite avec la participation de plusieurs personnes, tout utilisateur peut créer, modifier et supprimer des contenus de manière collaborative, chaque modification est sauvegardée et les versions historiques restent toujours accessibles.
- **Forum** : est un espace d'échange d'informations où les internautes posent ou répondent à une question donnée. Les différentes contributions forment un fil de discussion. Les forums sont classés par thèmes bien précis. Les messages publiés dans les forums par les internautes sont archivés. Ce qui leur permet d'y participer d'une manière asynchrone.
- **Microblog** : est un type de blog dans lequel les utilisateurs peuvent publier de petits morceaux de contenu numérique comme les images, les vidéos, ou l'audio sur internet ces publications appelées micro-messages, sont immédiatement accessibles à une petite communauté ou au grand publique le point de différence entre eux est la longueur du contenu qui est plus petite.
- **Les réseaux sociaux numériques** : un réseau social numérique est un site internet qui permet aux internautes de créer une page personnelle pour partager et échanger des informations, des médias avec leurs communautés d'amis ainsi leur réseau de connaissances qui les réunissent via des échanges personnalisés, chacun peut lire les messages de tel ou tel autre utilisateur.

1.2.2. Le contenu généré par la pratique

Le contenu généré par la pratique comprend des informations communicatives qui fournissent indirectement des renseignements sur les interactions, les émotions, les relations et les comportements sociaux. Ces informations peuvent être obtenues de différentes manières :

- **Les traces des utilisateurs** : Les activités de navigation en ligne des utilisateurs, telles que les pages web visitées, les clics effectués, la durée des visites, peuvent révéler leurs préférences et leurs domaines de recherche

- **Les données personnelles :** Lors de leur inscription sur les réseaux sociaux, les utilisateurs fournissent des informations personnelles qui peuvent être exploitées pour comprendre leurs caractéristiques et leurs intérêts.

- **Les liens sociaux :** Les plateformes sociales établissent des relations entre les utilisateurs selon des règles spécifiques. Certaines plateformes, comme Twitter,

permettent des liens sociaux sans restriction, sauf si le compte est privé. En revanche, d'autres plateformes comme Facebook exigent un consentement mutuel pour partager des informations entre utilisateurs [Damek,2014].

2. Présentation de Twitter.

Twitter a été créé en 21 mars 2006 à San Francisco au sein de la société américaine startup Odeo fondée par Noah Glass et Evan Williams, et Jack Dorsey. Cette société proposait une plateforme d'hébergement, de diffusion et d'enregistrement de podcast. L'idée de départ lancée par Jack Dorsey était de permettre aux utilisateurs de partager facilement leurs petits moments de vie avec leurs amis.

3. Spécificités de la recherche d'information dans les microblogs

Les microblogs sont des plateformes qui permettent aux utilisateurs de partager de courts messages, généralement moins de 200 caractères, qui peuvent comprendre du texte, des liens, des images, des GIFs, des vidéos, et plus encore. Ils sont généralement utilisés pour des mises à jour rapides et fréquentes, et ils encouragent l'interaction et le partage d'information en temps réel. Twitter est probablement le meilleur exemple d'une plateforme de microblogging.

Le moteur de recherche des microblogs offre une expérience spécifique, qui diffère de celle des moteurs de recherche traditionnels comme Google. Il prend en compte non seulement les mots-clés, mais aussi des comptes utilisateurs, des hashtags et des URLs. Les résultats présentés dépendent du type de données recherchées, affichant soit le profil d'un utilisateur si un compte est sélectionné, soit une liste de microblogs contenant les termes recherchés. Par défaut, les résultats sont présentés en ordre chronologique inverse, mais ils peuvent être triés par pertinence, basée sur leur popularité ou leur fréquence de retweets.

Une étude de [Teevan et *al*., 2011] a révélé que les utilisateurs de Twitter recherchent principalement des informations récentes (49 % des participants), des informations sociales (26 %) et des informations sur des sujets spécifiques (36 %). Les recherches effectuées sur Twitter diffèrent de celles effectuées sur les moteurs de recherche traditionnels en termes de longueur des requêtes, de la présence de noms de célébrités ou de hashtags, de la fréquence de soumission des requêtes et de la durée des sessions de recherche.

En résumé, les plateformes de microblogging, comme Twitter, représentent une nouvelle source d'information en constante évolution grâce à leurs caractéristiques uniques.

Cela inclut le partage d'informations en temps réel, les abonnements sans restriction, la faible longueur des messages et l'utilisation d'un jargon spécifique à Internet. Ces nouvelles fonctionnalités et formes ont conduit à de nouveaux usages par les individus et les organisations, comme le suivi des célébrités, l'analyse de l'humeur en temps réel, et la participation à distance à des conférences [Damak, 2014].

4. La recherche d'information Adhoc dans les microblogs

La recherche d'informations Ad hoc, également connue sous le nom de recherche Ad hoc, est une méthode de recherche d'informations qui vise à répondre à une requête spécifique de manière instantanée et ponctuelle à partir d'une vaste collection de documents tels que des articles, des pages web, des livres, etc. Elle se concentre sur la satisfaction immédiate d'une demande d'information précise plutôt que sur une recherche continue ou à long terme [Efron,2011].

Pour effectuer ce type de recherche, l'utilisateur formule une requête en utilisant des mots-clés ou des expressions qui décrivent le sujet recherché. Les systèmes de recherche Ad hoc, tels que les moteurs de recherche, utilisent des algorithmes et des index pour évaluer rapidement une grande quantité de documents et fournir les résultats les plus pertinents. Leur objective est de fournir efficacement et précisément les informations recherchées en tenant compte du contexte et des besoins spécifiques de la requête.

Cette approche est largement utilisée dans divers domaines, tels que la recherche d'information dans les Microblogs. Elle permet aux utilisateurs de trouver rapidement les informations qu'ils recherchent dans les vastes collections. Des facteurs de pertinence sont utilisés pour améliorer la pertinence de la recherche, par la suite nous décrivant les plus importants.

4.1. Facteur de pertinence temporelle :

Il s'agit d'une nouvelle tendance dans la recherche d'informations dans les tweets. En raison de la brièveté du texte des tweets, les résultats de recherche liés uniquement à la pertinence du contenu ne peuvent pas satisfaire les besoins d'information des utilisateurs. La recherche temporelle révèle des performances d'extraction des tweets pertinent.

Les travaux présentés dans ce domaine peuvent être divisés en deux catégories. La première considère les tweets récents comme pertinents pour la requête et présente des modèles pour les sélectionner. La deuxième considère que les tweets pertinents sont ceux qui

figurent dans les grandes concentrations des tweets, dans ce cas aussi plusieurs approches ont été également proposées.

4.2. Facteur de pertinence social

Les microblogs peuvent être triés et classés en exploitant le réseau social sous-jacent aux plateformes de microblogging. Cette méthodologie suggère que la pertinence d'un tweet est déterminée par la crédibilité de son auteur. Les critères pour évaluer cette crédibilité peuvent inclure le nombre de tweets d'un auteur, le nombre de retweets, les citations, les abonnements et les abonnés [Zhao et *al*, 2011 ; Damak et *al*., 2011].

Des techniques d'apprentissage machine, comme les SVM [Joachims, 2005] et la régression linéaire peuvent être utilisées pour traiter ces critères. De plus, des graphes de liens sociaux peuvent être générés pour représenter les différentes relations et interactions au sein des plateformes de microblogging [Jabeur et *al*., 2012].

Enfin, une autre approche consiste à exploiter les informations sociales de l'utilisateur recherchant l'information, en les comparant avec les informations sociales liées aux tweets pour obtenir des résultats personnalisés [Feng et Wang, 2013].

Cependant, comme le montre l'étude de [Kwak et *al*.,2010], il existe une discordance entre les résultats des différentes approches, indiquant un besoin d'études plus approfondies pour définir l'importance d'un utilisateur dans le contexte du microblogging.

4.3. Facteur de pertinence textuelle.

La question principale de la pertinence textuelle dans la recherche de microblogs est intrinsèquement liée à leur courte longueur. Les modèles de recherche d'information (RI) traditionnels, qui dépendent généralement de facteurs tels que la fréquence des termes dans les documents et la longueur des documents, sont restreints par la brièveté des microblogs où les termes n'apparaissent généralement qu'une seule fois.

Certaines études ont constaté que l'application de ces facteurs dans le modèle BM25 non seulement est inefficace, mais peut aussi nuire aux résultats [Ferguson et *al*., 2012]. Pour pallier ces problèmes, d'autres approches, comme celle proposée par [Lin et *al*., 2012], se sont concentrées sur la co-occurrence des termes pour établir des scores de pertinence.

L'amélioration de la représentation des termes est une autre stratégie adoptée pour contrecarrer les limitations des microblogs. Cela inclut l'extension des requêtes avec des

termes fréquents ou temporellement pertinents [Efron, 2011], ainsi que l'enrichissement des microblogs avec du contenu similaire ou l'établissement d'un profil temporel pour chaque microblog [Efron et *al*., 2011].

4.4. Facteur de pertinence d'hypertextualité

Dans les microblogs, les utilisateurs partagent régulièrement des URLs qui ajoutent une dimension informative à leurs publications. Ces URLs sont utilisées pour améliorer la qualité des résultats de recherche, soit par la simple présence ou par la fréquence de ces URLs. Elles aident également à caractériser l'écosystème des microblogs en mesurant l'importance et la fiabilité des tweets. Enfin, le contenu des URLs est utilisé pour enrichir le vocabulaire des tweets, offrant ainsi une meilleure qualité de résultats lors de leur utilisation.

4.5. Autres facteurs de pertinence

D'autres facteurs peuvent être utilisés dans la recherche sur le microblogging. Le facteur de qualité de Weibo est indépendant de la requête. Distinctif Microblogging (qualité du langage, brièveté du texte...), ces critères sont essentiels pour évaluer la qualité de microblog. Voici les normes les plus populaires dans la littérature :

- Longueur du microblog
- Fréquence de Retweets
- Fréquence de hashtags
- Qualité du langage

5. Travaux voisins

Il existe plusieurs travaux qui ont contribué dans le domaine de recherche d'information dans les tweets par l'usage de l'aspect temporel comme facteur de pertinence, nous résumons ci-après les plus importants dans le tableaux (2.2) :

Titre	Résumé	Méthode	Référence
Incorporating Temporal Informationing Microblog Retrieval	Ils ont proposé trois méthodes pour la recherche temporelle des tweets, la première favorise les termes récents ayant une cooccurrence élevée avec tous les termes de la requête, la deuxième favorise les tweets pertinents qui appartiennent aux périodes de grande concentration des tweets, la troisième favorise les termes qui appartiennent à des tweets pertinents qui figurent dans les grandes concentrations des tweets et qui ont une occurrence élevée avec tous les termes de la requête.	-Peak-Finding. - Fraicheur. -Burst.	[Willis,2012]
Temporal Feedback for Tweet Search with Non-Parametric Density Estimation	Ces derniers ont hypothèse qu'il existe une densité **fq** au cours du temps de corpus, de sorte que **fq** est grand pour les moments où les documents pertinents sont susceptibles d'apparaître et de petits dans le cas inverse. Alors pour promouvoir les tweets dont leur temps coïncide avec une grande valeur de la densité. Ils ont utilisé la densité du noyau d'une loi normal. Comme ils ont pondéré chaque kernel par le score thématique du tweet correspondant vit à vie la requête. Se la va permettre d'amplifier la densité des régions temporelles ou figure des tweets pertinents.	-Fraicheur -Estimations de la densité du noyau (KDE) avec Trois pondérations différentes. -La méthode « The moving window ».	[Efron,2014]
Combining Temporal and Content Aware Features for Microblog Retrieval	Dans cet article, ils ont proposé une méthode pour redéfinir le résultat de la recherche en fonction des caractéristiques temporelles, des fonctionnalités liées au compte et des fonctionnalités spécifiques au twitter, ainsi que des fonctionnalités textuelles des tweets. Ils ont également appliqué une technique d'expansion de la requête en deux étapes pour améliorer la	-Modèle de langue avec lissage Dirichlet -Modèle d'espace vectoriel -URL - Compte	[Chy,2015]

	pertinence de sélection des tweets. Ils ont effectué leurs expériences sur la collection TREC 2011.	Retweet -Compte de statu	
TAKer: Fine-Grained Time-Aware Microblog Search with Kernel Density Estimation	Ont proposé un nouveau cadre pour le reclassement temporel des documents, le modèle temporel prenant en considération pour la récupération le prédicteur temporel des mots en plus du prédicteur temporel des documents. Ils ont proposé une estimation via la densité du noyau basé sur l'aspect temporelle.	Densité de noyau Modèle de langue Modèle BM25	[Chen ,2018]

TABLEAU 2. 2: TRAVAUX VOISIN

D'après les travaux liés à la recherche d'information dans les Microblogs (tweets) pertinents qui ont été citées précédemment, on peut sortir avec les points suivants :

- Le tweet est un document court. Ecrit souvent avec un langage mal orthographie, contenant des abréviations et des argots afin de transcrire l'information avec un nombre minimum de caractères. Ceci présente un véritable défi pour la restitution des documents pertinent avec les systèmes de recherche d'information classique.
- Pour les requêtes non sensibles au temps, il est inutile d'introduire l'aspect temporel.
- Pour les requêtes sensibles au temps, l'introduction de preuves temporelles en conjonction avec des preuves thématiques est la nouvelle tendance des travaux récents pour surmonter les inconvénients des modèles IR traditionnels dans les tweets et pour améliorer les performances de la recherche. Le temps est souvent représenté par les horodatages des tweets (temps de du tweet), le temps de soumission de la requête et les expressions temporelles.
- L'expansion de la requête est parmi les techniques les plus utilisées pour une meilleure contextualisation de la requête.
- Selon le type temporel de la requête, plusieurs méthodes ont été proposées ou utilisées pour la recherche des tweets pertinents.

- Pour les requêtes sensibles aux tweets récents, les techniques utilisées partent du principe que les tweets récents sont les plus pertinents.
- Pour les requêtes sensibles aux tweets soumis suite à un événement. La question est : comment trouver le moment important pour cet événement ?

Pour améliorer la précision des systèmes de recherche d'information dans les Microblogs, ça serait intéressent d'utiliser l'expansion de la requête via la conjonction des aspects temporel et thématique.

Conclusion :

Dans ce chapitre nous avons présenté les notions principales auxquelles nous faisons appel comme support pour la modélisation de nos propositions. Nous souhaitons apporter des contributions pour améliorer la recherche d'informations dans les grandes collections de tweets. Via l'usage de l'aspect thématique et l'aspect temporel.

Chapitre 3 : Conception

1. Introduction :

Dans ce chapitre, nous allons détailler notre approche pour l'expansion des requetés afin d'améliorer les performances d'appariement tweet-requête. Notre travail est constitué de plusieurs phases (collecte de tweets, prétraitement du corpus, indexation du corpus, typage de la requête, détection de bursts, détection de topics, expansion de la requête). Notre objectif est de trouver les tweets les plus pertinents temporellement et thématiquement à la requête. Ensuite utiliser les meilleurs termes de ces tweets pour élargir la requête. La nouvelle requête va garantir une meilleure restitution des tweets pertinents.

2. Approche proposée :

Plusieurs techniques d'expansion de la requête sont proposées par les travaux de l'état de l'art pour améliorer la sélectivité de la recherche des tweets pertinents. Parmi ces contributions, il y a une catégorie qui se base sur l'hypothèse termes clustering « documents relevant to a query Q will form clusters in a term space » [Jardine and Rijsbergen 1971] cela signifie que les documents pertinents partagent le même vocabulaire. La sélection des meilleurs termes de ces documents permet un bon enrichissement de la requête, ce qui permet d'améliorer le classement des documents pertinents dans la liste de résultats de la recherche. Une deuxième catégorise qui se base sur l'hypothèse de temporal clustering « documents relevant to a query will form clusters along a timeline » (Efron et al. 2014), cette hypothèse se base sur l'aspect temporal pour estimer la pertinence des documents. Elle considère que les documents pertinents qui contiennent évidemment les meilleurs termes d'expansion sont ceux qui se concentrent dans des intervalles temporels avant le temp de soumission de la requête. Cette hypothèse a montré largement son efficacité dans les travaux (Efron et al. 2014 ; Rao et al. 2015 ; Rao and Lin 2016 ; Rao et al. 2017).

De notre tour, nous proposons l'hypothèse suivante qui est une hybridation des hypothèses précédentes : « les tweets pertinents se regroupent ensemble dans des intervalles temporels dont partagent le même vocabulaire ».

À fin de réaliser notre hypothèse, nous proposons l'architecture ci-dessous qui présente les étapes nécessaires pour l'enrichissement de la requête. Par la suite nous allons détailler ses différents composants.

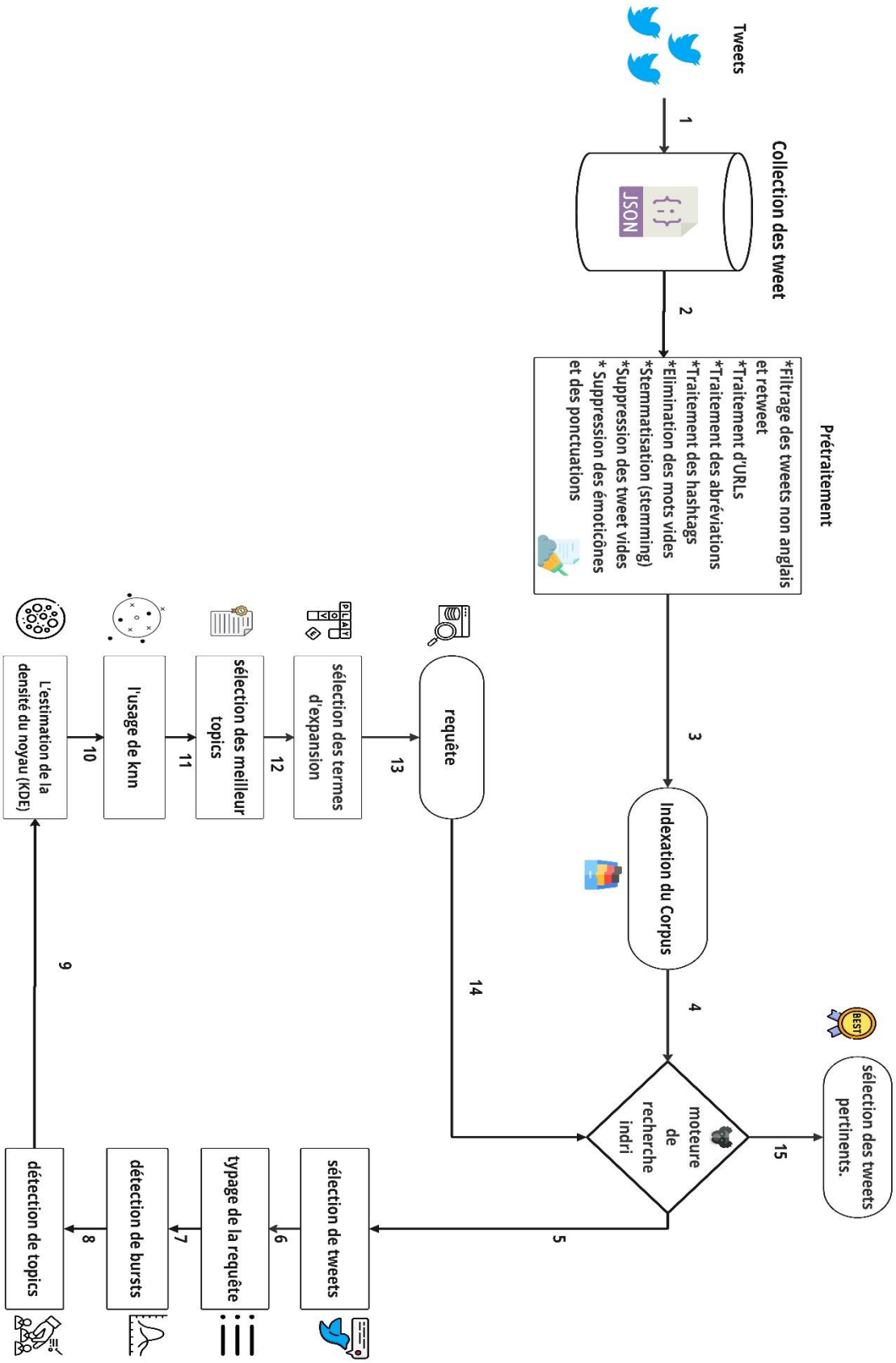


FIGURE 3. 1: ARCHITECTURE FONCTIONNELLE GLOBALE DE NOTRE SYSTEME

3. La collecte de données

Cette étape est la plus lente du notre projet, elle consiste à télécharger le corpus des tweets **TREC2011** qui contient 16 millions de tweets, publié entre le 23 janvier 2011 et le 7 février 2011.

Le corpus de tweets utilisé dans la piste Microblog de TREC 2011, est distribué sous forme de 15 répertoires, chacun contenant environ 100 fichiers .DAT, chacun contenant une liste de (tweet id, user Name, …ect). Chacun de ces fichiers est appelé bloc d'état (c'est-à-dire bloc de tweets). Nous avons utilisé l'API gratuit open source « Twitter Tools »[3] pour le télécharger.

3.1. Prétraitement :

Le contenu textuel des tweets contient des données non structurées, des abréviations, des émoticons, des mots vides de sens, des typographique, … etc. Tout cés donnés nécessitent un prétraitement et un nettoyage.

Cette phase a une très grande importance car la recherche doit se faire sur un corpus épuré sinon les résultats n'auront aucun sens.

Le prétraitement est divisé en plusieurs étapes citées ci-dessous :

- **Filtrage des tweets non anglais :**

  Nous n'avons conservé que les tweets rédigés en anglais dans le corpus TREC 2011 microblog tweet, étant donné que l'anglais est la langue principale de ce corpus. Par conséquent, tous les tweets rédigés dans d'autres langues ont été supprimés.

- **Suppression des retweets :**

  Pour éviter la redondance des tweets, qui va forcément influencer le résultat de la recherche, nous avons décidé de supprimer les retweets.

- **Traitement d'URLs**

  Nous avons mené une série d'étapes pour remplacer chaque URL qui figure dans le tweet par les mots clés de sa page web. Au départ, nous avons identifié toutes les URLs dans les tweets grâce à une expression régulière, puis nous les avons stockées dans un dictionnaire où chaque clé est un ID de tweet et chaque valeur est une liste

[3] https://github.com/lintool/twitter-tools

d'URLs pour ce tweet. Ensuite, nous avons extrait les mots clés depuis les balises 'meta-name="keywords"' et '<title>'. En utilisant le dictionnaire que nous avons créé, nous avons remplacé chaque URL dans un tweet par ses mots clés.

Cette tâche est représentée dans la figure (3.2) :

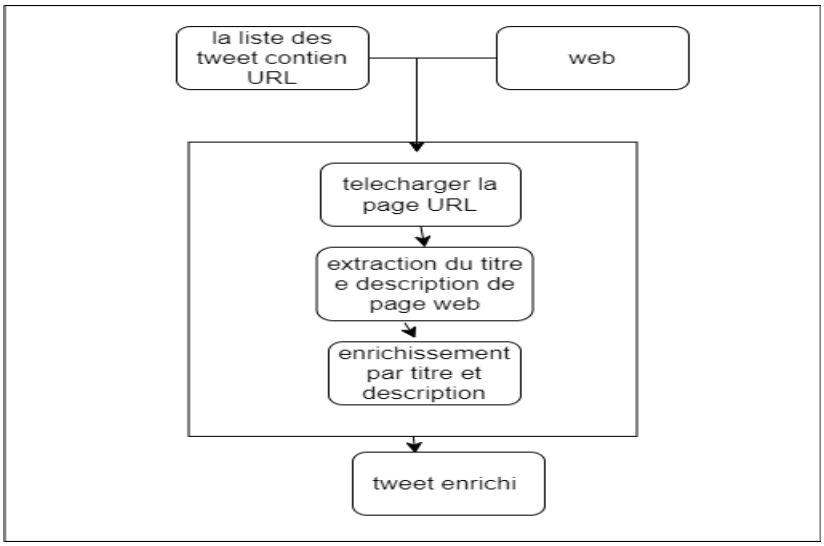


FIGURE 3.2 : TRAITEMENT D'URL

- **Traitement des abréviations**

Un traitement des abréviations utilisées dans les tweets a été effectué. Une liste des abréviations couramment employées sur Twitter a été créée, et un algorithme a été développé pour remplacer ces abréviations par leur signification respective dans les tweets. Cette tâche vise à améliorer la lisibilité des données et à faciliter leur analyse ultérieure en évitant les ambiguïtés causées par les abréviations. L'efficacité de l'algorithme a été évaluée en le testant sur un ensemble de tweets représentatifs.

Par exemple, l'abréviation "2day" est remplacée par "today" et "4ever" devient "forever". Ces transformations démontrent comment l'algorithme permet de traduire les abréviations couramment utilisées sur Twitter en leur signification complète, contribuant ainsi à une meilleure compréhension des tweets.

- **Traitement des hashtags**

Les hashtags sont recueillis du corps du tweet en utilisant les informations d'entités incluses dans ce dernier. Par la suite, chaque hashtag est décomposé en mots distincts grâce à l'emploi de la bibliothèque "wordsegment". Par exemple, le hashtag

"#ukrainewar " serait décomposé en "Ukraine" et "war ". Ces segments sont alors réinsérés dans le texte initial du tweet, ce qui préserve la structure globale du tweet tout en segmentant les hashtags. Cette démarche vise à augmenter la richesse du texte des tweets en y incluant davantage d'informations, pour ainsi améliorer la lisibilité du contenu des tweets.

- **Elimination des mots vides (stop words)**

Les mots vides (ou mots vides, en anglais) sont des mots très courants qui sont utilisés dans presque tous les textes. Leur présence peut dégrader les performances de l'algorithme de recherche. En anglais, les mots vides peuvent être : « are », « after », « and », « before »… etc.

Ce mot apparaît avec une fréquence semblable dans chacun des textes de la collection. Pour les filtrer et les supprimer de notre corpus nous avons utilisé une liste qui contient ces mots vides.

- **Stemmatisation (stemming)**

La stemmatisation est le processus d'élimination de suffixes des mots afin d'obtenir leur racine commune. Cela permet de générer la forme de base (souvent tronquée) appelée le stem (Racine en français). Par exemple : {computer, computing, computation} devient « comput ».

- **Suppression des tweet vides**
- **Suppression des émoticônes et des ponctuations**

Cette étape consiste à éliminer les émoticônes telle que (☺) ainsi que les ponctuations (, ; ! ?).

Dès que cette étape terminée, notre corpus de Tweet sera prêt à l'utilisation, ce qui n'est pas le cas avant le prétraitement.

4. Indexation du Corpus Tweets2011 :

Plusieurs moteurs de recherche d'information open source ont vu la lumière dans ces dernières années. Parmi les, nous citons : Indri, IXE, Lucene. Chacun de ces moteurs présente à la fois des avantages et des inconvénients

Dans l'objectif d'indexer le corpus Tweets2011 et d'effectuer une première recherche pour chaque requête. Nous avons choisi le moteur de recherche Indri qui fait partie du projet

LEMUR [4] mené par un laboratoire de l'université Massachusetts, il est basé sur un modèle de langue, qui est un type de modèle de recherche probabiliste. Le modèle de langue est une approche statistique pour la recherche d'information qui traite le contenu des documents et des requêtes comme des échantillons de distributions de probabilité.

Nous avons choisi ce moteur en raison de sa capacité à gérer et indexer de vastes collections de documents. De plus, il offre la possibilité de prendre en charge les requêtes structurées, permettant aux utilisateurs de combiner diverses requêtes et opérateurs pour une recherche plus précise et efficace.

Le score d'appariement d'un document par rapport à une requête est calculé par Indri via la formule suivante :

$$\text{Score}(d|q) = \Sigma(\text{poids}(t) * P(t|d) * \log(P(t|q)))$$

Tel que :

- **t** : est le terme
- **P(t|d) :** représente la probabilité d'occurrence du terme t dans le document.
- **P(t|q) :** représente la probabilité d'occurrence du terme t dans la requête.
- **poids(t) :** est le poids attribué au terme t, qui peut être calculé en utilisant des techniques de pondération telles que TF-IDF.

5. Typage des requêtes :

Dans cette phase, nous procédons à la classification des différentes requêtes en trois catégories distinctes : requête de type évènement, récent et non sensible au temps. Selon le processus suivant :

1. La récupération des 1000 tweets résultat de la recherche avec Indri pour la requête Q.
2. L'établissement de l'histogramme de la requête :
    - L'axe des cordonnés représente : la proportion de tweets publiés chaque jour par rapport au nombre total de tweets (pour un jour donnée la proposition est : le nombre de tweets publiés dans ce jour/ 1000).
    - L'axe des abscisses représente le temps (notre corpus et étalé sur 15 jours).
3. Établissement de deux seuils :

[4] URL: http://www.lemurproject.org/

- P : la proportion minimale de tweets dans un jour donnée pour qu'un évènement se produise.
- S : un seuil utilisé pour déterminer s'il s'agit d'un évènement du type récent.

4. Selon S et P nous déterminons si la requête est insensible au temps, de type récent ou de type événement.

5.1. Les requêtes insensibles au temps (Time-Insensitive)

Ce type de requête est caractérisé par le fait qu'il n'est pas sensible au temps, c'est-à-dire que les réponses à cette requête ne changent pas ou ne dépendent pas significativement du moment où la requête est effectuée. Pour illustrer ce concept, prenons l'exemple de la requête 6, "NSA" (figure3.3). Les réponses où les tweets relatifs à la "NSA" sont relativement constants et ne dépendent pas du moment où la requête est sommée.

De manière plus générale, une requête est probablement classée comme insensible au temps, si elle cherche des informations stables et non variables, comme des faits universellement reconnus ou des définitions de concepts largement acceptés. Ces types de données ne subissent pas de modifications significatives au fil du temps.

Dans le contexte de notre travail, une requête peut être considérée comme insensible au temps si le Pic n'atteint pas un certain seuil prédéfini noté **P**. Dans notre cas, la valeur de **P** est 0.05. Cette valeur détermine qu'il n'y a pas une concentration significative de tweets à un moment particulier.

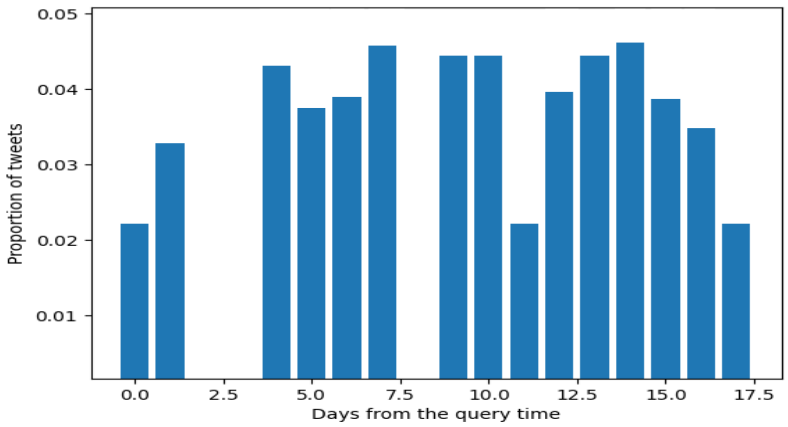


FIGURE 3. 3 : HISTOGRAMMES DE LA REQUETE 6

5.2. Les requêtes de type récent :

Les requêtes de ce type sont généralement associées à un sujet ou un événement qui est très actuel ou qui vient de se produire. Il peut s'agir d'une demande d'informations sur une

nouvelle découverte scientifique, une récente sortie de produit, ou un événement majeur récent dans le monde.

Par exemple, une requête comme "Keith Olbermann new job" (figure3.4) pourrait être classée comme une requête de type récent. En général, ce type de requête se manifeste par une hausse significative des tweets et des discussions en ligne peu de temps après un événement ou une annonce, indiquant une forte concentration des tweets autour du sujet récent.

Dans notre travail, une requête est classée comme étant du type récent si la concentration maximale des tweets se produit dans une période de deux jours avant le temps de soumission de la requête.

Pour qu'une requête soit classée comme étant type récent, nous suggérons que le plus grand Pic doit être plus élevé de 1.5 fois (S= 1.5 fois) que le deuxième plus grand Pic, avec (P > 0.05). Cela produit une concentration significative de tweets, indiquant un sujet d'actualité ou un événement récent.

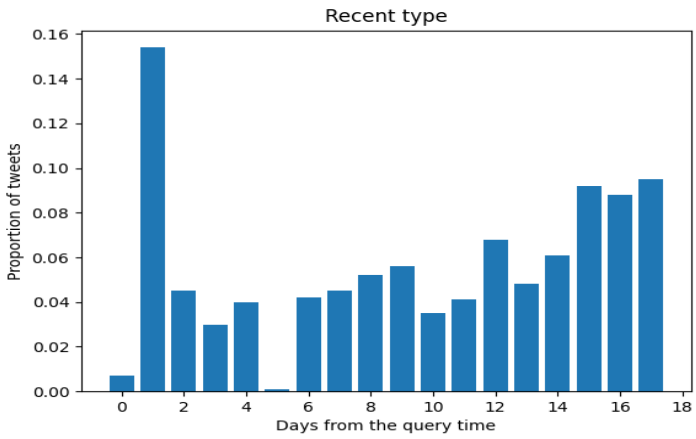


FIGURE 3. 4 : HISTOGRAMME D'UNE REQUETE DE TYPE RECENT

5.3. La requête de type événement :

Les requêtes de type événement sont caractérisées par des modèles d'activité ou de réponses qui sont liés à des occurrences temporelles spécifiques. Ces occurrences peuvent être récurrentes ou uniques, mais elles sont généralement associées à des événements ou des moments précis qui sont pertinents pour la nature de la requête.

Dans notre travail, une requête serait classée comme de type événement si elle montre des pics d'activité qui coïncident avec les moments où les événements associés à la requête

sont susceptibles de se produire. Alors si la requête n'est pas du type récent ou insensible au temps. Alors elle est considérée comme de type évènement.

Pour la requête 2 « FIFA Soccer 2022 » par exemple (voir la figure 3.5), on pourrait observer une augmentation significative d'activité sur tweeter aux moments où le lancement du jeu est annoncé, se déroule où se termine.

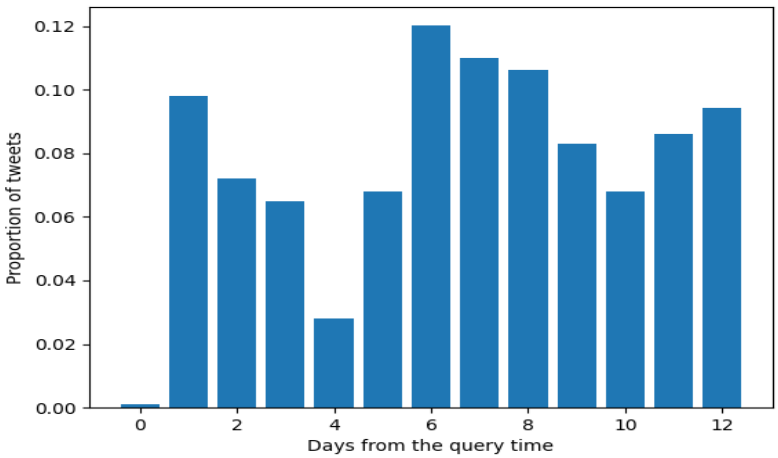


FIGURE 3.5 : HISTOGRAMME D'UNE REQUETE DE TYPE EVENEMENT

Ses caractéristiques temporelles spécifiques permettent de distinguer les requêtes de type événement des autres types de requêtes, en mettant en évidence leur lien direct avec des moments clés et des événements importants.

6. Détection de Bursts

L'objectif du notre travail est de détecter les "bursts" ou les pics d'activité dans les résultats de la recherche pour une requête Q avec Indri dans le corpus Tweets2011. Ces pics d'activité représentent les clusters temporels, aussi les intervalles temporels où les tweets pertinents se présentent.

Nous visons à sélectionner les trois "bursts" les plus significatifs pour une analyse ultérieure. Ces "bursts" sont choisis en se basant sur leur intensité, c'est-à-dire que nous retenons les trois bursts qui ont présenté le plus grand volume de tweets au sein de leur fenêtre temporelle respective.

Cette sélection permet de se concentrer sur les moments où l'interaction des utilisateurs avec l'information était à son paroxysme, ce qui est susceptible de révéler les tweets les plus pertinents sur la propagation de l'information et les comportements des utilisateurs.

Pour réaliser cette tâche, nous avons adapté pour l'algorithme Offline Peak-Finding, tel que présenté par Adam Marcus et ses collaborateurs [Adam Marcus, 09] pour détecter les pics ainsi les bursts.

L'algorithme est basé sur les deux points essentiels :

1. **Détection des pics d'activité** :

L'algorithme analyse une séquence de tweets sur une période donnée pour identifier les fenêtres temporelles où le volume dépasse un certain seuil (τ). Ces fenêtres sont considérées comme des "bursts" ou pics d'activité. Ce processus est réalisé en suivant la séquence de données et en marquant le début d'un "burst" lorsqu'une augmentation significative de la densité des tweets est observée. Le "burst" se termine lorsque la densité des tweets tombe en dessous du seuil τ ou ne montre plus d'augmentation. Plus précisément, un "burst" commence lorsque les conditions suivantes sont satisfaites :

$$\frac{|Ci-mean|}{meandev} > \tau \text{ et } Ci > C_{i+1} \qquad (3.1)$$

Tel que :

- $\mathbf{C_i}$ représente la densité des tweets à l'itération actuelle i
- **Mean** est la moyenne actuelle de la densité des tweets,
- **meandev** est l'écart moyen courant
- $\boldsymbol{\tau}$ est le seuil spécifiquement défini pour la détection des "bursts"

2. **Mise à jour de la moyenne et de la déviation moyenne :**

À chaque nouvelle observation, l'algorithme actualise la moyenne et la déviation moyenne pour refléter les données les plus récentes. Ceci est accompli en utilisant le paramètre alpha (α) qui agit comme un facteur de pondération. Ce facteur détermine l'importance relative des nouvelles observations par rapport aux anciennes dans le calcul de la moyenne et de la déviation moyenne. En ajustant constamment ces deux paramètres en fonction des nouvelles observations, l'algorithme parvient à détecter les "bursts" d'activité de

manière plus précise et réactive. Les formules mathématiques utilisées pour mettre à jour ces valeurs sont :

$$\boldsymbol{newmean = \alpha \times Ci + (1 - \alpha) \times mean} \quad (3.2)$$

$$\boldsymbol{diff = |mean - Ci|} \quad (3.3)$$

$$\boldsymbol{newmeandev = \alpha \times diff + (1 - \alpha) \times meandev} \quad (3.4)$$

Où :

- Newmean : la nouvelle moyenne de la densité des tweets.
- Diff : la différence absolue entre la moyenne actuelle et la densité des tweets à l'itération i.
- Newmeandev **:** la nouvelle déviation moyenne de la densité des tweets.
- α: est le facteur de pondération qui détermine l'importance relative des nouvelles observations par rapport aux anciennes lors de la mise à jour de la moyenne et de la déviation moyenne.

En utilisant cet algorithme nous avons identifié les clusters temporels où les tweets publiés discutent un événement important, qui suscitent l'attention des utilisateurs et entraînent une augmentation significative du volume de publications.

Pour illustrer l'efficacité de cet algorithme dans la détection des "bursts" d'activité, nous allons prendre l'exemple de la requête1. Nous avons appliqué notre algorithme sur les tweets correspondant à cette requête et nous avons pu identifier trois "bursts" significatifs qui indiquent une activité importante des utilisateurs (voir le tableux 3.1)

Burst	Date début	Date fin	Nombre de tweet
Burst 1	27/01/2011	29/01/2011	142
Burst 2	30/01/2011	01/02/2011	118
Burst 3	06/02/2011	07/02/2011	132

TABLEAU 3. 1 : LES BURSTS DE LA REQUETE 1

Le tableau ci-dessus montre les dates de début et de fin pour chacun des trois bursts détectés pour la requête1. Ces intervalles temporels correspondent aux périodes pendant lesquelles l'activité des utilisateurs autour de la requête1 était à son paroxysme. En examinant ces "bursts", nous avons pu identifier les moments clés où les informations pertinentes à la requête1 se sont propagées.

7. Détection de Topics

Après la détection des clusters temporels (bursts) les plus importants, qui figure dans la liste des tweets résultat de la recherche effectuée par Indri. Notre objectif est de grouper les tweets de chaque cluster temporel dans des clusters thématiques. Dans l'optique d'avoir des groupes de tweets, où les documents de chaque groupe partagent le même vocabulaire.

Cette partie de notre approche consiste à appliquer le clustering thématique sur les tweets de chaque cluster temporel pour identifier les topics. Cette modélisation thématique est une technique de traitement du langage naturel (NLP), vise à découvrir les thèmes ou les sujets cachés dans une collection de documents. Les thèmes sont découverts en fonction de la distribution des mots dans les documents. Un modèle thématique attribue chaque mot d'un document à un thème spécifique.

Plusieurs modèles à topic ont été conçus comme Latent Semantic Indexing (LSI) Deerwester et al. (1990), Probabilistic Latent SemanticAnalysis (PLSA) Hofmann (1999) et Latent Semantic Analysis (LSA) Ando and Lee (2001). Un modèle à topic très intéressant a été introduit dernièrement par Blei et al. (2003), nommer Latent Dirichlet Allocation (LDA). Tous les modèles cités auparavant ont montré leur efficacité pour les documents longs [Blei,2007].

Dans le cadre de notre projet, nous avons privilégié l'utilisation du Biterm Topic Model (BTM) [Yan et al,2013] en raison de son efficacité avec les textes courts, tels que les tweets. Le BTM est une méthode avancée de découverte de thèmes dans des textes courts, qui peuvent ne pas présenter une distribution claire des thèmes ou contenir suffisamment de mots pour une analyse statistique solide.

Un "Biterm" dans la terminologie BTM, se définit comme une paire de mots co-occurrents dans un même texte court. Ces biterms fournissent des informations précieuses sur la corrélation sémantique entre les mots, et cette information est utilisée pour découvrir les thèmes. Le BTM, en tant que modèle thématique, est donc particulièrement adapté à l'analyse de textes courts comme les tweets, où la découverte de thèmes peut être plus délicate.

L'algorithme BTM s'articule en deux étapes, pour découvrir les thèmes cachés à partir des textes courts :

- **Modélisation des thèmes :**

Dans la phase de modélisation des thèmes, le BTM utilise une approche d'échantillonnage de Gibbs pour estimer les thèmes. Au départ, chaque biterm est attribué à un thème de manière aléatoire.

Ensuite, en utilisant les thèmes actuellement attribués aux différents biterms, le BTM calcule les probabilités de chaque thème possible pour un biterm spécifique.

Le modèle retire temporairement l'attribution de thème actuel de ce biterm, puis utilise ces probabilités pour réattribuer un thème. Cette réattribution est faite de telle sorte qu'un thème ayant une probabilité plus élevée à la plus grande chance d'être attribué au biterm. Ce processus d'échantillonnage de Gibbs est répété à travers plusieurs itérations, où l'attribution des thèmes à chaque biterm est ajustée et améliorée. Au fur et à mesure des itérations, les attributions de thèmes se stabilisent, convergeant vers une distribution où chaque biterm est attribué au thème qui lui correspond le mieux en fonction du corpus de texte.

- **Agrégation des thèmes** :

Le thème d'un tweet est déterminé en examinant les thèmes de tous les biterms qui composent ce tweet. Le BTM regroupe (ou agrège) ces thèmes pour obtenir une vue d'ensemble des thèmes du tweet. Si un certain thème apparaît plus souvent parmi les biterms du tweet, alors ce thème sera probablement considéré comme le thème principal du tweet.

Après l'application de la méthode BTM sur l'ensemble des tweets, nous avons obtenu des vecteurs de topic. Chaque vecteur donne une distribution des mots sur un topic, avec les probabilités associées (figure3.6). De plus, chaque court texte est représenté par une distribution de thèmes, offrant ainsi une indication des thèmes abordés dans le texte.

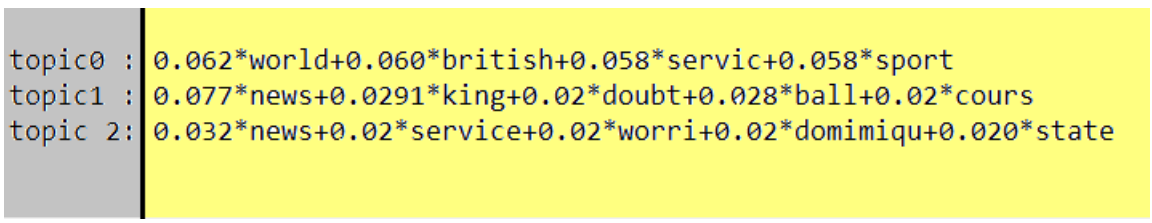


FIGURE 3. 5 : TERMES DES TOPIC

La figure ci-dessus illustre les différents thèmes associés à un "burst" particulier de la requête1. En étudiant ces termes, nous pouvons comprendre les sujets principaux discutés pendant ce "burst" d'activité.

Dans la section suivante nous présentent notre approche pour la sélection des meilleurs topics pour chaque cluster temporel.

7. L'expansion de la requête

Après la détection des différents topics (clusters thématiques) dans les bursts (cluster temporels), nous visons à de sélectionner le plus pertinent topic dans chaque burst. Pour expansé la requête avec les termes les plus fréquents de chaque cluster thématique. Ce qui va permettre une meilleure contextualisation de la requête. Cela va améliorer le classement des tweets pertinents dans la liste des résultats de la première recherche.

Notre méthode se compose de plusieurs étapes :

7.1. L'estimation de la densité du noyau (KDE) de chaque tweet dans chaque topic :

L'estimation par noyau (ou encore kernel density estimation ou KDE)[silverman,1996] est une méthode non-paramétrique d'estimation de la densité de probabilité d'une variable aléatoire. Elle se base sur un échantillon d'une population statistique et permet d'estimer la densité en tout point du support [Efront, 2014].

Dans notre travail, nous avons utilisé la densité du noyau (Kernel Density Estimation – KDE) pour évaluer la pertinence temporelle des tweets de chaque topic. Afin d'estimer la pertinence temporelle de chaque topic.

Sachant que l'application du Biterme fournit pour chaque topic les termes les plus fréquents classés selon leur probabilité. Les termes les plus pertinents sont utilisés pour extraire des tweets de chaque topic. Les dates de publication de ces tweets sont ensuite utilisées comme des données (x1, x2, ..., x_n).

Pour l'estimation de la densité de la date de publication x1 d'un tweet t1 nous avons utilisé la formule 3.5. Dans notre travail x1, x2, ..., x_n représentent les temps de publication des tweets t1, t2.....tn d'un burst Bi.

Les formules utilisées pour effectuer cette estimation sont :

$$F(x) = \frac{1}{nH}\sum_{i=1}^{n} K\left(\frac{x - xi}{H}\right) \quad (3.5)$$

$$K(x) = \frac{1}{\sqrt{2\pi}} e^{\frac{-1}{2}^{x^2}} \quad (3.6)$$

$$H = \left(\frac{4\sigma^5}{3n}\right)^{\frac{-1}{5}} \quad (3.7)$$

$$\sigma = \sqrt{\frac{1}{N}\sum_{i=1}^{N}(x_i - m)^2} \quad (3.8)$$

$$\text{m} = \frac{1}{N}\sum_{i=1}^{N} xi \quad (3.9)$$

Où

- n : est le nombre de tweet d'un burst B1,
- h: est un paramètre nommé fenêtre, qui régit le degré de lissage de l'estimation.
- K : est la densité d'une fonction gaussienne.
- σ: c'est l'écart type des valeurs xi et m c'est la moyenne des écarts.

7.2. La création des classes à partir des topics.

L'algorithme des k plus proches voisins (k-NN) [Diallo et al,2022] est une méthode non paramétrique utilisée pour la classification et la régression. Dans le cadre de notre étude, nous avons utilisé cet algorithme dans le but de convertir chaque topic identifié préalablement par l'algorithme de Bitrerme à un cluster. L'algorithme regroupe tous les tweets d'un topic dans un cluster. Cette transformation est vitale car elle nous permit de déterminer le centroïde de chaque topic.

7.3. L'usage de KNN pour le calcul du centroïde de chaque classe.

Une fois les topics sont convertis en cluster, l'étape suivante est de déterminer les centroïdes de chaque cluster. Les centroïdes sont calculés comme étant les valeurs moyennes du temps de publication et de la densité des tweets dans chaque cluster, offrant une représentation centrale au topic. Les centroïdes sont déterminés par la formule suivant :

$$C = \frac{1}{n} * \sum_{i=1}^{n} P_i \qquad (3.10)$$

- n : est le nombre total de points dans le cluster
- ∑Pi : représente la somme des coordonnées des points.

Dans notre cas, les coordonnées de chaque point sont (x, y) où x est la date de publication du tweet et y est sa densité.

Les distances entre les points et les centroïdes sont ensuite calculées par la formule suivante :

$$\boldsymbol{d(a,b) = \sqrt{(x_b - x_a)^2 + (y_b - y_a)^2}} \qquad (3.11)$$

- (x_a, y_a) représentent la date de publication et la densité du tweet a.
- (x_b, y_b) représentent la date de publication et la densité du tweet b.

Cette méthode nous a permet d'identifier les caractéristiques (densité, temps) centrales de chaque groupe.

7.4. Le choix du meilleur topic pour chaque burst et l'expansion de la requête

Pour chaque burst et pour chaque topic dans ce dernier, nous avons calculé la distance temporelle (la différence entre la caractéristique temporelle du centroïde et la caractéristique temporelle du burst) afin de sélectionner le meilleur topic dans chaque burst. Nous n'avons considéré que le topic le plus proche temporellement au burst comme pertinent, comme nous avant considère les termes les plus probables de se dernière comme étant les plus appropriés pour l'expansion de la requête.

Pour illustrer notre approche, nous présentons ici un exemple concret avec la requête1, mentionnée dans les chapitres précédents. Initialement, notre requête1 peut se présenter sous la forme suivante :" **BBC World Service staff cuts** "

Après avoir identifié les "bursts" les plus significatifs pour cette requête et extrait les topics pertinents de chaque "burst", nous avons calculé la distance temporelle entre chaque topic et le centroïde de son "burst". Le topic ayant la distance temporelle la plus faible est considéré comme le plus proche du "burst", et donc comme le plus pertinent pour l'expansion de la requête.

En utilisons les termes les plus probables de ces topics (news egypt british), nous avons ensuite procédé à l'expansion de la requête 1. La requête élargie pourrait ressembler à ceci " **BBC World Service staff cuts news egypt british** "

En incorporons les termes clés des topics les plus pertinents, nous pouvons améliorer la pertinence de notre requête et ainsi obtenir des résultats de recherche plus précis et plus informatifs.

Conclusion :

Dans ce chapitre nous avons décrit en détail notre approche. Dans le prochain chapitre, nous allons présenter l'implémentation de notre approche ainsi que l'évaluation de nos résultats.

Chapitre 4 :
test
et
implémentation

Introduction

Dans ce présent chapitre, nous décrivons les différents environnements et outils utilisés afin d'implémenter le modèle présenté dans le chapitre 2. Ensuite, nous décrivons le cadre expérimental de notre approche, pour terminer par l'affichage des résultats obtenus et les discuter.

1. Présentation de l'environnement de travail :

Dans cette partie, nous présentons les outils utilisés pour l'implémentation de notre modelé.

1.1. Le langage de programmation python

Le langage Python est un langage de programmation de haut niveau créé par Guido van Rossum et publié pour la première fois en 1991. Grâce à sa syntaxe claire et concise, Python est facile à lire et à écrire, ce qui le rend souvent recommandé comme premier langage de programmation pour les débutants. Il est largement employé dans divers domaines, tels que le développement web, l'automatisation, le calcul scientifique, l'analyse de données et l'apprentissage machine[5].

Python possède plusieurs versions. Pour implémenter notre approche nous avons choisi la version python 3.11.2 qui est sortie en Février 2023.

1.2. Visuel studio code

Afin de programmer avec le langage Python, nous avons choisi L'environnement de développement intégré **visuel studio code** EDI[6] qui permet de développer des applications dans de nombreux langages de programmation. Le choix de cet IDE vient de sa simplicité d'utilisation et du nombre important de possibilités proposées par ce dernier par exemple la génération automatique des composants graphiques et des interfaces.

1.3. Linux

Linux est un système d'exploitation open source (OS) basé sur UNIX créé par Linus Torvalds en 1991. Les utilisateurs peuvent modifier et créer des variations du code source, connues sous le nom de distributions, pour les ordinateurs et autres périphériques. L'utilisation la plus courante est en tant que serveur, mais Linux est également utilisé dans les ordinateurs

[5] URL: https://python.sdv.univ-paris-diderot.fr/01_introduction/

[6] URL : https://visualstudio.microsoft.com/fr/

de bureau, les smartphones, les lecteurs de livres électroniques et les consoles de jeux, etc. Nous somme choisi kali-linux 2023.22a pour notre travail.

2. Les bibliothèques :

2.1. Json

JSON[7] est une bibliothèque python simple pour traiter Json, lire et écrire des données Json, elle utilise Map et List pour traiter les données.

Nous pouvons utiliser Json pour analyser les données et écrire dans le fichier JSON. L'une des meilleures fonctionnalités de json est qu'il ne dépend pas de bibliothèques tierces. Json est une API très légère qui fonctionne bien pour les exigences JSON simples.

2.2. Matplotlib

Matplotlibest[8] une bibliothèque open source disponible pour Python qui permet aux utilisateurs de générer facilement des graphiques et des diagrammes. Il est particulièrement utile pour les utilisateurs qui ont besoin de régénérer des dessins qui changent fréquemment.

2.3. Nltk

La bibliothèque NLTK[9] (Natural Language Toolkit) est une bibliothèque populaire et puissante pour le traitement du langage naturel (NLP) en Python. Elle fournit des outils et des ressources pour travailler avec des données textuelles et effectuer des tâches telles que la tokenization (découpage en unités linguistiques), le stemming (réduction de mots à leur forme de base), le lemmatization (recherche de la forme canonique d'un mot), la partie du discours (POS) tagging (étiquetage grammatical), la reconnaissance d'entités nommées, l'analyse de sentiments, la classification de texte, et bien d'autres.

Il fournit également des outils pour explorer, nettoyer et prétraiter les données textuelles, ainsi que pour appliquer des algorithmes de traitement du langage naturel.

2.4. Pandas : La bibliothèque Pandas[10] est une bibliothèque open-source populaire pour la manipulation et l'analyse de données en Python. Elle fournit des structures de données flexibles et performantes pour traiter des données tabulaires, telles que des tableaux et

[7] URL: https://docs.python.org/fr/3/library/json.html

[8] URL: https://datascientest.com/matplotlib-tout-savoir

[9] URL: https://nltk.org/

[10] URL: http://pandas.pydata.org/

des DataFrames, qui permettent de stocker et de manipuler des données de manière efficace.

2.5. Scipy

Scipy est une bibliothèque open source pour le langage de programmation Python qui est utilisée pour effectuer des calculs scientifiques et techniques. Le nom SciPy est un portemanteau de "Scientific" et "Python". La bibliothèque SciPy est construite sur la bibliothèque NumPy qui fournit des fonctionnalités pour manipuler des tableaux et des matrices de données.

2.6. BeautifulSoup

BeautifulSoup est une bibliothèque Python pour l'analyse syntaxique de documents HTML et XML. Elle crée des parse trees (arbres d'analyse) qui sont utiles pour extraire les données de ces types de documents structurés. Elle offre une interface simple et Pythonique pour naviguer, rechercher et modifier l'arbre d'analyse

2.7. Wordsegment

Wordsegment est une bibliothèque Python dédiée au segmentage de mots dans un texte, principalement utilisée pour les chaînes de caractères sans espaces. Cette bibliothèque est très utile dans le contexte du traitement du langage naturel (NLP) et des applications de recherche d'information, telles que la correction orthographique, la suggestion de recherche et le SEO.

2.8. Sklearn :

C'est une bibliothèque de machine learning en Python. Elle fournit un ensemble d'outils pour la modélisation prédictive, y compris la classification, la régression, le clustering et la réduction de dimensionnalité.

2.9. Numpy :

C'est une bibliothèque très populaire pour le calcul numérique en Python. Elle fournit des fonctionnalités pour le calcul matriciel, les transformations de Fourier, les

2.10. Sklearn.neighbors.KNeighborsClassifier :

Cette bibliothèque fait partie de Scikit-learn, une bibliothèque pour l'apprentissage automatique en Python. KNeighborsClassifier est une implémentation de l'algorithme de

classification par les k plus proches voisins. Il est utilisé pour la classification de données en groupes prédéfinis.

2.11. Biterm

La bibliothèque "biterm" en Python est utilisée pour la modélisation de sujets dans les textes. C'est une implémentation de l'algorithme Biterm Topic Model (BTM), qui est un modèle de sujets particulièrement utile pour les collections de textes courts, comme les tweets ou les titres.

3. Evaluation

Pour évaluer nos résultats de recherche, on va utiliser l'outil TREC-eval :

3.1. TREC-eval :

Trec_eval est un outil dédié à l'évaluation des systèmes de classement, que ce soit pour des documents ou tout autre type d'information organisée par pertinence. L'évaluation se fait en se basant sur deux fichiers spécifiques : le premier appelé "qrels" (query relevance) énumère les jugements de pertinence pour chaque requête effectuée. Le deuxième fichier contient le classement des documents fourni par notre système de Recherche d'Information (RI).

./trec_eval [fichier_qrels] [fichier_resultats]

- Trec_eval : c'est le nom du programme exécutable.
- fichier_qrels : chemin du fichier qrel
- fichier_resultats: chemin du fichier résultat de requête

La figure suivante représente les résultats de TREC-eval pour la requête 34 :

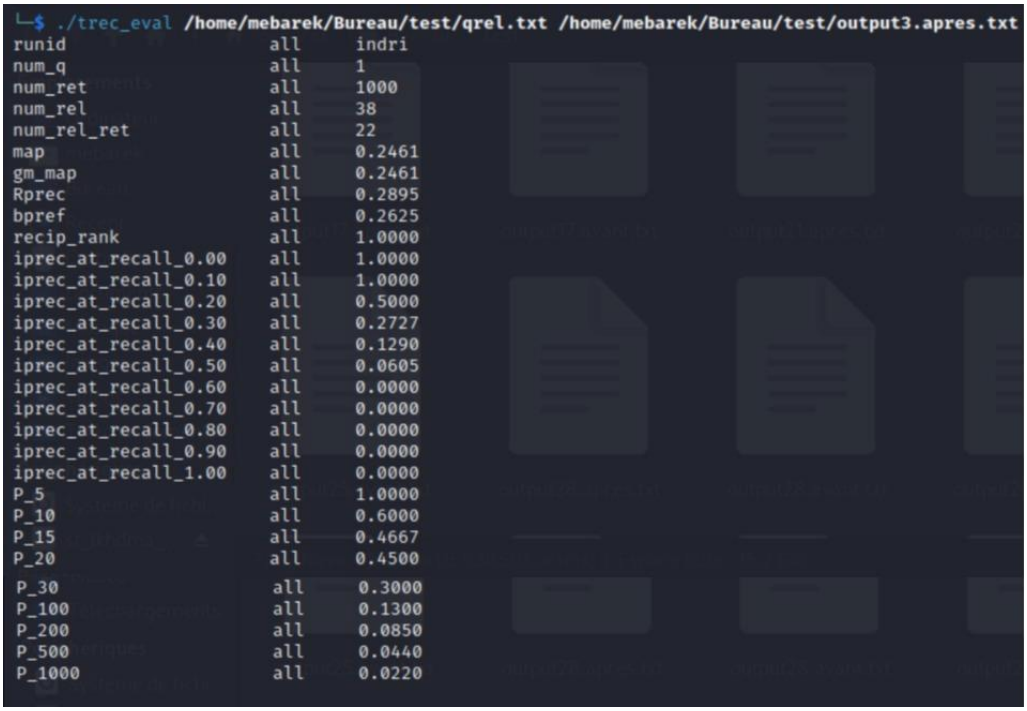


FIGURE 4. 1:LES RESULTATS DE TREC-EVAL POUR LA REQUETE 34

3.2.Les fichiers qrel_file

Ce fichier renferme un répertoire de documents jugés pertinents pour chaque requête. Ce jugement de pertinence est réalisé manuellement par des individus qui sélectionnent les documents qui doivent être retrouvés lorsqu'une requête spécifique est exécutée. Ce fichier peut être perçu comme la "solution idéale", et les documents obtenus par votre système de Recherche d'Information (RI) devraient s'en approcher autant que possible.

Le format de ce fichier est le suivant :

Query-id 0 document-id pertinence

Le champ query-id est une séquence alphanumérique qui sert à identifier la requête. Le champ id-document est également une séquence alphanumérique, mais elle est utilisée pour identifier le document évalué. En ce qui concerne le champ "pertinence", il s'agit d'un chiffre qui indique le niveau de pertinence du document par rapport à la requête : 0 signifie non pertinent, tandis que 1 indique la pertinence et 2 signifie plus pertinent. Le second champ "0"

n'est actuellement pas utilisé, mais il doit être inclus dans le fichier pour des raisons de format. Ces différents champs peuvent être séparés par un espace ou une tabulation.

La forme de fichier qrel représente dans la figure suivante :

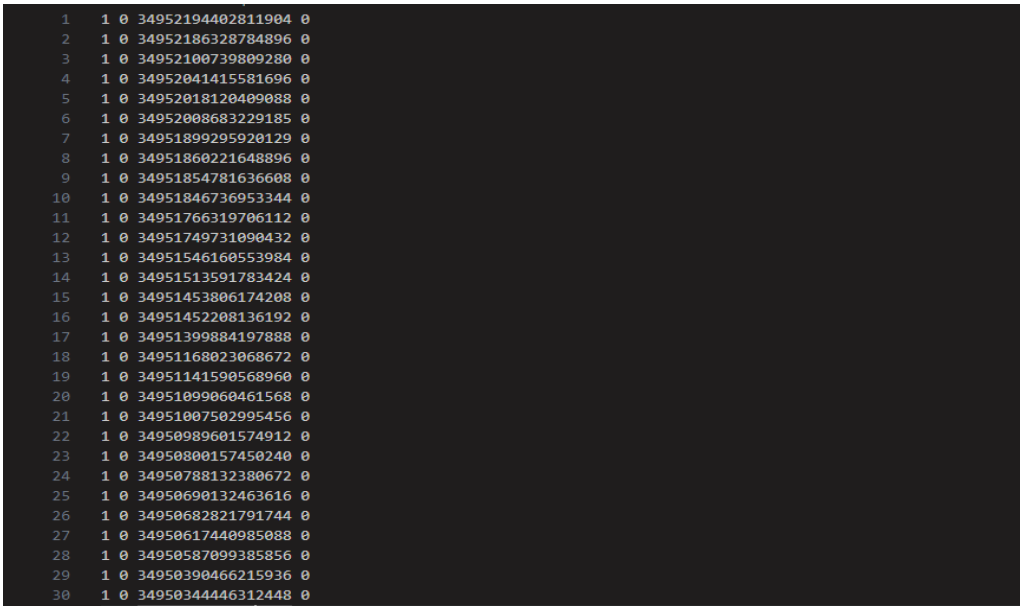


FIGURE 4. 2:LE FICHIER Q_REL

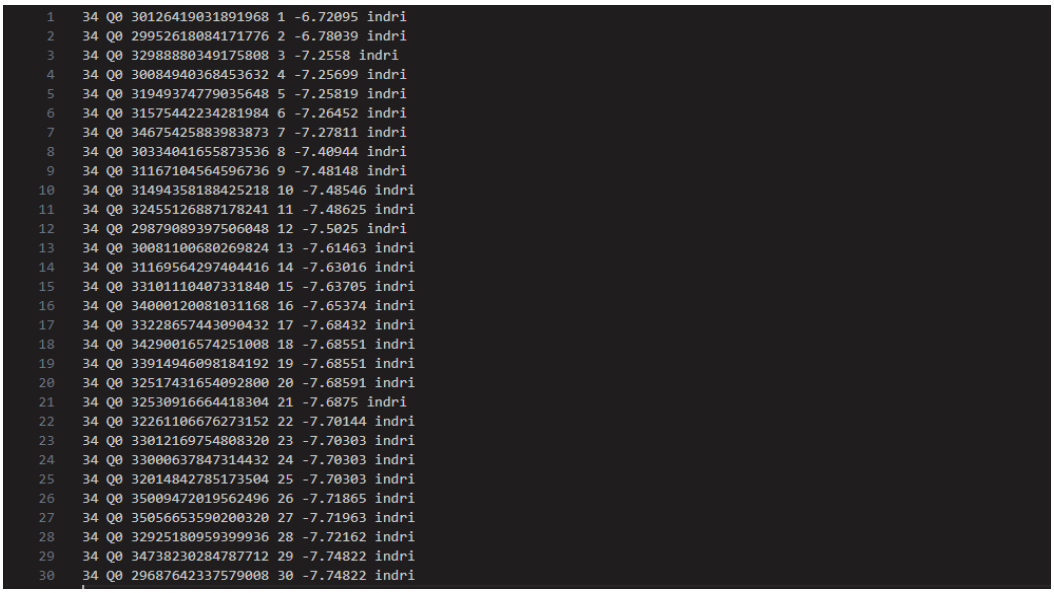
3.3. Les fichiers resultas_file :

Le fichier des résultats contient un classement des documents pour chaque requête automatiquement générée par notre application. C'est le fichier qui sera évalué par trec_eval en fonction de la « réponse correcte » fournie par le premier fichier. Ce fichier a le format suivant :

Query-id Q0 document-id classement score STANDARD

Le champ "query-id" est une séquence alphanumérique destinée à identifier la requête. Le deuxième champ, portant la valeur "Q0", est actuellement omis par trec_eval, mais doit être inclus dans le fichier pour des raisons de format. Le champ "document-id" est une autre séquence alphanumérique utilisée pour identifier le document récupéré. Le champ "classement" est une valeur numérique entière représentant la position du document dans le classement. Toutefois, ce champ est également ignoré par trec_eval. Le "score" peut être une valeur entière ou décimale et indique le niveau de correspondance entre le document et la requête, de sorte que les documents les plus pertinents obtiennent les scores les plus élevés. Enfin, le dernier champ, portant la valeur "STANDARD", est uniquement utilisé pour identifier cette instance particulière d'exécution (ex : nom de moteur de recherche).

La forme de fichier résultat représente dans la figure suivante :



```
34 Q0 30126419031891968 1 -6.72095 indri
34 Q0 29952618084171776 2 -6.78039 indri
34 Q0 32988880349175808 3 -7.2558 indri
34 Q0 30084940368453632 4 -7.25699 indri
34 Q0 31949374779035648 5 -7.25819 indri
34 Q0 31575442234281984 6 -7.26452 indri
34 Q0 34675425883983873 7 -7.27811 indri
34 Q0 30334041655873536 8 -7.40944 indri
34 Q0 31167104564596736 9 -7.48148 indri
34 Q0 31494358188425218 10 -7.48546 indri
34 Q0 32455126887178241 11 -7.48625 indri
34 Q0 29879089397506048 12 -7.5025 indri
34 Q0 30081100680269824 13 -7.61463 indri
34 Q0 31169564297404416 14 -7.63016 indri
34 Q0 33101110407331840 15 -7.63705 indri
34 Q0 34000120081031168 16 -7.65374 indri
34 Q0 33228657443090432 17 -7.68432 indri
34 Q0 34290016574251008 18 -7.68551 indri
34 Q0 33914946098184192 19 -7.68551 indri
34 Q0 32517431654092800 20 -7.68591 indri
34 Q0 32530916664418304 21 -7.6875 indri
34 Q0 32261106676273152 22 -7.70144 indri
34 Q0 33012169754808320 23 -7.70303 indri
34 Q0 33000637847314432 24 -7.70303 indri
34 Q0 32014842785173504 25 -7.70303 indri
34 Q0 35009472019562496 26 -7.71865 indri
34 Q0 35056653590200320 27 -7.71963 indri
34 Q0 32925180959399936 28 -7.72162 indri
34 Q0 34738230284787712 29 -7.74822 indri
34 Q0 29687642337579008 30 -7.74822 indri
```

FIGURE 4. 3:LE FICHIER RESULTAT DE LA REQUETE 34

3.4. Mesures d'évaluation

Pour évaluer notre modèle définis précédemment, nous avons opté à utiliser les mesures d'évaluation standards, à savoir :

- Le MAP
- La R-Précision
- La précision@X

Nous avons aussi utilisé les mesures de base Rappel, précision.

3.5. Résultat de l'évaluation

3.5.1. Evaluation de la recherche pour la requête 1 (évenement)

Pour évaluer notre approche d'expansion de requêtes, nous avons utilisé les termes des meilleurs topics de chaque burst. Il est noté que chaque topic généré de nombreux termes, mais pour nos tests, nous avons choisi de nous concentrer sur les cinq termes les plus fréquents. Nous avons mené une série d'expérimentations sur plusieurs requêtes, nous présentons ici l'exemple de la requête1 de type événement, dont le texte d'origine est "BBC World Service staff cuts".

Nous avons progressivement augmenté le nombre de termes utilisés pour l'expansion de la requête, en commençant par un seul terme, deux et trois termes. Les termes sélectionnés sont (news king doubt ball cours news egypt staf live dead servic sport world british intern).

Nous avons élargi la requête1 via un seul terme du meilleur topic de chaque burst, pour obtenir "BBC World Service staff cuts news egypt british".

Les résultats de cette comparaison sont présentés dans les deux tableaux (4.1 et 4.2) :

- **Les mesures : R-Précison, MAP**

	MAP	**R-Précision**
INDRI Baseline	**0.1214**	**0.2239**
1 terme d'expansion	**0.1546**	**0.2290**
2 termes d'expansion	**0.0927**	**0.2090**
3 termes d'expansion	**0.326**	**0.0749**

TABLEAU 4. 1 : R-PRECISION, MAP DE LA REQUETE 1

FIGURE 4. 4:HISTOGRAMME R-PRECISONS, MAP DE LA REQUETE

- **La précision @X**

	P@5	**P@10**	**P@30**
INDRI Baseline	**0.40**	**0.40**	**0.16**
1 terme d'expansion	**0.80**	**0.60**	**0.30**
2 termes d'expansion	**0.20**	**0.30**	**0.13**
3 termes d'expansion	**0.20**	**0.10**	**0.13**

TABLEAU 4. 2 : PRECISION DE LA REQUETE 1

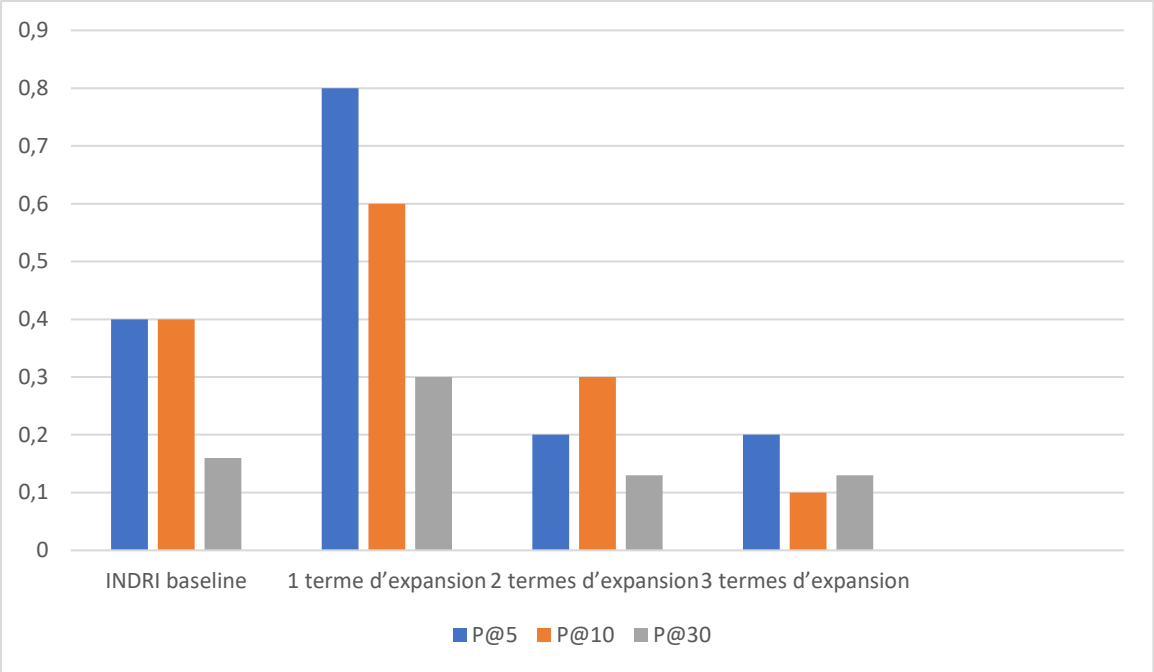


FIGURE 4. 5:HISTOGRAMME DE PRECISION DE LA REQUETE 1

D'après les résultats obtenus dans (tableau 4.1 et 4.2), nous pouvons observer que l'expansion de la requête avec un seul terme a permis d'obtenir les meilleurs résultats en termes de précision aux rangs 5 et 10, ainsi qu'en précision moyenne (MAP) et R-précision. En augmentons le nombre de termes d'expansion, les performances ont tendance à diminuer. Dans notre cas, l'expansion de la requête avec un seul terme du meilleur topic de chaque burst semble être l'approche la plus efficace.

3.5.2. Evaluation de la recherche pour la requête 30 (récent)

Dans le cadre de l'évaluation de notre système, nous présentons par la suite une étude de cas sur la requête 30, intitulée " Keith Olbermann new job ». Nous avons détecté un seul burst significatif qui est survenu un jour avant la date de la requête. Son intervalle temporel est présenté dans le tableau ci-dessus.

burst	Date début	Date fin
Burst1	06/02/2011	07/02/2011

Le texte de la requête 30 est " Keith Olbermann new job". Après l'application de notre approche, nous avons obtenu les termes d'expansion suivant « urban super » ".La requête expansé est " Keith Olbermann new job urban super ".

Les résultats de la précision et du Mean Average Précision (MAP) pour la requête initiale et la requête enrichie sont présentés dans le tableau suivant.

	p@5	P@10	P@30	MAP
Indri	0.800	0.500	0.5657	0.3068
Notre approche	0.800	0.500	0.6657	0.3284

TABLEAU 4. 3 : PRECISON, MAP DE LA REQUETE 30

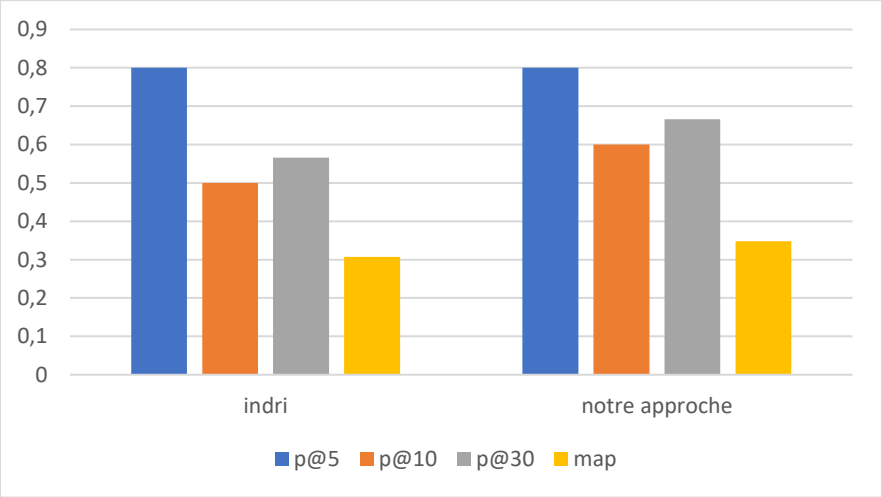


FIGURE 4. 6:HISTOGRAMME LES MESURES : PRECISONS, MAP DE LA REQUETE 30

En observant (le tableau 4.3), nous remarquons que la précision après l'expansion n'a pas connu une amélioration que dans p@30. Alors que l'amélioration dans le MAP est significative.

3.5.3. Evaluation de la recherche pour 5 requetés

Dans le but de valider notre approche et d'évaluer sa performance, nous avons comparé nos résultats avec ceux d'Indri et Lucene. Ces tests ont été réalisés en utilisons un ensemble de cinq requêtes choisi aléatoirement. L'histogramme de (la figure 4.8) montre la différence en précision entre les trois systèmes tandis que l'histogramme de (la figure 4.7) illustre la différence en MAP.

Les résultats de cette comparaison sont présentés dans le tableau ci-dessous :

	Requête 1		Requête 3		Requête 7		Requête 17		Requête 42	
	MAP	P@30	MAP	P@30	MAP	P@30	MAP	P@30	MAP	P@30
Indri baseline	0.1214	0.1667	0.053	0.0000	0.2443	0.833	0.179	0.266	0.1497	0.3333
Lucene baseline	0.0401	0.0000	0.0781	0.0661	0.0384	0.1000	0.2801	0.3333	0.0314	0.0333
Notre approche	0.1564	0.4500	0.2461	0.0000	0.2879	0.833	0.1670	0.266	0.2527	0.3667

TABLEAU 4. 4 : RESULTAT D'EVALUATION DE 5 REQUETES

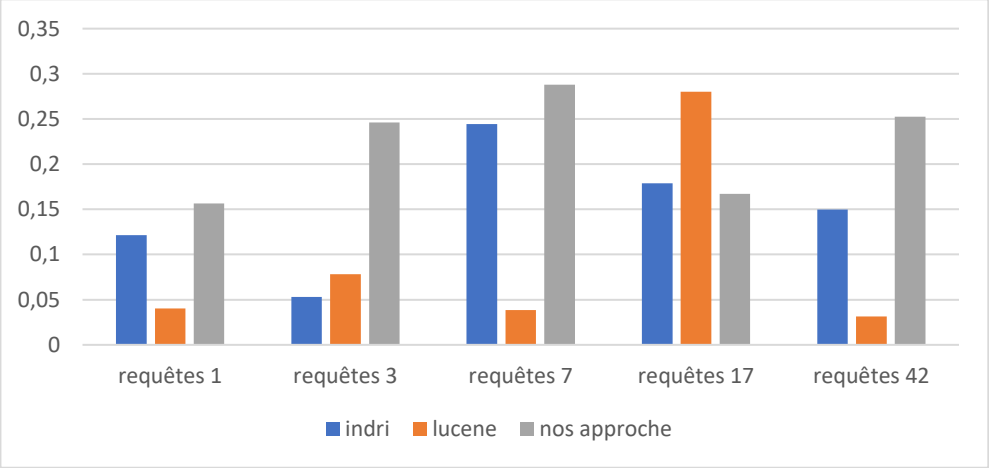


FIGURE 4. 7:LES RESULTATS DE MAP

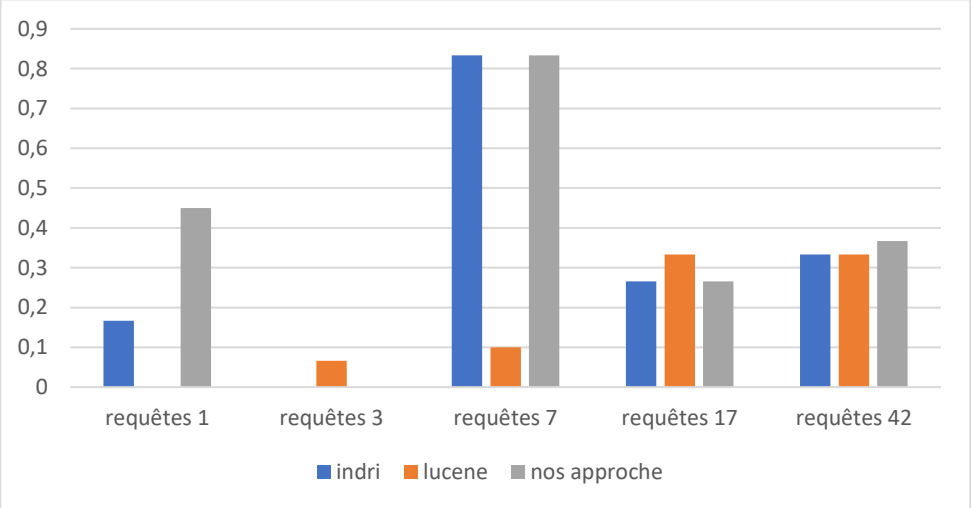


FIGURE 4. 8:LES RESULTATS DE P@30

Ces résultats indiquent que notre méthode présente une performance supérieure par rapport aux méthodes classique (fondues sur un modèle de recherche d'information de base). Cela confirme l'efficacité de notre approche pour améliorer la sélection des tweets pertinents.

La comparaison avec Lucene et Indri est particulièrement importante, car ces deux outils sont largement utilisés dans le domaine de la recherche d'informations.

3.5.4. Evaluation de la recherche pour le système.

Le (tableau 4.5) met en évidence les résultats découlant de nos expérimentations. Nous avons appliqué notre méthode axée sur l'expansion de termes à différents niveaux du système.

Nous avons comparé nos résultats avec ceux d'[Efron,2014], [Chen,2018], [Willis,2012] qui sont des références majeures dans ce domaine, afin de valider l'efficacité de notre approche.

Les résultats sont mentionnés dans le tableau (4.5) :

Mesure d'évaluation	MAP	P@30
INDRI Baseline	**0.1280**	**0.2381**
LUCENE Baseline	**0.1413**	**0.1007**
[efron,2014]	**0.3618**	**0.2457**
[willis ,2012]	**0.4231**	**0.3833**
[HEMIS.2023]	**0.1904**	**0.3987**
[Chen,2018]	**0.154**	**0.154**
Notre approche	**0.1994**	**0.3167**

TABLEAU 4. 5 : RESULTAT D'EVALUATION DE SYSTEME

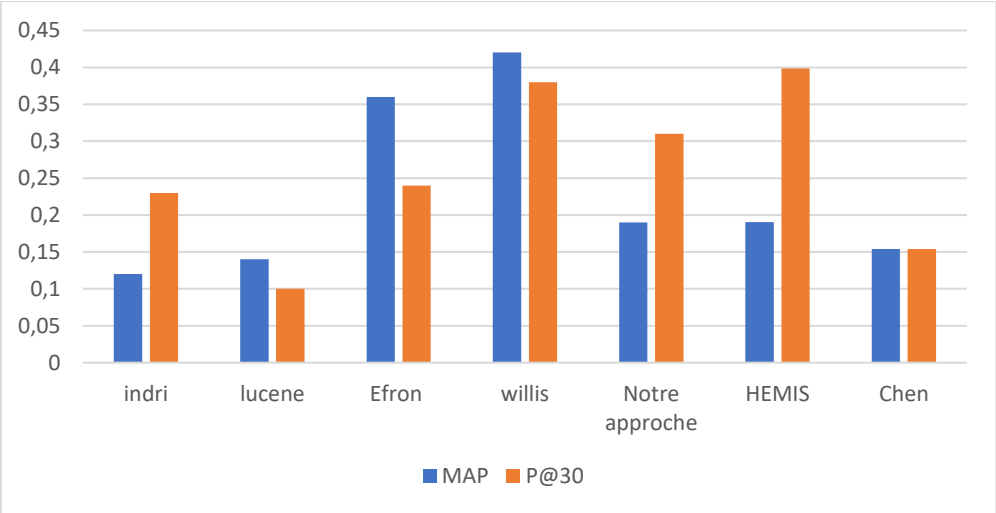


FIGURE 4. 9:HISTOGRAMME DE RESULTAT D'EVALUATION DE SYSTEME

D'après les résultats présentés, on peut voir que notre approche a surperformé les deux baselines (Indri, Lucerne), [Chen, 2018] et [Willis, 2012] en termes de P@30, indiquant que notre approche est plus précise lorsqu'il s'agit de récupérer les 30 premiers documents pertinents. Cependant, en ce qui concerne le MAP, notre approche a surpassé la baseline Indri, baseline Lucene, [Willis, 2012] et [Hemis, 2023] mais elle est restée derrière [Efron, 2014] et [Chen,2018].

Cela signifie que bien que notre approche ait montré une bonne performance en récupérant les documents les plus pertinents, elle pourrait encore être améliorée pour la récupération de tous les documents pertinents dans l'ensemble du corpus.

Ces résultats soulignent l'efficacité relative de notre approche par rapport aux méthodes existantes. Cependant, ils indiquent aussi qu'il y a encore de la place pour l'amélioration, en particulier en termes de récupération de tous les documents pertinents.

Conclusion

Nous avons évalué dans ce chapitre le facteur de pertinence expansion de la requête par l’usage des aspects temporel et thématique. Nous avons montré expérimentalement l’amélioration en précisons et MAP après l’expansion de la requête. Nous avons comparé nos résultats avec ceux des travails connexes citer dans notre état de l’art. on terme de précision p@30 nous avons surpassé le baseline Indri et Lucene, [Willis,2012] et [Chen,2018]. Et ce concerne le MAP on a restée derrière [Efron, 2014] et [Chen,2018].

Conclusion générale :

Nous nous sommes concentrés sur l'amélioration de la recherche d'informations ad hoc (IR) dans les microblogs, dans le but d'identifier les microblogs qui répondent aux besoins d'information d'un utilisateur. Par l'usage des aspects temporel et thématique, via notre hypothèse : «les tweets pertinents pour une requête se regroupent ensemble dans des intervalles temporels comme ils partagent le même vocabulaire». Pour mener nos expériences, nous avons utilisé l'ensemble de données fourni par la tâche Microblog 2011 de TREC.

Dans ce travail, nous avons passé en revue de manière approfondie les systèmes IR de pointe existant dans les microblogs. Notre objectif était de combler les lacunes des approches précédentes et d'améliorer le processus de recherche d'informations dans les microblogs. La nature concise des microblogs a entraîné des correspondances limitées entre le contenu du microblog et les termes de la requête, même lorsqu'ils étaient sémantiquement similaires ce qui a faussé les résultats de la recherche. Pour améliorer le classement des tweets pertinents dans la liste des résultats, nous avons proposé une méthode de reclassement qui se base sur l'expansion de la requête via l'usage des aspects temporel et thématique. Le travail présenté s'articule dans les points suivants : téléchargement du corpus de test, prétraitement du corpus de test, la détection du type temporel de la requête, utilisé le modèle de topic pour grouper les tweets dans des topics, utiliser la densité du kernel et les algorithmes du clustering pour pondérer les topics, la sélection des meilleurs topics dans les trois grands bursts, utiliser les terms les plus classés du meilleur topic dans chaque burst pour élargir la requête.

Le travail présenté ouvre plusieurs perspectives de recherche pour une exploration future. Il sera intéressant de :

- Enrichissez chaque tweet non seulement avec le contenu de la balise titre et description des pages web liée, mais également avec le contenu de la balise texte
- Dans l'objectif d'améliorer l'appariement tweet/requête et pour une meilleure contextualisation de ces derniers. Nous proposons l'enrichissement du contenu sémantique de la requête et des tweets par l'utilisation du Word Embedding.
- Enrichir notre collection du test par l'usage du corpus de la conférence TREC 2012.

Annexe

1. Créer un compte Twitter :

a. **Accédez à la page d'accueil de Twitter.** Ouvrez un navigateur web et rendez-vous sur le site https://www.twitter.com

b. **Cliquez sur "S'inscrire"**. Vous trouverez ce bouton sur la page d'accueil.

c. **Entrez vos informations**. On vous demandera de fournir votre nom et votre numéro de téléphone ou votre adresse e-mail. Il est également possible de vous inscrire avec un compte Google ou Apple.

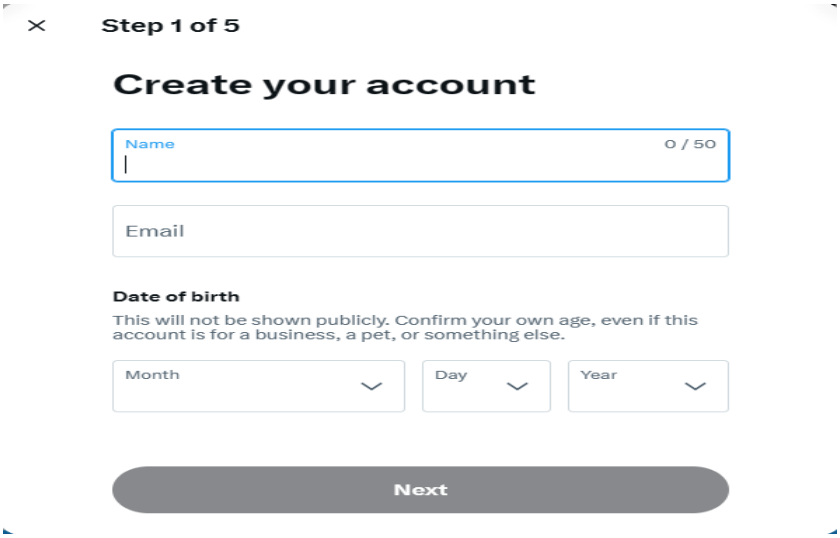


FIGURE 2. 1: FORMULAIRE D'INSCRIPTION TWEETER

d. **Créez un mot de passe.** Assurez-vous que votre mot de passe est unique et sécurisé.

e. **Choisissez un nom d'utilisateur.** Ce sera votre identifiant unique sur Twitter. Il doit comporter moins de 15 caractères. Si le nom d'utilisateur que vous souhaitez est déjà pris, Twitter vous suggérera des alternatives.

f. **Acceptez les conditions d'utilisation.** Lisez et acceptez les conditions d'utilisation de Twitter et la politique de confidentialité.

g. **Vérifiez votre compte.** Twitter vous enverra un code par e-mail ou par SMS, selon l'option que vous avez choisie. Entrez ce code dans le champ prévu à cet effet pour vérifier votre compte.

h. **Choisissez vos centres d'intérêt**. Twitter vous suggère des comptes à suivre basés sur vos centres d'intérêt.

i. **Complétez votre profil**. Ajoutez une photo de profil, une bio et d'autres détails si vous le souhaitez.

j. **Commencez à tweeter !** Vous pouvez maintenant commencer à publier des tweets, à suivre d'autres utilisateurs, et à interagir avec le contenu sur Twitter.

2. Le vocabulaire spécifique de Twitter :

Twitter a introduit un ensemble unique de termes et de jargon qui sont spécifiques à la plateforme. Voici quelques-uns des termes les plus couramment utilisés :

b. **Tweet :** Un message posté sur Twitter. Il peut contenir jusqu'à 280 caractères, ainsi que des images, des vidéos, et des liens.

c. **Retweet (RT) :** Un moyen de partager le Tweet d'un autre utilisateur avec vos propres abonnés. Vous pouvez simplement retweeter le message original, ou vous pouvez ajouter vos propres commentaires en utilisant la fonction "Retweet avec commentaire".

d. **Follower :** Un utilisateur de Twitter qui s'est abonné à votre compte afin de voir vos tweets dans son flux.

e. **Following :** Les comptes que vous avez choisis de suivre sur Twitter. Leurs tweets apparition dans voter flux.

f. **Mention (@) :** Utilisé pour désigner un autre utilisateur dans un tweet, par exemple "@nom_utilisateur". L'utilisateur mentionné reserve une notification.

g. **Hashtag (#) :** Un mot ou une phrase précédée d'un symbole dièse (#). Les hashtags sont utilisés pour regrouper les tweets autour d'un même sujet.

h. **Trending Topic :** Sujet populaire ou tendance sur Twitter à un moment donné. Les sujets tendances peuvent être personnalisés en fonction de votre localisation et de qui vous suivez.

i. **Direct Message (DM) :** Un moyen de communiquer en privé avec un autre utilisateur de Twitter.

j. **Handle :** Votre identifiant unique sur Twitter, précédé d'un @. Par exemple, "@nom_utilisateur".

k. **Timeline :** Le flux de tweets que vous voyez quand vous vous connectez à Twitter. Il est composé des tweets des personnes que vous suivez, ainsi que de tout contenu que Twitter pense que vous pourriez trouver intéressant.

Bibliographie

[Azzoug .2013] W. Azzoug: Contribution à la définition d'une approche d'indexation sémantique de documents textuels. Mémoire de Magister, Université Boumerdes, 2013.

[Belkin,1992] J.N Belkin, P. Ingwersen, A.M. « Proceedings of the 15 th annual International ACM SIGIR, In Conference on Research and Development in Information Retrieval». Copenhagen, Denmark, June, pages 21-24, ACM 1992.

[Blei,2007] D. M. Blei and J. D. Lafferty, "A correlated topic model of science," Annals of Applied Statistics, vol. 1, no. 1, pp. 17–35, 2007.

[Chen, 2018] Chen, Q., Hu, Q., Huang, J. X., & He, L. (2018). Taker: Fine-grained time-aware microblog search with kernel density estimation. *IEEE Transactions on Knowledge and Data Engineering*, *30*(8), 1602-1615.

[Damak et al, 2013] Damak, F., Pinel-Sauvagnat, K., Cabanac, G., et Boughanem, M. (2013). Effectiveness of State-of-the-art Features for Microblog Search. In SAC'13 : ACM Symposium on Applied Computing. ACM.

[Damak et al., 2011]. Damak, F., Jabeur, L. B., Cabanac, G., Pinel-Sauvagnat, K., Lechani, L., et Boughanem, M. (2011). IRIT at TREC Microblog 2011. In E. M. Voorhees et (Eds.), Text Retrieval Conference (TREC), Gaithersburg, USA,. National Institute of Standards and Technology (NIST).

[Damak, 2014] F. Damak.2014 "Etude des facteurs de pertinence dans la recherche de microblogss". Thèse de doctorat on informatique l'Université Toulouse 3 paul sabatier (UT3 Paul Sabatier).

[Diallo et al,2022] Diallo, A., Camara, G., Camara, F., & Mboup, A. (2022). Artificial Intelligence approaches to predict COVID-19 infection in Senegal. *Procedia Computer Science*, *201*, 764-770.

[Efron et al ,2014] Efron, M., Lin, J., He, J., & De Vries, A. (2014, July). Temporal feedback for tweet search with non-parametric density estimation. In *Proceedings of the 37th international ACM SIGIR conference on Research & development in information retrieval* (pp. 33-42).

[Efron,2011] Efron, M, et Golovchinsky, G. (2011). Estimation methods for ranking recent information. In Proceedings of the 34th international acm sigir conference on research and development in information retrieval (pp. 495–504). New York, NY, USA: ACM.

[Feng et Wang, 2013] Feng, W., et Wang, J. (2013). Retweet or not? Personalized tweet re-ranking. In Proceedings of the sixth acm international conference on web search and data mining (pp. 577–586). New York, NY, USA : ACM

[Ferguson et al., 2012] Ferguson, P., O'Hare, N., Lanagan, J., Phelan, O., et McCarthy, K. (2012). An investigation of term weighting approaches for microblog retrieval. In Proceedings of the 34th European conference on advances in information retrieva (pp. 552–555). Berlin, Heidelberg : Springer-Verlag.

[Furnas et al. 1987] Furnas, G. W., Landauer, T. K., Gomez, L. M., et Dumais, S. T. (1987). The vocabulary problem in human-system communication. Commun. ACM, 30 (11), 964–971.

[Hammache.2013] Hammache A : Recherche d'Information : un modèle de langue combinant mots simples et mots composés. Thèse doctorat, UMMTO, 2013.

[HEMIS ,2023] Hemis, H. (2023). Un nouveau modèle social de recherche d'information dans les microblogs. Université Saad Dahleb de Blida 1, Blida, Algérie.

[Jabeur et al, 2012] abeur, L., Tamine, L., et Boughanem, M. (2012). Featured tweet search: Modelingtime and social influence for microblog retrieval. In IEEE/WIC/ACM International Conference on Web Intelligence, Macau, China (pp. 166–173). IEEE Computer Society - Conference Publishing Services.

[Jardine and Rijsbergen, 1971] Jardine, N., & van Rijsbergen, C. J. (1971). The use of hierarchic clustering in information retrieval. Information storage and retrieval, 7(5), 217-240.

[Jian-Yun,2001] Nie, J.-Y., & Simard, M. (2001). Study of Query Expansion Techniques in the Context of the TREC. Proceedings of the Text Retrieval Conference.

[Jinxi Xu and W. Bruce Croft, 1996] Xu, J., & Croft, W. B. (1996). Query expansion using local and global document analysis. In Proceedings of the 19th annual international ACM SIGIR conference on Research and development in information retrieval (pp. 4-11)

[Joachims, 2005] Joachims, T. (2005). A support vector method for multivariate performance measures. In Proceedings of the 22nd international conference on machine learning (pp. 377–384). New York, NY, USA: ACM.

[Kwak et al.2010] Kwak, H., Lee, C., Park, H., et Moon, S. (2010). What is twitter, a social network or a news media ? In Proceedings of the 19th international conference on world wide web (pp. 591–600). New York, NY, USA: ACM.

[Lin et al. 2012] Lin, Y., Li, Y., Xu, W., et Guo, J. (2012). Microblog retrieval based on termsimilarity graph. In Computer science and network technology (iccsnt), 2012 2nd international conference on (p. 1322-1325).

[Rijsbergen, 1979] C.J. Van Rijsbergen «Information Retrieval », Butterworth-Heinemann, Newton, MA, USA, 2nd edition, 1979.

[Salton ,1975] G Salton, A Wong, CS Yang «A vector space model for automatic indexing», Communications of the ACM vol 18, n°11, p 613-620. November 1975.
[silverman,1996] B. W. Silverman. Density Estimation for Statistics and Data Analysis. Monographs on Statistics and Applied Probability. Chapman & Hall, Boca Raton, 1996.

[Teevan et al. 2011] Teevan, J., Ramage, D., et Morris, M. R. (2011). #twittersearch : a comparison of microblog search and web search. In Wsdm'11: Proceedings of the fourth acm international conference on web search and data mining (pp. 35–44). New York, NY, USA: ACM.

[Voorhees, 2006] E. M. (2006). Overview of the trec 2006. In TREC'06: 6th Text Retrieval Conference.

[Yan et al,2013] Yan, X., Guo, J., Lan, Y., & Cheng, X. (2013, May). A biterm topic model for short texts. In *Proceedings of the 22nd international conference on World Wide Web* (pp. 1445-1456).

[Zhao et al, 2011] Zhao, L., Zeng, Y., et Zhong, N. (2011). A weighted multi-factor algorithm for microblog search. In Proceedings of the 7th international conference on active media technology (pp. 153–161). Berlin, Heidelberg : Springer-Verlag.

Webographie

[1] https://www.proinfluent.com/nombre-utilisateurs-twitter

[2] https://www.blogdumoderateur.com/chiffres-reseaux-sociaux/

[3] https://python.sdv.univ-paris-diderot.fr/01_introduction/

[4] https://visualstudio.microsoft.com/fr/

[5] https://docs.python.org/fr/3/library/json.html

[6] https://datascientest.com/matplotlib-tout-savoir

[7] https://nltk.org/

[8] http://pandas.pydata.org/