N° d'ordre : 16/2023 - D/In

République Algérienne Démocratique et Populaire
Ministère de L'Enseignement Supérieur et de la Recherche Scientifique
Université des Sciences et de la Technologie HOUARI BOUMEDIENE

Faculté d'Informatique

THÈSE DE DOCTORAT EN SCIENCES
Présentée pour l'obtention du **grade** de **DOCTEUR**
En : Informatique
Spécialité : Informatique

Par : **LEBIB Fatma Zohra**

Thème :

Une approche optimale basée Social Computing pour la sélection de sources d'information dans un environnement de recherche multi-sources

Soutenue publiquement, le 18/06/2023, devant le jury composé de :

M. BOUKHALFA Kamel	Professeur	à l'USTHB	Président
M. MEZIANE Abdelkrim	Directeur de Recherche	au CERIST	Directeur de thèse
Mme. BOUGHACI Dalila	Professeur	à l'USTHB	Examinateur
M. KECHID Samir	Professeur	à l'USTHB	Examinateur
M. NOUALI Omar	Directeur de Recherche	au CERIST	Examinateur
M. BOULIF Menouar	Professeur	à l'UMBB	Examinateur

Remerciements

Je souhaite ici rendre hommage et exprimer ma profonde gratitude à tous ceux qui, de près ou de loin, ont contribué à la réalisation de cette thèse et à son aboutissement.

Mes remerciements vont tout d'abord à mon directeur de thèse, le directeur de recherche Monsieur MEZIANE Abdelkrim qui m'a guidé tout au long de ce parcours. Je tiens à le remercier pour ses conseils avisés et ses encouragements.

Mes remerciements s'adressent aussi au Professeur BOUKHALFA Kamel, Professeur à l'USTHB, pour l'honneur qu'il me fait en président ce jury.

Je tiens également à témoigner toute ma reconnaissance au Professeur BOUGHACI Dalila, Professeur KECHID Samir, Directeur de recherche NOUALI Omar et Professeur BOULIF Menouar pour l'honneur qu'ils m'ont fait en acceptant d'évaluer ce travail de recherche et de faire partie du jury. Je les remercie également pour leur précieuses recommandations et leurs remarques constructives.

Je tiens également à remercier mes collègues pour leur soutien au quotidien.

Enfin je renouvelle toute mon amitié et ma sympathie à ceux qui m'ont accordé du temps et m'ont témoigné un soutien constant dans ce long travail de recherche.

Merci pour tout...

LEBIB F.Z.

Louange à Dieu tout puissant, qui m'a permis de voir ce jour tant attendu.

Dédicaces

Avec tous mes sentiments de respect, d'amour et de profonde gratitude, je dédie cette thèse :

À l'âme de mon père et de ma mère pour leur affection et leur amour inépuisables. Que Dieu vous accorde la paix éternelle et vous accueille dans son vaste paradis.

À mon époux et à mes chers enfants MOHAMED EL AMINE, SEIF EDDINE, HOCINE & HASNA. Puisse Dieu vous préserve et vous procure santé et bonheur.

À mes chers frères et sœurs et leurs enfants. Que Dieu Tout Puissant vous protège et vous accorde santé, bonheur et longue vie.

À toute ma famille et toutes les personnes que j'aime.

Fatma Zohra

Les seules choses qui sont impossibles à finir sont celles que l'on ne commence pas.

Lynn Johnston

Abstract

The aim of a multi-source retrieval system (also called a distributed information retrieval system) is to retrieve documents from a set of distributed sources through a centralized broker. The multi-source retrieval system encompasses a body of research investigating solutions for searching online content that cannot be discovered using standard Web crawling techniques and is often referred to as the "deep Web" or "hidden Web". To enable retrieval, the broker maintains a representative description of the content held in each source. A critical issue in multi-source search is source selection. Because a multi-source retrieval system may consist of a large number of sources, the only way to ensure timely and economic retrieval is to search a small number of sources which are likely to contain relevant documents for a query. Source selection is also critical for information retrieval accuracy (relevance- efficiency- effectiveness) for the simple reason that searching the wrong sources containing few or no relevant documents will result in retrieval failure for a query.

There are a plethora of information sources available on the Internet in recent years, what makes that the conventional information retrieval methods cannot meet the current needs of users with different profiles. The same query results are not satisfactory for all users with different interests, therefore, personalized query results are needed. This is the well-known problem of personalized search.

On the other hand, social networks are now the network of users' interests and their relationships with each other. Information retrieval should consider this social community in the information retrieval process to satisfy different users with personalized results.

In multi-source information retrieval, there is a whole body of literature on source selection problem, various solutions have been proposed which use different models and techniques. In this thesis, we address the problem of source selection from the perspective of personalization. The objective is to satisfy the user with results close to his interests when this user is included in social relationships with other users and when the search environment includes a wide number of information sources. At first, we formulized the problem of sources selection as a combinatorial optimization problem, which invoves finding the near-optimal combination (a selection of sources) in a prohibitive search space containing a huge number of possible solutions (combinations). This problem is solved by the use of an intelligent approach, in particular the genetic algorithm. Then, we proposed to exploit social data to improve the intelligent approach proposed for source selection. The improvement of selection accuracy is assured based on the user's track through the use of sources, to say that source description is enriched with tags from the tagging history. And to solve the problem of lack of user data, we proposed an approach for analysising a large amount of data contained in the log files of a multi-source retrieval system using data mining techniques. The goal is to capture, model and analyze the behavioral patterns and profiles of users who interact with the system. The main contribution of this thesis is to provide a solution to the source selection problem to personalize the multi-source

search. We have proposed a multidimensional approach based on LDA (Latent Dirichlet Allocation) topic modeling, which considers both the social dimension and the intelligence dimension in order to adapt the source selection to the user's topics of interest. We combine bio-inspired methods based on genetic algorithms and LDA topic modeling methods to find the "optimal" and "personalized" source selection based on a user's profile in social networks. LDA-based topic modeling method is used to analyze user behavior in social networks and discover users' latent topics of interest from a large collection of data from social tagging systems, which is very effective in personalizing search in a multi-source environment.

All the contributions of this thesis are evaluated on real datasets extracted from a multi-source retrieval system and social tagging networks and the results showed the effectiveness of our proposals compared to state-of-the-art approaches using precision-based performance evaluation metrics.

Keywords : source selection, distributed information retrieval, bio-inspired methods, genetic algorithm, social tagging, topic modeling, Latent Dirichlet Allocation, knowledge extraction

Résumé

L'objectif d'un système de recherche multi-sources (également appelé système de recherche d'information distribuée) est de récupérer des documents à partir d'un ensemble de sources distribuées via un courtier centralisé. Le système de recherche multi-sources englobe un ensemble de recherches portant sur des solutions pour rechercher du contenu en ligne qui ne peut pas être découvert à l'aide de techniques d'exploration Web standard et qui est souvent appelé "Web profond" ou "Web caché". Pour permettre la recherche, le courtier conserve une description représentative du contenu présent dans chaque source. Un problème critique dans la recherche multi-sources est la sélection des sources. Étant donné qu'un système de recherche multi-sources peut être constitué d'un grand nombre de sources, la seule façon d'assurer une recherche rapide et économique est de rechercher un petit nombre de sources susceptibles de contenir des documents pertinents pour une requête. La sélection des sources est également essentielle pour la précision de la recherche d'information (pertinence-efficience-efficacité) pour la simple raison que la recherche de sources erronées contenant peu ou pas de documents pertinents entraînera un échec de la recherche pour une requête.

Il existe depuis quelques années une pléthore de sources d'information disponibles sur Internet, ce qui fait que les méthodes classiques de recherche d'information ne peuvent répondre aux besoins actuels des utilisateurs aux profils différents. Les mêmes résultats de requête ne sont pas satisfaisants pour tous les utilisateurs ayant des intérêts différents, par conséquent, des résultats de requête personnalisés sont nécessaires. C'est le problème bien connu de la recherche personnalisée.

D'autre part, les réseaux sociaux sont désormais le réseau d'intérêts des utilisateurs et de leurs relations les uns avec les autres. La recherche d'information doit prendre en compte cette communauté sociale dans le processus de recherche d'information pour satisfaire les différents utilisateurs avec des résultats personnalisés.

Dans la recherche d'information multi-sources, il existe toute une littérature sur le problème de sélection des sources, diverses solutions ont été proposées qui utilisent différents modèles et techniques. Dans cette thèse, nous abordons le problème de la sélection des sources sous l'angle de la personnalisation. L'objectif est de satisfaire l'utilisateur avec des résultats proches de ses intérêts lorsque cet utilisateur est inclus dans des relations sociales avec d'autres utilisateurs et que l'environnement de recherche comprend un grand nombre de sources d'information. Pour commencer, nous avons formulé le problème de la sélection des sources comme un problème d'optimisation combinatoire, qui consiste à trouver la combinaison (une sélection des sources) quasi-optimale dans un espace de recherche prohibitif contenant un grand nombre de solutions possibles (combinaisons). Ce problème est résolu par l'utilisation d'une approche intelligente, notamment l'algorithme génétique. Ensuite, nous avons proposé d'exploiter les données sociales pour améliorer l'approche intelligente proposée pour la sélection des sources. L'amélioration de la précision de la sélection est assurée en fonction de la trace de l'utilisateur lors de l'utilisation de

sources, c'est-à-dire que la description de la source est enrichie de balises issues de l'historique de marquage. Et pour résoudre le problème de manque de données utilisateurs, nous avons proposé une approche d'analyse d'une grande quantité de données contenues dans les fichiers journaux d'un système de recherche multi-sources en utilisant des techniques de fouille de données (Data Mining). L'objectif est de capturer, modéliser et analyser les schémas comportementaux et les profils des utilisateurs qui interagissent avec le système. La principale contribution de cette thèse est de fournir une solution au problème de sélection des sources pour personnaliser la recherche multi-sources. Nous avons proposé une approche multidimensionnelle basée sur la modélisation des sujets LDA (Latent Dirichlet Allocation), qui considère à la fois la dimension sociale et la dimension intelligence afin d'adapter la sélection des sources aux thèmes d'intérêt de l'utilisateur. Nous combinons des méthodes bio-inspirées basées sur des algorithmes génétiques et des méthodes de modélisation des sujets LDA pour trouver la sélection des sources "optimale" et "personnalisée" en fonction du profil d'un utilisateur dans les réseaux sociaux. La méthode de modélisation de sujets basée sur LDA est utilisée pour analyser le comportement des utilisateurs dans les réseaux sociaux et pour découvrir les thèmes d'intérêt (topics of interest) latents des utilisateurs à partir d'une grande collection de données provenant de systèmes de marquage social, ce qui est très efficace pour personnaliser la recherche dans un environnement multi-sources.

Toutes les contributions de cette thèse sont évaluées sur des ensembles de données réels extraits d'un système de recherche multi-sources et de réseaux de marquage social et les résultats ont montré l'efficacité de nos propositions par rapport aux approches de l'état de l'art en utilisant des métriques d'évaluation des performances basées sur la précision.

Mots clés : sélection des sources, recherche d'information distribuée, méthodes bio-inspirées, algorithme génétique, tagging social, modélisation de sujets, allocation latente de Dirichlet, extraction de connaissances

Trouver l'information est un art, pas une science.

Jean-Pierre Lardy

Table des matières

Table des figures

Liste des tableaux

Introduction

1 Contexte général et problématique

La recherche d'information (RI) sur Internet est l'une des activités les plus populaires sur le Web. Plus de 80% des chercheurs sur Internet utilisent des moteurs de recherche pour satisfaire leurs besoins en information [146]. Ces moteurs utilisent des programmes appelés robots (ou araignées) pour indexer les documents de différentes sources d'information, afin de faciliter ultérieurement la recherche de contenu sur ces sources. La croissance intense du volume d'informations disponibles sur le Web rend difficile, voire impossible, leur indexation complète. Il y a aussi une partie importante du web qui est cachée et qui est tout à fait non accessible par les moteurs de recherche traditionnels. Cette partie de web est connue sous le nom du **web caché** (hidden web) [18, 126] ou **Web profond** (Deep Web) [104]. La recherche d'information distribuée [29, 139] ou la recherche d'information multi-sources (RIMS) [1], aussi connue comme la recherche fédérée (Federated Search) [142] offre une solution pour les problèmes cités ci-dessus. Elle offre aux utilisateurs la possibilité de rechercher simultanément plusieurs sources d'information via une interface unifiée.

Dans les systèmes de recherche d'information multi-sources, les sources sont différentes et leur contenu est de type multimodal dans le sens où nous pouvons avoir un serveur de base de données comme une source, un site web comme une autre source, un site intranet, etc. Le processus de recherche d'information multi-sources comprend trois étapes importantes, à savoir la description de la source, la sélection de la source et la fusion des résultats [29, 139]. La première étape consiste à acquérir une description du contenu de chaque source. La description de la source fournit des informations utiles pour les algorithmes de sélection des sources. La deuxième étape est la sélection des sources, son but est de déterminer quelles sources sont les plus susceptibles de contenir des informations pertinentes pour une requête d'utilisateur donnée. L'idée est que l'information pertinente n'est pas nécessairement contenue dans chaque source, ce qui rend inutile et coûteux l'envoi de la requête de l'utilisateur à toutes les sources. En général, le nombre de sources choisies est prédéterminé et est bien inférieur au nombre total de sources. Plusieurs algorithmes de sélection des sources sont proposés dans la littérature, nous citerons les plus connus dans les chapitres suivants. Enfin, dans la troisième étape, le système agrège ou fusionne les résultats renvoyés par les sources sélectionnées en une seule liste unifiée et classée avant de la présenter à l'utilisateur final.

Dans les systèmes classiques de recherche d'information, l'utilisateur est généralement représenté par sa requête seule ; le système répond avec la même liste de résultats pour deux utilisateurs qui ont envoyé la même requête et qui ont pourtant des besoins d'information différents. Les requêtes peuvent être ambiguës, car les utilisateurs peuvent utiliser

1. L'expression "recherche d'information multi-sources" est utilisée dans ce manuscript pour faire référence à la recherche d'information distribuée.

la même requête même s'ils s'attendent à des ressources pertinentes différentes [132].

Récemment, avec l'explosion de la quantité d'informations sur Internet, d'une part, qui a conduit à une grave surcharge d'informations, rendant difficile pour les utilisateurs d'identifier les informations qu'ils recherchent, et avec l'émergence des réseaux sociaux et des relations sociales entre les utilisateurs, d'autre part, ont fait que la recherche traditionnelle ne peut pas satisfaire tous les utilisateurs aux profils différents. Pour faire face au problème de la surcharge d'information et pour mieux se rapprocher des besoins exprimés par l'utilisateur, des systèmes de personnalisation ont été proposés et déployés pour le domaine de la recherche d'information. La recherche d'information personnalisée tend à modéliser l'utilisateur selon un profil (ensemble de préférences et d'intérêts, relations sociales, etc.) puis à l'intégrer dans la chaîne d'accès à l'information. La recherche d'information personnalisée, en tant qu'outil efficace pour résoudre les problèmes ci-dessus, est devenue l'un des services Web les plus importants dans un environnement de réseau moderne [158].

Dans la recherche d'information multi-sources, la personnalisation de la recherche est un enjeu important et une tâche difficile à réaliser. Dans ce travail, nous nous concentrons sur l'étape la plus importante de la recherche multi-sources, qui est la sélection des sources. Nous considérerons des sources différentes et géographiquement distribuées telles que des bases ou des banques de données, des sites de Web invisible, des moteurs de recherche à usage général ou autres. L'objectif est d'adapter la sélection des sources en fonction du profil de l'utilisateur engagé dans une relation sociale. Le choix de la source doit sans doute tenir compte du profil de la source elle-même représenté par des statistiques du contenu de la source (par exemple le nombre de documents que contient la source et la fréquence des documents pour chaque mot) et de la requête de l'utilisateur représentée par une liste de termes ou de mots-clés, qui seront mis en correspondance avec le profil de l'utilisateur ou ses relations sociales.

Nous proposons dans un premier temps d'utiliser des algorithmes génétiques pour trouver la sélection "optimale" des sources à interroger en considérant un espace de recherche très large des solutions potentielles. Ensuite, nous nous concentrons sur l'amélioration des performances et de l'efficacité de l'algorithme génétique proposé en intégrant des données extraites des systèmes de marquage social, ces dernières étant utilisées pour enrichir la description des sources, ce qui améliore la précision et l'efficacité de la sélection des sources. Puis nous abordons le problème de personnalisation de la recherche multi-sources en proposant d'adapter la sélection des sources à interroger en fonction des thèmes d'intérêt de chaque utilisateur. L'objectif est d'améliorer la recherche d'information multi-sources en assurant la satisfaction des utilisateurs en fournissant des résultats plus pertinents et plus proches de leurs intérêts. Les thèmes d'intérêt général pour chaque utilisateur sont générés à partir d'une grande collection de balises (tags) à l'aide de la modélisation de sujets LDA [22]. La sélection personnalisée des sources qui maximise la similarité entre la solution et la requête de l'utilisateur et ses thèmes d'intérêt est ensuite générée par un algorithme génétique. Et enfin nous proposons d'exploiter les fichiers journaux d'un système de recherche multi-sources pour extraire de nouvelles connaissances sur les utilisateurs qui interagissent avec le système. Pour cela, nous proposons une méthode d'analyse des fichiers journaux basée sur un nettoyage en profondeur. Les données analysées sont principalement utilisées pour améliorer l'expérience de recherche des utilisateurs.

2 Contributions de cette thèse

Notre contribution dans le cadre de la recherche d'information multi-sources se situe au niveau de la phase de sélection des sources d'information afin de répondre aux problématiques présentées dans la section précédente. La contribution est répartie selon les quatre points importants suivants :

1. **Approche intelligente pour la sélection des sources** [93] :
   Nous considérons la recherche d'information à grande échelle où un très grand nombre de sources d'information sont disponibles. Nous avons modélisé le problème de sélection des sources comme un problème d'optimisation combinatoire, qui consiste à trouver la combinaison quasi-optimale parmi plusieurs solutions possibles. La combinaison représente "une sélection", composée d'un nombre prédéfini de sources d'information. Pour résoudre ce problème, nous avons proposé d'utiliser des algorithmes naturels et bio-inspirés, en particulier des algorithmes génétiques. Un algorithme génétique est utilisé pour trouver la solution quasi-optimale (solution approchée de bonne qualité) en explorant un espace de recherche prohibitif contenant un grand nombre de solutions possibles (le nombre de combinaisons est faramineux lorsque le nombre de sources disponibles est très élevé). La solution trouvée doit maximiser la similarité entre la requête de l'utilisateur et la description des sources qui constituent la solution. Cette contribution est expliquée plus en détail au chapitre 4.
2. **Approche intelligente enrichie de données sociales pour améliorer la sélection des sources** [94] :
   Notre objectif est d'intégrer les données extraites des réseaux sociaux dans la sélection des sources afin d'améliorer la précision de la méthode de sélection des sources intelligente proposée (contribution 1) en considérant la trace des utilisateurs lors de l'utilisation des sources. Les données de marquage social peuvent contribuer à améliorer la qualité de la description des sources par les tags associés par les utilisateurs aux sources. La pertinence de la solution de l'algorithme génétique proposé prend en compte à la fois le contenu de la source et les balises attribuées à cette source. La meilleure sélection des sources maximise la fonction de fitness adaptée, ce qui augmente les performances de la recherche. La contribution est expliquée plus en détail au chapitre 5.
3. **Une approche multidimensionnelle pour adapter la sélection des sources aux thèmes d'intérêt de l'utilisateur** [96] :
   Nous abordons ici le problème de la sélection des sources sous l'angle de la personnalisation. L'objectif est de personnaliser les résultats de la recherche multi-sources en fonction du profil social de l'utilisateur. Une approche multidimensionnelle pour la sélection des sources est proposée qui intègre la dimension sociale et la dimension intelligence. L'approche proposée s'appuie sur les techniques de modélisation des sujets LDA (Latent Dirichlet Allocation) pour personnaliser la sélection des sources en fonction des thèmes (sujets) d'intérêt de l'utilisateur.

   L'utilisation de techniques de modélisation de sujets (telles que LDA) peut jouer un rôle important dans la découverte de structures cachées liées au comportement des utilisateurs dans les réseaux sociaux. Nous avons utilisé le modèle LDA pour découvrir la structure thématique latente présente dans un corpus de données de tagging social. Les tags sont regroupés en thèmes ce qui élimine le bruit, l'ambiguïté et la

redondance dans la collection de tags. Nous construisons alors un profil d'utilisateur concis de dimensions z (ou z est le nombre de thèmes). Les thèmes d'intérêt découverts sont ensuite utilisés pour comprendre les besoins réels des utilisateurs et pour trouver la sélection des sources "optimale" et "sur mesure" pour chaque utilisateur en utilisant un algorithme génétique. La contribution est expliquée plus en détail au chapitre 6.

4. **Découverte de connaissances à partir de l'analyse des fichiers journaux d'un système de recherche multi-sources basée sur un nettoyage en profondeur** [95] :

   Nous avons proposé une méthode d'analyse approfondie des fichiers journaux pour un système de recherche multi-sources afin de découvrir des informations pertinentes sur les caractéristiques des utilisateurs interagissant avec le système. La méthode proposée utilise des techniques de fouille de l'utilisation du Web (web usage mining) et est composée de plusieurs étapes, dont une étape importante est le nettoyage en profondeur. Le nettoyage en profondeur nettoie les fichiers journaux et élimine les lignes inutiles et ce en analysant la structure des requêtes des sources disponibles sur la base d'un dictionnaire de motifs. L'étape de nettoyage en profondeur accélère considérablement l'étape de traitement et génère des résultats plus précis. Les données générées peuvent être utilisées à diverses fins, y compris la création d'un profil d'utilisateur précis nécessaire pour améliorer les performances de la recherche multi-sources. Cette contribution est expliquée plus en détail au chapitre 7.

Toutes ces contributions sont évaluées sur des ensembles de données réels extraits d'un système de recherche multi-sources et de réseaux de marquage social et les résultats ont montré l'efficacité de nos propositions par rapport aux approches de sélection des sources de l'état de l'art en utilisant des métriques d'évaluation des performances basées sur la précision.

3 Travaux publiés

Les contributions de cette thèse ont fait l'objet d'articles publiés dans les conférences et revues internationales suivantes :

- Fatma Zohra Lebib, Habiba Drias, Hakima Mellah : Selection of Information Sources Using a Genetic Algorithm (2017). WorldCIST (1 :60-70.
- Fatma Zohra Lebib, Hakima Mellah, Habiba Drias (2017). Enhancing information source selection using a genetic algorithm and social tagging. Int. J. Inf. Manag. 37(6) :741-749.
- Fatma Zohra Lebib, Hakima Mellah, Abdelkrim Meziane (2019). Knowledge Discovery from Log Data Analysis in a Multi-source Search System based on Deep Cleaning. WEBIST'2019 : 257-264.
- Fatma Zohra Lebib, Hakima Mellah, Abdelkrim Meziane : A Multi-Dimensional Source Selection Based on Topic Modelling. J. Inf. Sci. Eng. 38(3) : 619-644 (2022).

4 Organisation de la thèse

Cette thèse est organisée en deux parties. L'objectif de la première partie est de présenter un aperçu de la littérature sur le contexte de la recherche d'information multi-sources

selon les chapitres suivants.

Dans le chapitre 1, nous commençons par décrire la recherche multi-sources et ses différentes étapes, puis nous présentons les différentes catégories de méthodes qui sont proposées pour la sélection des sources d'information. Enfin, nous présenterons les métriques utilisées pour l'évaluation des performance des systèmes de recherche d'information multi-sources.

Le chapitre 2 présente la recherche d'information qui prend en compte la dimension sociale des utilisateurs. Nous commençons par définir la recherche d'information sociale, puis nous montrons comment la dimension sociale est intégrée dans le processus de recherche d'information. Enfin, nous présenterons les mesures utilisées pour l'évaluation de la performance des systèmes qui considèrent l'aspect social dans la recherche d'information.

Le chapitre 3 présente l'aspect intelligence dans la recherche d'information. Nous commençons par introduire l'utilisation de méthodes d'intelligence artificielle pour résoudre des problèmes liés à la recherche d'information, puis nous décrivons les algorithmes génétiques qui représentent une classe de technologies intelligentes utilisées pour résoudre de nombreux problèmes complexes. Enfin, nous discutons des domaines d'application des algorithmes génétiques et de quelques travaux de recherche intéressants pour chacun de ces domaines.

La seconde partie décrit les différentes approches adoptées pour la recherche d'information dans un environnement multi-sources :

Dans le chapitre 4, nous présentons l'approche de sélection des sources basée sur un algorithme génétique. Nous définissons le problème de sélection des sources comme un problème d'optimisation combinatoire, qui consiste à trouver la meilleure combinaison dans un espace de recherche prohibitif. Pour résoudre ce problème, nous proposons d'utiliser des algorithmes bio-inspirés, en particulier des algorithmes génétiques. Nous expliquons l'algorithme proposé, que nous appelons GASS, et nous le comparons aux algorithmes de sélection des sources de l'état de l'art.

Dans le chapitre 5, nous présentons l'approche de sélection des sources basée sur un algorithme génétique amélioré appelé IGASS. Nous considérons le comportement des utilisateurs dans les réseaux de tagging social pour enrichir la description des sources ce qui améliore la précision de l'algorithme de sélection des sources. Nous présentons les résultats de la comparaison de l'algorithme IGASS avec l'algorithme GASS et les algorithmes de sélection des sources de l'état de l'art.

Dans le chapitre 6, nous proposons une approche de sélection des sources multidimensionnelle basée sur des techniques de modélisation de sujets LDA et des algorithmes génétiques. L'algorithme de sélection des sources proposé est appelé PGASS-based-LDA. L'objectif est de personnaliser la recherche multi-sources en combinant l'aspect social et l'aspect intelligence. Nous comparons PGASS-based-LDA aux algorithmes de sélection des sources personnalisés et non-personnalisés de l'état de l'art.

Dans le chapitre 7, nous présentons une méthode basée sur les techniques de Data Mining pour l'analyse des fichiers journaux du système de recherche multi-sources afin de découvrir de nouvelles connaissances sur les utilisateurs du système. Nous expliquons les différentes étapes de la méthode proposée et nous présentons la mise en œuvre de la méthode et les résultats d'analyse obtenus par les tests.

Partie I : La Recherche d'Information dans un Environnement Multi-Sources

Chapitre 1

Sélection des sources dans la recherche d'information multi-sources

Sommaire

1 Introduction

Un système de recherche d'information multi-sources permet aux utilisateurs d'accéder efficacement à des informations provenant de plusieurs sources d'information. Le système reçoit la demande d'un utilisateur et l'envoie aux sources appropriées. La requête est traitée dans chacune des sources, produisant des listes avec un ensemble de résultats individuels. Ces listes de résultats sont fusionnées en une seule liste de documents pertinents qui est ensuite retournée à l'utilisateur. La sélection des sources est une étape essentielle d'un système de recherche d'information multi-sources, dans laquelle le système sélectionne parmi les sources disponibles que celles qui contiennent des informations pertinentes pour une requête donnée. En général, le nombre de sources choisies est bien inférieur à l'ensemble total de sources, de sorte que le système peut bénéficier des coûts

réduits de ne pas avoir à accéder à toutes les sources. Plusieurs algorithmes de sélection des sources sont proposés dans la littérature, nous citerons ci-après les plus connus.

Dans la première partie de ce chapitre, nous présentons une description détaillée de la recherche d'information multi-sources. En particulier, nous décrivons les différentes étapes du processus de recherche d'information multi-sources. Dans la deuxième partie, nous passerons en revue les approches de sélection des sources d'information. La dernière partie de ce chapitre présente des méthodes d'évaluation des systèmes de recherche d'information multi-sources.

2 La recherche d'information multi-sources

2.1 Définition

La Recherche d'Information Multi-Sources (RIMS) ou la recherche d'information distribuée [29, 139], offre aux utilisateurs la possibilité de rechercher simultanément plusieurs sources d'information (bibliothèques de recherche universitaires, sites d'achat en ligne, bases de données, etc.) via une interface unique. Le contenu de ces sources d'information est souvent généré dynamiquement en réponse aux requêtes des utilisateurs, ce qui rend leur recherche via les moteurs de recherche conventionnels difficile, voire impossible.

Le processus de RIMS comprend les phases suivantes : (*i*) la représentation (description/résumé) de la source d'information [30, 142], où une description précise est créée pour chaque source. (*ii*) Sélection des sources [31, 142], où, compte tenu de la requête d'un utilisateur, un sous-ensemble de sources pertinentes est sélectionné pour repondre à la requête. (*iii*) Fusion de résultats [117, 141], où les résultats obtenus à partir des sources sélectionnées sont combinés en une seule liste de résultats qui est renvoyée à l'utilisateur. Un schéma général d'un système RIMS est illustré à la Figure 1.1.

2.2 Différentes phases de la recherche d'information multi-sources

Dans cette section, nous discutons brièvement des principales phases de la recherche d'information multi-sources, à savoir la description de la source, la sélection de la source et la fusion des résultats.

2.2.1 Description de la source

Dans la phase hors ligne de la recherche d'information multi-sources, une description représentative est construite pour chaque source d'information disponible (Figure 1.2). La description peut inclure un contenu complet d'une source (ou seulement un échantillon de ses documents dans des environnements non coopératifs [30]), des métadonnées de statistiques de termes et de documents (si disponibles) et d'autres descripteurs du contenu de la source.

Le protocole STARTS (Stanford Protocol for Internet Retrieval and Search) [66] est une des solutions d'acquisition des descriptions de sources. Cela necessite une coopération explicite de toutes les sources d'information. Bien que STARTS soit une bonne solution dans des environnements où la coopération peut être garantie, il ne fonctionne pas dans les environnements multipartites tels qu'Internet ou les grands réseaux d'entreprise, où une coopération totale est peu probable. L'échantillonnage basé sur des requêtes (Query Based Sampling -QBC) [29, 30] est une solution alternative pour acquérir des descriptions

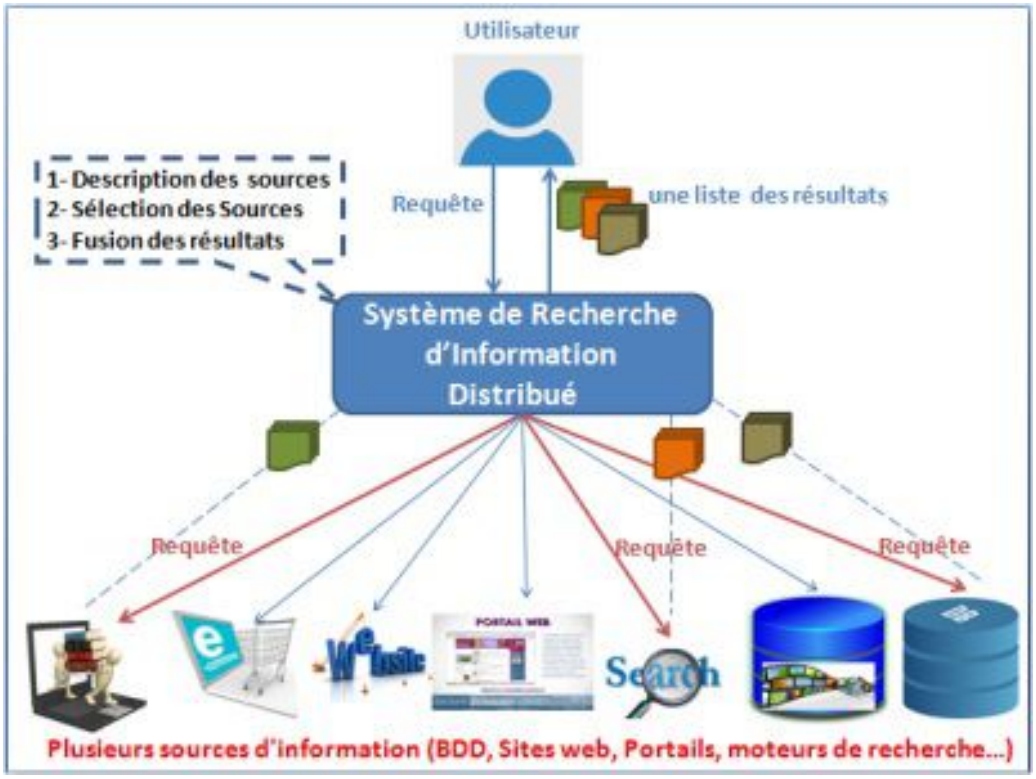


Figure 1.1 – Système de recherche d'information multi-sources.

de sources car elle ne nécessite pas la coopération explicite des sources d'information. QBS utilise uniquement le processus normal d'exécution de requêtes et de récupération d'une liste de documents téléchargeables pour créer des descriptions de sources.

L'échantillonnage basé sur des requêtes QBC est mis en œuvre avec un algorithme simple, décrit ci-dessous [30].

1. Sélectionner un terme de la requête initial.
2. Exécuter la requête à terme unique sur la source.
3. Récupérer les N premiers documents renvoyés par la source.
4. Mettre à jour la description de la source en fonction des caractéristiques des documents récupérés.
   a) Extraire les mots et les fréquences des N premiers documents renvoyés par la source ; et
   (b) Ajouter les mots et leurs fréquences à la description de cette source.
5. Si un critère d'arrêt n'est pas encore atteint,
   (a) Sélectionner un nouveau terme de la requête ; et
   (b) Passer à l'étape 2.

L'algorithme implique plusieurs choix, par exemple, comment les termes de la requête sont sélectionnés, combien de documents sont examinés par requête et quand arrêter l'échantillonnage.

Les requêtes utilisées par l'algorithme QBS sont généralement des mots simples échantillonnés à partir de la représentation actuelle de la source [30]. Une autre approche consiste à sélectionner ces requêtes à partir d'une ressource externe. Cette approche peut produire des échantillons plus représentatifs, mais elle est moins efficace car certaines

requêtes peuvent ne renvoyer aucun résultat. Concernant le nombre de documents à analyser, il a été suggéré que 300 à 500 documents suffisent pour échantillonner une source [30]. Des extraits peuvent être utilisés à la place des documents, afin d'éliminer le besoin de télécharger des documents à partir de la source[153].

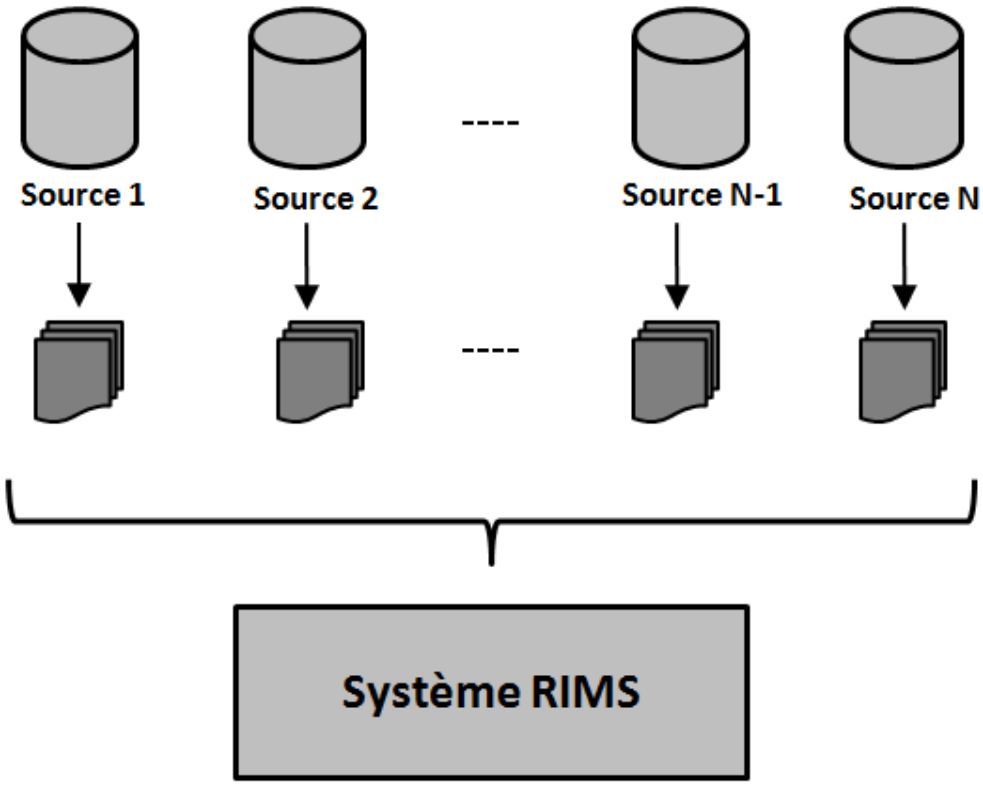


Figure 1.2 – La phase de description des sources.

Les descriptions de toutes les sources disponibles sont gérées de manière centralisée par un courtier (système de RIMS) et sont utilisées pour les phases suivantes, telles que la sélection des sources et la fusion des résultats.

2.2.2 Sélection de la source

Compte tenu de la requête d'un utilisateur et des descriptions des sources, le courtier RIMS sélectionne les sources les plus pertinentes pour la requête (Figure 1.3). Évidemment, ignorer cette étape et envoyer la requête à toutes les sources connues est une solution possible, mais cette méthode est très coûteuse en termes de ressources et peut augmenter la latence de l'utilisateur. Ainsi, l'objectif de la sélection des sources est de réduire au maximum le nombre de sources à interroger, sans diminuer l'efficacité de la recherche [54].

Afin de sélectionner un sous-ensemble de sources susceptibles de contenir des documents pertinents pour une requête donnée, la plupart des techniques de sélection des sources calculent un score pour chacune des sources, en fonction de leur pertinence (ou utilité) pour la requête soumise. Les sources sont ensuite classées par ordre décroissant de leurs scores de pertinence. Enfin, le système peut sélectionner soit K sources les mieux classées, soit les sources dont les scores de pertinence dépassent un certain seuil de pertinence. Ce type de sélection des sources implique la collecte d'informations sur les sources

disponibles afin de calculer les scores de pertinence de chaque source pour une requête d'utilisateur. Ainsi, les approches de sélection des sources existantes diffèrent dans la nature de ces informations ou la manière dont elles sont acquises, et dans la méthode utilisée pour estimer l'utilité des sources d'information disponibles pour la requête d'un utilisateur.

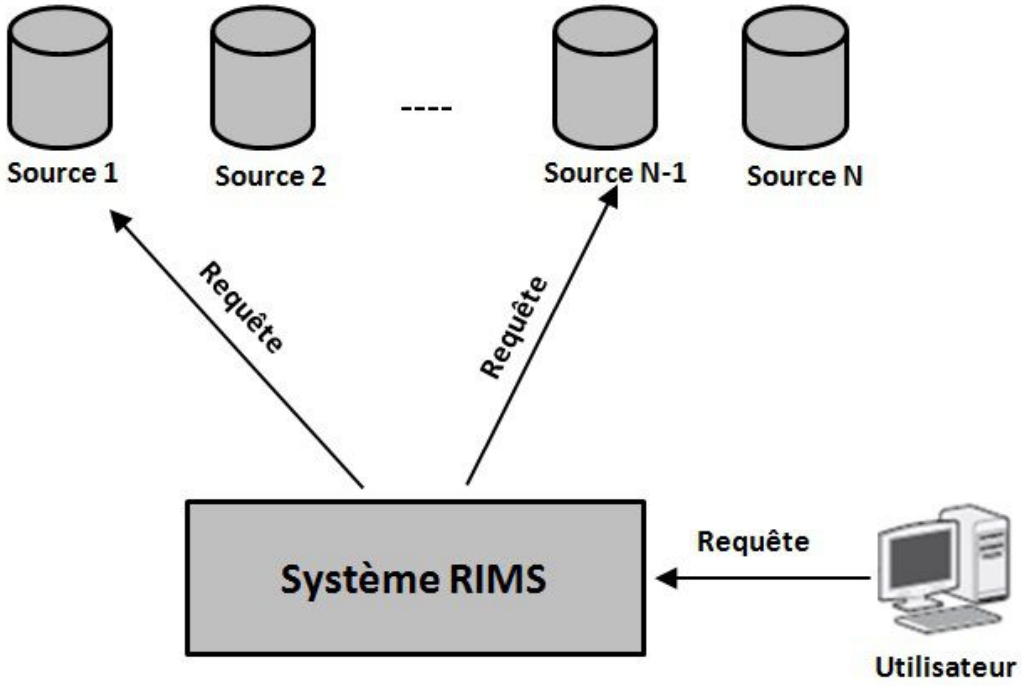


Figure 1.3 – La phase de sélection des sources.

Une variété d'approches différentes de la sélection des sources ont été proposées et évaluées au cours des dernières décennies [37, 68, 79, 124, 137, 142, 152]. Ces approches peuvent être divisées en trois grandes familles en fonction de leur stratégie de sélection. Ces trois grandes familles d'approches seront décrites dans la Section 3.

2.2.3 Fusion des résultats

La requête de l'utilisateur est transmise aux sources sélectionnées et les résultats récupérés à partir des sources spécifiques sont fusionnés en une seule liste à l'aide des méthodes de fusion et de normalisation des scores. La liste finale des résultats est présentée à l'utilisateur (Figure 1.4).

Les sources d'information utilisent des statistiques et des modèles de recherche différents, ce qui rend les scores des documents, calculés par ces sources, non comparables et qui doivent être normalisés pour pouvoir fusionner les différents résultats [106].

Les méthodes de normalisation des scores exigent que les scores de pertinence des documents soient fournis par les sources. Si les scores ne sont pas disponibles, ou si les scores des sources ne sont pas fiables, ce qui est souvent le cas dans les applications de recherche réelles, la fusion peut être basée sur les rangs des documents récupérés [141]. Dans ce cas, les documents échantillonnés peuvent être utilisés pour recalculer le score localement, mais cela nécessite un échantillon significatif de chaque source.

Les techniques de fusion des résultats qui utilisent des descriptions de sources fournies soit implicitement, par exemple CORI [107], soit explicitement, telles que SSL [141] et

SAFE [140].

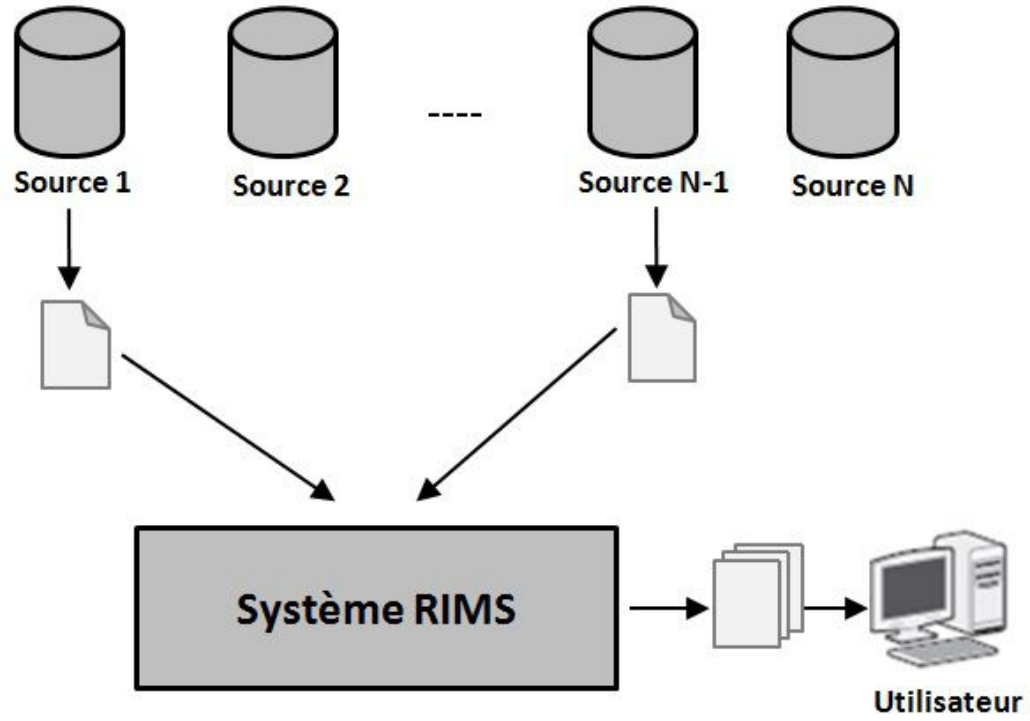


Figure 1.4 – La phase de fusion des résultats.

3 Approches de sélection des sources

Un certain nombre de solutions ont été proposées pour la sélection des sources, qui peuvent être regroupées en trois grandes catégories, à savoir les approches grand-document(Big document approaches), les approches petit-document (Small-document approaches) et les approches basées sur la classification (Classification based approaches).

3.1 Approches grand-document

La première catégorie est ainsi appelée car les sources d'information sont représentées par un grand document virtuel qui est la concaténation de tous les documents représentatifs des sources disponibles. Dans les environnements non coopératifs où les sources ne partagent pas leur contenu, des documents échantillonnés sont utilsés pour construire des descriptions de sources (voir la technique d'échantillonnage à la section 2.2.1). Les algorithmes de recherche de document standard sont adaptés pour classer les sources d'information par rapport à une requête donnée. Par conséquent, les approches grand-document diffèrent essentiellement par la manière dont les descriptions des sources sont classées, par exemple, en utilisant un réseau d'inférence bayésien (CORI) [31], le modèle d'espace vectoriel (vGlOSS) [68], ou des modèles de langage [143, 160].

CORI. CORI (collection retrieval inference network) [31] utilise un réseau d'inférence bayésien. Les feuilles dans un réseau CORI représentent les sources et elles sont connectées à un groupe de nœuds pour les ensembles de représentations des sources

au deuxième niveau du graphe. Le nœud de représentation de chaque source contient les termes qui apparaissent dans cette source.

La similarité entre les ensembles de représentation et une requête est mesurée par le système de recherche INQUERY [5]. INQUERY a été conçu à l'origine pour classer les documents, mais dans CORI, il est légèrement modifié pour devenir applicable à la sélection des sources. Dans INQUERY, la fréquence du terme est remplacée par la fréquence du document, et la fréquence inverse du document est remplacée par la fréquence inverse de la source.

Dans CORI, la croyance de la $i^{ème}$ collection associée au mot t, se calcule comme suit :

$$T = \frac{df_{t,i}}{df_{t,i} + 50 + 150 + \frac{cw_i}{avg_{cw}}} \tag{1.1}$$

$$I = \frac{log(\frac{N_c+0.5}{cf_t})}{log(N_c + 1.0)} \tag{1.2}$$

$$p(t \setminus c_i) = b + (1 - b) \times T \times I \tag{1.3}$$

Où $df_{t,i}$ est le nombre de documents de la $i^{ème}$ collection contenant t, cf_t est le nombre de collections qui contiennent t, N_c est le nombre total de collections disponibles, cw_i est le nombre total de mots dans la $i^{ème}$ collection, et avg_{cw} est le moyen de cw de toutes les collections. Enfin, b est la croyance par défaut, qui est généralement fixée à 0.4. La croyance $P(Q \setminus c_i)$ est utilisée par l'algorithme CORI pour classer les collections. La façon la plus courante de calculer la croyance $P(Q \setminus c_i)$ est d'utiliser la valeur moyenne des croyances de tous les termes de la requête.

vGlOSS. vGlOSS est la version espace vectoriel de GlOSS(glossary of servers server) [68]. Dans vGlOSS, les sources sont classées en fonction de leurs valeurs de qualité (*goodness*). la valeur *goodness* d'une source pour une requête est calculée en additionnant les valeurs de similarité de ses documents avec la requête (Équation 1.4).

$$Goodness(q, l, c) = \sum_{d \in Rank(q,l,c)} sim(q, d) \tag{1.4}$$

Où $sim(q, d)$ est la similarité cosinus [129, 130] des vecteurs pour le document d et la requête q. Pour éviter un éventuel bruit produit par des documents à faible similarité, vGlOSS utilise un seuil de similarité l.

KL. Xu and Croft [160] proposent une méthode de sélection des sources basée sur la classification des documents et la modélisation du langage. La classification des documents est utilisée pour organiser les sources autour de thèmes. La modélisation du langage est utilisée pour la représentation des thèmes et pour sélectionner les thèmes appropriés pour une requête donnée. La mesure de divergence Kullback-Leibler (KL) est utilisée pour prédire les thèmes pour une requête donnée.

La formule KL suivante est utilisée pour mesurer comment un modèle de sujet pour le sujet T prèdit une requête Q.

$$KL(Q, T) = \sum_{f(Q,w_i) \neq 0} \frac{f(Q, w_i)}{\mid Q \mid} log \frac{\frac{f(Q,w_i)}{|Q|}}{p_i} \tag{1.5}$$

Où $f(Q, w_i)$ est le nombre d'occurrences du mot w_i dans la requête Q et $\mid Q \mid$ est la longueur de Q en mots. w_i est un mot dans l'ensemble de vocabulaire du modèle de langage. p_i est la fréquence à laquelle un mot w_i est utilisé dans le texte de T lorsqu'il est observé avec une quantité illimitée de données. Pour un ensemble de documents disponibles D sur T, p_i est estimé comme suit :

$$p_i = \frac{f(D, w_i) + 0.01}{\mid D \mid + 0.01n} \tag{1.6}$$

Où $f(D, w_i)$ est le nombre d'occurrences de w_i dans D, $\mid D \mid$ est la taille de D en mots et n est la taille du vocabulaire. La petite valeur 0.01 empêche les probabilités nulles car la divergence KL décrite ci-dessus implique des logarithmes.

Si et al. [143] ont proposé une méthode de sélection des sources qui construit des modèles de langage à partir des ensembles de représentation des sources disponibles, et il classe les sources en calculant la divergence de Kullback-Leibler (KL)entre le modèle de requête et les modèles de source.

Taily. Taily [6] modélise la distribution des scores d'une requête dans chaque fragment en tant qu'une distribution Gamma, et il sélectionne les fragments avec des scores de document élevés dans la queue de la distribution. Taily pré-calcule deux paramètres de mise à l'échelle θ et K de la distribution gamma pour ajuster le score du document pour chaque requête à terme unique par rapport à chaque fragment. En stockant la distribution de score d'une requête à terme unique sur chaque fragment, il peut estimer la distribution de score d'une requête utilisateur avec plusieurs termes.

Ces paramètres sont calculés en fonction de la moyenne et de la variance des fonctionnalités de la fonction de score dans les collections et dans les fragments.

Les méthodes de grand-document ne tiennent pas compte des frontières des documents dans les sources et peuvent donc sous-estimer une source volumineuse contenant de nombreux documents pertinents [75]. La modélisation thématique a été proposée par des travaux plus récents pour pallier cette limitation [12].

3.2 Approches petit-document

La décision de supprimer les limites des documents dans l'ensemble de représentations de documents peut avoir un impact sur les performances de la sélection des sources [160], un certain nombre d'études empiriques ont soutenu cette affirmation[71, 142, 151]. En conséquence, des approches petit-document ont été proposées pour conserver les limites des documents. Les approches petit-document sont généralement conçues pour les environnements non coopératifs où les informations complètes du lexique des sources ne sont pas disponibles. Cependant, ces approches pourraient également être appliquées dans des environnements coopératifs [139]. Les documents échantillonnés obtenus à partir de chaque source sont indexés au niveau d'un courtier pour former un index centralisé partiel, qui représente une approximation de l'index virtuel global de la source. Ainsi, étant donné une requête utilisateur, les documents de l'index centralisé sont classés en premier. Ce classement des documents est ensuite utilisé pour prédire quelles sources ont le plus grand nombre de documents pertinents, cela aide à guider le processus de décision pour sélectionner un sous-ensemble de sources pertinentes pour la recherche. Une étude fournie par Markov et Crestani [109] présente une analyse détaillée des approches petit-document.

Des algorithmes tels que ReDDE [142], DTF [55], CRCS [137], SUSHI [152] et [91] ont été proposés, ils utilisent différentes méthodes pour pondérer les documents les mieux classés et pour estimer la probabilité de pertinence.

ReDDE. L'algorithme ReDDE (relevant document distribution estimation) [142] a été conçu pour sélectionner un petit nombre de sources avec le plus grand nombre de documents pertinents. Pour atteindre cet objectif, ReDDE estime explicitement la distribution des documents pertinents dans toutes les sources et classe les sources en conséquence. Pour estimer le nombre de documents pertinents dans chaque source, ReDDE utilise un index d'échantillon centralisé, composé de tous les documents qui sont échantillonnés dans la phase d'échantillonnage. Chaque requête est comparée à cet index avant d'être soumise aux sources. Le nombre de documents pertinents dans chaque collection est estimé en fonction de la contribution des sources dans les documents les mieux classés de l'index central, et la valeur de *goodness* de chaque source (collection) est calculée comme suit :

$$goodness(q, c_i) = \frac{\widehat{rel(q, c_i)}}{\sum_i \widehat{rel(q, c_i)}} \tag{1.7}$$

$\widehat{rel(q, c_i)}$ est le nombre estimé de documents pertinents dans la collection c pour la requête q et est calculé comme suit :

$$\widehat{rel(q, c_i)} = \sum_{d \in |S_{c_i}|} P(rel, d) \times \frac{\widehat{|\ c_i\ |}}{|\ S_{c_i}\ |} \tag{1.8}$$

Où, $P(rel, d)$ est la probabilité de pertinence du document d dans la collection c_i, $|\ S_{c_i}\ |$ est le nombre de documents échantillonnés téléchargés par QBS à partir de la collection c_i, et $\widehat{|\ c_i\ |}$ est le nombre estimé de documents dans c_i.

Différentes variantes des algorithmes ReDDE ont émergé qui pondèrent les documents les mieux classés et estiment la probabilité de pertinence de différentes manières, telles que CRCS [137] et SUSHI [152] qui sont décrits plus en détail plus loin dans cette section.

DTF. Le DTF (decision-theoretic framework) [55] vise à minimiser le coût total de la recherche, y compris les coûts de recherche de chaque source, de récupération des résultats et de présentation des documents.

Dans DTF, l'efficacité de la recherche de sources peut être apprise en utilisant à l'avance un ensemble de requêtes d'apprentissage. La méthode DTF nécessite un grand nombre de requêtes d'apprentissage, mais elle possède l'un des modèles théoriques les plus solides parmi les techniques de sélection des sources disponibles. Elle combine coûts (monétaires, réseau) et pertinence dans un cadre théorique de décision. La principale limite de DTF dans les applications pratiques est la nécessité de spécifier à l'avance les fonctions de coût.

CRCS. Comme dans ReDDE [142], CRCS (central-rank-based collection selection) [137] exécute la requête sur un index centralisé de tous les documents échantillonnés et classe les sources en fonction des rangs de ses documents échantillonnés qui sont dans les premiers résultats γ. Les documents classés après les premiers résultats γ sont considérés comme moins pertinents et n'auront aucun impact sur les pondérations des sources. L'impact d'un document échantillonné sur le poids de sa source originale

est calculé en fonction de la position de ce document dans les premiers résultats γ. Dans sa forme la plus simple, cela peut être calculé linéairement comme suit :

$$R(d_j) = \begin{cases} \gamma - j & \text{si } j < \gamma \\ 0 & \text{sinon} \end{cases} \tag{1.9}$$

Où $R(d_j)$ représente l'impact du document d au rang j des résultats retournés par un modèle de recherche appliqué à l'index de tous les documents échantillonnés des collections.

CRCS calcule goodness (poids) de chaque collection comme suit :

$$Goodness(c_i) = \frac{\mid c_i \mid}{\mid c^{max} \mid \times \mid S_{c_i} \mid} \times \sum_{d \in S_{c_i}} R(d_j) \tag{1.10}$$

Où, $\mid c_i \mid$ est la taille de la collection i estimée par la méthode de l'historique des captures (capture-history method). Les tailles de collection sont normalisées en divisant la taille de chaque collection par la taille de la plus grande collection impliquée ($\mid c^{max} \mid\mid$). La taille du résumé de la collection i, c'est-à-dire le nombre de documents téléchargés par échantillonnage basé sur une requête à partir de cette collection, est représentée par $\mid S_{c_i} \mid$. Le poids de chaque collection est calculé en additionnant les valeurs d'impact de ses documents échantillonnés.

SUSHI. La plupart des techniques de sélection des sources supposent des valeurs de coupure fixes. En d'autres termes, le nombre de sources sélectionnées pour toutes les requêtes est le même. Thomas et Shokouhi [152] ont assoupli cette hypothèse dans SUSHI (Scoring Scaled Samples for Server Selection). Les auteurs ont ajusté plusieurs courbes à la distribution des scores des documents échantillonnés afin de vérifier le nombre de sources à sélectionner pour une requête. Les auteurs ont montré que SUSHI peut atteindre des performances comparables à ReDDE et CRCS, tout en sélectionnant moins de sources.

Ces approches considèrent que les sources sont comparables à travers les scores attribués en fonction de la correspondance des termes. Cette hypothèse peut être problématique lorsque les sources contiennent des informations sur différents supports (par exemple image ou vidéo), appartenant à des thèmes différentes. Le modèle de sélection des sources doit également tenir compte des informations incomplètes et de la désambiguïsation des termes. C'était une motivation clé pour l'introduction de techniques supervisées plus sophistiquées pour la sélection des sources, dont nous parlerons dans la section suivante.

3.3 Approches basées sur la classification

Les approches basées sur la classification considèrent la sélection des sources comme un problème de classification [8, 37, 42, 43, 75, 78, 84]. Un modèle de classification peut être appris à partir d'un ensemble de requêtes d'apprentissage et est utilisé pour prédire l'adéquation d'une source pour des requêtes de test.

Dans une série d'articles, Ipeirotis et Gravano [77–79] ont proposé une technique basée sur la classification pour la sélection des sources en exploitant la relation thématique entre les sources. Les auteurs attribuent les sources aux branches d'un arbre de classification hiérarchique selon les termes de leurs documents échantillonnés. Chaque branche représente une catégorie de thème qui peut être liée à plusieurs sources. Les statistiques de

terme des ensembles de représentation source sont propagées pour générer des résumés de catégorie. Pour la sélection des sources, la requête est comparée aux résumés de catégories, la requête est envoyée aux sources de catégories ayant les scores les plus élevés.

Arguello et al. ont proposé une approche basée sur la classification pour la sélection des sources pour la recherche fédérée [8] ou la recherche verticale [9]. Un modèle de classification est appris à partir d'un ensemble de requêtes d'apprentissage et est utilisé pour prédire la pertinence d'une source pour des requêtes de test. Les auteurs ont considéré que les sources ont une pertinence binaire, c'est-à-dire pertinente ou non pertinente par rapport à chaque requête. Dans leur algorithme, des fonctionnalités telles que la similarité des sources, la requête de l'utilisateur, la description des ressources des sources et la relation entre les sources sont utilisées pour classer les sources comme pertinentes ou non pertinentes.

Cetintas et al.[37] ont montré que l'utilisation des résultats de recherche des requêtes précédentes améliore l'efficacité de la sélection des sources. Chaque requête est comparée à l'ensemble de toutes les requêtes précédentes. Les sources sont classées en fonction de la moyenne pondérée de leurs performances pour les requêtes précédentes les plus similaires. La valeur de similarité pour chaque paire de requêtes est calculée par rapport à un index centralisé des documents échantillonnés. Les requêtes qui renvoient des listes classées similaires sont considérées comme similaires. De plus, les performances des sources pour les requêtes précédentes sont estimées en fonction de la position de leurs documents dans un classement centralisé des documents échantillonnés. Par conséquent, l'approche proposée ne repose pas sur des jugements de pertinence pour les requêtes de formation.

Dans le travail de Hong et al. [75], les auteurs proposent un nouveau modèle de classification probabiliste conjoint pour la sélection des sources dans la recherche textuelle fédérée. Leur l'hypothèse est qu'une source d'information a tendance à être pertinente pour une requête d'utilisateur, si elle est similaire à une autre source avec une forte probabilité d'être pertinente. Ainsi, le modèle proposé estime les probabilités de pertinence de manière conjointe en combinant à la fois les données des sources individuelles et la relation entre les sources d'information.

Pour la recherche verticale, Kang et al. [84] ont proposé une nouvelle formulation de pertinence multi-aspect, qui calcule la pertinence entre une requête et un document selon plusieurs aspects de pertinence, tels que la correspondance du texte, la réputation et la distance. Afin d'apprendre une fonction de classement en utilisant la pertinence multi-aspects, les auteurs ont étudié deux types d'approches basées sur l'apprentissage pour estimer le compromis entre ces aspects de pertinence, à savoir une méthode d'agrégation d'étiquettes (labels) et une méthode d'agrégation de modèles. les résultats expérimentaux ont montré que la formulation de pertinence multi-aspects est prometteuse et que les approches d'agrégation proposées sont très efficaces pour apprendre le compromis entre les différents aspects.

Des techniques d'apprentissage supervisé basées sur des méthodes d'apprentissage d'ordonnancement (learning to rank)ont également été exploitées pour la sélection et le classement de sources dans la recherche distribuée. Xu et Li [161] ont proposé de nouvelles fonctionnalités qui sont utilisées pour apprendre à classer les sources en utilisant deux algorithmes de classeemnt SVM (Support Vector Machine) et RankingSVM. Dai et al. [42] ont étudié l'application de l'approche d'apprentissage d'ordonnancement pour classer les sources dans la recherche sélective. Ils ont développé de nouvelles fonctionnalités pour le classement des sources et ils ont proposé une approche d'apprentissage qui ne nécessite pas de jugements de pertinence humains. ils entraînent le modèle SVMrank pour classer

les sources en combinant des fonctionnalités indépendantes des requêtes, des fonctionnalités basées sur des termes et des fonctionnalités d'échantillons de documents (l'index d'échantillons centralisé). Des résultats des expériences ont montré que l'approche proposée est efficace pour classer des centaines de fragments produits par la recherche sélective et qu'il s'agit d'une amélioration par rapport aux approches de sélection des sources de l'état de l'art. Ils ont également montré aussi que l'algorithme de sélection des sources appris produit une précision de recherche comparable à une recherche exhaustive jusqu'au rang 1000. Wu et al. [157] proposent un algorithme de sélection des sources basé sur l'apprentissage d'ordonnancement appelé LTRRS. Leur modèle combine trois fonctionnalités, la fonctionnalité de correspondance des termes, la fonctionnalité de pertinence du sujet (thème) et les fonctionnalités basées sur l'index central des échantillons (CSI : Central Sample Index). Ils sont utilisé le modèle LambdaMART comme modèle d'apprentissage d'ordonnancement dans l'algorithme proposé LTRRS.

3.4 Autres approches de sélection des sources

La plupart des recherches précédentes se sont concentrées sur l'évaluation de la pertinence d'une source en analysant des informations statiques de cette source[75]. Des travaux plus récents prennent en compte d'autres informations importantes telles que : les résultats des requêtes précédentes [37], la diversification des résultats [74], l'importance et la fiabilité des sources et des résultats [13–15], la pertinence et la nouveauté [127], la qualité de la source [45, 46, 98–100], le contexte de la requête [36], et la représentation sémantique de la source et la distance sémantique entre la requête et la source [16].

Dans [13–15], les auteurs ont considérés l'importance et la fiabilité (confiance) des sources et des résultats pour la sélection des sources dans le web invisible. Ils ont estimé que la concordance entre les réponses est utile pour évaluer l'importance et la fiabilité des sources et des résultats. Pour évaluer la qualité de la source, la concordance entre les sources est calculée à partir de la concordance des réponses renvoyées par ces sources. Cet accord est modélisé par un graphe où les sommets représentent les sources. Sur ce graphique d'accord, un score de qualité de source, appelé **SourceRank**, est calculé comme la probabilité de visite stationnaire d'une marche aléatoire. SourceRank est calculé hors ligne et peut être combiné avec une mesure de pertinence de la source pour une requête donnée pour le classement final de la source. Rehatsinas et al. [127] ont étudié le problème de la sélection des sources en considérant des sources de données dynamiques dont le contenu évolue dans le temps. Ils ont défini un ensemble de mesures dépendant du temps, y compris la couverture, la fraîcheur et la précision, pour estimer la qualité de l'intégration des données. Leur approche permet de sélectionner un sous-ensemble optimal de sources qui à la fois maximise la qualité de l'intégration et dont leur fréquence de mise à jour est optimale. Dong et al. [46] ont proposé une approche pour sélectionner des sources de données de bonne qualité pour une intégration à faible coût. Une nouvelle source n'est intégrée que si son gain marginal, souvent lié à l'amélioration de la qualité de l'intégration, est supérieur au coût marginal associé aux coûts d'achat de données et aux ressources d'intégration. Dans [45], les auteurs ont proposé une méthode de sélection des sources de données multimédia basée sur les commentaires des utilisateurs (feedbacks). Ils ont pris en compte les dimensions de la qualité de la source de données qui peuvent être calculées objectivement, telles que l'efficacité ou la précision des requêtes, la taille de la source, la convergence des résultats et le délai correspondant à l'intervalle de temps entre l'envoi d'une requête et le retour des résultats. Lin et al. [99, 100] ont étudié le problème de

sélection des sources dans le Big Data, pour l'intégration d'une grande masse de sources de données. Les auteurs ont proposé une méthode de sélection des sources qui tient compte à la fois l'efficacité et l'efficience. La méthode proposée permet d'obtenir des résultats de bonne qualité à partir d'une petite partie des sources de données sélectionnées et permet de garantir une évolutivité vers des sources de données massives. Ils ont proposé un modèle de couverture probabiliste pour évaluer la qualité de la source de données et ils ont formulé le problème de sélection des sources comme un problème d'optimisation qui maximise la contribution et minimise les coûts. L'algorithme **greedy** est utilisé pour trouver la solution à ce problème. Catania et al. [36] ont considéré la qualité de données dépendant du contexte selon plusieurs dimensions telles que la précision, la fraîcheur et l'exhaustivité pour sélectionner la source de qualité élevée. L'approche proposée est basée sur l'utilisation de graphes nommés imbriqués pour associer des méta-données de qualité à une source de données selon différents contextes et à différents niveaux de granularité. Les auteurs de [16] ont proposé une méthode de sélection des sources basée sur les connaissances pour améliorer la représentation de la source et la métrique de calcule de la similarité source-requête. Les auteurs ont ont pris en compte la structure et les relations entre les mots dans l'index de l'échantillon central pour modéliser la source comme un ensemble d'entités pondérées. Ils ont proposé une mesure de similarité source-requête qui agrège les distances sémantiques en fonction du contexte et de la structure entre les entités de requête et les entités source.

De plus, certaines méthodes de sélection des sources [87, 90] visent à améliorer l'efficacité de la sélection des sources en utilisant des stratégies appropriées telles que des méthodes d'équilibrage de charge (load balancing methods). Ces méthodes accordent plus d'attention à l'efficacité du système. Récemment, les auteurs de [57] proposent un modèle de sélection des sources basé sur le word embedding en utilisant l'approche Word2Vec qui apprend la similarité (syntaxique et sémantique) entre les requêtes actuelles et les requêtes passées. Nous résumons dans Table 1.1 les différentes approches de sélection des sources.

4 Évaluation des méthodes de sélection des sources

La performance d'un système de recherche multi-sources dépend de la performance des trois étapes qui le constituent. Les techniques de description de source sont souvent évaluées sur l'exhaustivité de leurs ensembles de représentation, et les méthodes de sélection des sources et de fusion des résultats peuvent être évaluées sur la qualité des résultats finaux fusionnés. Cependant, l'efficacité de chaque étape dépend également de la performance des étapes précédentes. Par exemple, il n'est pas possible de comparer l'efficacité de différentes méthodes de fusion lorsque les sources sélectionnées ne contiennent pas de documents pertinents [139].

Pour évaluer la performance des méthodes de selection des sources, plusieurs mesures ont été proposées qui sont generalement orientées rappel et précision. Autrement dit, les techniques de sélection des sources sont comparées en fonction du nombre de documents pertinents disponibles dans les sources sélectionnées [142]. Le résultat de la recherche doit contenir moins de bruit en évitant de sélectionner des sources contenant des documents non pertinents. De plus, assurer moins de silence dans le résultat final si des sources contenant des documents pertinents sont sélectionnés. L'objectif est de sélectionner un petit ensemble de sources contenant autant de documents pertinents que possible (rappel élevé), et d'assurer la précision dans la liste finale des documents fusionnés (précision

Table 1.1 – Différentes approches de sélection des sources.

Approches	**Principe**	**Références**
Grand document	Traiter les sources comme un gros sac de mots et les classer en fonction de leur similarité lexicale avec la requête.	CORI [31], GlOSS [68], Taily [6], KL [160]
Petit document	Utiliser un index de documents échantillonnés de chaque source et classer les sources en fonction du classement de leurs documents échantillonnés pour la requête.	ReDDE [142], DTF [55], CRCS [137], SUSHI [152]
Basées sur la classification	Classer les requêtes/sources en catégories/sujets et utiliser l'apprentissage automatique pour prédire la pertinence d'une source pour une requête donnée.	[8, 37, 42, 43, 75, 78, 84]
Autres	Considérer la qualité des documents/sources.	[45, 46, 98–100]
	Utiliser la modélisation des thèmes de source.	[12]
	Considérer la confiance (Trust) des sources.	[13–15]
	Tenir compte du contexte de la requête.	[36]
	S'intéresser aux sources de données dynamiques dont le contenu évolue dans le temps.	[127]
	Utiliser la description sémantique des sources.	[16]
	Considérer l'importance de l'efficacité du système de recherche sélective.	[87, 90]
	Apprendre à partir des requêtes passées en utilisant des techniques de word embedding.	[57]

élevée).

Rappel : La précision de la sélection de la source peut être mesurée à l'aide de la métrique R_k basée sur le rappel [123, 142]. R_k est une mesure du pourcentage global de documents pertinents contenus dans les k premières sources consultées [30].

$$R_k = \frac{\sum_{i=1}^{k} E_i}{\sum_{i=1}^{k} B_i} \tag{1.11}$$

Où E_i est le nombre de documents pertinents de la source i classés selon un algorithme de sélection des sources, tandis que B_i est le nombre de documents pertinents de la source i classés selon un classement parfait (optimal) basé sur la pertinence (c'est-à-dire classer

la source en fonction du nombre réel de documents pertinents qu'elle contient). Pour une valeur fixe de k, par exemple $k = 20$ [12, 137, 142]), une valeur plus grande de R_k indique un meilleur classement.

Précision : La précision P_k est la proportion de sources sélectionnées qui sont pertinentes.

Gravano et al. [67] supposent que toute source s contenant au moins un document correspondant à une requête q est une source appropriée pour cette requête. Ils ont défini $Goodness(1, q, s)$ comme le nombre de documents pertinents dans s, qui présentent une similarité avec la requête q au-dessus d'un seuil donné l. En supposant que k est le nombre de sources à sélectionner, la valeur de précision pour la sélection des sources est calculée par la formule suivante.

$$P_k = \frac{|s \in Top_k(E)/Goodness(l, q, s) > 0|}{|Top_k(E)|} \tag{1.12}$$

Cela représente la fraction des meilleures sources k du classement qui ont un nombre non nul de documents pertinents.

Sogrine et al. [144] a combiné les valeurs de précision P_k et de rappel R_k dans une seule métrique appelée $maxF_k$ comme suit :

$$maxF_k = max_k \frac{2}{\frac{1}{R_k} + \frac{1}{P_k}} \tag{1.13}$$

Les auteurs ont comparé les méthodes de sélection des sources en fonction de leurs valeurs $maxF_k$ pour toutes les valeurs possibles de k.

MSE (Mean square error) : Callan et al. [31] ont mesuré l'erreur quadratique moyenne (MSE) des méthodes de sélection des sources par rapport à une base de référence optimale. Pour une requête q donnée, l'efficacité d'un classement de sélection des sources Ω peut être calculé comme suit :

$$\frac{1}{N_s} . \sum_{i \in S} (O_i - \Omega_i)^2 \tag{1.14}$$

Où,

N_s est le nombre total de sources, Ω_i et O_i représentent les positions de la $i^{ème}$ source respectivement dans les classements d'une méthode de sélection des sources et d'une base de référence optimale. Dans le classement optimal, les sources sont classées en fonction du nombre de documents pertinents qu'elles contiennent. Les classements avec des valeurs MSE faibles sont considérés comme efficaces.

SRCC (Spearman rank correlation coefficient) : Le coefficient de corrélation des rangs de Spearman SRCC est utilisé pour mesurer la qualité des échantillons de sources [30]. Une version simplifiée du coefficient de corrélation des rangs de Spearman a été proposée pour comparer les classements produits par les méthodes de sélection des sources à celui d'une base de référence optimale [54], donnée par la formule suivante :

$$SRCC = 1 - \frac{6 \sum_{i=1}^{N_s} (Q_i - \Omega_i)^2}{N_s(N_s^2 - 1)} \tag{1.15}$$

N_s est le nombre total de sources, Ω_i et O_i sont respectivement les positions de la $i^{ème}$ source dans les classements d'une technique de sélection des sources, et d'une méthode de référence.

5 Conclusion

A travers les travaux de la littérature, nous avons présenté dans ce chapitre les principales catégories d'approches de sélection des sources dans un environnement de recherche multi-sources, à savoir les approches grand-document, approches petit-document et les approches basées sur la classification. La sélection des sources est une fonction essentielle d'un système de recherche multi-sources, dans laquelle le système essaie d'envoyer des requêtes uniquement aux sources qui contiennent (potentiellement)des informations pertinentes afin d'assurer de bonnes performances de recherche en évitant d'interroger des sources qui ne contiennent pas de documents pertinents. La performance d'un système de recherche multi-sources mesure la capacité du système à satisfaire le besoin d'information de l'utilisateur, c'est à dire la pertinence des résultats de la recherche. Cette performance est évaluée selon deux facteurs importants que sont le rappel et la précision. Le rappel mesure la capacité du système à sélectionner tous les documents pertinents et la précision mesure la capacité du système à rejeter tous les documents non pertinents.

Chapitre 2

L'aspect social dans la recherche d'information

Sommaire

1 Introduction

Les systèmes traditionnels de recherche d'information sont basés sur la recherche par mot-clés, qui vise à fournir des ressources pertinentes à l'utilisateur en fonction de son besoin d'information représenté par sa requête seule, généralement composée de quelques mots-clés. Cependant, il a été démontré que les requêtes peuvent être ambiguës car des utilisateurs ayant des besoins différents peuvent utiliser la même requête même s'ils attendent des ressources pertinentes différentes [132]. Les systèmes conventionnels sont incapables de traiter efficacement de telles requêtes, ce qui diminue l'efficacité de la recherche. La recherche d'information personnalisée est une solution au problème cité ci-dessus, elle améliore la recherche sans changer la façon dont les utilisateurs spécifient leurs besoins.

La recherche personnalisée prend en compte non seulement la requête, mais également d'autres informations fournies directement ou non par l'utilisateur [3]. Ces systèmes nécessitent des informations sur les utilisateurs afin d'apprendre et de répondre à leurs besoins réels en information. Chaque système modélise et construit indépendamment le profil de l'utilisateur. Le profil d'utilisateur est une représentation des informations connues sur cette personne, y compris les données démographiques, les intérêts, les préférences, les objectifs et l'historique des recherches passées. Le profil de l'utilisateur est alors utilisé pour étoffer ou reformuler la requête, réordonner les résultats, ou pour recommander un document ou un lien, afin d'améliorer l'efficacité de l'accès à l'information et de satisfaire tous les utilisateurs.

L'explosion du web 2.0 et des réseaux sociaux a crée une source d'information énorme et enrichissante qui a motivé les chercheurs de différents domaines à l'exploiter. Dans le domaine de la recherche d'information, les informations générées par les réseaux sociaux sont utilisées dans la construction d'un profil d'utilisateur enrichi d'une dimension sociale, qui est exploité dans un processus de personnalisation et de recommandation [65]. La recherche d'information sociale est alors devenue un domaine émergent et une voie prometteuse pour la conception et la mise en œuvre d'une nouvelle génération de systèmes de recherche d'information, permettant aux utilisateurs d'obtenir des informations répondant à leurs besoins d'information spécifiques en exploitant les connaissances et les expériences de recherche de la communauté sociale.

2 Recherche d'information sociale

L'avènement des plateformes communautaires sociales en ligne telles que Flickr, del.icio.us, Facebook, LinkedIn, etc. a changé la façon dont les utilisateurs interagissent avec Internet. Alors que la plupart des utilisateurs étaient auparavant de simples consommateurs d'information, ces plateformes offrent aux utilisateurs un moyen simple et facile de publier également leur propre contenu, ce qui permet également aux utilisateurs de produire de l'information [17]. Sur ces plateformes sociales, les utilisateurs sont invités à partager des photos, des vidéos, des opinions, à évaluer le contenu, mais aussi à explorer la communauté en ligne et à rechercher des personnes ayant des profils d'intérêt similaires. En ce sens, les plateformes communautaires sociales en ligne modifient non seulement la façon dont les gens interagissent avec Internet, mais aussi la façon dont les utilisateurs interagissent entre eux en constituant un réseau social.

La grande quantité de données pertinentes générées par les utilisateurs des résaux sociaux conduit à profiter de cette précieuse source de connaissances pour identifier et classer les contenus sur le web (Figure 2.1). Cependant, les systèmes traditionnels de recherche sur le Web ne sont pas efficaces sur les réseaux sociaux, car ils ne prennent pas compte de la composante sociale et se concentrent uniquement sur la qualité du contenu. La recherche d'information sociale [24] est apparue comme un prolongement de la recherche d'information traditionnelle en prenant en compte le profil social de l'utilisateur. Le profil social de l'utilisateur est intégré dans le processus de recherche, par exemple dans la reformulation de la requête et/ou dans le reclassement des documents. Le classement d'un document dépend non seulement de la correspondance entre le document et la requête, mais aussi de la correspondance entre les intérêts de l'utilisateur et le document. La recherche d'information sociale exploite les différentes entités présentes dans les réseaux sociaux (utilisateur, document, etc.) et leurs relations mutuelles afin de personnaliser les résultats de la recherche [34, 154, 167, 168], découvrir des groupes (clusters) ou

des communautés [89, 125, 165] et recommander des items [83, 164].

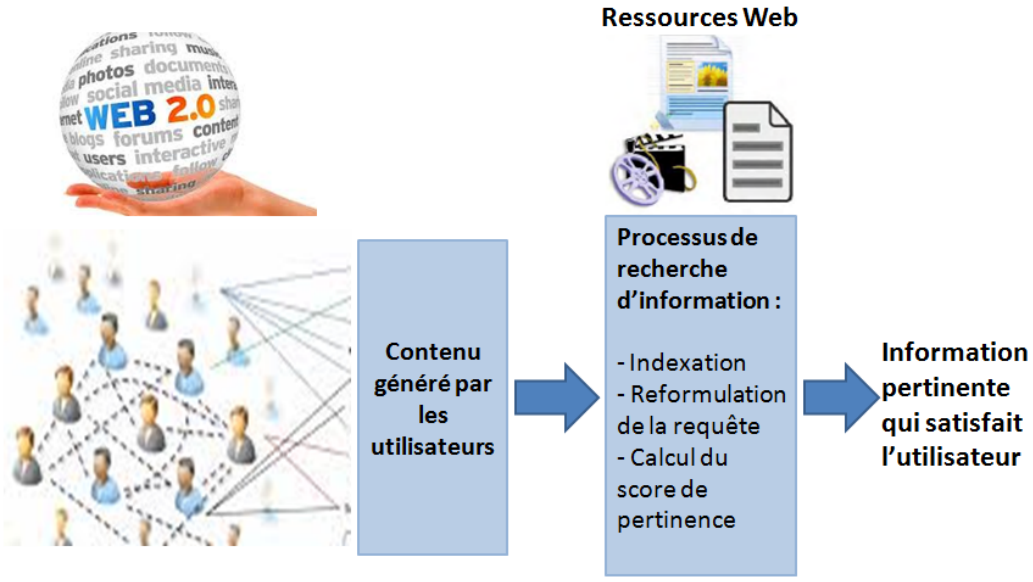


Figure 2.1 – Recherche d'information sociale.

En effet, l'exploitation des informations sociales présente un certain nombre d'avantages pour la recherche d'information [24] :

— Les commentaires (feedbacks) sur les réseaux sociaux sont fournis directement par l'utilisateur, de sorte que des informations précises peuvent être collectées lorsque les utilisateurs expriment leurs opinions sur les plateformes sociales.

— Une quantité considérable d'informations sociales est publiée et disponible avec l'accord des éditeurs. L'exploitation de ces informations ne doit pas violer la vie privée des utilisateurs, en particulier les informations de marquage social, qui ne contiennent pas d'informations sensibles sur les utilisateurs.

— Les ressources sociales sont souvent accessibles, car la plupart des réseaux sociaux fournissent des APIs pour accéder à leurs données (même si souvent un contrat monétisé doit être établi avant toute utilisation à grande échelle).

3 Le modèle utilisateur

3.1 Définition

Un modèle d'utilisateur (aussi appelé "profil d'utilisateur") est une structure de données qui représente les intérêts, les objectifs et les comportements des utilisateurs. Les profils qui peuvent être modifiés ou améliorés sont considérés comme dynamiques, contrairement aux profils statiques qui conservent les mêmes informations au fil du temps. Les profils dynamiques prenant en compte le temps permettent de distinguer les intérêts à court terme des intérêts à long terme. Les profils à court terme représentent les intérêts actuels de l'utilisateur, tandis que les profils à long terme indiquent des intérêts qui ne sont pas soumis à des changements fréquents dans le temps [58].

3.2 Création d'un modèle utilisateur

La Figure 2.2 illustre les principales étapes de la construction de profils d'utilisateurs, à savoir la collecte de données d'utilisateurs à partir de différentes sources d'information, et la construction du profil où les intérêts des utilisateurs seront extraits et représentés à l'aide de différentes méthodes. Le profil utilisateur est ensuite utilisé par différentes applications basées sur la personnalisation, telles que les systèmes de recommandation, les systèmes de recherche d'information et le commerce électronique.

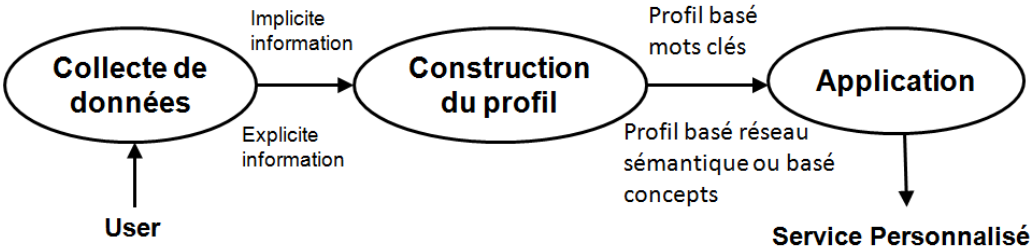


Figure 2.2 – Processus de construction du profil utilisateur [58].

3.2.1 Acquisition et collecte de données utilisateur

Deux techniques principales peuvent être utilisées pour l'acquisition des données et préférences des utilisateurs, à savoir l'approche explicite et l'approche implicite [58, 113].

L'approche explicite : l'utilisateur est invité à communiquer ses données et préférences directement au système, par le biais d'un questionnaire ou d'un texte descriptif. Ces informations comprennent des informations démographiques, telles que le nom, l'âge, l'adresse, la situation familiale, le travail ou les intérêts personnels. En plus des simples cases à cocher et des champs de texte, l'utilisateur peut aussi exprimer son opinion en sélectionnant une valeur dans une plage. La principale limite de cette approche est que l'utilisateur ne souhaite généralement pas participer à de tels formulaires ou évaluations. Ainsi, un tel profil construit peut ne pas fournir d'informations précises [116]. De plus, comme l'utilisateur exprime généralement ces informations lors de la première interaction avec le système, le profil créé est statique. Cependant, il est nécessaire de disposer d'informations à jour sur l'utilisateur afin d'améliorer son expérience de recherche actuelle.

L'approche implicite : consiste à analyser le comportement de l'utilisateur interagissant avec un système, en exploitant l'historique de recherche [145], les journaux de requêtes [70], ou les données des réseaux sociaux [34, 154, 166]. L'avantage de cette approche par rapport à la méthode explicite est qu'aucun effort n'est requis de la part de l'utilisateur.

Selon le domaine d'application, différent types de retours utilisateurs implicites peuvent être analysés :

- Sites web : pages web visitées, temps passé sur une page, date et heure de la visite, etc.
- Sites de commerce et de recommandation : achats, avis et articles recommandés, etc.

— Moteurs de recherche : requêtes, résultats de recherche correspondants, historique de navigation Web, etc.

— Applications sociales (réseaux sociaux, applications de social tagging, blogs, etc.) : annotations, messages, signaux sociaux (like, partager, retweet, favori, etc.)

Des études ont été menées pour trouver la source d'information la plus efficace sur laquelle construire des profils. Ces études montrent qu'il n'y a pas de réponse claire quant à savoir si les profils créés implicitement sont plus ou moins précis que ceux créés explicitement [58]. Étant donné que l'approche implicite impose moins de charge à l'utilisateur et se met à jour automatiquement lorsque l'utilisateur interagit avec le système, cela semble être la méthode préférable pour collecter des informations sur les utilisateurs.

3.2.2 Représentation du profil utilisateur

Il existe généralement deux groupes de techniques de représentation des profils utilisateurs : les modèles basés sur des mots-clés pondérés et les modèles basés sur un réseau sémantique riche [169]. Dans le premier groupe, le profil est représenté par des termes automatiquement extraits des documents et informations fournies par les utilisateurs [39]. Ces mots-clés sont souvent associés à des pondérations pour représenter les intérêts de recherche de l'utilisateur. Ces mots-clés peuvent également être des termes conceptuels/catégoriels tirés d'une sorte de source de connaissances [28]. Dans ce cas, des ressources externes doivent être incluses dans l'ensemble du processus.

Le profil de l'utilisateur peut également être représenté à l'aide d'une structure de réseau sémantique riche [166]. En plus des mots-clés pondérés, ce type de profil utilisateur peut représenter des relations pondérées entre des termes et/ou des concepts pour décrire avec précision les intérêts de l'utilisateur. Dans ce cas, le profil utilisateur utilise des nœuds et des nœuds associés qui capturent les termes et leurs termes sémantiquement associés, respectivement. Des pondérations peuvent être attribuées aux nœuds, aux nœuds associés et aux liens entre eux. Cependant, comparées aux méthodes basées sur des mots clés pondérés, les méthodes basées sur des réseaux sémantiques sont souvent complexes et difficiles à mettre en œuvre.

Selon la représentation souhaitée du profil utilisateur , différentes techniques peuvent être utilisées pour construire le profil utilisateur. Ces techniques sont basées sur l'apprentissage automatique ou sur la recherche d'information.

Les différents types de modèles de profil utilisateur pouvant être créés sont :

Modèle basé sur des vecteurs : le profil est représenté sous la forme d'un vecteur pondéré de mots clés. Les mots-clés sont pondérés avec le schéma de pondération TF-IDF (Term Frequency-Inverse Document Frequency) largement utilisé dans la recherche d'information [129]. Les vecteurs créés de cette manière peuvent être comparés à l'aide d'une mesure de similarité, par exemple, la formule cosinus [129]. L'un des principaux inconvénients des profils basés sur des mots clés est que de nombreux mots ont plusieurs significations. En raison de cette polysémie, les mots clés du profil utilisateur sont ambigus, ce qui rend le profil inexact.

Modèle à base d'un réseau sémantique : afin de résoudre le problème de polysémie inhérent aux profils basés sur des vecteurs de mots-clés, les profils peuvent être représentés par un réseau sémantique pondéré dans lequel chaque nœud représente

un concept, ce qui permet de mettre en évidence les relations sémantiques entre les informations de l'utilisateur.

Modèle basé sur des concepts : les profils basés sur le concept sont similaires au profil basé sur le réseau sémantique dans le sens où les deux sont représentés par des nœuds conceptuels et des relations entre ces nœuds. Cependant, dans les profils basés sur des concepts, les nœuds représentent des sujets abstraits considérés comme intéressants pour l'utilisateur, plutôt que des mots spécifiques ou des ensembles de mots apparentés. Les profils de concept sont également similaires aux profils de mots-clés dans la mesure où ils sont souvent représentés comme des vecteurs de caractéristiques pondérées, mais les caractéristiques représentent des concepts plutôt que des mots ou des ensembles de mots. Divers mécanismes sont appliqués pour exprimer l'intérêt de l'utilisateur pour chaque sujet. La technique la plus simple est une valeur numérique, ou pondération, associée à chaque sujet.

Les concepts sont tirés d'une source de connaissances, telle que Wikipedia [1] , ODP [2] , SUMO [3] , WordNet [4].

4 Accès personnalisé à l'information dans un environnement multi-sources

Les systèmes classiques de recherche d'information sont basés sur la recherche par mot-clé, étant donné un ensemble de ressources et un besoin d'information de l'utilisateur, ces systèmes visent à fournir des ressources pertinentes à l'utilisateur. Le plus souvent, ce besoin d'information est exprimé par une requête composée de quelques mots clés (en général, moins de trois mots). Cependant, il a été démontré que les requêtes peuvent être ambiguës puisque des utilisateurs ayant des besoins différents peuvent utiliser la même requête même s'ils attendent des ressources pertinentes différentes [132].

D'autre part, avec l'explosion continue de la quantité d'informations sur Internet, le problème de la surcharge d'information est devenu de plus en plus grave, ce qui rend difficile pour les utilisateurs d'identifier l'information souhaitée [32].

Par conséquent, la recherche d'information personnalisée est considérée comme l'un des outils les plus importants pour résoudre les problèmes cités ci-dessus. La recherche d'information personnalisée, tenant compte des besoins réels des utilisateurs, peut aider les utilisateurs à obtenir leurs données cibles avec précision et efficacité à partir des informations massives sur Internet, basées sur les mots-clés fournis par les utilisateurs, combinés avec d'autres informations fournies, directement ou non, par l'utilisateur [27].

Le système de recherche d'information personnalisée peut contenir une grande quantité d'informations sur les profils d'intérêts, les préférences, la localisation et les relations sociales des utilisateurs. Ces informations de l'utilisateur sont intégrées dans le processus de recherche d'informations afin d'adapter les résultats de recherche aux profils des utilisateurs, par exemple lors de l'expansion/reformulation de la requête, ou lors de l'indexation et de la classification de documents [27]. L'efficacité du système de recherche personnalisé dépend fortement de la quantité et de la qualité des informations disponibles sur l'utili-

1. http ://www.wikipedia.org/.
2. Open Directory Project : http ://www.dmoz.org.
3. Suggested Upper Merged Ontology : http ://www.ontologyportal.org/.
4. http ://wordnet.princeton.edu/.

sateur. Plus les informations de l'utilisateur sont précises, plus la réponse personnalisée peut être efficace [119].

De nombreuses recherches ont abordé le problème de la personnalisation de la recherche, dans le but de prendre en compte le profil de l'utilisateur dans le processus d'évaluation de la pertinence des requêtes de l'utilisateur afin d'améliorer l'efficacité de l'accès à l'information et de fournir à l'utilisateur des résultats proches de ses intérêts et de ses préférences.

Cependant, dans un environnement de recherche multi-sources, comme il existe de très nombreuses sources d'information distribuées appartenant à des thèmes variés et avec un vocabulaire très riche, la tâche de personnaliser la recherche pour un utilisateur spécifique devient de plus en plus difficile. Dans la section suivante, nous décrivons comment le problème de personnalisation de la recherche multi-sources peut être abordé dans un tel environnement.

4.1 Utilisation du profil utilisateur pour personnaliser la recherche d'information multi-sources

Le profil de l'utilisateur peut être intégré à chacune des trois phases de la recherche multi-sources afin de personnaliser la recherche selon les intérêts et les préférences de l'utilisateur. Dans la phase de description de la source, les informations du profil de l'utilisateur peuvent être utilisées pour construire une représentation de la source. Les termes de requêtes de sonde nécessaires à l'algorithme d'échantillonnage sont extraits des données de profil. Dans ce cas, les documents échantillonnés seront proches des centres d'intérêts de l'utilisateur, ce qui peut influencer la performance de la sélection des sources. Dans la sélection de la source, les données du profil de l'utilisateur peuvent être utilisées pour calculer le degré de correspondance entre la requête de l'utilisateur et les sources consultées par cet utilisateur, afin de générer un score personnalisé pour chaque source qui peut être combiné avec le score non personnalisé pour le calcule finale de la pertinence d'une source par rapport à la requête de l'utilisateur. De cette manière, les préférences de l'utilisateur pour certaines sources par rapport à d'autres peuvent être capturées. Dans la fusion des résultats, l'historique d'utilisation des sources par un utilisateur peut être utilisé pour donner de l'importance aux documents provenant de sources fiables, par exemple en reclassant les documents en fonction de la similitude de leur contenu avec la distribution des termes dans le profil de l'utilisateur.

4.2 Approches de personnalisation de la recherche d'information multi-sources

Il existe toute une littérature sur la personnalisation de la recherche [4, 25, 48, 159, 170], mais peu qui considèrent la recherche dans un environnement multi-sources, peut-être en raison de la difficulté (le coût) de mettre en place un tel système qui gère plusieurs sources d'information distribuées en l'absence d'autorité centrale. Nous citons ci-dessous les plus en rapport avec nos travaux de recherche.

Carman et Crestani [33] ont proposé quelques idées préliminaires sur la recherche fédérée personnalisée. Leur objectif est de personnaliser les trois phases du processus de recherche multi-sources pour d'abord déterminer quelles phases sont les mieux adaptées à la personnalisation. Par exemple, une approche d'échantillonnage personnalisé permet d'échantillonner des documents plus proches des domaines d'intérêt de l'utilisateur.

Lu et Callan [103] ont proposé une approche qui modélise les intérêts des utilisateurs pour améliorer l'efficacité de la recherche fédérée en texte intégral dans les réseaux peer-to-peer. Leur objectif est d'améliorer la qualité de sélection des hubs [5] dans les réseaux Peer-to- Peer afin de lancer la recherche dans le bon voisinage, c'est à dire, avec un rayon de recherche plus petit. Pour cela ils ont modélisé les intérêts à long terme de l'utilisateur en fonction des requêtes précédentes. Ce modèle est utilisé pour guider la sélection des hubs pour les nouvelles requêtes qui représentent des intérêts similaires aux requêtes passées.

Dans [85], les auteurs ont proposé une approche de personnalisation de la recherche d'information dans un environnement distribué basée sur la technologie multi-agents. Leur approche consiste à intégrer à la fois le profil de l'utilisateur et le profil de la source dans le processus de sélection de la source et dans les étapes de fusion des résultats. Le profil d'utilisateur est construit à partir de données spécifiées explicitement par l'utilisateur lui-même, consistant en données personnelles, historique de recherche et préférences. Les résultats des expérimentations ont montré que l'approche proposée améliore la pertinence des résultats, réduit le temps de réponse et améliore l'extensibilité du système.

Dans [133], le profil utilisateur constitué de données sociales est utilisé pour personnaliser et améliorer la recherche d'information distribuée. Une structure de folksonomy entre les trois entités (utilisateur, document, balise) est exploitée pour améliorer l'expansion des requêtes et pour personnaliser la sélection des sources et la fusion des résultats. Les auteurs ont considéré le profil à court terme de l'utilisateur en exploitant la requête actuelle et les relations frères dans le marquage social.

Ces dernières années, les auteurs de [1] ont proposé une approche centrée sur l'utilisateur qui aborde la sélection des sources comme un problème multicritère. Leur approche utilise une méthodologie d'aide à la décision pour permettre aux utilisateurs de formaliser leurs préférences en précisant l'importance relative des différents critères. Les préférences de l'utilisateur sont ensuite utilisées dans un cadre d'optimisation pour trouver les sources les plus appropriées pour l'utilisateur.

5 Évaluation des méthodes de recherche d'information personnalisée

Afin d'évaluer un système de recherche d'information personnalisée, nous avons besoin d'un banc d'essai avec deux propriétés : premièrement, l'ensemble de documents doivent être répartis sur (ou séparés en) plusieurs sources. Deuxièmement, nous avons besoin d'un flux de requêtes d'utilisateurs et de jugements de pertinence personnelle correspondants. Les moteurs de recherche commerciaux stockent les données personnelles des utilisateurs sous la forme de journaux de requêtes et de données de clic [70] qui sont inaccessible au public. Ces données représentent l'historique de recherche des utilisateurs qui peut être utilisé pour construire des jugements de pertinence personnelle [35] nécessaires à l'évaluation des systèmes de recherche d'information personnalisés. Chaque clic sur une URL d'un document est considéré comme un vote implicite d'un utilisateur confirmant la pertinence de ce document pour sa requête. En absence ou l'indisponibilité de ces données qui sont privées et personnelles, la communauté de la recherche d'information a cherché d'exploiter d'autres données qui sont accessibles au public, et qui peuvent remplacer les données de clics et les journaux de requêtes. Les informations contenues dans les systèmes de réseaux

5. Hub : service d'annuaire, qui se charge d'acheminer les requêtes et les réponses entre les consommateurs et les fournisseurs

sociaux ont été utilisées pour évaluer les systèmes de recherche d'information personnalisée et pour améliorer les résultats de recherche [34]. Les services de Crowdsourcing (par exemple, Amazon Mechanical turk , Crowdflower) se sont révélés être un outil important pour optimiser les ressources nécessaires à l'évaluation des systèmes de recherche d'information [155]. Les notes, représentées par les degrés de préférence des résultats de recherche ou par le classement des résultats selon un ensemble prédéfini de catégories ou de thèmes, fournies par les utilisateurs des services de Crowdsourcing sont utilisées pour mesurer la précision des méthodes de personnalisation par rapport aux intérêts réels des utilisateurs [155].

5.1 Métriques d'évaluation basées sur des ensembles

En règle générale, les mesures d'évaluation des systèmes de recherche d'information nécessitent une collection de documents et une requête, alors que chaque document est soit pertinent, soit non pertinent pour une requête particulière.

Selon la Figure 2.3, soit P le sous-ensemble de documents pertinents par rapport à une requête q, et R l'ensemble de documents récupérés. En utilisant ces fonctionnalités, nous introduisons ci-dessous les métriques Précision, Rappel et F-Mesure.

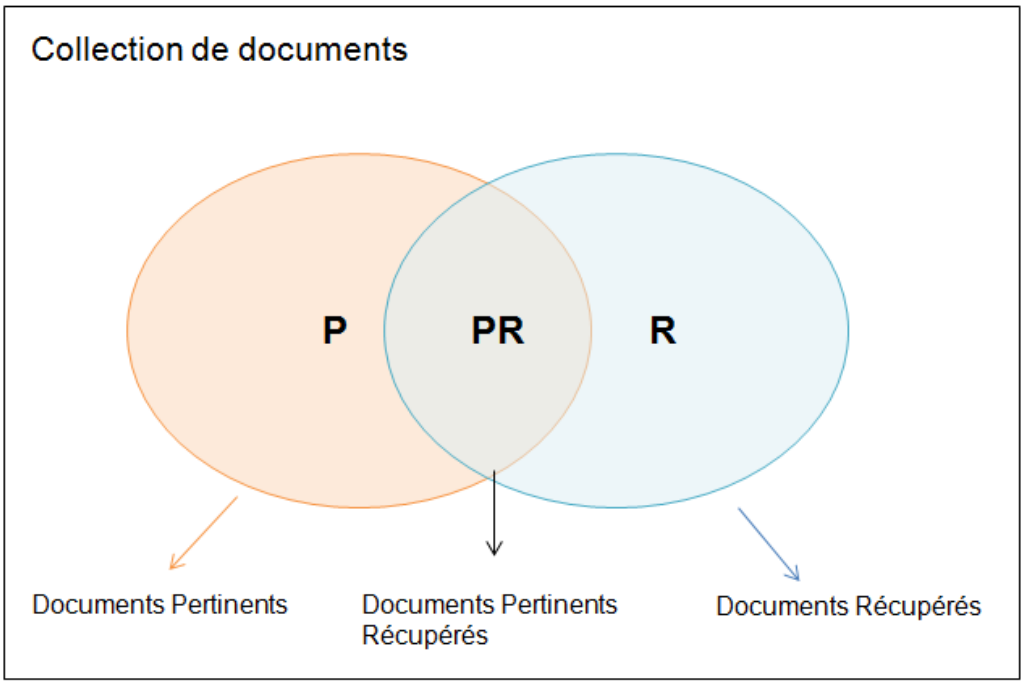


Figure 2.3 – Relation entre les documents pertinents et les documents récupérés.

— **Précision :** La précision est la partie des documents récupérés qui sont pertinents pour le besoin d'information de l'utilisateur. En fait, cela montre la capacité d'un système à sélectionner tous les documents pertinents dans la collection. Elle est donnée par le nombre de documents récupérés pertinents divisé par le nombre total de documents récupérés :

$$Précision = \frac{|PR|}{|R|} \tag{2.1}$$

— **Rappel :** Le rappel dans la recherche d'information est la partie des documents récupérés avec succès qui sont pertinents pour la requête. En effet, elle met l'accent sur la capacité d'un système à ne sélectionner que les documents pertinents. Il est donné par le ratio de documents récupérés pertinents divisé par le nombre de documents pertinents pour la requête :

$$Rappel = \frac{|PR|}{|P|} \tag{2.2}$$

— **F-Mesure :** Habituellement, il y a un compromis entre la précision et le rappel, c'est-à-dire que plus le rappel est élevé, plus la précision a tendance à être faible. Ainsi, un système de recherche d'information se distingue par le rapport de la précision au rappel appelé F-Mesure ou F-Score, estimé par la formule suivante.

$$F - mesure = \frac{2.Précision.Rappel}{Précision + Rappel} \tag{2.3}$$

5.2 Métriques d'évaluation basées sur le classement

Plusieurs mesures de classement ont été proposées dans le domaine de la recherche d'information afin d'évaluer les systèmes de recherche basés sur le classement. Nous présentons celles couramment utilisées, à savoir MAP, NDCG et MRR.

— **MAP@r :** La précision moyenne MAP (Mean Average Precision) au rang r mesure, pour une requête d'évaluation q_i, la moyenne sur les valeurs de précision calculées à chaque point du classement où un document pertinent apparaît.

En utilisant un ensemble de requêtes d'évaluation Q, la précision moyenne est estimée par l'expression suivante :

$$MAP = \frac{1}{Q} \sum_{q_i \in Q} \frac{1}{r} \sum_{r \in R} Précision(q_i)@R \tag{2.4}$$

Où Q est le nombre de requêtes, r représente le nombre de documents pertinents pour une requête q_i et R est le rang d'un document pertinent.

— **NDCG@r :** Le NDCG (Normalized Discounted Cumulative Gain) au rang r évalue la capacité d'un système a retourner des documents pertinents par degré de pertinence [81]. Le DCG (Discounted Cumulative Gain) est une mesure qui donne plus de poids aux mieux classés documents et permet l'incorporation de différents niveaux de pertinence. Le DCG est mesuré pour chaque requête q_j à la $n^{ème}$ position :

$$DCG_j^n = rel_1 + \sum_{i=2}^{n} \frac{rel_i}{\log_2 i} \tag{2.5}$$

Où rel_i est une fonction de pertinence affectée au $i^{ème}$ document. Le NDCG peut être estimé à partir du DCG appliqué au classement parfait des jugements de pertinence selon leur degré, noté $IDCG_j^n$:

$$NDCG@n = \frac{\sum_{q_j \in Q} DCG_j^n}{\sum_{q_j \in Q} IDCG_j^n} \tag{2.6}$$

— **MRR@r :** Le rang inverse moyen MRR (Mean Reciprocal Rank) au rang r favorise l'hypothèse que les documents pertinents doivent être retournés en premier dans la liste d'ordonnancement par rapport aux r premiers documents retournés. Ainsi, cette métrique estime le rang moyen $Rang(l_h)@r$ du premier document pertinent dans les listes de résultats $l_i \in L$ de r documents retournés en réponse à la requête q_i :

$$MRR = \frac{1}{|Q|} \sum_{q_i \in Q} \sum_{l_i \in L} \frac{1}{Rang(l_i)@r} \tag{2.7}$$

6 Conclusion

Nous avons abordé dans ce chapitre l'intégration de l'aspect social dans la recherche d'information afin d'aider l'utilisateur à obtenir des informations qui répondent à ses besoins réels d'information. Les réseaux sociaux sont devenus une source d'information importante permettant de créer des modèles d'utilisateurs précis nécessaires à la personnalisation de l'accès à l'information.

Pour la recherche multi-sources, la personnalisation de l'accès à l'information reste un enjeu majeur, compte tenu du nombre prohibitif de sources d'information disponibles sur Internet et de la diversité des informations susceptibles d'intéresser les différents profils d'utilisateurs.

Notons enfin que l'évaluation des performances des méthodes de personnalisation de la recherche d'information est confrontée au problème de l'acquisition de jugements personnels pour évaluer la pertinence des résultats fournis par ces méthodes.

Chapitre 3

L'aspect intelligence dans la recherche d'information

Sommaire

1 Introduction

Le calcul évolutionnaire (evolutionary computation, en anglais) [11] est l'un des domaines de l'intelligence artificielle (ou, plus précisément, de l'intelligence computationnelle) qui a connu une croissance fulgurante ces dernières années. Le calcul évolutionnaire est basé sur l'utilisation de modèles de processus évolutionnaires pour la conception et la mise en œuvre de systèmes informatiques de résolution de problèmes. Il s'applique principalement aux problèmes d'optimisation, il s'inspire des mécanismes biologiques tels que la reproduction, la mutation, la recombinaison, la sélection naturelle et le comportement animal collectif. Les différents modèles proposés dans cette philosophie sont appelés d'une manière générique algorithmes évolutionnaires (evolutionary algorithms, en anglais)[11].

Divers algorithmes évolutionnaires ont été proposés et étudiés, principalement les algorithmes génétiques (en anglais, genetic algorithms) [62], les stratégies d'évolution (en anglais, evolution strategies) [60], la programmation génétique (en anglais, genetic programming) [88] et la programmation évolutionnaire (en anglais, evolutionary programming) [53].

L'application des algorithmes génétiques a beaucoup attiré l'attention de la communauté de la recherche d'information. Abu Kausar et al. [2] ont montré que les métaheuristiques telles que les algorithmes génétiques peuvent être des méthodes appropriées pour résoudre de nombreux problèmes liés à la recherche d'information. En effet, plusieurs travaux de recherche ont prouvé que l'utilisation de ces algorithmes améliore l'efficacité de la recherche, selon différents domaines de leur application.

Dans ce chapitre, nous présentons d'abord une brève description des algorithmes génétiques, puis nous donnons un aperçu de leur application pour résoudre divers problèmes dans le domaine de la recherche d'information.

2 Les algorithmes génétiques

2.1 Définition

Les algorithmes génétiques (AGs), inventés par John Holland [73], puis développés par David Goldberg [62], sont une variante d'algorithmes évolutionnaires. Ces techniques sont basées sur le principe de "**la survie du meilleur**", qui modélisent des phénomènes naturels liés à la génétique darwinienne [44]. Les algorithmes génétiques sont considérés comme des algorithmes de recherche et d'optimisation robustes et efficaces [49], largement utilisé et très réussis pour trouver des solutions optimales à de nombreux problèmes difficiles en s'appuyant sur des opérateurs bio-inspirés tels que la mutation, le croisement et la sélection.

2.2 Fonctionnement d'un algorithme génétique

Le fonctionnement d'un algorithme génétique est extrêmement simple. On part d'une population de solutions initiales (représentées par des chromosomes ou des individus) choisie aléatoirement. La performance de chaque individu de la population est évaluée à l'aide d'une fonction d'évaluation spécifique appelée fonction de fitness. Sur la base de ces performances, une nouvelle population de solutions est créée à l'aide d'opérateurs évolutionnaires simples : sélection, croisement et mutation. On répète ce cycle jusqu'à trouver une solution satisfaisante [59] (Figure 3.1).

L'algorithme 1 décrit les principales étapes d'un algorithme génétique.

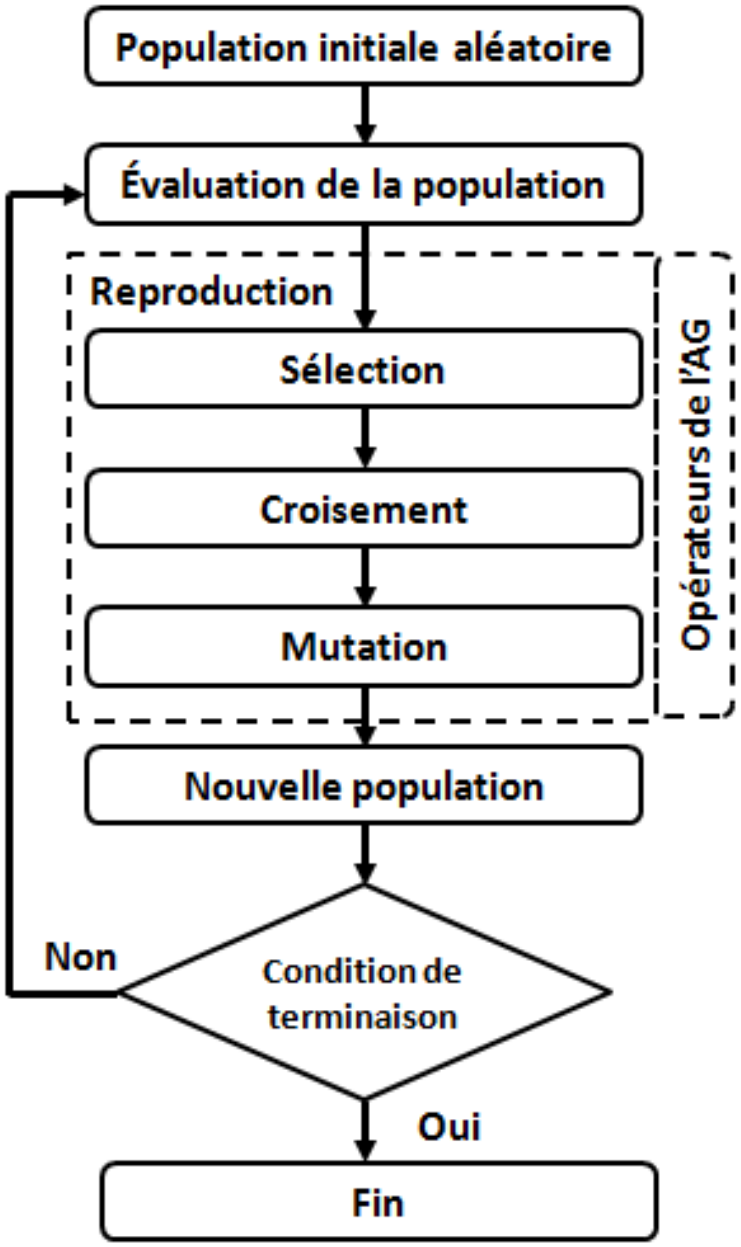


Figure 3.1 – Structure générale d'un algorithme génétique.

Algorithm 1 Algorithme Génétique

1: Générer une population initiale d'individus de taille fixe
2: Évaluation de la population courante : calcule les chances de survie (ou de fitness) de chaque individu dans la population
3: Reproduction : Appliquer les opérateurs génétiques (sélection, croisement et mutation)
4: Selon les probabilités associées à chaque opérateur de croisement et de mutation, appliquez ces opérateurs
5: Placer les individus produits dans la nouvelle population
6: Remplacer l'ancienne population d'individus par la nouvelle (favoriser les meilleurs individus)
7: Vérifier si le critère de terminaison est atteint. Si oui, terminer, sinon retourner à l'étape 2

Les principaux éléments sont à considérer lors de la conception d'algorithmes génétiques, à savoir le codage des paramètres, la fonction d'évaluation, les opérateurs génétiques et d'autres paramètres nécessaires à l'exécution de l'algorithme génétique.

2.2.1 Le codage

L'application d'un algorithme génétique à un problème commence par le codage. Le codage spécifie une mise en correspondance qui transforme une solution possible du problème en une structure contenant un ensemble de variables de décision pertinentes pour le problème à résoudre. Une solution particulière au problème peut ensuite être représentée par une affectation spécifique de valeurs aux variables de décision. L'ensemble de toutes les solutions possibles est appelé l'espace de recherche et une solution particulière représente un point dans cet espace de recherche. En pratique, ces structures peuvent être représentées sous différentes formes, notamment des chaînes, des arbres et des graphiques. Il existe également une variété de valeurs possibles qui peuvent être attribuées aux variables de décision, y compris les valeurs binaires, k-aires, réelles et de permutation.

Traditionnellement, les algorithmes génétiques utilisent principalement des structures de chaînes contenant des variables de décision binaires (Figure 3.2). La structure qui encode une solution s'appelle un chromosome ou un individu. Une variable de décision est appelée un gène et sa valeur est appelée un allèle.

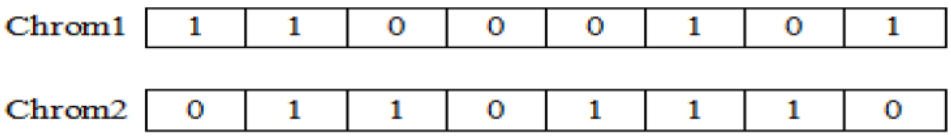


Figure 3.2 – Technique de codage binaire.

2.2.2 La fonction d'évaluation (ou d'adaptation)

La fonction d'adaptation, ou de *fitness*, associe une valeur numérique (habituellement réelle)à chaque individu. Cette valeur a pour but d'évaluer si un individu est mieux adapté qu'un autre à son environnement. Ce qui signifie qu'elle quantifie la réponse apportée au problème pour une solution potentielle donnée. Ainsi les individus peuvent être comparés entre eux.

2.2.3 Les opérateurs génétiques

ce sont les opérateurs de reproduction permettant l'évolution des individus de la génération actuelle à la suivante. Les algorithmes génétiques exploitent principalement trois types d'opérateurs visant chacun un objectif spécifique relatif à la couverture de l'espace des solutions. Ces opérateurs sont la sélection, le croisement et la mutation. Alors que l'opérateur de sélection favorise la propagation des meilleurs chromosomes, les opérateurs de croisement et de mutation se préoccupent de favoriser l'exploration de nouvelles régions de l'espace de recherche qui constitue les solutions possibles.

L'opérateur de sélection. L'opérateur de sélection mime le processus de la sélection naturelle, c'est-à-dire que les individus les mieux adaptés ont tendance à se reproduire plus fréquemment.

Il existe différents mécanismes pour mettre en œuvre cet opérateur, appartenant essentiellement à quatre types de méthodes de sélection :

- Méthode de "loterie biaisée" (roulette wheel) de Goldberg [62],
- La méthode "élitiste",
- Sélection par tournois,
- Sélection universelle stochastique.

L'opérateur de croisement. C'est un opérateur de combinaison qui agit généralement par paires en déterminant un ou plusieurs points de coupure, délimitant les frontières des parties à échanger. La paire d'individus d'origine s'appelle les parents et la paire d'individus résultante s'appelle les enfants. Cet opérateur est un processus appliqué aléatoirement avec une forte probabilité. La Figure 3.3 illustre un opérateur de croisement couramment utilisé : *le croisement en un point.*

La littérature définit plusieurs opérateurs de croisement. Ils différent selon le type de codage adapté et la nature du problème traité.

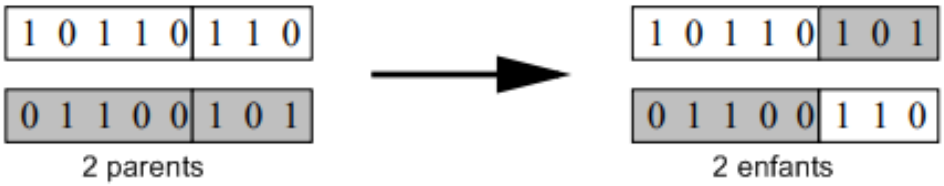


Figure 3.3 – Croisement en un point de deux individus (point de coupure à la 5^{ieme} position).

L'opérateur de mutation. La mutation modifie au hasard la valeur du gène d'un individu. Pour un codage binaire des individus, cela revient à changer un 0 en un 1 ou inversement. La mutation est généralement utilisée avec une très faible probabilité. Elle garantit que la diversité n'est jamais perdue, quelle que soit la position du gène. La Figure 3.4 montre un exemple de mutation.

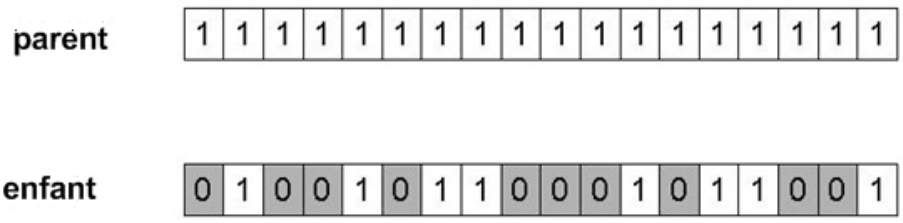


Figure 3.4 – Mutation d'un individu (10 gènes sont modifiés).

2.2.4 Autres paramètres

Les opérateurs de l'algorithme génétique sont guidés par un certain nombre de paramètres fixés à l'avance. La valeur de ces paramètres influe sur la réussite ou non d'un algorithme génétique. Ces paramètres sont :

— **La taille de la population, N.** Si N est trop grand le temps de calcul de l'algorithme peut être très long, et si N est trop petit, il peut converger trop rapidement sur le mauvais chromosome.

— **La probabilité de croisement $\mathbf{p_c}$.** Cela dépend de la forme de la fonction de fitness. Son choix est généralement heuristique. Plus elle est élevée, plus les changements subis par la population sont importants. Les valeurs généralement acceptées se situent entre $0,5$ et $0,9$.

— **La probabilité de mutation $\mathbf{p_m}$.** Ce taux est généralement faible car un taux élevé risque de conduire à une solution sous-optimale.

— **Le nombre maximum de génération.** Généralement, un algorithme génétique se termine après un certain nombre de générations ou une fois une convergence satisfaisante est atteinte. Il est également possible de terminer l'exécution de l'algorithme lorsqu'une certaine condition est atteinte, par exemple lorsque la qualité (fitness) d'un individu dépasse un certain seuil.

3 Application des algorithmes génétiques à la recherche d'information

Depuis plusieurs années, il y avait un intérêt croissant pour l'application des algorithmes génétiques dans différents domaines de la recherche d'information [56]. Les algorithmes génétiques ont été utilisés avec succès pour l'indexation des documents [64], l'extension ou la reformulation de la requête utiliateur [110, 134], l'adaptation des fonctions d'appariement [19], l'optimisation des paramètres de recherche [56], l'optimisation de l'exploration Web [38, 50, 114, 138], etc. Drias et. al. [47] ont montré, à travers des expérimentations faites pour deux algorithmes génétiques proposés, que les techniques de recherche heuristiques surpassent les approches traditionnelles en termes de qualité et de temps d'exécution pour la recherche d'information à grande échelle.

Nous présentons dans ce qui suite les différents domaines de la recherche d'information où les algorithmes génétiques ont été utilisés avec succès.

3.1 Description des documents et indexation

Gordon [64] a proposé l'utilisation des algorithmes génétiques pour déduire la description d'un document. Les documents sont représentés par un ensemble de mots-clés. Il a choisi un schéma de codage binaire où chaque description est un vecteur binaire de longueur fixe qui évolue dans le temps par la sélection naturelle et les opérateurs génétiques. La population génétique est composée de différentes descriptions pour un même document. La fonction de fitness est basée sur le calcul de la similarité entre la description courante du document et chacune des requêtes. Le résultat final est la meilleure chaine décrivant le document.

3.2 La description de la requête

Ce groupe est le plus étendu des applications des AGs dans la recherche d'information. Il consiste à modifier requête précédente (ajout et suppression des termes ou modification des poids des termes existants de la requête) en tenant compte des jugements de pertinence des documents récupérés par cette requête.

Lourdes Araujo et Joaquin Perez-Iglesias [7] ont développé un classificateur pour l'expansion des requêtes courtes ou non spécifiques. Dans leur travail, les jugements de pertinence de l'utilisateur sur un ensemble de documents sont utilisés comme fonction de fitness. La requête est prolongée par un ensemble de termes appropriés tirés à partir des meilleurs documents de la liste de classement initiale.

Le travail proposé par Sathya and Simon [134] consiste à récupérer les documents pertinents en utilisant la meilleure combinaison de la liste de termes (mots clés). Les auteurs ont utilisé le robot d'exploration de documents pour collecter et extraire des informations à partir de documents accessibles à partir de bases de données en ligne et d'autres bases de données. L'algorithme génétique est ensuite utilisé pour générer la meilleure combinaison de termes, qui est ensuite utilisée dans un système de recherche d'information. les auteurs ont montré que le système de recherche proposé est plus efficace dans un domaine spécifique.

Al Mashagba et al. [110] ont mené une étude comparative de différentes approches basées sur les algorithmes génétiques pour l'optimisation de la requête de l'utilisateur pour la recherche d'information basée sur le modèle vectoriel. Différentes stratégies (fonctions de fitness, mesures de similarité, opérateurs de mutation et de croisement) ont été comparées sur une collection de données en langue arabe. Les résultats des expérimentations ont montré que la meilleure approche est celle utilisant l'opérateur de croisement en un point, la mutation en un point et la similarité basée sur le produit scalaire.

Dans [76], les auteurs ont présenté un modèle hybride GA-Particle Swarm Optimization (HGAPSO), qui combine l'algorithme génétique avec la méthode d'optimisation par essaims de particules pour l'optimisation des requêtes dans la recherche d'information sur le Web. Les mots clés sont utilisés pour générer de nouveaux mots-clés liés à la recherche de l'utilisateur.

3.3 Adaptation de la fonction d'appariement

L'objectif de l'apprentissage de la fonction d'appariement est d'utiliser un algorithme génétique pour générer une mesure de similarité qui améliore la performance de recherche d'un système de recherche d'information. Ceci constitue une nouvelle philosophie de retour de pertinence puisque ce sont les fonctions de matching qui sont adaptées à la place des requêtes.

Dans [121], les auteurs ont proposé une nouvelle fonction d'appariement pondérée, qui est la combinaison linéaire des différentes fonctions de similarité existantes (Dice, Jaccard, Cosinus, etc.). Un sous-ensemble de documents de pertinence connue a été utilisé comme entrée dans l'algorithme génétique pour trouver la meilleure combinaison de poids pour la fonction d'appariement finale.

Dans [19], l'algorithme génétique a été utilisé pour trouver un ensemble optimal de poids des composants de la mesure de similarité. Cette dernière combine les différentes mesures de similarité standard qui sont utilisées pour la classification des documents.

Dans [80], Les auteurs ont proposé une mesure de similarité sensible à la requête (Query-Sensitive Similarity Measure), qui consiste à mesurer la similarité de deux documents pour une requête donnée. Cette mesure permet l'utilisation simultanée du produit et de la somme pondérée pour fusionner les informations provenant des sources identifiées au préalable. Un algorithme génétique est utilisé pour apprendre les valeurs optimales des paramètres de cette mesure pour une collection spécifique.

3.4 Optimisation des paramètres de recherche

Certaines études ont optimisé les paramètres indépendamment des modèles de recherche, permettant à l'ensemble de paramètres optimisés d'illustrer les caractéristiques des collections de test.

Fan et al. [51, 52] ont proposé une approche basée sur la programmation génétique pour générer automatiquement des stratégies de pondération des termes pour différents contextes à des fins de classification des documents. L'algorithme proposé utilise la rétroaction de la pertinence de l'utilisateur. Les auteurs ont montré que chaque contexte spécifique nécessite une stratégie de pondération des termes différente. Le nouveau cadre proposé a été testé sur des données TREC [1] et les résultats ont été très prometteurs.

Dans [56], un algorithme génétique a été utilisé pour estimer et optimiser les paramètres de recherche pour la recherche en langue japonaise. L'approche proposée optimise les paramètres de retour (feedback) et les paramètres de base de notation (scoring) indépendamment du modèle de recherche. Quatre collections de test ont été utilisées pour valider la proposition.

3.5 Construire des robots d'exploration

L'algorithme génétique est utilisé pour optimiser l'exploration du Web et sélectionner les pages Web les plus appropriées à récupérer par le robot d'indexation [38, 50, 114, 138].

Dans [38], Chen a comparé un crawler basé sur le meilleur-premier avec un autre basé sur l'algorithme génétique. Sur la base de la liste initiale des pages d'accueil fournie par l'utilisateur, le robot utilisant ces deux méthodes parcourt Internet et rapporte les résultats correspondant aux intérêts de l'utilisateur. Les expérimentations ont montré que le robot basé sur l'algorithme génétique atteint la valeur de rappel la plus élevé, tandis que les valeurs de précision des deux robots sont presque égales.

Dans [138], les mots clés initiaux utilisés dans l'exploration du web sont prolongés par des termes pertinents générés par l'algorithme génétique.

Dans [114], Nhan et al. ont proposé un system d'indexation des pages web vietnamiennes basé sur l'algorithme génétique pour estimer le meilleur chemin à suivre. Partant d'un ensemble initial de mots-clés, le robot élargit l'ensemble de mots-clés en ajoutant les termes les plus appropriés qui sont intelligemment sélectionnés lors du processus d'exploration par l'algorithme génétique.

Dans [50], les auteurs ont proposé une nouvelle stratégie d'exploration du web basée sur l'algorithme génétique et une technique de niche. Leur stratégie considère à la fois le contenu des pages Web visitées ainsi que la structure des liens hypertextes des pages explorées. L'approche proposée utilise des hyperliens (URL) en tant qu'individus génétiques et le modèle d'espace vectoriel pour évaluer la fitness de ces individus. La technique de niche est utilisée pour éliminer les URL non pertinents pour des thèmes prédéfinis.

3.6 Amélioration du profil utilisateur

La génération de profil de l'utilisateur peut être considéré comme un problème d'optimisation. Les algorithmes évolutionnaires peuvent être considérés comme une approche appropriée pour construire le profil de l'utilisateur.

1. http ://trec.nist.gov/

Dans la recherche d'images [136], les auteurs ont proposé un système de requête adaptatif composé de deux composants principaux, à savoir un système de requête personnalisé (en ligne) et un mécanisme d'apprentissage basé sur l'algorithme génétique (hors ligne). L'algorithme génétique est utilisé pour améliorer la précision du profil de l'utilisateur grâce à la rétroaction de l'utilisateur. Les chromosomes représentent les profils possibles d'un utilisateur. La fonction de fitness utilise les commentaires des utilisateurs pour la pertinence. Ces commentaires ne sont fournis qu'au stade initial de l'évolution. Le meilleur chromosome est décodé dans un profil utilisateur pour remplacer le profil actuel dans la base de données. Les résultats expérimentaux ont indiqué que la précision de la recherche est augmentée de façon significative par le mécanisme d'apprentissage basé sur l'algorithme génétique.

Dans [105], un système de recherche d'information adaptatif basé sur le paradigme multi-agent est proposé. La modélisation de l'utilisateur pour la recherche d'information est réalisée grâce à un algorithme génétique permettant de faire évoluer et adapter les vecteurs requêtes qui sont des modèles représentatifs des besoins d'information de l'utilisateur. Le profil utilisateur (chromosome) est représenté par un vecteur de mots-clés et leurs poids associés. La fonction de fitness est basée sur la combinaison de deux mesures de pertinence, à savoir la mesure quantitative (algorithmique) et la mesure qualitative (subjective). La mesure quantitative est donnée par la similarité entre le chromosome du modèle de l'utilisateur et les documents récupérés. Chaque document récupéré fait l'objet d'une évaluation qualitative, de manière interactive avec l'utilisateur. Les résultats ont montré que l'apprentissage par renforcement interactif améliore les performances du système proposé par rapport à la rétroaction de pertinence traditionnelle, et ce avec l'utilisation d'une approche évolutive pour la modélisation des utilisateurs.

Dans le cadre du filtrage de l'information, Bouchachia et al. [26] ont proposé une approche évolutionnaire pour générer et adapter des profils d'utilisateurs. L'algorithme proposé est incrémental et met à jour le profil à mesure que de nouveaux commentaires deviennent disponibles. Afin d'améliorer la précision du système de filtrage, les auteurs ont proposé une stratégie multi-profils qui génère plusieurs profils distincts pour un même utilisateur, chacun correspond à un thème (sport, politique, science). Un individu de la population est considéré comme un profil possible. Un individu ou un chromosome est constitué d'un ensemble de gènes, chaque gène représente un terme et le poids qui lui est associé.

4 Conclusion

L'aspect intelligence est un facteur important qui a été considéré dans le domaine de la recherche d'information depuis plusieurs années. Les techniques d'algorithmes évolutionnaires ont suscité une attention considérable en raison du potentiel qu'elles offrent pour résoudre des problèmes complexes. Ces techniques basées sur le principe puissant de "la survie des meilleurs", qui modélisent les phénomènes naturels liés à la génétique darwinienne. Elles constituent une catégorie intéressante d'heuristiques modernes de recherche et d'optimisation.

L'application des méthodes intelligentes, telles que les algorithmes génétiques, a saisi beaucoup l'attention de la communauté de la recherche d'information pour résoudre divers problèmes d'optimisation et de recherche. En effet plusieurs travaux de recherche ont prouvé que l'utilisation de ces algorithmes améliore l'efficacité de la recherche en fonction de leurs applications, comme pour la construction des indexes des documents, l'expan-

sion de la requête utilisateur, pour l'optimisation des paramètres de recherche ou pour l'exploration du Web.

Partie II : Approches Adoptées pour la Recherche d'Information dans un Environnement Multi-Sources

Chapitre 4

Approche intelligente pour la sélection des sources d'information

Sommaire

1 Introduction

Une phase importante dans la recherche d'information multi-sources est la phase de sélection des sources d'information qui sont généralement réparties géographiquement. Pour une requête utilisateur donnée, les sources susceptibles de contenir les informations pertinentes sont sélectionnées pour être interrogées. Les résultats de chacune de ces sources

sont fusionnés et envoyés à l'utilisateur. Avec le nombre important de sources qui existent, il est important de sélectionner un nombre optimal de sources les plus pertinentes pour satisfaire au mieux l'utilisateur.

Dans ce chapitre, nous abordons le problème de la sélection des sources dans ce contexte, où le nombre de sources d'information est très élevé. Étant donné que la dimension de l'espace de recherche, définie par le nombre de sources d'information disponibles, est élevée, le chalenge est de trouver l'approche qui peut explorer cet espace de recherche de manière intelligente afin de trouver la solution quasi-optimale, ou à défaut une solution approximative de bonne qualité. L'objectif est d'optimiser la sélection des sources d'information afin de restreindre l'envoi de la requête à un nombre limité de sources susceptibles de contenir l'information désirée par l'utilisateur. L'approche que nous proposons dans ce travail consiste à utiliser une méthode intelligente pour répondre au problème d'optimisation combinatoire que nous définirons ci-dessous.

De nos jours, il existe une pléthore d'outils intelligents pour faire face à ce type de problème, y compris les algorithmes nature et bio-inspirés. Premièrement, nous abordons la question avec des algorithmes génétiques, qui ont été les plus répandus dans le monde, peut-être en raison de leur conception basée sur l'évolution naturelle des espèces biologiques, mais aussi pour leur simplicité de mise en œuvre. Dans un second temps, nous envisageons d'utiliser des algorithmes beaucoup plus récents. Il faut savoir que la modélisation de notre problème avec un algorithme évolutionnaire est identique pour tous les autres algorithmes évolutionnaires. La différence résidera dans la simulation du comportement des agents naturels impliqués dans l'algorithme. Ainsi, le codage de la solution, l'espace de recherche (espace des solutions potentielles) ainsi que la fonction d'adaptation sont les mêmes pour toute cette catégorie d'algorithmes.

Nous considérons que les algorithmes évolutionnaires [11] sont appropriés pour le problème de sélection des sources dans la recherche d'information multi-sources pour les raisons suivantes [19] :

— Le nombre de sources d'information qui ne cessent d'augmenter de jour en jour rend la sélection d'un ensemble de sources susceptibles de contenir l'information pertinente plus complexe et difficile à traiter par des méthodes analytiques.

— Le problème de sélection des sources peut être considéré comme un problème de recherche et d'optimisation dont l'objectif est de trouver pour une requête donnée un ensemble de sources pertinentes parmi un grand nombre de sources disponibles de manière optimale.

— Dans un espace de recherche élevé, il est important d'explorer et d'exploiter toutes les directions de l'espace de recherche pour trouver la bonne source. Des opérateurs comme le croisement et la mutation effectuent de telles opérations.

2 Une approche basée sur un algorithme génétique pour la sélection des sources d'information

Nous formalisons le problème de sélection des sources comme un problème de recherche et d'optimisation qui consiste à rechercher la solution quasi-optimale dans un espace de recherche prohibitif ; donc plusieurs solutions possibles. Pour répondre à ce problème d'optimisation, nous avons utilisé des algorithmes génétiques.

Dans les sous-sections suivantes, nous présenterons notre définition du problème de sélection des sources ainsi que le principe et les différentes étapes de l'algorithme génétique proposé.

2.1 Définition du problème

Nous formalisons le problème de sélection des sources comme un problème d'optimisation combinatoire que nous définissons comme suit :

Donnée : un ensemble $S = \{s1, s2, \ldots, sn\}$ de sources d'information et une requête q de l'utilisateur.

Question : déterminer un sous ensemble S' de S telle que la similarité entre ses éléments et q est maximale, où $|S'| = k < |S| = n$

Une solution au problème défini est une sélection des sources, donc un ensemble de sources. On recherche la sélection qui maximise la similarité requête-sélection, calculée entre la description de la requête de l'utilisateur et la description des sources qui composent une sélection. A notre connaissance, aucun travail de la littérature n'a abordé la problématique de la sélection des sources d'information de la manière formelle que nous proposons.

2.2 L'espace de recherche

Nous allons adopter une technique d'intelligence artificielle pour trouver une solution au problème de sélection des sources d'information dans un environnement multi-sources où l'espace de recherche est prohibitif. Ce dernier regroupe un nombre de solutions égal au nombre de combinaisons possibles égal à :

$$C_n^k = \binom{n}{k} = \frac{n!}{k!(n-k)!}$$

Quand n est très grand, le nombre de combinaisons possibles est faramineux. C'est la raison pour laquelle, on a recours à des techniques d'intelligence artificielle. Les algorithmes génétiques sont bien adaptés aux problèmes de grande complexité comme les problèmes d'optimisation combinatoire [23, 149], plusieurs approches ont utilisé des algorithmes génétiques pour trouver la meilleure combinaison à partir d'un très large espace de solutions potentielles [121, 134].

2.3 Algorithme génétique pour la sélection des sources

La Figure 4.1 montre l'approche proposée pour la sélection des sources basée sur un algorithme génétique. Pour une requête utilisateur donnée, l'algorithme proposé génère la solution quasi-optimale de sources à sélectionner parmi un ensemble de solutions potentielles.

L'algorithme génétique est initié par un ensemble de k-combinaisons possibles de sources représentant les solutions possibles au problème. Chaque chromosome ou individu qui représente une solution potentielle est évalué par la fonction de fitness. Les opérateurs génétiques (sélection, croisement et mutation) sont utilisés pour générer une nouvelle population à partir de la population actuelle. Une fois une nouvelle génération

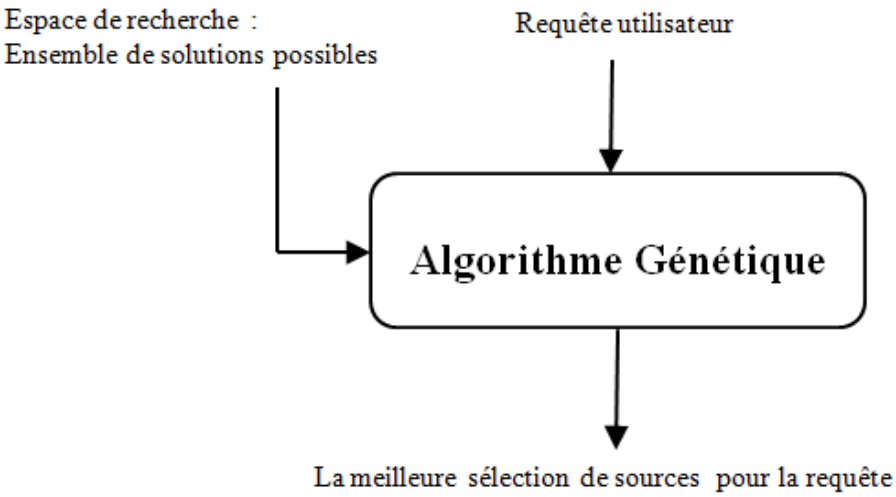


Figure 4.1 – L'approche de sélection des sources.

est créée, le processus génétique est répété de manière itérative jusqu'à ce qu'une solution optimale au problème soit trouvée.

2.4 Le codage de la solution

Une solution au problème défini précédemment est une sélection des sources composée d'un ensemble de k sources. Une façon de représenter une solution est donc un vecteur de longueur k contenant des sources d'information (un vecteur de sources). Ces dernières seront codées par des nombres entiers pour simplifier leur manipulation. Ainsi une source s_i est comprise entre 1 et n et une solution possible est un vecteur de k entiers entre 1 et n.

La Figure 4.2 représente le schéma de codage des chromosomes utilisé par l'algorithme génétique pour la résolution du problème de sélection des sources, de sorte que chaque solution engendré par ce chromosome représente une sélection possible de k sources. Cet encodage doit répondre à certaines conditions suivantes :

— L'ordre des gènes d'un chromosome n'est pas important (individus similaires).

— La valeur de chaque gène correspond à une source d'information (un entier compris entre 1 et n).

— La valeur d'un gène spécifique ne doit apparaître qu'une seule fois dans un chromosome (pour éviter les gènes en double).

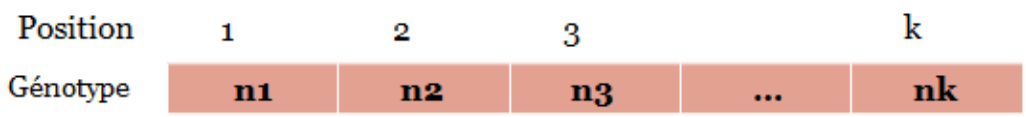


Figure 4.2 – Représentation des solutions.

Exemple :

Si le nombre de sources d'information est égal à 5 et que le nombre de sources à sélectionner est égal à 3, une solution peut être :

$\{1, 4, 5\}$ ou bien

$\{2, 3, 5\}$
$\{3, 4, 5\}$ ...

2.5 La fonction de fitness

La fonction de fitness (ou d'adaptation) est une fonction de mesure de la performance qui évalue la qualité de chaque solution (chromosome). Le choix de la fonction de fitness est crucial pour le bon fonctionnement de l'algorithme et dépend du problème. Dans notre cas, la fonction de fitness est la similarité entre la requête utilisateur q et une solution de l'espace de recherche appelée *sol* et qui est un ensemble de sources d'information. Cette similarité est calculée par la Formule 4.1.

Pour calculer $Similarity(sol, q)$, nous considérons *sol* comme une collection de documents représentant les sources et nous calculons la similarité entre q et la collection *sol*.

$$Similarité(sol, q) = \frac{\sum_{h \in sol} Similarité(h, q)}{k} \tag{4.1}$$

$Similarité(sol, q)$: la similarité entre la requête q et la solution *sol*.
$Similarité(h, q)$: la similarité entre la requête q et la source h de la solution.
k : nombre de sources dans la sélection.

La similarité entre une requête utilisateur et une source d'information peut être calculée par la mesure **cosinus** du modèle de recherche vectorielle [130]. La source d'information est considérée comme un ensemble de termes. Nous représentons la requête et la source par des vecteurs de poids de termes dans un espace de dimensions m correspondant aux termes présents dans l'espace de recherche (termes d'index). Cette similarité est obtenue en calculant le cosinus entre le vecteur de la source et le vecteur de la requête.

Ainsi la similarité entre une source h et une requête q est donnée comme suit :

$$Similarité(s_h, q) = \frac{\sum_{j=1,m}(t_{hj} * t_{qj})}{\sqrt{\sum_{j=1,m}(t_{hj})^2 * \sum_{j=1,m}(t_{qj})^2}} \tag{4.2}$$

t_{hj} et t_{qj} sont les poids du terme j dans la source h et la requête q respectivement, calculés avec la formule TF-IDF.

Le poids t_{hj} du terme j dans la source h est calculé comme suit.

$$t_{hj} = tf_{hj} * idf_j \tag{4.3}$$

Tel que :
tf_{hj} : la fréquence du terme j dans la source h,

Le poids t_{qj} du terme j dans la requête q est calculé comme suit.

$$t_{qj} = tf_{qj} * idf_j \tag{4.4}$$

Tel que :
tf_{qj} : la fréquence du terme j dans la requête q,
idf_j : la fréquence inverse de document du terme j (nombre de documents qui contiennent le terme j), calculée par la formule de base suivante :

$$idf_j = \log(\frac{n}{df_j}) \tag{4.5}$$

Où n est le nombre total de sources et df_j est le nombre de sources dans l'espace de recherche où le terme j apparaît.

2.5.1 Normalisation de tf

Le nombre tf est généralement normalisé pour éviter un biais vers des sources plus longues (qui peuvent avoir un nombre de terme plus élevé, quelle que soit l'importance réelle de ce terme dans la source) pour donner une mesure de l'importance du terme j dans la source particulière h.

$$tf_{hj} = \frac{tf_{hj}}{\max(tf_h)} \tag{4.6}$$

tf_h : la fréquence maximale des termes dans la source h.

2.5.2 Normalisation de idf

Inverse Document Frequency idf estime la rareté d'un terme dans l'ensemble de l'espace de recherche. Elle est calculée par la Formule 4.5 dans le cas où le nombre de sources où apparaît le terme j, $df_j \neq 0$. Si le terme n'est pas trouvé dans l'espace de recherche, cela conduira à une division par zéro. Il est donc courant d'utiliser $1 + df_j$ La Formule 4.5 devient :

$$idf_j = \log(\frac{n}{1 + df_j}) \tag{4.7}$$

2.6 L'algorithme génétique proposé

Nous avons proposé un algorithme génétique pour la sélection des sources d'information, appelé GASS (Genetic Algorithm for Source Selection). L'algorithme 2 décrit le fonctionnement de GASS.

Dans ce qui suit, nous décrivons les différents composants de l'algorithme.

2.6.1 La population initiale

Le processus d'évolution commence par une population initiale de taille $Taille_{Pop}$ générée aléatoirement à partir de toutes les combinaisons possibles. Elle se compose de $Taille_{Pop}$ chromosomes ; chacun dénote une solution au problème qui est représentée par un vecteur de k sources codées par des entiers de 1 à n.

Lors de la génération de la population, les mêmes sources sont évitées dans le même chromosome (gènes en double). Par exemple dans le chromosome {2,5,3,2}, le chiffre 2 est répété, une telle construction de chromosomes est à éviter. Il faut aussi éviter de répéter le même chromosome dans la population (chromosomes dupliqués), par exemple {3,4,5,8} est le même que {8,3,5,4} car l'ordre n'est pas important.

Algorithm 2 GASS : Genetic Algorithm for Source Selection

Input: un ensemble de n sources et une requête utilisateur q
Output: la sélection quasi-optimale de k sources ($sol_{optimale}$) pour la requête q

1: Générer aléatoirement une population initiale de taille $Taille_{pop}$ à partir des k-combinaisons possibles de sources ($Taille_{pop}$ chromosomes)
2: Évaluer chaque solution dans la population initiale à l'aide de la fonction de fitness donnée par l'équation 4.1
3: **while** (non fin de la reproduction) {Appliquer les opérateurs génétiques à la population pour créer de nouveaux individus ou chromosomes} **do**
4: Sélectionner les chromosomes appropriés pour la reproduction (parents)
5: Appliquer l'opérateur de croisement sur les parents selon une probabilité de croisement pour produire de nouveaux chromosomes (descendants)
6: Appliquer l'opérateur de mutation sur les chromosomes selon une probabilité de mutation
7: Ajouter les chromosomes nouvellement constitués à la population
8: **end while**
9: Évaluer la population actuelle en utilisant la fonction de fitness (l'équation 4.1)
10: Sélectionner la nouvelle population pour la prochaine génération (de taille $Taille_{pop}$)
11: Vérifier les critères de terminaison (nombre maximum d'itérations atteint) ; si les critères ne sont pas satisfaits, retourner à (3)

2.6.2 Opérateurs génétiques

Des opérateurs génétiques (sélection, croisement et mutation) sont appliqués sur les vecteurs de sources (les sélections).

La sélection. L'opérateur de sélection simule *la survie du plus apte*. Il existe différents mécanismes pour mettre en œuvre cet opérateur, et l'idée est de privilégier les meilleurs chromosomes. Nous avons utilisé la sélection naturelle qui prend les meilleurs chromosomes à la génération suivante. Les meilleurs chromosomes sont identifiés en évaluant leur valeur de fitness.

Le croisement. C'est un opérateur génétique qui combine deux chromosomes ensemble pour former une nouvelle progéniture. Cela ne se produit qu'avec une probabilité de croisement P_c. Les chromosomes qui ne sont pas croisés restent inchangés. L'intuition derrière le croisement est l'exploration de nouvelles solutions et l'exploitation d'anciennes solutions. Nous utilisons le croisement à point unique sans "doublons". Ainsi, les valeurs des gènes dans le chromosome généré ne doivent pas être répétées. L'algorithme 3 présente l'algorithme de croisement proposé.

Algorithm 3 Algorithme de Croisement

Input: Soient Y = (y_1, y_2, ... y_k) et X = (x_1, x_2, ..., x_k) deux chromosomes à croiser.
Output: Y' = (y'_1, y'_2, ... y'_k) et X' = (x'_1, x'_2, ..., x'_k) deux chromosomes croisés.

1: Choisir un nombre aléatoire r dans l'ensemble {0, 1, 2 ... k-1},
deux nouveaux chromosomes X' and Y' sont créés selon la règle suivante :

$$x'_i = \begin{cases} x_i & \text{si } i < r \\ y_i & \text{sinon} \end{cases} \qquad y'_i = \begin{cases} y_i & \text{si } i < r \\ x_i & \text{sinon} \end{cases}$$

2: Supprimer, avant le point de coupe (r), les sources qui sont déjà placées après le point de coupe
3: Identifier les sources qui n'apparaissent pas dans chacun des deux chromosomes
4: Remplir au hasard les trous de chaque chromosome

Exemple :

On considère $n = 9$ et $k = 6$.

$S = \{1, 2, 3, 4, 5, 6, 7, 8, 9\}$, S' sous-ensemble de S de longueur 6. La Figure 4.3 montre un exemple de croisement de deux chromosomes.

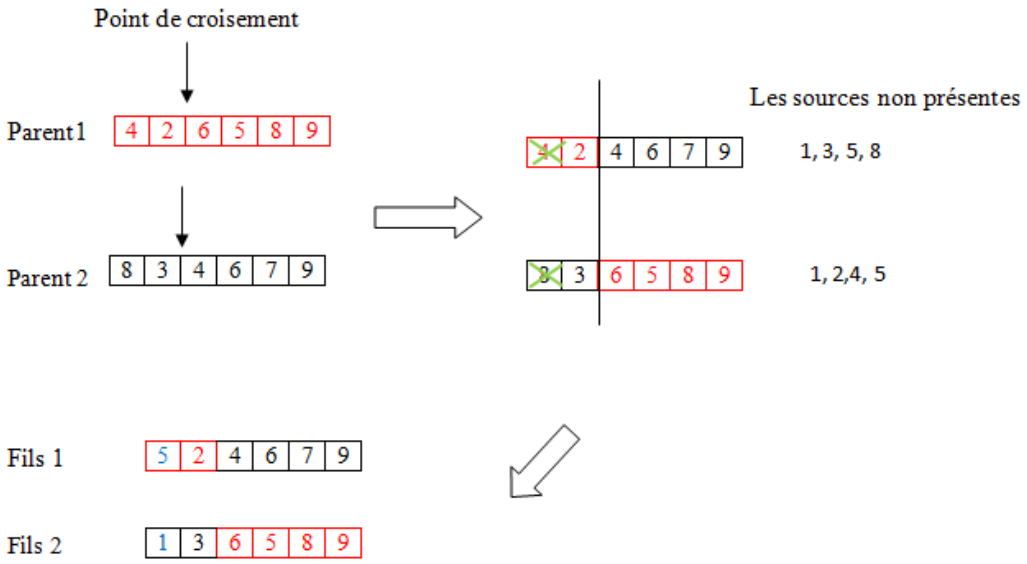


Figure 4.3 – Le croisement selon un point.

L'application de l'opérateur de croisement sur le parent 1 et le parent 2 permet d'obtenir de nouveaux individus (nouvelle progéniture) et donc de nouvelles sélections possibles de sources.

La mutation. La mutation est le processus de modification aléatoire des gènes d'un chromosome particulier. La mutation consiste à modifier les valeurs génétiques d'une solution avec une certaine probabilité P_m. L'objectif de la mutation est de restaurer les données perdues et d'explorer une variété de données. Nous avons utilisé une mutation en un seul point. Un gène est modifié avec une certaine probabilité par un nombre aléatoire généré dans l'intervalle $[1, n]$, tout en évitant la duplication des gènes. L'opérateur de mutation est décrit par l'algorithme 4.

Algorithm 4 Algorithme de Mutation

Input: La population actuelle
Output: La nouvelle population après l'opération de mutation

for chaque chromosome dans la population actuelle **do**
  Générer un nombre aléatoire r sur l'intervalle [0 1] {pour sélectionner le chromosome à muter}
  if $(r < p_m)$ { appliquer l'opérateur de mutation à ce chromosome} **then**
    Sélectionner le gène à modifier (générer un nombre aléatoire i entre 0 et $k-1$)
    Choisir la nouvelle valeur à placer {générer un nombre aléatoire v entre 1 et n ; v doit être différent des valeurs déjà existantes dans le chromosome sinon refaire la génération}
    Remplacer la valeur du gène i par la valeur v
    Insérer le nouveau chromosome dans la nouvelle population
  else
    insérer le chromosome dans la nouvelle population {le chromosome est inséré dans la nouvelle population sans changement}
  end if
end for

La Figure 4.4 montre un exemple d'un point de mutation dans lequel un seul gène est modifié.

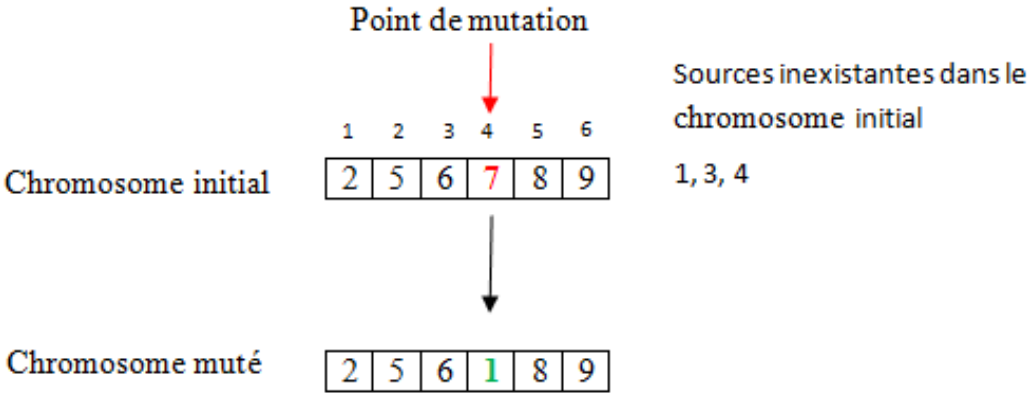


Figure 4.4 – 1- point de mutation (sur le $4^{ème}$ gène).

2.6.3 La condition de terminaison

Le processus de génération est répété jusqu'à ce qu'une condition de terminaison soit satisfaite. Les conditions de terminaison les plus courantes sont : une solution satisfaisant aux critères minimaux est trouvée ou un nombre fixe de générations est atteint. Dans notre cas, la condition de terminaison sera le nombre maximum d'itérations ou de générations à atteindre. Sachant qu'un nombre de générations plus élevé pour tous les opérateurs augmentera le temps de calcul, le nombre optimal de générations lorsque l'algorithme converge peut être défini empiriquement.

La solution optimale au problème sera celle ayant la valeur maximale de la fonction de similarité 4.1 c'est-à-dire celle ayant la meilleure valeur de la fonction de fitness de la population de la dernière génération.

3 Expérimentations

Toutes les expérimentations sont réalisées sur une machine ALFATRON configurée comme suit : Processeur Intel Core i5, CPU = 3.10 GHz, RAM =4 Go et système Windows 7. Nous avons implémenté l'algorithme génétique proposé en utilisant un environnement java et la bibliothèque d'algorithmes génétiques java JGAP [1]. Dans cette section, nous décrivons les données et les mesures utilisées pour l'évaluation de la performance de notre approche.

3.1 Bases de données de test

Pour évaluer notre approche de sélection des sources d'information, nous avons utilisé des bases de données de documents de recherche scientifique couvrant différents domaines (informatique, médecine, juridique, etc.). Ces sources d'information sont considérées comme des données du web invisible car elles fournissent leurs propres interfaces de recherche. La recherche de documents ou d'articles de recherche scientifique sur ces bases de données s'effectue sur une interface mise à disposition par l'éditeur. Certains documents sont disponibles gratuitement et d'autres nécessitent un abonnement auprès du fournisseur afin de pouvoir télécharger les documents souhaités.

L'accès aux ressources de ces bibliothèques peut être assuré par le portail SNDL [2] du Centre de Recherche sur l'Information Scientifique et Technique CERIST [3]. SNDL permet à tous les utilisateurs étudiants, enseignant/chercheurs d'accéder aux documents nécessaires à leur recherche via un compte utilisateur fourni par la plateforme.

Nous avons sélectionné 10 bases de données dont l'accès à ces sources était possible via un compte SNDL. Ces bases de données sont décrites dans Table 4.1.

3.2 Construction des descriptions de sources de test

La méthode d'échantillonnage basé sur les requêtes (Query-Based Sampling) [30] est utilisée pour construire la description des sources d'information utilisées. Des requêtes de sonde composées chacune d'un seul terme sont envoyées à chacune des sources. Les requêtes sont choisies en fonction des domaines auxquels appartiennent les sources. Pour chaque requête (dans un ensemble de 15 requêtes) les 4 premiers documents sont téléchargés à partir de chaque source. Les documents téléchargés à partir d'une source sont utilisés pour représenter cette source.

Nous avons utilisé Indri [4], un système d'indexation et de recherche d'information gratuit et open-source, pour indexer ces sources et rechercher les documents. La liste des mots vides (data/stoplist.dft [5]) de la norme INQUERY [5] est utilisée dans l'indexation des sources. Cette liste contient 418 mots très fréquents.

Le fichier d'index est filtré pour garder uniquement les termes les plus significatifs.

Le choix des termes d'indexation est basé sur la fréquence des termes dans les documents (nombre de documents dans lesquels le terme apparaît), les paramètres classiquement utilisés par salton et al. [131], où les termes dont la fréquence dans les documents

1. JGAP est un cadre pour les algorithmes génétiques et de la programmation génétique écrit en Java (http ://jgap.sourceforge.net/)
2. https ://www.sndl.cerist.dz/
3. https ://www.cerist.dz/
4. http ://www.lemurproject.org/indri
5. https ://github.com/fedorn/lemur/blob/master/data/stoplist.dft

Table 4.1 – Les sources d'information de test.

Numéro de source	**Source**	**Domaines**
1	ACM Digital Library	Informatique
2	ClinicalKey	Médicine
3	Edward Elgar Products	L'économie, les finances, les affaires et la gestion, le droit et la politique publique
4	IEEE, Institute of Electrical and Electronics Engineers	Informatique, Électronique, Télécommunication
5	IOP science Extra of IOP Publishing	Physique, Sciences des matériaux, Mathématiques Appliquées
6	JSTOR	Multidisciplinaire
7	Royal Society of Chemistry	Chimie, Sciences des matériaux, environnement, Biologie
8	ScienceDirect of Elsevier	Multidisciplinaire
9	SpringerLink	Multidisciplinaire
10	SpringerProtocols	Sciences et technique, Sciences de la vie et de la terre

est comprise entre N/100 et N/10 ; N est le nombre de documents dans le corpus. Dans notre cas, nous avons retenu les termes dont la fréquence de document est comprise entre 5 et 300.

Les vecteurs de sources et de requêtes sont construits en utilisant l'approche $tf - idf$ [129] en remplaçant tf (la fréquence du terme dans un document) par df et idf (fréquence inverse de document) par icf. Le poids d'un terme t dans une requête q ou dans une collection c (ou une source), noté $Weight_t(q/c)$ est défini comme suit :

$$Weight_t(q/c) = df(t) * icf(t) \tag{4.8}$$

Où,

$df(t)$ est la fréquence de document dans la source où le terme t apparait (pour calculer le vecteur de source). Pour le calcule du vecteur de requête, cette fréquence est calculée par la proportion du nombre d'occurrence du terme dans la requête par rapport au nombre total de termes de la requête.

$icf(t)$ est la fréquence inverse de collection ou de source, peut être calculée comme suit :

$$icf(t) = log\frac{N}{cf(t)} \tag{4.9}$$

Où,

N est le nombre total des sources, et $cf(t)$ est le nombre de sources où le terme t apparait.

3.3 Paramètres de l'algorithme génétique

Nous avons fixé les valeurs indiquées dans Table 4.2 des paramètres de l'algorithme génétique proposé.

Table 4.2 – Configuration des paramètres de l'algorithme génétique

Paramètre	**Valeur**
Taille de la population	50
Taux de croisement	60%
Taux de mutation	10%
Nombre de générations	1500

3.4 Mesures d'évaluation

Deux facteurs sont généralement utilisés pour évaluer les systèmes de recherche d'information, qui sont la précision et le rappel. Le rappel mesure la capacité d'un système à sélectionner tous les documents pertinents de la collection. Et la précision mesure la capacité d'un système à ne sélectionner que les documents pertinents. Le principal problème de la mesure de rappel est la nécessite de connaitre tous les documents pertinents pour une requête.

Ces mesures peuvent être utilisées dans un environnement de test qui inclut des jugements de pertinences. Cependant, leur utilisation devient difficile dans un environnement avec de nombreuses sources d'information et aucun jugement de pertinence ; l'effort manuel impliqué dans l'évaluation de la pertinence de documents devient irréalisable.

En raison du manque de jugements de pertinence concernant le nombre total de documents pertinents dans l'ensemble de données utilisé dans les expérimentations, nous avons utilisé la mesure précision pour évaluer les performances de l'algorithme de sélection des sources proposé. La mesure de précision calcule la proportion de sources pertinentes par rapport aux sources sélectionnées, donnée par la formule suivante :

$$Précision_k = \frac{|Sources\ pertinentes\ sélectionnées|}{|Sources\ sélectionnées(k)|} \tag{4.10}$$

tel que $|Sources pertinentes sélectionnées|$ est le nombre de sources pertinentes parmi k sources sélectionnées.

Pour déterminer les sources pertinentes, des requêtes de test sont envoyées aux source d'information et nous comptons uniquement les documents pertinents retournés par chaque source. Nous analysons les Top T documents de chaque source, ainsi la source est marquée pertinente si elle contient un certain nombre de documents pertinents au-dessus d'un seuil τ.

Nous avons considéré les 20 premiers documents retournés par chaque source ($T = 20$). Nous avons demandé aux utilisateurs de juger de la pertinence des documents retournés. Une source est marquée pertinente si elle renvoie au moins 3 documents pertinents pour la requête (τ =3). La moyenne de la précision peut alors être calculée sur l'ensemble des requêtes de test.

3.5 Méthodes utilisées pour la comparaison

Nous avons choisi de comparer notre approche de sélection des sources basée sur un algorithme génétique (GASS) avec CORI [29, 31]. CORI fait partie des algorithmes de sélection des sources pionniers qui est largement utilisé pour la sélection des sources. De plus, CORI est considéré comme une méthode de sélection des sources robuste et efficace [54], qui a fait l'objet de comparaison dans plusieurs études, notamment : [6, 8, 14, 85, 118, 133, 142, 152]. Les paramètres par défaut de l'algorithme CORI sont utilisés.

La méthode CORI consiste à considérer chaque collection comme un document unique. Ce document gigantesque est la concaténation de tous les documents de la collection en question.

Dans CORI, la pertinence de la collection pour le terme t, est calculée par la formule suivante.

$$P(t/c) = \Phi + (1 + \Phi) * T * I \tag{4.11}$$

T et I sont calculés par les formules suivantes.

$$T = \frac{df_{t,c}}{df_{t,i} + 50 + 150 * \frac{cw_c}{avg_{cw}}} \tag{4.12}$$

$$I = \frac{log(\frac{N_c+0.5}{cf_t})}{log(N_c + 1.0)} \tag{4.13}$$

$df_{t,c}$: le nombre de documents dans la collection c qui contiennent le terme t
cf_t : le nombre de collections qui contiennent t
N_c : le nombre total de collections disponibles
cw_c : le nombre total de mots dans la collection c
avg_{cw} est la moyenne cw de toutes les collections
Φ : Le composant de croyance minimum lorsque t est disponible dans c, qui est généralement fixé à $0,4$ (valeur par défaut)

La croyance $P(q/c)$ est utilisée par l'algorithme CORI pour classer les collections. Le calcul de $P(q/c)$ consiste à utiliser la valeur moyenne des croyances de tous les termes de la requête.

3.6 Résultats des expérimentations

Afin d'évaluer les performances de l'algorithme de sélection des sources proposé et de le comparer à CORI, nous avons utilisé 20 requêtes de test qui sont présentées dans Table A.1 en annexe. Nous avons choisi des requêtes générales qui renvoient des résultats et nous évitons les requêtes qui ne renvoient aucune réponse.

La Table 4.3 montre un exemple de requêtes de test et les sources pertinentes associées à chacune.

Nous avons fait varier le nombre de sources à sélectionner ($k = 2, 4, 6, 8, 9$) pour calculer la précision moyenne atteinte pour chaque algorithme de sélection des sources, à savoir GASS et CORI. Nous rapportons dans Table 4.4 la précision moyenne obtenue par les deux algorithmes sur les 20 requêtes de test.

Les résultats obtenus indiquent que l'approche GASS est plus efficace que CORI en termes de précision moyenne, en particulier lorsque $k = 4, 6$ et 8 (Figure 4.5). Cela confirme que GASS est plus efficace pour la sélection des sources d'information les plus pertinentes et les plus appropriées pour des requêtes spécifiques.

Table 4.3 – Exemple de requêtes de test avec des jugements de pertinence.

Numéro de requête	**Requête**	**Sources pertinentes**
q1	Java programming language initiation	1,6,8,10
q2	mammography women risk factor breast cancer	1,2,5,6,7,8,9,10
q3	protein absorption	2,6,8,9,10
q4	aids HIV-1	1,2,3,4,5,6,7,8,9
q5	corporatism economy	3, 6, 8,9

Table 4.4 – La précision moyenne des deux algorithmes (GASS et CORI).

Sources séléctionnées Algorithme	**k=2**	**k=4**	**k=6**	**k=8**	**k=9**
GASS	0.875	**0.875**	**0.79165**	**0.733**	0.68345
CORI	0.95	0.8375	0.76665	0.7125	0.68345

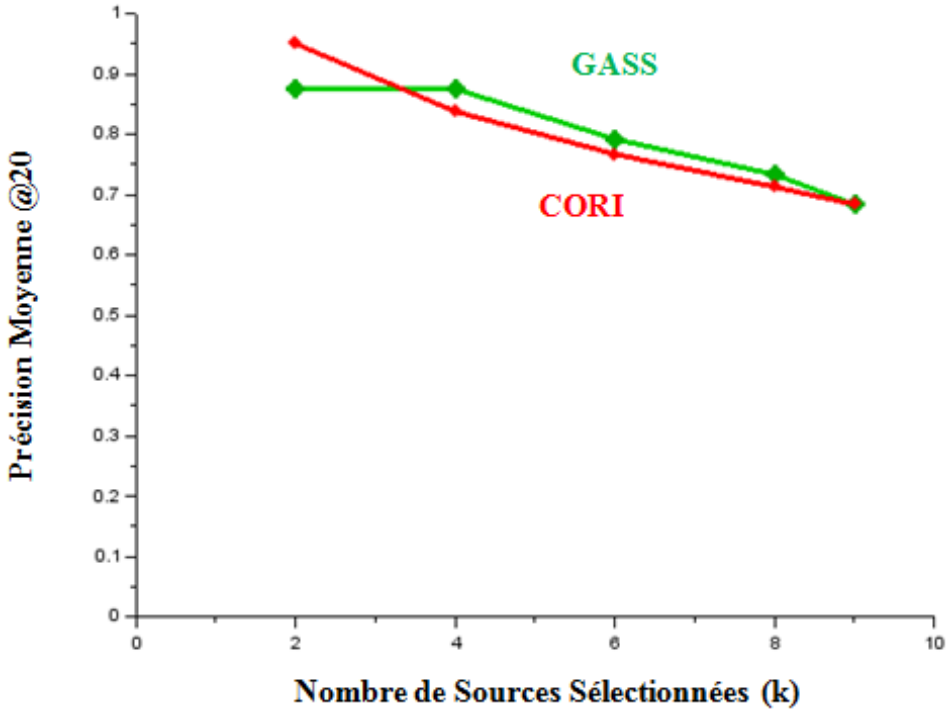


Figure 4.5 – Performances des deux méthodes de sélection des sources en calculant la précision moyenne sur 20 requêtes.

On peut conclure que l'approche basée sur l'algorithme génétique peut fournir des solutions efficaces aux problèmes de sélection des sources dans des environnements multi-sources en offrant une bonne précision par rapport à l'algorithme de sélection des sources

CORI de l'état de l'art.

4 Conclusion

Nous avons montré dans ce travail comment les méthodes bio-inspirées et plus spécifiquement les algorithmes génétiques peuvent apporter des solutions au problème de sélection des sources dans un environnement multi-sources. Tout d'abord, nous avons utilisé un algorithme génétique pratiquement simple pour trouver les sources optimales pour les requêtes des utilisateurs. les premières expérimentations montrent une amélioration des performances en termes de précision de l'algorithme proposé par rapport à l'algorithme CORI. Ceci affirme que cette approche peut être efficace dans la recherche d'information multi-sources.

Malgré ces résultats encourageants, beaucoup de travail reste à faire pour améliorer les performances de l'algorithme génétique proposé. Il serait intéressant d'opter pour un autre codage chromosomique plus efficace (par exemple un codage binaire), d'améliorer la population initiale en incorporant certains chromosomes générés par des règles de priorité, et d'utiliser des opérateurs de recombinaison améliorés pour obtenir une convergence plus rapide vers des solutions de qualité.

Il serait également intéressant d'augmenter le nombre de sources d'information pour tester les performances de l'algorithme proposé pour la recherche d'information à grande échelle.

Chapitre 5

Approche intelligente enrichie de données sociales pour améliorer la sélection des sources

Sommaire

1 Introduction

Dans les plates-formes collaboratives et sociales d'aujourd'hui, les utilisateurs peuvent souvent jouer un rôle actif dans la génération de contenu et l'annotation des ressources grâce à des balises qui constituent collectivement la folksonomie [111]. C'est un moyen d'organiser les ressources pour une meilleure navigation, un filtrage ou une recherche future. Les utilisateurs annotent généralement les éléments qu'ils jugent pertinents, de sorte que les balises qu'ils fournissent peuvent être considérées comme une description de leurs intérêts et de leurs besoins. De plus, on peut supposer que plus la balise est souvent utilisée, plus cette balise sera importante pour l'utilisateur. De même, les balises attribuées aux éléments décrivent généralement leur contenu. Plus les utilisateurs annotent un élément avec une balise spécifique, mieux cette balise décrit le contenu de l'élément.

Les balises de l'utilisateur peuvent alors identifier les sujets des ressources annotées ou l'opinion de l'utilisateur sur la ressource [63].

Dans un système de recherche multi-sources, la sélection d'une source d'information est basée sur la description de son contenu qui permet d'évaluer sa pertinence pour une requête donnée. Plus la description de la source est précise, plus la source sélectionnée est pertinente. Nous proposons dans ce travail d'intégrer la trace des utilisateurs lors de l'utilisation des sources dans la sélection des sources, en tirant parti de l'historique de marquage social. Les balises utilisateur peuvent fournir des informations supplémentaires sur le contenu, le thème, les concepts ou d'autres aspects pertinents de la source. Nous considérons les balises qui décrivent une ressource, où une ressource est une source d'information particulière. Nous encourageons les utilisateurs à fournir des balises aux sources utilisées pour créer un nuage de balises pour chaque source. Avec ces balises, les utilisateurs décrivent le contenu des sources à l'aide des annotations sémantiques qui sont utiles pour trouver des sources pertinentes pour les futures requêtes des utilisateurs.

2 Exploitation des données sociales dans la sélection des sources d'information basée sur l'algorithme génétique

Nous proposons d'améliorer l'algorithme génétique de sélection des sources (GASS [93]), proposé au chapitre 4, en exploitant les données de marquage social. L'amélioration concerne principalement la fonction d'adaptation de l'algorithme GASS, en intégrant la description du contenu des sources et les balises attribuées aux sources d'information dans l'évaluation de la performance des solutions potentielles afin d'estimer plus précisément la qualité (fitness) des individus dans la population. Nous considérons que les utilisateurs des sources disponibles attribuent des tags aux sources consultées, ce qui se traduit par un nuage de tags permettant de décrire le contenu de ces sources. Le nouveau algorithme est appelé IGASS (Improved Genetic Algorithm for Sources Selection). IGASS est initié par une population de solutions initiales (représentées par des chromosomes ou des individus) choisies de manière aléatoire. Chaque solution est évaluée à l'aide de la fonction de d'adaptation (fitness) améliorée. Les opérateurs génétiques (sélection, croisement et mutation) sont utilisés pour générer une nouvelle population à partir de la population actuelle. Une fois qu'une nouvelle génération est créée, le processus génétique est répété de manière itérative jusqu'à ce que l'on trouve une solution satisfaisante (une solution quasi-optimale)(Figure 5.1).

2.1 Définition du problème

Nous définissons le problème de sélection des sources comme suit :

Donnée : un ensemble $S = \{s1, s2, \ldots, sn\}$ de sources d'information, une requête q de l'utilisateur et un ensemble de balises ou balises $T(S) = \{T(s1), T(s2), \ldots, T(sn)\}$ attribuées aux sources d'information.

Question : déterminer un sous ensemble S' de S telle que la similarité entre (S', $T(S')$) et q est maximale.

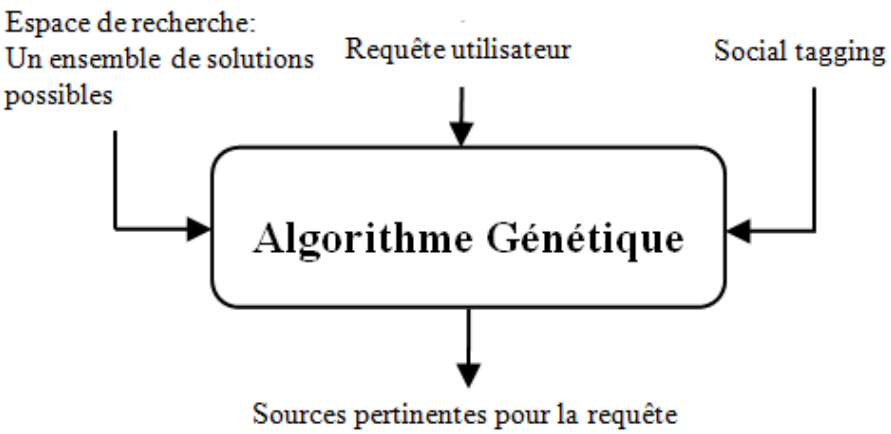


Figure 5.1 – L'approche de sélection des sources.

2.2 Représentation sociale des sources d'information

Chaque source est représentée par un nuage de tags (tag-cloud) qui est une agrégation des tags de tous les utilisateurs associés à cette source.

Soit $T(S)$: ensemble de balises utilisées par les utilisateurs pour annoter les sources disponibles.

$T(h)$: ensemble de balises utilisées par les utilisateurs pour annoter la source h, tel que :

$h \in S$ et $T(h) \subset T(S)$, $T(h) = \{t_1, t_2, ..., t_m\}$, t_i est la balise i attribuée à la source h.

La source h est représentée par des paires $(t_j, frèquence_{t_j})$, $t_j \in T(h), j = 1...m$.

Où $frèquence_{t_j}$ est le nombre de fois que la balises t_j est utilisée pour annoter la source h, qui est calculé indépendamment de l'utilisateur.

Le poids $tw(t_i)$ d'une balise t_i utilisée pour annoter la source h est calculé et normalisé comme suit :

$$tw(t_i) = \frac{frèquence(t_i)}{\sum_{t_j \in T(h)} frèquence(t_j)} \tag{5.1}$$

2.3 Fonction de fitness

La fonction de fitness mesure la similarité entre la solution et la requête de l'utilisateur. Rappelons ici que la solution au problème, notée *sol*, est une sélection composée d'un ensemble de k sources (un vecteur de sources). Pour estimer la similarité entre une solution *sol* et une requête utilisateur q, nous considérons à la fois le contenu des sources et les balises (tags) attribuées aux sources. Cette similarité est calculée sur la base de deux mesures de similarité, à savoir la similarité des balises et la similarité des termes, elle est donnée par la formule suivante :

$$Similarité(sol, q) = similarité^{termes}(sol, q) + similarité^{tags}(sol, q) \tag{5.2}$$

Où,

$Similarité^{termes}(sol, q)$: la similarité des termes entre la solution *sol* et la requête q.

$Similarité^{tags}(sol, q)$: la similarité des balises entre la solution *sol* et la requête q.

2.3.1 Calcule la similarité des termes

La similarité des termes entre une solution *sol* et une requête q est donnée par la formule suivante :

$$Similarité^{termes}(sol, q) = \frac{\sum_{h \in sol} sim^{termes}(h, q)}{k} \tag{5.3}$$

Où,

$Sim^{termes}(h, q)$: la similarité des termes entre la source h dans la solution et la requête q calculée par la mesure cosinus du modèle de recherche vectoriel en utilisant le vocabulaire de la source.

k : le nombre de sources dans la solution.

2.3.2 Calcule la similarité des balises

La similarité des balises entre une solution *sol* et la requête q est donnée par la formule suivante :

$$Similarité^{tags}(sol, q) = \frac{\sum_{h \in sol} sim^{tags}(h, q)}{k} \tag{5.4}$$

Où,

$Sim^{tags}(h, q)$: la similarité des balises entre la source h dans la solution et la requête q.

k : le nombre de sources dans la solution.

Pour calculer la similarité des balises entre une source et la requête de l'utilisateur, toutes les balises utilisées pour annoter cette source sont prises en compte. La représentation sociale de la source est donc utilisée.

La similarité entre la requête q et la source h sur un ensemble de balises utilisées pour annoter cette source est définie alors comme suit :

$$sim^{tags}(h, q) = \sum_{t_i \in T(h)} W(t_i, q) \tag{5.5}$$

Avec,

$$W(t_i, q) = \begin{cases} tw(t_i) & \text{si } t_i \in q \\ 0 & \text{sinon} \end{cases} \tag{5.6}$$

Où, $tw(t_i)$ est le poids de la balise t_i dans l'ensemble de balises de la source h.

2.4 Algorithme génétique proposé

L'algorithme 5 décrit l'algorithme génétique proposé pour la sélection des sources appelé IGASS (Improved Genetic Algorithm for Sources Selection).

Algorithm 5 IGASS : Improved Genetic Algorithm for Sources Selection

Input: un ensemble de n sources, une requête utilisateur q, un ensemble de balises par source

Output: la sélection quasi-optimale de k sources ($sol_{optimale}$) pour la requête q

1: Générer aléatoirement une population initiale de taille $Taille_Pop$ à partir des k combinaisons possibles de sources
2: Évaluer chaque solution dans la population initiale à l'aide de la fonction de fitness donnée par la formule 5.2
3: **while** (non fin de la reproduction) {évolution génétique} **do**
4: Sélectionner les chromosomes appropriés pour la reproduction (les parents)
5: Appliquer l'opérateur de croisement sur les parents selon une probabilité de croisement pour produire de nouveaux chromosomes (descendants)
6: Appliquer l'opérateur de mutation sur le chromosome courant selon une probabilité de mutation
7: Ajouter les nouveaux chromosomes à la population
8: **end while**
9: Évaluer la population actuelle à l'aide de la fonction de fitness (Formule 5.2)
10: Sélectionner la nouvelle population pour la prochaine génération
11: Vérifier les critères de terminaison (nombre maximum d'itérations atteint) ; si les critères ne sont pas satisfaits, retourner à (3)

L'algorithme génétique IGASS utilise les mêmes opérateurs de sélection, de croisement et de mutation décrits au chapitre 4.

3 Expérimentations et méthodes d'évaluation

Pour nos expérimentations, nous avons utilisé les sources d'information SNDL décrites dans la Section 3.1 du Chapitre 4 (la Table 4.1). Et nous avons également utilisé des données de marquage social que nous décrirons dans la section suivante. Nous avons comparé l'algorithme proposé (IGASS) avec GASS [93], l'algorithme présenté au Chapitre 4. GASS ne prend pas en compte l'aspect social dans l'évaluation de la pertinence des sources pour une requête d'un utilisateur. A travers cette comparaison, nous voulons montrer l'impact des données de marquage social sur les performances de l'algorithme génétique. Nous avons également utilisé l'algorithme CORI [31] dans les comparaisons car il est considéré comme l'un des algorithmes de sélection des sources les plus efficaces. Les paramètres par défaut de l'algorithme CORI sont utilisés.

Nous définissons les paramètres des algorithmes génétiques IGASS et GASS par les valeurs présentées dans Table 5.1.

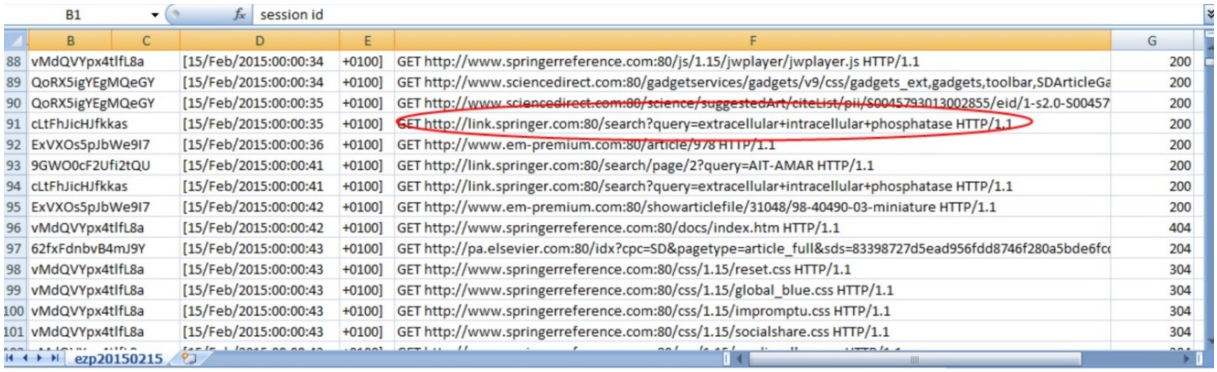
3.1 Données de marquage social de test

En raison du manque de données de marquage social concernant les balises attribuées aux sources d'information, et de la difficulté et du temps requis pour collecter des annotations des différents utilisateurs via l'utilisation de sources de test, et afin d'éviter le problème de démarrage à froid, nous avons proposé d'exploiter les fichiers journaux (log files, en anglais) du système de recherche SNDL. Les fichiers log contiennent des traces d'accès des utilisateurs aux sources SNDL. Le système de recherche SNDL enregistre les

Table 5.1 – Configuration des paramètres des algorithmes génétiques.

Parametre	Valeur
Taille de la population	50
Taux de croisement	60%
Taux de mutation	10%
Nombre de générations	1500

requêtes (HTTP) soumises par les utilisateurs, les URL des documents téléchargés par les utilisateurs et d'autres informations dans les fichiers journaux, comme illustré dans la Figure 5.2.



	B	C	D	E	F	G
88	vMdQVYpx4tIfL8a		[15/Feb/2015:00:00:34	+0100]	GET http://www.springerreference.com:80/js/1.15/jwplayer/jwplayer.js HTTP/1.1	200
89	QoRX5igYEgMQeGY		[15/Feb/2015:00:00:34	+0100]	GET http://www.sciencedirect.com:80/gadgetservices/gadgets/v9/css/gadgets_ext,gadgets,toolbar,SDArticleGa	200
90	QoRX5igYEgMQeGY		[15/Feb/2015:00:00:35	+0100]	GET http://www.sciencedirect.com:80/science/suggestedArt/citeList/pii/S0045793013002855/eid/1-s2.0-S00457	200
91	cLtFhJicHJfkkas		[15/Feb/2015:00:00:35	+0100]	GET http://link.springer.com:80/search?query=extracellular+intracellular+phosphatase HTTP/1.1	200
92	ExVXOs5pJbWe9I7		[15/Feb/2015:00:00:36	+0100]	GET http://www.em-premium.com:80/article/978 HTTP/1.1	200
93	9GWO0cF2Ufi2tQU		[15/Feb/2015:00:00:41	+0100]	GET http://link.springer.com:80/search/page/2?query=AIT-AMAR HTTP/1.1	200
94	cLtFhJicHJfkkas		[15/Feb/2015:00:00:41	+0100]	GET http://link.springer.com:80/search?query=extracellular+intracellular+phosphatase HTTP/1.1	200
95	ExVXOs5pJbWe9I7		[15/Feb/2015:00:00:42	+0100]	GET http://www.em-premium.com:80/showarticlefile/31048/98-40490-03-miniature HTTP/1.1	200
96	vMdQVYpx4tIfL8a		[15/Feb/2015:00:00:42	+0100]	GET http://www.springerreference.com:80/docs/index.htm HTTP/1.1	404
97	62fxFdnbvB4mJ9Y		[15/Feb/2015:00:00:43	+0100]	GET http://pa.elsevier.com:80/idx?cpc=SD&pagetype=article_full&sds=83398727d5ead956fdd8746f280a5bde6fc	204
98	vMdQVYpx4tIfL8a		[15/Feb/2015:00:00:43	+0100]	GET http://www.springerreference.com:80/css/1.15/reset.css HTTP/1.1	304
99	vMdQVYpx4tIfL8a		[15/Feb/2015:00:00:43	+0100]	GET http://www.springerreference.com:80/css/1.15/global_blue.css HTTP/1.1	304
100	vMdQVYpx4tIfL8a		[15/Feb/2015:00:00:43	+0100]	GET http://www.springerreference.com:80/css/1.15/impromptu.css HTTP/1.1	304
101	vMdQVYpx4tIfL8a		[15/Feb/2015:00:00:43	+0100]	GET http://www.springerreference.com:80/css/1.15/socialshare.css HTTP/1.1	304

Figure 5.2 – Structure du fichier journal du système de recherche SNDL.

Chaque requête http est constituée d'une liste de mots-clés saisis par un utilisateur particulier. Nous considérons les termes de requête ou les mots-clés envoyés à une source comme des balises associées à cette source. Par exemple, pour la requête envoyée à la source Springer comme le montre la Figure 5.2, les mots-clés extracellular, intracellular et phosphatase sont considérés comme des balises qui décrivent cette source.

Pour ces expérimentations, nous avons analysé des centaines de requêtes http à partir de fichiers journaux SNDL pour extraire les mots-clés de toutes les requêtes des utilisateurs qui sont envoyées aux sources de test. Ces mots clés sont utilisés pour représenter ces sources. Chaque source de test est alors représentée par un vecteur de paires <mot clé, fréquence>.

3.2 Méthodes d'évaluation

L'évaluation des performances d'un système de recherche multi-sources est généralement basée sur des mesures de rappel et de précision (décrites au chapitre 1, section 4). Pour évaluer l'approche proposée pour la sélection des sources, nous avons utilisé la mesure de rappel au niveau de la source, donnée par la formule suivante.

$$Précision = \frac{|Sources\ pertinentes\ sélectionnées|}{|Sources\ sélectionnées|} \tag{5.7}$$

La précision moyenne est calculée sur 20 requêtes de test. Les requêtes de test sont sélectionnées manuellement en tenant compte du contenu des sources et des données de

marquage social afin de montrer la performance de l'approche proposée. Nous choisissons les requêtes qui renvoient des résultats et nous évitons les requêtes qui ne renvoient aucune réponse (Table A.1 en annexe). La Table 5.2 montre un exemple de requêtes de test. Nous avons fait varier le nombre de sources à sélectionner entre 2 et 9 ($k = 2, 4, 6, 8, 9$).

Table 5.2 – Exemples de requêtes de test

Numéro de requête	**Requête**
q1	developing prostate cancer
q2	oxygen respiration and pollution of environment
q3	antioxidant food natural
q4	intelligent bio inspired algorithms
q5	genome analysis

Pour identifier les sources pertinentes pour une requête, nous avons analysé les 20 premiers documents renvoyés par chaque source en réponse à cette requête. Nous avons demandé aux utilisateurs de juger de la pertinence des documents restitués. Une source est marquée comme pertinente si elle renvoie au moins 3 documents pertinents pour la requête. La Table 5.3 montre les sources pertinentes pour les requêtes de test.

Table 5.3 – Sources pertinentes pour les requêtes de test

Numéro de requête	**Sources pertinentes pour la requête**
q1	ACM, ClinicalKey, IEEE , IOP , JSTOR , RSC , ScienceDirect , SpringerLink , SpringerProtocols
q2	Elgar , JSTOR , RSC, ScienceDirect , SpringerLink , SpringerProtocols
q3	ClinicalKey , JSTOR , RSC, ScienceDirect, SpringerLink , SpringerProtocols
q4	ACM, IEEE , JSTOR , ScienceDirect , SpringerLink
q5	ACM, ClinicalKey , IEEE , JSTOR , ScienceDirect, SpringerLink , SpringerProtocols

3.3 Résultats des expérimentations

Nous rapportons dans Table 5.4 la précision moyenne obtenue par chaque algorithme de sélection des sources sur 20 requêtes de test pour les 10 sources SNDL.

Table 5.4 – Précision moyenne des trois algorithmes sur 20 requêtes.

Sources sélectionnées / Algorithme	k=2	k=4	k=6	k=8	k=9
GASS	0.875	**0.8625**	0.7917	0.733	0.68345
CORI	0.95	0.8375	0.7667	0.7125	0.68345
IGASS	**0.95**	**0.8625**	**0.8167**	**0.75**	**0.6944**

La Table 5.4 montre une amélioration de la précision de l'algorithme IGASS par rapport à l'algorithme GASS (Figure 5.3). En effet, l'intégration des données de marquage social dans le processus de sélection des sources permet d'identifier les sources les plus pertinentes à la requête de l'utilisateur, les balises fournies par les utilisateurs améliorent considérablement la qualité de la description des sources et par conséquent, la précision de la recherche est également améliorée. On constate que les deux algorithmes génétiques peuvent atteindre la même précision (cas de $k = 4$) et ce dans l'indisponibilité des données de marquage social.

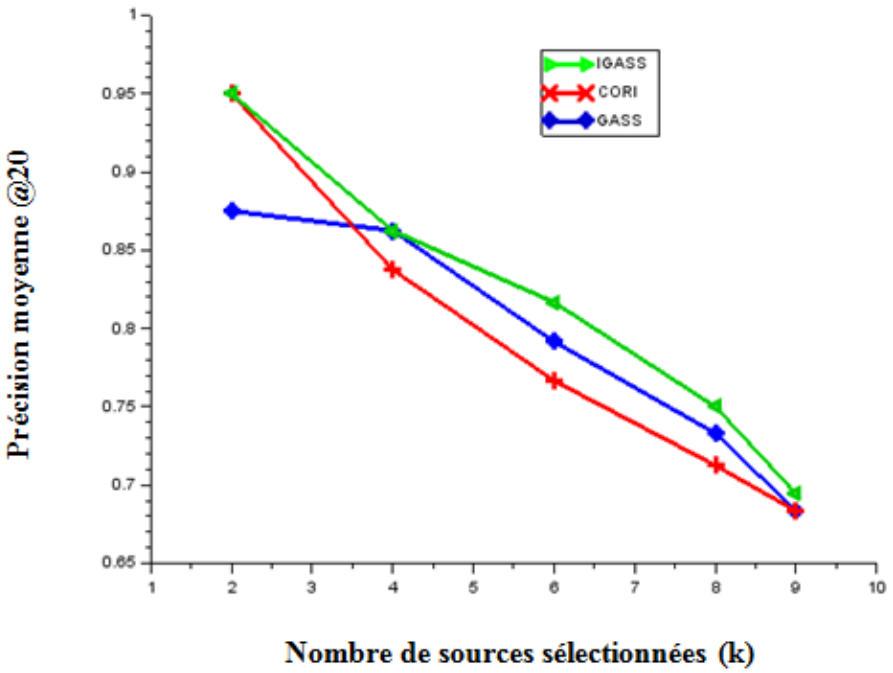


Figure 5.3 – Précision moyenne des trois algorithmes GASS, IGASS et CORI sur les données SNDL.

Notons également que les algorithmes génétiques (GASS et IGASS)sont meilleurs que l'algorithme CORI en termes de précision (Figure 5.3). On peut conclure que les algorithmes génétiques offrent une bonne solution au problème de sélection des sources dans un environnement multi-sources.

4 Conclusion

Dans ce chapitre, nous avons montré que les données de marquage social peuvent considérablement améliorer la sélection des sources dans un environnement de recherche multi-sources. Dans un premier temps, nous avons essayé d'améliorer la description des sources par les tags attribués par les utilisateurs aux sources. L'algorithme génétique de sélection des sources est enrichi d'une dimension sociale pour atteindre des performances de recherche plus élevées. La fonction de fitness de l'algorithme génétique proposé considère à la fois le vocabulaire et la représentation sociale des sources pour évaluer la qualité d'une solution potentielle à une requête donnée.

Les résultats des expérimentations ont montré que les performances de l'algorithme proposé dépassent les performances des deux algorithmes de sélection de source GASS et CORI. Ceci affirme que cette approche est efficace pour la recherche d'information multi-sources. Nous avons supposé que l'utilisateur décrit mieux ou de manière sémantique une source d'information, il serait intéressant d'étudier et d'analyser le niveau d'expertise de l'utilisateur dans le domaine afin de filtrer les balises non significatives et de ne sélectionner que les meilleurs tags qui décrivent d'une façon précise les sources.

Dans les travaux futurs, nous prévoyons de déterminer les relations entre les sources en fonction du contexte social pour regrouper les sources similaires en clusters afin de permettre la coopération entre les sources. L'étude des relations sociales source-utilisateur est une perspective intéressante pour personnaliser la sélection des sources.

Chapitre 6

Une approche multidimensionnelle pour adapter la sélection des sources aux thèmes d'intérêt de l'utilisateur

Sommaire

1 Introduction

Dans la recherche d'information, le paradigme "taille unique" signifie que les mêmes résultats sont fournis aux mêmes requêtes, peu importe qui les a soumises : le système

fournit des informations qui répondent strictement aux critères de la requête. Cependant, différents utilisateurs peuvent avoir des intérêts et des informations différents même s'ils utilisent la même requête. Par exemple, un informaticien peut utiliser la requête "apple" pour trouver des informations sur une marque d'ordinateur, tandis qu'un nutritionniste peut utiliser la même requête pour trouver la description d'un fruit. Lorsqu'une telle requête est soumise, le système renvoie une liste de résultats qui mélangent différents sujets pour tous les utilisateurs, même s'ils ont des intérêts différents. La requête seule ne représente pas le besoin réel d'information d'un utilisateur donné, il est donc nécessaire de comprendre les besoins spécifiques de l'utilisateur et d'adapter les résultats de la recherche en conséquence. L'accès à l'information pertinente en fonction des besoins des utilisateurs est devenu plus qu'une nécessité, d'où l'émergence de la personnalisation de l'information.

Le principal objectif de la recherche d'information personnalisée est la satisfaction complète de l'utilisateur. Pour obtenir des résultats proches de l'utilisateur, la recherche d'information tend à modéliser l'utilisateur selon un profil qui peut être explicite ou implicite (extrait de sa requête, son comportement, ses interactions, etc.) puis à l'intégrer dans la chaîne d'accès à l'information.

Lors de la recherche dans un environnement multi-sources, où les informations pertinentes sont réparties sur plusieurs sources d'information, il est essentiel de sélectionner uniquement les sources qui sont pertinentes pour la requête d'un utilisateur donné. Cela revient à filtrer les sources non pertinentes et à ne rechercher que celles susceptibles de contenir des documents pertinents [108]. La sélection des sources est une phase importante pour un système de recherche multi-sources qui peut avoir un impact sur le résultat final. Les approches classiques de sélection des sources évaluent la pertinence d'une source en fonction d'une mesure locale de similarité entre la requête et le contenu de la source, sans tenir compte de l'utilisateur qui a soumis la requête ; de ses préférences, de ses interactions, de ses relations sociales, etc.

Aujourd'hui les réseaux sociaux sont devenus partie intégrante de la vie des utilisateurs pour partager et diffuser l'information. Les utilisateurs partagent leurs opinions sur un sujet ou un événement et reçoivent des commentaires, des recommandations de pairs, d'amis, etc. Une grande masse de données est générée chaque jour par les réseaux sociaux qui représentent une source précieuse d'informations qui contribue à améliorer la recherche d'information personnalisée. L'avantage d'exploiter ce type d'informations est qu'il permet aux systèmes de recherche personnalisés d'acquérir une connaissance approfondie des intérêts et des préférences de leurs utilisateurs en raison de la richesse des informations disponibles sur les sites Web sociaux. De plus, étant donné qu'une grande partie des informations partagées sur les sites sociaux sont publiques, l'utilisation de ce contenu public ne doit pas constituer une menace pour la vie privée des utilisateurs.

Dans ce chapitre, nous proposons une nouvelle approche de sélection des sources pour personnaliser la recherche multi-sources en tenant compte de l'aspect social des utilisateurs dans les réseaux sociaux. Nous décrivons dans la section suivante le problème auquel nous sommes confrontés et nous donnons les objectifs importants de la solution proposée.

2 Problématique de recherche et objectifs

Le principal problème que nous abordons dans ce chapitre est de savoir comment satisfaire un utilisateur en ne sélectionnant que des sources proches de ses intérêts dans un environnement de recherche multi-sources. Un utilisateur peut s'intéresser à plusieurs domaines tout comme une source peut couvrir plusieurs domaines. En effet, chaque uti-

lisateur a des besoins spécifiques et nécessite une adaptation spécifique, par conséquent, nous visons dans ce travail à adapter la sélection des sources en fonction des intérêts de chaque utilisateur. Nous proposons un modèle multidimensionnel pour personnaliser la sélection des sources, qui combine la dimension intelligence avec la dimension sociale. La dimension intelligence consiste à utiliser des méthodes d'intelligence artificielle pour améliorer les performances de la recherche d'information multi-sources et pour optimiser la sélection des sources lorsque le nombre de sources disponibles est énorme. Des méthodes intelligentes basées sur des algorithmes génétiques ont montré leur efficacité pour résoudre le problème de sélection des sources et trouver la meilleure sélection [93, 94]. La dimension sociale exploite les interactions et les activités des utilisateurs sur les réseaux sociaux (amis communs, tagging, commentaires et avis, Likes, etc.) pour détecter les intérêts des utilisateurs qui sont les éléments clés pour l'adaptation [112].

Dans les environnements multi-sources, le comportement de l'utilisateur à travers l'utilisation des sources d'information peut être utile pour comprendre les sources plus ou moins intéressantes pour lui. Dans un système de marquage social, l'utilisateur peut annoter les sources qu'il utilise avec un ensemble de balises ou d'étiquettes (tags). Les mégadonnées générées à partir des balises des utilisateurs peuvent contenir de nombreux synonymes et polysémies, ce qui rend difficile leur analyse pour comprendre les différents intérêts des utilisateurs. Une solution consiste à regrouper les caractéristiques de ces balises en "thèmes d'intérêt" ou "clusters", constitués d'ensembles de balises apparentées qui partagent une signification sémantique dans chaque cluster. Il existe quelques extensions notables des méthodes de clustering de base, telles que l'analyse sémantique latente probabiliste (Probabilistic Latent Semantic Analysis , PLSA) ou l'allocation de Dirichlet latente (Latent Dirichlet Allocation , LDA) [22]. Ces méthodes utilisent des modèles probabilistes adaptés pour regrouper les contenus. L'hypothèse principale de ces techniques est l'existence de certains sujets ou objectifs implicites. Ils allouent tout le contenu à ces sujets [86].

Au lieu de considérer les balises d'utilisateurs brutes, nous nous concentrons sur des sujets (thèmes) d'intérêt général des utilisateurs, extraits de ces balises à l'aide de la technique de modélisation des sujets LDA. Le modèle LDA fournit un outil puissant pour découvrir et exploiter les structures thématiques cachées dans des vastes archives de documents [21]. L'objectif général de ce modèle est de produire des représentations de documents interprétables qui peuvent être utilisées pour découvrir des sujets ou une structure dans une collection de documents non étiquetés.

L'approche de sélection des sources proposée utilise d'abord le modèle LDA pour faire émerger des thèmes pertinents dans un grand corpus de balises, ce qui permet de réduire la dimensionnalité des données en considérant les thèmes dans les balises au lieu des balises individuelles, et permet également de résoudre le problème du bruit et de l'ambiguïté des données.

Les thèmes d'intérêt (topic of interest) des utilisateurs découverts sont ensuite intégrés dans le processus de sélection des sources pour générer la sélection quasi-optimale adaptée à chaque utilisateur à l'aide d'un algorithme génétique.

3 La méthode LDA (Latent Dirichlet Allocation)

La modélisation de sujets ou thèmes (topic modelling en anglais) [21, 69] fournit des méthodes pour découvrir des connaissances latentes, trouver des relations entre les

données, comprendre et synthétiser un énorme corpus de données [82]. Les modèles de sujet sont un excellent moyen d'explorer et de structurer automatiquement un grand nombre de documents : ils regroupent des documents en fonction des mots qui y figurent. Comme les documents sur des sujets similaires ont tendance à utiliser un sous-vocabulaire similaire, les groupes de documents résultants peuvent être interprétés comme traitant de différents «sujets» ou "topics".

Le modèle d'allocation de Dirichlet latent LDA, proposé par Blei et al. [22], est l'une des méthodes de modélisation de sujet les plus populaires utilisées pour découvrir des sujets sémantiques latents dans de grandes collections de textes. L'idée de base est que les documents sont représentés comme des mélanges aléatoires sur des sujets latents, où chaque sujet est caractérisé par une distribution sur les mots. La Figure 6.1 illustre le modèle LDA. Un certain nombre de "sujets", qui sont des distributions sur des mots, existent pour l'ensemble des collections (indiqués à gauche de la figure). Chaque document est supposé être généré comme suit. Choisissez d'abord une distribution sur les sujets (l'histogramme à droite) ; puis, pour chaque mot, choisissez une affectation de sujet (les bulles colorées) et choisissez le mot de sujet correspondant.

LDA est un modèle non supervisé qui ne nécessite pas d'informations sur les sujets dans les documents et que les documents ne sont pas non plus étiquetés avec des sujets ou des mots-clés, il est donc largement utilisé pour résoudre de nombreux problèmes dans divers domaines, notamment dans la recherche d'information [12, 35, 70, 128, 156] et dans les systèmes de recommandation [61, 83, 164].

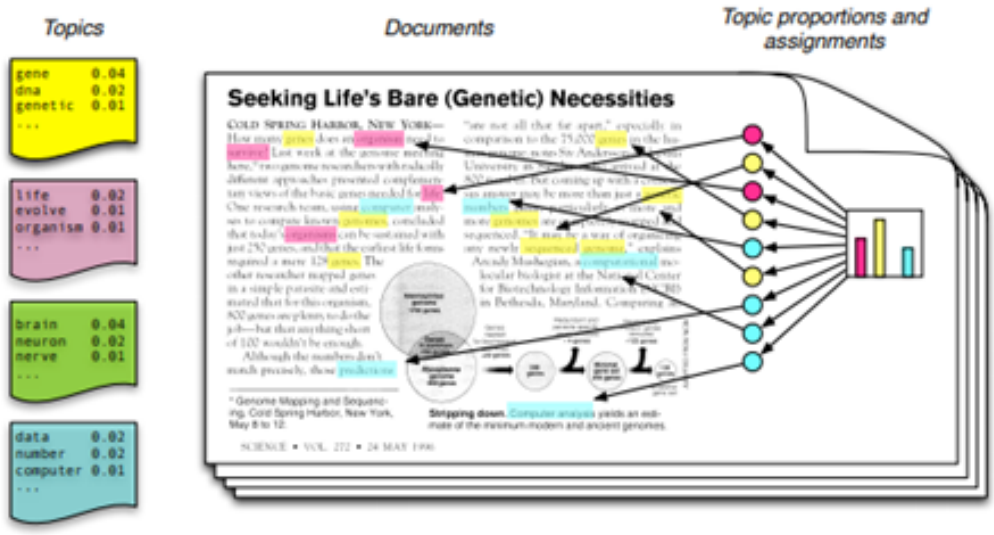


Figure 6.1 – Le schéma illustre le modèle LDA [20].

LDA est un modèle bayésien hiérarchique qui utilise des a priori de Dirichlet pour estimer les variables latentes insolubles du modèle. À un niveau élevé, LDA est basé sur un modèle génératif dans lequel chaque mot d'un document d'entrée d'un corpus est choisi en sélectionnant d'abord un sujet qui correspond à ce mot, puis en sélectionnant le mot à partir d'une distribution de sujet sur les mots. Chaque distribution de sujet sur les mots et distribution des mots sur les sujets est tirée de sa distribution Dirichlet respective. La définition formelle de l'algorithme génératif sur un corpus est (voir le modèle graphique de la Figure 6.2) :

1. Pour chacun des k sujets ϕ_k :

2. Choisir $\phi_k \sim Dir(\beta)$
3. Pour chacun des D documents d :
4. Choisir $N_d \sim Poisson(\xi)$
5. Choisir $\Theta_d \sim Dir(\alpha)$
6. Pour chacun des N_d mots $w_{n,d}$:
7. Choisir $z_{n,d} \sim Multinomiale(\Theta)$
8. Choisir $w_{n,d} \sim Multinomiale(z_{\phi_{n,d}})$

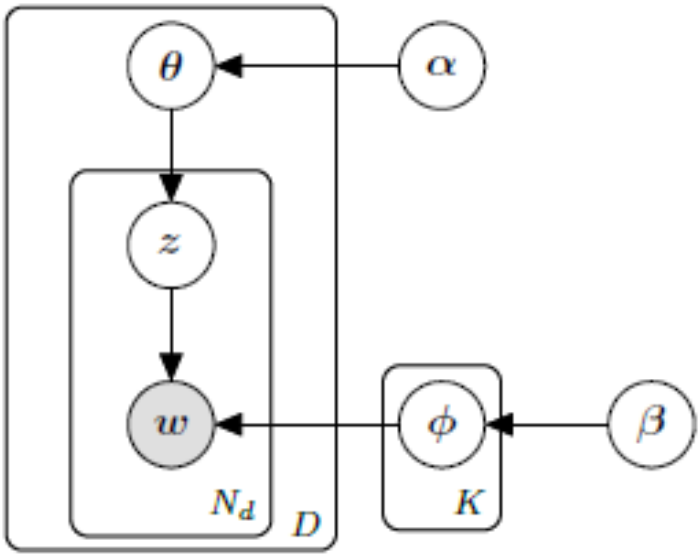


FIGURE 6.2 – Modèle graphique pour l'allocation Dirichlet latente.

Le modèle LDA est représenté sous la forme d'un modèle graphique probabiliste dans la Figure 6.2. Comme le montre clairement la figure, la représentation LDA a trois niveaux, à savoir le niveau corpus, le niveau document et le niveau mot. Les paramètres et les variables du modèle sont décrits ci-dessus.

α est la distribution des sujets par document,
β est la distribution des mots par sujet,
Θ est la distribution des sujets du document d,
ϕ est la distribution des mots pour le sujet k,
z est le sujet du nième mot du document d, et
w est le mot spécifique.

Il existe donc deux processus de génération dans le modèle LDA : l'un est la génération de la distribution de sujets de document et l'autre est la génération de la distribution de mots de sujet.

4 L'approche de sélection des sources proposée

Une nouvelle approche de sélection des sources qui combine deux dimensions, à savoir sociale et intelligence, est proposée. Cette approche vise à trouver la sélection quasi-optimale de sources correspondant aux sujets (thèmes) d'intérêt d'un utilisateur donné. La modélisation de sujets LDA est utilisée pour représenter les intérêts des utilisateurs

sur un espace de sujets latents de faible dimension. LDA traditionnellement utilisée pour modéliser le contenu textuel, est utilisée pour trouver des modèles de comportement des utilisateurs dans le marquage social et les regrouper en conséquence. Cela signifie que l'espace des sujets est extrait directement de toutes les balises utilisées par les utilisateurs pour annoter les sources d'information disponibles. Un profil utilisateur multidimensionnel sur une diversité de thèmes est alors créé, ce qui permet de mieux comprendre les différents domaines d'intérêt de l'utilisateur et donc de personnaliser la recherche multi-sources en ne sélectionnant que les sources les plus proches de ses thèmes. Un algorithme génétique est utilisé pour trouver la solution (sélection) quasi-optimale et adaptée pour chaque utilisateur. L'algorithme proposé est identique à l'algorithme GASS [93] (Chapitre 4) concernant le codage des solutions ainsi que les opérateurs génétiques, cependant, la différence réside dans l'évaluation des solutions potentielles de nouveau algorithme qui intègre l'aspect social de l'utilisateur comme un paramètre important dans la sélection des sources (Figure 6.3).

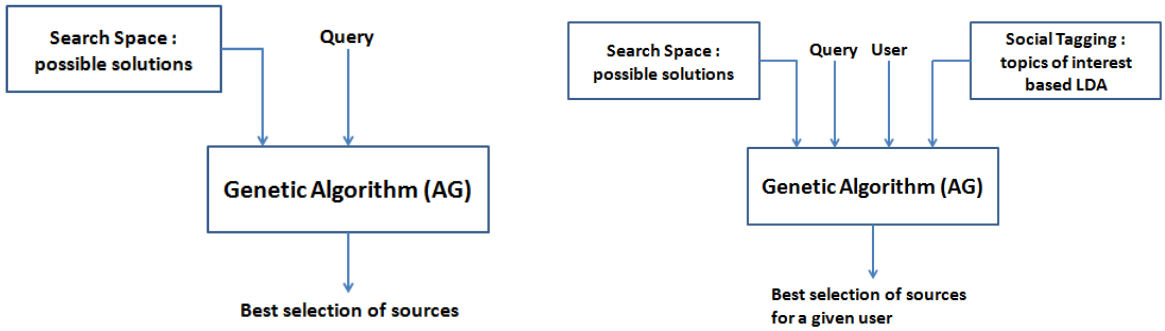


Figure 6.3 – Approche simple basée sur l'AG (à gauche) et l'approche proposée (à droite).

Avant d'introduire la description détaillée de l'approche proposée, une définition formelle du problème de sélection des sources d'information est donnée dans la sous-section suivante.

4.1 Définition du problème

Nous définissons le problème de sélection des sources d'information comme suit :

Donnée : un ensemble $S = \{s_1, s_2, ..., s_n\}$ de sources d'information ($|S| = n$), un ensemble $T(U) = \{T(u_1), T(u_2), ..., T(u_p)\}$ de balises utilisées par les utilisateurs pour annoter les sources d'information, un utilisateur u_i et une requête $q(u_i)$ de l'utilisateur u_i.

Question : déterminer un sous-ensemble S' de S pour l'utilisateur u_i, telle que la similarité entre les éléments de S' et la paire ($q(u_i)$, intérêts de u_i) soit maximale, où $|S'| = k < n$.

Nous rappelons que l'espace de recherche des solutions possibles, noté E, est composé de k-combinaisons, telle que $|E| = \binom{n}{k} = \frac{n!}{k!(n-k)!}$

Une solution à ce problème est une sélection adaptée à l'utilisateur. On note $sel^k_{u_i}$ un sélection de k sources d'information adaptée à l'utilisateur u_i. Une source d'information est considérée comme un document volumineux représenté par des termes extraits de ses documents échantillonnés [31, 68, 163].

4.2 Description de l'approche

L'approche proposée comprend deux étapes principales, comme le montre la Figure 6.4.

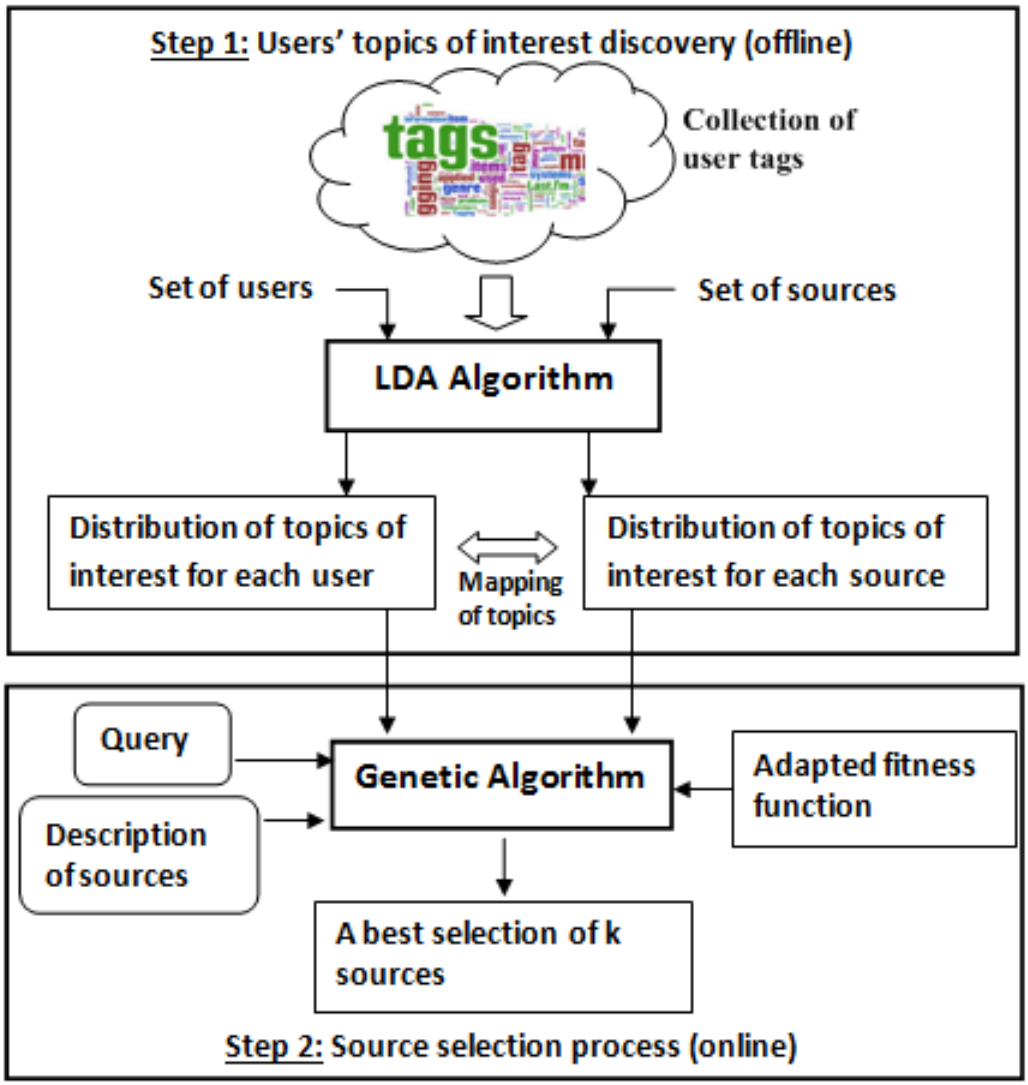


FIGURE 6.4 – Aperçu de l'approche proposée.

La première étape est la « découverte des thèmes d'intérêt des utilisateurs » qui consiste à découvrir les thèmes d'intérêt général de chaque utilisateur à partir d'une large collection de balises utilisées par tous les utilisateurs pour annoter les sources disponibles, pour cela, le modèle LDA est utilisé. La sortie de l'algorithme LDA consiste en la distribution des probabilités sur tous les thèmes d'intérêt découverts. La sortie du modèle LDA est l'entrée de la deuxième étape, qui est le « processus de sélection des sources ». Dans cette étape, les thèmes d'intérêt déduits par le modèle LDA sont utilisés pour générer pour un utilisateur donné la sélection "optimale" de sources en utilisant un algorithme génétique. La solution composée de k sources doit maximiser la fonction d'adaptation de l'algorithme génétique ce qui augmente la satisfaction des utilisateurs. Ces deux étapes sont décrites en détail ci-dessous.

Notez que la modélisation des sujets LDA est une étape hors ligne qui est effectuée indépendamment de l'algorithme génétique.

4.2.1 Découverte des thèmes d'intérêt des utilisateurs

Grâce à l'utilisation de sources d'information, les utilisateurs peuvent annoter des sources (ou des documents extraits des sources) à l'aide d'un ensemble de termes ou de balises, générant ainsi une énorme collection de données de marquage. Cette collection de balises est considérée pour capturer et découvrir les différents intérêts de tous les utilisateurs. La modélisation de sujets basée sur LDA est utilisée dans l'analyse de ces données de marquage pour extraire les thèmes d'intérêt de tous les utilisateurs en combinant toutes les balises de tous ces utilisateurs. Sachant que chaque balise est associée à un document ou une source particulière. Cette association permet de déduire le ou les thèmes auxquels la source est liée, ainsi les thèmes de toutes les sources disponibles sont inférées à partir de ces différents liens d'association.

Selon le modèle LDA, un utilisateur est considéré comme "un document" et toutes les balises utilisées par cet utilisateur comme des "mots" contenant ce document. Ainsi, chaque thème découvert peut être interprété comme "un thème d'intérêt" pour l'utilisateur. Le processus de modélisation LDA implique de trouver un mélange de thèmes d'intérêt pour chaque utilisateur. Ces thèmes d'intérêt sont générés à partir de balises postées par tous les utilisateurs pour les sources, ce qui garantit une plus grande précision.

Nous considérons z le nombre de thèmes d'intérêt à découvrir. Le nombre de thèmes z doit être défini avant de construire le modèle LDA.

Soit $T(U) = \{T(u_1), T(u_2), ..., T(u_p)\}$ une collection de balises générées par p utilisateurs, où $T(u_i)$ est la sous-collection de balises utilisées par l'utilisateur u_i, telle que :

$$T(u_i) = \{t_1, t_2, .., t_m\} \ ((u_i \in U \text{ et } |T(u_i)| = m).$$

La collection de balises $T(U)$ constitue le vocabulaire introduit en entrée du modèle LDA. Le résultat fourni en sortie du modèle LDA est constitué de :

— Une représentation de chaque thème $t \in 1, .., z$ par un mélange de mots. Un thème est représenté par une distribution sur des mots appartenant au vocabulaire. Les mots associés aux thèmes peuvent être utilisés pour interpréter les thèmes et découvrir ce qu'ils représentent en se basant sur les scores de probabilité des mots. Nous présentons la distribution des M mots de vocabulaire d'un thème i comme suit :

$Topic_t = \{(mot_j, poids_j)/j = 1...M\}$, où mot_j est un mot de vocabulaire et $poids_j$ le poids de mot_j pour le $t^{ème}$ thème.

— Une représentation de chaque utilisateur $u_i, i \in 1, ..., p$ par un mélange de thèmes. Un utilisateur est représenté par une distribution sur z thèmes. Nous représentons la distribution de thèmes d'un utilisateur u_i comme suit :

$$D(u_i) = \{w_1, w_2, \ldots, w_z\}, u_i \in U \text{ et } |D(u_i)| = z$$

$D(u_i)$ représente la distribution des thèmes d'intérêt (topics of interest, en anglais) pour l'utilisateur u_i, où w_i est le taux d'intérêt de l'utilisateur pour le $t^{ème}$ thème.

4.2.2 Déduire l'intérêt de l'utilisateur pour les sources disponibles

Le modèle LDA généré est utilisé pour déduire l'appartenance des sources d'information aux thèmes d'intérêt. Chaque source s_j représentée par une collection de tags est mappée sur les thèmes générés par LDA pour en déduire sa distribution de thèmes en utilisant la technique *inférence* du modèle LDA. La distribution de probabilités sur les z thèmes d'intérêt de la source s_j est donnée comme suit :

$$D(s_j) = \{v_1, v_2, \ldots, v_z\}, s_j \in S \text{ et } |D(s_j)| = z$$

$D(s_j)$ décrit l'importance de la source pour chaque thème, où v_i est le taux d'appartenance de la source au $t^{ème}$ thème.

La modélisation LDA est utilisée pour modéliser les relations thème-utilisateur et thème-source. Comme le montre la Figure 6.5), une autre relation (ou lien) cachée pourrait être déduite entre un utilisateur u_i et une source s_j. Cette relation interprète l'intérêt de l'utilisateur pour les sources disponibles qui peut être estimé sur la base de leur similarité sur les thèmes d'intérêt.

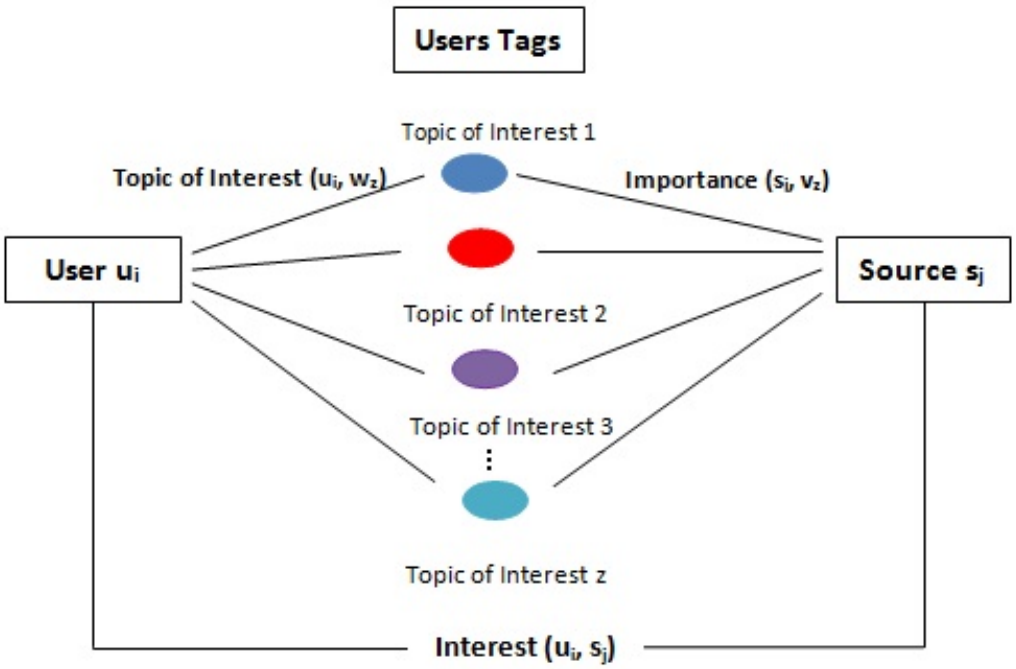


Figure 6.5 – Les relations générées à l'aide de la modélisation LDA sur les balises des utilisateurs.

Pour calculer l'intérêt de l'utilisateur u_i pour une source s_j (noté, $Intérêt(u_i, s_j)$), la formule cosinus [129, 130] est utilisée, elle calcule la similarité de thème entre $D(u_i)$ et $D(s_j)$, les deux vecteurs de distribution des sujets d'intérêt de l'utilisateur u_i et de la source s_j respectivement. Cette similarité est donnée par l'équation 6.1.

$$Intérêt(u_i, s_j) = Similarité^{thème}(u_i, s_j) = \frac{\sum_{h=1,z}(w_{ih} * v_{jh})}{\sqrt{\sum_{h=1,z}(w_{ih})^2 * \sum_{h=1,z}(v_{jh})^2}} \quad (6.1)$$

4.2.3 Processus de sélection des sources

Dans cette étape, un algorithme génétique est utilisé pour générer la meilleure solution ou "sélection" adaptée à un utilisateur donné, sur la base de la description des sources, de la requête de l'utilisateur et des thèmes d'intérêt (topics of interest)de l'utilisateur.

L'algorithme proposé reprend le schéma général de l'algorithme GASS, donc le codage des solutions potentielles, l'espace de recherche (espace des solutions potentielles) ainsi que les opérateurs génétiques de croisement et de mutation sont les mêmes (voir Chapitre 4). Les modifications portent notamment sur l'évaluation des solutions potentielles. Dans ce qui suit, nous présentons la fonction d'adaptation (fitness) proposée ainsi que le nouveau algorithme génétique de sélection des sources.

4.2.3.1 Fonction de fitness L'évaluation de la pertinence d'une solution au problème de sélection des sources $sel^k_{u_i}$ est basée sur la pertinence des k sources qui apparaissent dans $sel^k_{u_i}$, donnée par l'équation 6.2.

$$Pertinence(sel^k_{u_i}, q) = \frac{\sum_{j=1,k} Pertinence(s_j^{u_i}, q)}{k} \tag{6.2}$$

Où,

$Pertinence(s_j^{u_i}, q)$: la pertinence de la source s_j pour l'utilisateur u_i et pour la requête q .

k : le nombre de sources sélectionnées, $k = |sel^k_{u_i}|$.

L'évaluation de la pertinence d'une source considère le triplet (source, utilisateur, requête). La pertinence d'une source pour une requête est en effet relative pour chaque utilisateur elle est calculée en combinant linéairement la similarité entre la source et la requête qui considère le vocabulaire de la source, et l'intérêt de l'utilisateur pour la source qui considère les données de social tagging. Cette évaluation est donnée par l'équation suivante.

$$Pertinence(s_j^{u_i}, q) = (1 - \lambda)\ Similarité(s_j, q) + \lambda\ Intérêt(u_i, s_j) \tag{6.3}$$

Où,

— La $Similarité(s_j, q)$ est la similarité entre une source s_j et une requête q. Cette similarité est calculée par la formule cosinus entre les deux vecteurs de poids de la source et de la requête en considérant le vocabulaire de le source.

— $Intérêt(u_i, s_j)$ représente l'intérêt de l'utilisateur u_i à la source s_j calculé par l'équation 6.1. Cet intérêt est calculé en utilisant à la fois la distribution des thèmes pour la source et pour l'utilisateur.

— Le paramètre λ, compris entre zéro et un, contrôle l'effet des intérêts de l'utilisateur sur l'évaluation globale.

4.2.3.2 Algorithme génétique La sélection des sources est adaptée à chaque utilisateur, en tenant compte de ses thèmes d'intérêt. L'algorithme génétique proposé a pour but de trouver un ensemble de sources qui correspondent le mieux aux intérêts de l'utilisateur. La fonction de fitness qui évalue les performances de chaque solution combine de manière linéaire la similarité entre la requête et la source, et l'intérêt de l'utilisateur pour la source. Ainsi, pour une même requête, la meilleure sélection peut être différente pour deux utilisateurs ayant des intérêts différents.

L'algorithme proposé (Algorithme 6) est appelé algorithme génétique personnalisé pour la sélection des sources basé sur le modèle LDA, noté PGASS-based-LDA (Personalized Genetic Algorithm for Sources Selection based LDA).

Algorithm 6 PGASS-based-LDA : Personalized Genetic Algorithm for Sources Selection based LDA

Input: Un ensemble de n sources, thèmes d'intérêt de l'utilisateur u_i, et la requête q
Nous considérons,
p_m : taux de mutation
p_c : taux de croisement
Pop_{Size} : taille de la population
Gen_{Max} : nombre maximal d'itérations
Output: la sélection quasi-optimale de k-sources ($sel_{optimale}{}^{k}_{u_i}$) pour l'utilisateur u_i

1: Générer aléatoirement une population initiale de taille Pop_{Size} de solutions possibles
2: Évaluer chaque solution dans la population initiale à l'aide de la fonction de fitness donnée par l'équation 6.2
3: **while** (le nombre Gen_{Max} n'est pas atteint) {évolution génétique} **do**
4: Sélectionner les chromosomes appropriés pour la reproduction
5: Appliquer l'opérateur de croisement sur les paires (parents) en fonction de p_c pour produire de nouveaux chromosomes (enfants)
6: Ajouter les nouveaux chromosomes à la population
7: Appliquer l'opérateur de mutation pour chaque chromosome de la population selon p_m
8: Ajouter les chromosomes modifiés à la population
9: **end while**
10: Évaluer la population actuelle en utilisant la fonction de fitness donnée dans l'équation 6.2
11: Sélectionner les chromosomes les plus performants pour la génération suivante (utiliser la population nouvellement générée pour une nouvelle exécution de l'algorithme)

L'algorithme PGASS-based-LDA se termine après un nombre fixe de générations Gen_{Max}. La solution optimale trouvée $sel_{optimale}{}^{k}_{u_i}$ pour l'utilisateur u_i est celle qui a la valeur maximale de fitness dans la population de la dernière génération.

4.2.4 Sélection des sources pour un nouvel utilisateur

L'approche proposée basée sur la modélisation LDA utilise des données de marquage social pour déduire les thèmes d'intérêt des utilisateurs. Le manque de telles données sociales, pour un nouvel utilisateur ou pour des sources rarement ou non étiquetées par les utilisateurs, présente les limites de l'approche proposée, connues par les problèmes de rareté des données et de démarrage à froid.

Dans le cas d'un nouvel utilisateur, nous suggérons d'exploiter la relation utilisateur-utilisateur en utilisant le vocabulaire des requêtes précédentes pour identifier les utilisateurs similaires et procéder à la sélection des sources en fonction de la distribution des thèmes d'intérêt des utilisateurs les plus similaires, en suivant les étapes suivantes :

— Générer la distribution des probabilités des utilisateurs existants sur les éléments de la requête (par exemple en utilisant l'approche tf-idf).

— Pour chaque nouvel utilisateur, calculer sa distribution sur les éléments de la requête en considérant tous les termes des requêtes (même vocabulaire).

— Calculer la similarité entre la distribution d'un nouvel utilisateur et chaque distribution de tous les utilisateurs existants en utilisant la formule du cosinus.

— Sélectionner l'utilisateur le plus similaire au nouvel utilisateur.

— Sélectionner les sources pour le nouvel utilisateur en utilisant sa requête et la distribution des thèmes d'intérêt de l'utilisateur sélectionné (son similaire).

Il est à noter que la faisabilité de cette solution n'est pas démontrée dans cette thèse.

5 Expérimentations et tests

Cette section décrit les expérimentations menées pour évaluer l'approche proposée. L'approche personnalisée proposée est comparée à des approches personnalisées et non personnalisées choisies à partir d'algorithmes de sélection des sources de l'état de l'art. Tout d'abord, des détails sur les bases de référence sont fournis, puis les ensembles de données et les métriques utilisées dans l'évaluation des approches sont présentés avec la description de l'exécution de la modélisation LDA pour générer les thèmes d'intérêt des utilisateurs, et enfin les résultats et discussions sont présentés.

5.1 Bases de référence

Nous comparons les performances de l'approche de sélection des sources proposée avec quatre modèles de référence, l'un personnalisé et les autres non personnalisés. Les modèles de référence sont : l'algorithme le plus populaire CORI [31], l'algorithme de sélection des sources basé sur le vocabulaire Taily [6], l'algorithme de sélection des sources basé sur l'algorithme génétique GASS [93] et l'algorithme de sélection des sources personnalisé proposé dans [133].

— L'algorithme CORI considère chaque collection ou source comme un document volumineux, il est considéré comme l'un des algorithmes de sélection des sources les plus stables et les plus efficaces [124]. CORI est basé sur des réseaux d'inférence bayésiens. Dans CORI, les similarités entre une requête utilisateur et un ensemble de collections de documents sont calculées, afin de classer les collections. La similarité d'une requête avec une collection donnée est la somme des probabilités de croyance des termes de la requête apparaissant dans la collection. La similarité CORI entre une requête q et une collection c peut être calculée par l'équation suivante.

$$CORI(q,c) = \frac{\sum_{t \in q\&c}(d_b + (1-d_b).T_{c,t}.I_{c,t})}{|q|} \tag{6.4}$$

Où d_b est la composante de croyance minimale, est fixé à 0.4, $T_{c,t}$ est le poids du terme dans la collection, $I_{c,t}$ est la fréquence de collection inverse, et $|q|$ est le nombre de termes distincts dans la requête. La valeur $|q|$ peut être ignorée car elle est constante pour une requête donnée. La fréquence de collecte inverse $I_{c,t}$ et le poids $T_{c,t}$ sont calculés par les équations suivantes.

$$I = \frac{log(\frac{|C|+0.5}{cf_t})}{log(|C|+1.0)} \tag{6.5}$$

$$T = \frac{df_t}{df_t + tf_{base} + tf_{factor} \cdot \frac{cw}{avcw}} \tag{6.6}$$

Où C est le nombre total de collections disponibles, cf_t est le nombre de collections qui contiennent le terme t de la requête , df_t est le nombre de documents dans la i^{th} collection qui contiennent le terme t de la requête, cw est le nombre total de mots dans la i^{th} collection et $avcw$ est la moyenne de cw de toutes les collections. Les autres termes sont des constants : $tf_{base} = 50$, et $tf_{factor} = 150$.

— Taily est un algorithme de sélection des sources basé sur le vocabulaire. Taily modélise la distribution des scores d'une requête dans chaque fragment ou source sous la forme d'une distribution Gamma et sélectionne les fragments avec des documents très bien notés à la fin de la distribution. Taily estime les paramètres de distribution des scores en fonction de la moyenne et de la variance des caractéristiques de la fonction de score dans les collections et les fragments. L'algorithme Taily utilise deux paramètres, n_c et v, où n_c estime la profondeur de la liste de classement finale souhaitée, et v est le nombre de documents dans le top n_c auxquels un fragment doit être estimé comme contribuant pour être sélectionné. Au moment de la requête, la fonction de distribution cumulative de la distribution *Gamma* est utilisée pour estimer le nombre de documents dans chaque fragment qui auront un score supérieur à un seuil dérivé de n_c. Chaque fragment qui fournit v ou plusieurs documents est sélectionné [6]. Les valeurs des paramètres ($n = 400$ et $v = 50$) recommandées par Aly et al. sont utilisées dans les expérimentations.

— L'algorithme GASS décrit au chapitre 4, utilise un algorithme génétique pour la sélection des sources mais ne considère pas l'aspect social.

— Un algorithme de sélection des sources personnalisé présenté dans [133] (noté SaoudAlgo), leur approche intègre les informations de profil social dans le processus de sélection des sources. Les données de marquage social sont utilisées pour créer un profil social de chaque utilisateur. les sources sont classées selon un score qui combine deux mesures comme suit :

$$ScoreSource_s(u_m, q) = (1 - \alpha) \ * \ SimSource_s^{Terms}(q) + \alpha * SimSource_s^{Tags}(u_m, q) \tag{6.7}$$

Où,
$\alpha \in [0\ 1]$
$ScoreSource_s(u_m, q)$: est le score de la source s associée à l'utilisateur u_m pour la requête q.
$SimSource_s^{Terms}(q)$: représente le degré de similarité entre la source s et la requête de l'utilisateur q, selon tous les termes des documents de la source. Cette mesure est calculée avec la formule du cosinus.
$SimSource_s^{Tags}(u_m, q)$: représente le degré de similarité entre la source s et l'utilisateur u_m, selon l'ensemble des balises associées aux documents de la source. Cette similarité est calculée à l'aide de la formule suivante :

$$SimSource_s^{Tags}(u_m, q) = \sum_{j=1}^{k} \frac{tf(u_m, d_j)}{k} \tag{6.8}$$

Où

k : est le nombre de documents retournés,

d_j : est le document j de la source s,

$tf(u_m, d_j)$: la mesure de similarité basée sur la fréquence des balises utilisées par l'utilisateur u_m pour annoter les documents de la source s.

Nous avons implémenté les algorithmes génétiques dans un environnement java en utilisant la bibliothèque d'algorithmes génétiques java JGAP [1]. Les mêmes valeurs de paramètres présentées dans Table 6.1 sont utilisées pour les deux approches basées sur des algorithmes génétiques, à savoir GASS et PGASS-based-LDA.

Table 6.1 – Les valeurs des paramètres des deux algorithmes génétiques.

Paramètre	**Valeur**
Taille de la population	50
Taux de croisement	60%
Taux de mutation	10%
Nombre de générations	500

5.2 Données de test

5.2.1 Sources d'information

Des sources d'information réelles sont utilisées dans l'évaluation des approches de sélection des sources qui sont des bases de données en ligne d'articles de recherche scientifique de différents domaines tels que l'informatique, l'économie, la finance, etc. L'accès à ces sources se fait via la plateforme SNDL [2]. Huit (8) sources d'information sont sélectionnées pour les expériences, elles sont décrites dans Table 6.2.

Pour créer la description des sources, nous avons utilisé un compte utilisateur dans la plate-forme SNDL, qui nous a permis de rechercher et de télécharger des documents à l'aide de requêtes de sonde. Pour cela, la technique d'échantillonnage des requêtes [30] a été utilisée. 15 requêtes d'un seul mot sont envoyées à chaque source et nous avons téléchargé les 4 premiers documents, nous avons obtenu environ 15*4=60 documents par source. Le processus ci-dessus peut ne pas produire de représentants optimaux comme l'ont noté Thomas et Hawking [150], mais est devenu une pratique standard lors de l'évaluation des algorithmes de sélection des sources [137, 142]. Les sources d'information sont indexées avec Indri [148] qui est un système gratuit d'indexation et de recherche d'information de Lemur Project [41]. Un index commun rassemblant les termes de toutes les sources est construit. Ensuite, le fichier d'index est nettoyé pour supprimer les termes non significatifs. les vecteurs de sources et de requêtes sont construits en utilisant l'approche $tf * idf$.

1. JGAP est un framework d'algorithmes génétiques et de programmation génétique écrit en Java (http ://jgap.sourceforge.net/).

2. https ://www plate-forme .sndl.cerist.dz

Table 6.2 – Sources SNDL utilisées dans les expérimentations.

Numéro de la source	Nom de la source	Domaine(s)
1	ACM Digital Library	Informatique
2	Edward Elgar Products	l'économie, les finances, les affaires et la gestion, le droit et la politique publique
3	IEEE, Institute of Electrical and Electronics Engineers	Informatique, Électronique, Télécommunication
4	IOP science Extra of IOP Publishing	Physique, Sciences des matériaux, Mathématiques Appliquées
5	JSTOR	Multidisciplinaire
6	RSC, Royal Society of Chemistry	Chimie, Sciences des matériaux, environnement, Biologie
7	ScienceDirect of Elsevier	Multidisciplinaire
8	SpringerLink	Multidisciplinaire

5.2.2 Données sociales

Pour nos expérimentations, nous avons utilisé les données sociales de BibSonomy [3] . Le service social BibSonomy a été développé par une l'Université Kassel depuis janvier 2006. BibSonomy est un système de bookmarking social et de partage de publications. Il vise à intégrer les fonctionnalités des systèmes de bookmarking ainsi que la gestion des publications en équipe. BibSonomy offre aux utilisateurs la possibilité de stocker et d'organiser leurs bookmarks et leurs entrées de publication et prend en charge l'intégration de différentes communautés et personnes en fournissant une plate-forme sociale pour l'échange de littérature.

Les bookmarks et les entrées de publication peuvent être annotés pour aider à structurer et à retrouver des informations. Les termes descriptifs peuvent être choisis librement, l'attribution de balises (tags) de différents utilisateurs crée un vocabulaire spontané et incontrôlé appelé une folksonomie. Dans BibSonomy, la folksonomie évolue à partir de la participation de groupes de recherche, de communautés d'apprentissage et d'utilisateurs individuels, organisant leurs besoins d'information.

A des fins de recherche, la base de données BibSonomy est proposée sous forme d'un fichier dump SQL aux personnes intéressées, offrant un moyen simple de l'utiliser avec une base de données MySQL. L'accès à l'ensemble de données de BibSonomy nécessité la signature d'un contrat de licence. Nous avons utilisé la version $2017 - 01 - 01$ [4] de

3. http ://www.bibsonomy.org/

4. Knowledge and Data Engineering Group, University of Kassel : Benchmark Folksonomy Data from BibSonomy, version of January 01st, 2017. http ://bibsonomy.org/

l'ensemble de données BibSonomy. L'ensemble de données contient des bookmarks publics et des postes de publications organisés en quatre fichiers : tas, signet, bibtex et relation. Nous avons utilisé le fichier bibtex qui contient des informations sur les données BibTeX issues de publications scientifiques. La manipulation des données a été effectuée à l'aide des commandes Mysql. Un pré-traitement des données est effectué sur le fichier pour n'extraire que les données utiles aux expérimentations et pour filtrer les données non utilisées qui ne contiennent pas d'informations sur les sources d'information utilisées dans les tests. L'ensemble de données réduit qui en résulte est décrit plus en détail dans Table 6.3, où les éléments sont les documents des sources présentées dans Table 6.1 qui sont annotés par les utilisateurs.

Table 6.3 – Caractéristiques de l'ensemble de données sociales.

Données de test de BibSonomy			
Utilisateurs	**Éléments (URLs distinctes)**	**Tags individuels**	**Tags distincts**
2807	95014	254732	38291

La Table 6.4 récapitule pour chaque source le nombre de balises associées.

Table 6.4 – Nombre de balises associées aux sources.

Données de test de BibSonomy		
Numéro de la source	**Nom de la source**	**Nombre de balises**
1	ACM	70387
2	Elgar	421
3	IEEE	33628
4	IOP	236
5	JSTOR	6431
6	RSC	958
7	Sciencedirect	135338
8	Springerlink	7333

Nous donnons en annexe des exemples de balises associées aux sources d'information utilisées dans les tests (Table A.2).

5.3 Mesures d'évaluation

Les mesures d'évaluation des méthodes de sélection des sources sont généralement basées sur le rappel et la précision. En raison de l'indisponibilité des jugements de pertinence concernant le nombre total de documents pertinents disponibles dans les sources d'information utilisées dans les expériences, l'évaluation basée sur la précision est utilisée pour évaluer les performances des approches de sélection des sources, elle est donnée par l'équation suivante.

$$Précision = \frac{nombre\ de\ sources\ pertinentes}{nombre\ de\ sources\ séléctionnées\ (k)} \tag{6.9}$$

Deux autres mesures sont également utilisées pour évaluer la performance de l'approche proposée, à savoir :

— La moyenne de la précision moyenne (MAP : Mean Average Precision) qui est la précision moyenne sur plusieurs requêtes/classements.

— Le rang réciproque moyen (MRR : Mean Reciprocal Rank) : Cette métrique est utile pour montrer comment les meilleures sources pertinentes sont classées dans la position la plus élevée. Mathématiquement, cela est donné par :

$$MRR = \frac{1}{|Q|}\sum_{i=1}^{|Q|}\frac{1}{rank_i} \tag{6.10}$$

où :
$|Q|$ est le nombre total de requêtes.
$rank_i$ est le rang du premier résultat pertinent.

La pertinence d'une source s_j dépend de la paire (u_i, q) (l'utilisateur u_i soumet la requête q), c'est-à-dire les jugements de pertinence personnelle nécessaires pour évaluer toute approche de sélection des sources personnalisée. Pour construire ces jugements de pertinence, nous avons exploité des données sociales et nous avons fait l'hypothèse suivante : toute source s_j marquée par u_i avec au moins un terme de la requête q est considérée comme pertinente pour la paire (u_i, q). Ces jugements de pertinence nécessitent des efforts considérables pour générer les requêtes de test et étiqueter les sources pertinentes.

Le but de cette évaluation est de vérifier, pour chaque requête et chaque utilisateur, si une source étiquetée comme pertinente (c'est-à-dire la source que l'utilisateur a annotée en utilisant les termes de la requête) apparaît en premier rang de son résultat final comme une bonne solution.

Les cinq algorithmes CORI, Taily, GASS, SaoudAlgo et PGASS-LDA sont comparés à l'aide de requêtes de test composées de 2 à 6 termes chacune et d'un certain nombre d'utilisateurs sélectionnés dans l'ensemble de données sociales.

La précision moyenne de ces algorithmes est calculée pour 15 utilisateurs et 12 requêtes de test, pour chaque utilisateur et chaque requête la précision donnée par l'équation 6.9 est calculée puis la moyenne sur 12 requêtes de test est calculée avant de calculer la moyenne de précision finale sur 15 utilisateurs. Notez que les requêtes de test et les utilisateurs sont choisis avec soin afin de pouvoir montrer l'amélioration des performances de l'approche proposée. Les utilisateurs sont sélectionnés à partir de l'ensemble de données sociales utilisé dans les expérimentations, et les requêtes sont générées en tenant compte du contenu des sources et des balises d'utilisateurs.

5.4 Construire le modèle LDA

L'implémentation Java de LDA (JGibbLDA [5]) [122] est utilisée pour générer des thèmes intérêts cachés des utilisateurs. Cette implémentation repose sur la methode d'inference d'échantillonnage de Gibbs (Gibbs Sampling [69]) pour apprendre les distributions, ce qui nécessite les paramètres suivants. z : nombre de thèmes, β et α, les a

5. JGibbLDA, http ://Jgibblda.sourceforge.net/

priori de Dirichlet, et N, nombre d'itérations. Nous définissons les valeurs par défaut des hyper-paramètres, $\alpha = \frac{50,0}{z}$ et $\beta = 0,1$, où z est le nombre de sujets (thèmes) considérés ($z = 100$). Pour toutes les expériences, les opérations LDA sont exécutées sur $N = 1000$ itérations d'échantillonnage de Gibbs. Le nombre de mots les plus probables pour chaque thème est fixé à 20.

Les résultats du modèle LDA construit est constitué des fichiers suivants :

— $< model\_name >$ *.phi* : ce fichier contient les distributions de probabilité mot-thème. Chaque ligne est un thème et chaque colonne est un mot du vocabulaire

— $< model\_name >$ *.theta* : ce fichier contient les distributions de probabilité thème-utilisateur. Chaque ligne est un utilisateur et chaque colonne est un thème.

— $< model\_file >$ *.twords* : ce fichier contient *twords* mots les plus probables de chaque thème. Le nombre de mots *twords* est spécifié dans la commande.

En utilisant les modèles LDA précédemment estimés, les probabilités qu'une source appartienne à chacun des thèmes d'intérêt sont déduites. La source peut alors être représentée avec un vecteur de distribution sur les thèmes d'intérêt. Nous obtenons alors les vecteurs de probabilités $D(s_j)$ pour chaque source.

6 Résultats expérimentaux et discussion

Dans les expériences, nous avons fait varier le nombre de sources sélectionnées ($k = 2, 4, 6, 7$) parmi les huit (8) sources. L'approche proposée est évaluée sur 15 utilisateurs et 12 requêtes de test, qui présentent $12 * 15 = 180$ cas à évaluer. La Table A.3 en annexe montre les requêtes choisies pour nos expérimentations. Les mêmes utilisateurs et requêtes de test sont utilisés pour évaluer les modèles de référence. Dans cette section, nous analysons et discutons les points suivants : (1) l'impact du paramètre λ sur les performances de l'algorithme proposé PGASS-based-LDA, (2) la comparaison des performances de l'approche proposée avec les quatre modèles de référence et (3) complexité temporelle de l'approche proposée par rapport aux modèles de référence.

6.1 Impact du paramètre λ

Le paramètre λ de l'algorithme PGASS-based-LDA est utilisé pour évaluer l'adéquation de chaque solution (voir équation 6.3). Il montre l'effet de l'intégration des thèmes d'intérêt des utilisateurs lors de l'évaluation de la pertinence d'une source. En faisant varier les valeurs λ dans la plage $[0, 1]$, nous pouvons déduire si la pertinence de la source est plus ou moins liée aux intérêts de l'utilisateur ou à la requête. Si seules les sources proches des intérêts des utilisateurs sont prises en compte dans l'évaluation (λ= 1), alors le résultat peut être un ensemble petit ou vide alors que de bons résultats peuvent être trouvés dans d'autres sources qui sont peu ou rarement utilisées par un utilisateur, leur importance n'est pas perçue par l'utilisateur même si elles sont pertinentes pour sa requête. Nous avons testé l'effet de ce paramètre avec les valeurs $\lambda = 0.1$, $\lambda = 0.4$ et $\lambda = 0.55$. La Table 6.5 montre les performances de l'approche proposée en termes de précision pour $k = 2$, 4, 6 et 7. Lorsque $\lambda = 0.1$, l'approche proposée offre des performances élevées sur les ensembles de données de test(voir Figure 6.6). Nous avons défini $\lambda = 0.1$ pour comparer les approches de sélection des sources.

Table 6.5 – Précision moyenne de l'algorithme PGASS-based-LDA sur les données de test SNDL (en variant λ).

	Nombre de sources sélectionnées			
λ	$k = 2$	$k = 4$	$k = 6$	$k = 7$
$\lambda = 0.1$	**0.5778**	**0.4417**	**0.3250**	**0.2802**
$\lambda = 0.4$	0.5722	0.4111	0.3185	0.2794
$\lambda = 0.55$	0.5417	0.4028	0.3158	0.2762

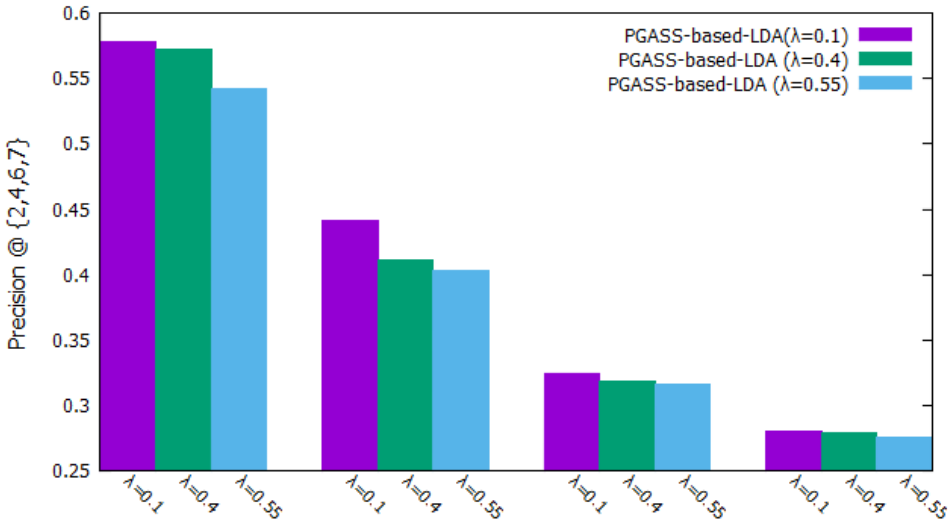


Figure 6.6 – Estimation de la valeur du paramètre λ pour l'algorithme PGASS-based-LDA sur les données de test SNDL.

6.2 Comparaison de différentes approches de sélection des sources

La Table 6.6 montre P@{2,4,6,7}, MAP et MRR des cing algorithmes, à savoir PGASS-based LDA, GASS, CORI, Taily et SaoudAlgo.

Table 6.6 – Comparaison des résultats de différents algorithmes sur des ensembles de données SNDL.

Algorithmes	**P@2**	**P@4**	**P@6**	**P@7**	**MAP**	**MRR**
PGASS-based-LDA (λ=0.1)	**0.5778**	**0.4417**	**0.3250**	**0.2802**	**0.4061**	**0.9123**
GASS	0.4639	0.3805	0.3148	0.2762	0.3588	0.8672
CORI	0.3305	0.3722	0.2898	0.2635	0.3140	0.6994
Taily (n=400, v=50)	0.3861	0.3555	0.2917	0.2643	0.3302	0.7662
SaoudAlgo ($\alpha = 0.1$)	0.45	0.3839	0.3148	0.2769	0.3577	0.7784

Les résultats sur les ensembles de données SNDL ont indiqué que l'algorithme proposé (PGASS-based-LDA) offre de meilleures performances rapport aux quatre autres

algorithmes. Cela montre que la prise en compte des informations des utilisateurs dans le processus de sélection des sources améliore la précision des résultats par rapport aux solutions non personnalisées (CORI, GASS et Taily) (voir la Figure 6.7). Et par rapport à l'approche personnalisée (SaoudAlgo), l'approche proposée est plus efficace que l'approche SaoudAlgo, cela est dû à la modélisation des thèmes d'intérêt des utilisateurs à partir des données des balises sociales au lieu d'utiliser directement les données brutes des balises.

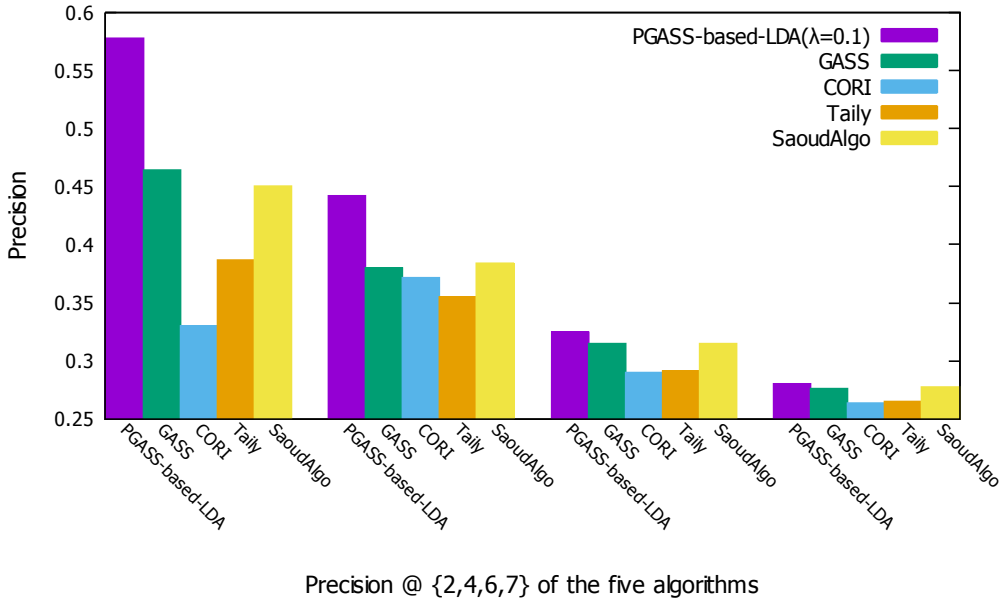


Figure 6.7 – Comparaison des algorithmes de sélection des sources sur les données de test SNDL

Les résultats ont également indiqué que l'algorithme PGASS-based-LDA était plus efficace en termes de MRR ce qui explique que les sources les plus intéressantes pour l'utilisateur montent dans le classement, elles changent de position par rapport aux positions initiales obtenues en utilisant les quatre autres algorithmes (CORI, Taily, GASS et SaoudAlgo) (Figure 6.8).

Les résultats des expérimentations montrent aussi que l'application de l'algorithme génétique offre de meilleures performances que les algorithmes CORI, SaoudAlgo et Taily.

6.3 Complexité de l'approche proposée

L'algorithme génétique s'exécute par itérations (ou générations). Initialement, un ensemble de solutions est généré aléatoirement (appelé population). Des opérations de croisement et de mutation sont effectuées sur les solutions à chacune des itérations. Les k meilleures solutions, évaluées à l'aide de la fonction de fitness, sont conservées dans la population pour la génération suivante. Après la dernière itération, la meilleure solution (par rapport à la fonction de fitness) est trouvée. La performance des algorithmes génétiques est généralement mesurée par le nombre d'évaluations de la fonction de fitness effectuées au cours d'une course. Pour des tailles de population fixes, le cas habituel dans les implémentations des algorithmes génétiques, le nombre d'évaluations de la fonction de fitness est donné par le produit de la taille de la population et du nombre de générations. [102]. Notons ici que le coût en temps d'une génération dépend des opérations internes (croisements, mutation, génération de solutions aléatoires, etc.) qui sont généralement simples

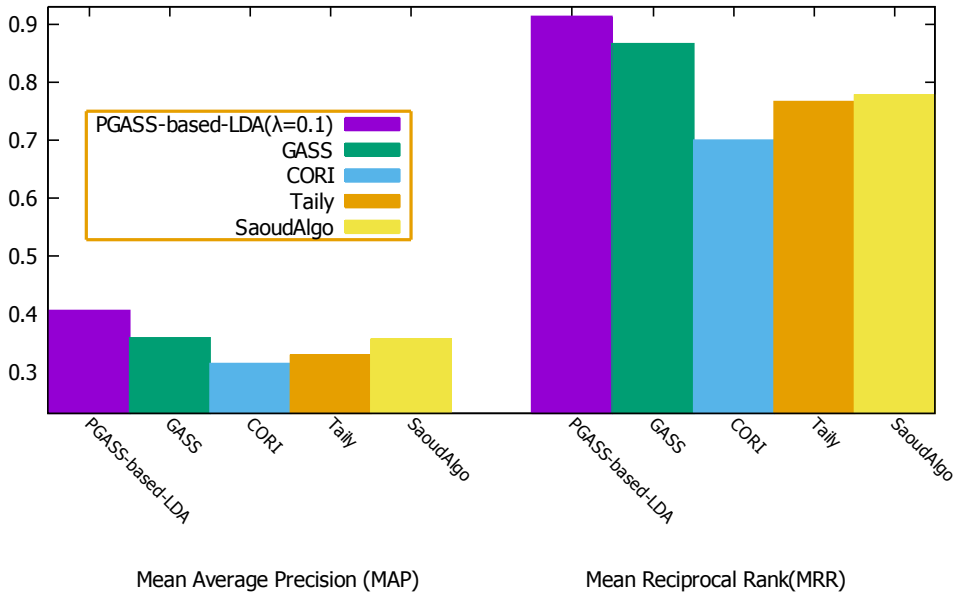


Figure 6.8 – MAP et MRR des algorithmes de sélection des sources sur les données de test SNDL

à mettre en œuvre, et également dépendantes du problème. En général, elles dépendent de la taille d'une solution.

Le temps d'exécution d'un algorithme génétique dépend également du nombre de générations [102]. Typiquement, on veut s'arrêter quand on converge vers une solution qui n'est guère améliorée. Comment trouver le nombre d'itérations qui garantissent cela ? il existe des analyses probabilistes pour trouver le temps de convergence moyen [115]. Dans de nombreux cas, le nombre d'itérations dans un algorithme génétique est déterminé empiriquement

Au cours de nos expérimentations, nous avons fait varier le nombre d'itérations de l'algorithme proposé et nous avons vérifié les résultats obtenus pour 12 requêtes de test. L'algorithme proposé converge vers la solution optimale lorsque l'algorithme atteint le nombre d'itérations (Nb.Iter) égal à 500, 100 et 50, dans chacun de ces cas, le coût en temps de l'algorithme est calculé. Le temps d'exécution est également calculé pour les autres algorithmes utilisés dans l'évaluation des performances. La Table 6.7 montre le temps d'exécution moyen de chaque algorithme pris pour répondre à une requête utilisateur.

Table 6.7 – Complexité temporelle des cinq algorithmes

Algorithmes	**Temps d'exécution (secondes)**		
	Nb.Iter=500	Nb.Iter=100	Nb.Iter=50
PGASS-based-LDA	102.43	20.86	10.72
GASS	107,16	21.36	10.92
CORI		0.042	
Taily		5.83	
SaoudAlgo		4.79	

Les approches basées sur des algorithmes génétiques (PGASS-based-LDA et GASS) sont plus complexes que CORI, Taily et SaoudAlgo en termes de temps mais offrent de meilleures performances en termes de qualité des solutions générées (comme indiqué

dans Table 6.6). Pour un nombre d'itérations égal à 50, les algorithmes génétiques ont convergé vers l'optimum avec un temps d'exécution acceptable (environ 10 secondes) ce qui démontre les performances des algorithmes proposés.

7 Conclusion

Dans ce chapitre, nous avons étudié l'application de l'algorithme génétique et de la modélisation de sujets LDA pour la personnalisation de la recherche d'information dans un environnement multi-sources. Nous avons proposé une approche de sélection des sources multidimensionnelle basée sur LDA, qui prend en compte les thèmes d'intérêt des utilisateurs dans les réseaux sociaux.

Des thèmes d'intérêt général des utilisateurs sont découverts à partir de balises associées aux sources d'information à l'aide du modèle LDA. Le modèle LDA a aussi permis de déduire les sources les plus proches des intérêts de l'utilisateur. Un algorithme génétique est ensuite utilisé pour trouver la sélection quasi-optimale de sources qui maximise la similarité entre la solution et la requête de l'utilisateur en tenant compte de ses thèmes d'intérêt.

Les résultats des expérimentations sur les ensembles de données des sources d'information de SNDL et des données sociales de BibSonomy ont montré que l'approche proposée offre une bonne précision comparée aux approches de sélection des sources personnalisées et non personnalisées de l'état de l'art en utilisant les métriques Precision@{2,4,6,7}, MAP et MRR.

Notons que le modèle LDA est construit sur un vocabulaire composé uniquement de tags issus des utilisateurs du système de tagging social, il serait plus intéressant d'utiliser également le contenu des documents annotés par les utilisateurs afin d'améliorer la précision des thèmes générés. Un prétraitement des données serait également nécessaire pour une meilleure représentation des thèmes. A noter aussi que la relation utilisateur-source est exploitée pour la sélection des sources. Les relations utilisateur-utilisateur et source-source peuvent aussi être utilisées pour déterminer des utilisateurs et des sources similaires, ce qui peut contribuer à améliorer la sélection des sources.

L'évaluation des performances des systèmes de recherche personnalisés nécessite des jugements de pertinence personnelle qui sont subjectifs et dépendent de l'utilisateur. Pour évaluer les performances de l'approche proposée, nous avons exploité le réseau de marquage social pour construire des jugements de pertinence personnelle à partir des tags des utilisateurs. D'autres données d'utilisateur peuvent également être utilisées, telles que des opinions ou des évaluations qui peuvent être fournies par les utilisateurs sur diverses autres plateformes sociales.

Il convient également de noter que les valeurs des paramètres de l'algorithme génétique (taille de la population, nombre de générations) ont un impact sur les performances et l'efficacité de l'algorithme. Ces paramètres dépendent de nombreux autres paramètres tels que la taille du problème et l'espace de recherche. Les valeurs de ces paramètres peuvent être ajustées dans les expérimentations pour augmenter les performances et l'efficacité de l'algorithme proposé.

Enfin, il serait intéressant d'utiliser de grandes collections de tests publiques pour montrer que la solution est évolutive et efficace dans un environnement multi-sources à grande échelle.

Chapitre 7

Découverte de connaissances à partir de l'analyse des fichiers journaux d'un système de recherche multi-sources basée sur un nettoyage en profondeur

Sommaire

1 Introduction

Dans un système de recherche multi-sources, comprendre les intérêts des différents utilisateurs est essentiel pour améliorer la recherche et adapter les résultats en fonction de chaque profil d'utilisateur. Des informations intéressantes caractérisant les utilisateurs peuvent être cachées dans de gros fichiers journaux (log files), qui doivent être extraits pour créer un profil précis de chaque utilisateur.

Les comportements des utilisateurs lors de l'utilisation du système sont enregistrés dans un format spécifique dans les fichiers journaux. Un fichier journal peut contenir toutes les traces d'accès de tous les utilisateurs, telles que l'identité de l'utilisateur, la date et l'heure de l'accès, les requêtes soumises au système, les ressources récupérées, etc. La taille de ces fichiers journaux peut atteindre des dimensions prodigieuses. De plus, les fichiers journaux contiennent des données brutes qui sont inutilisables dans leurs états, par conséquent, des techniques d'analyse efficaces sont nécessaires pour analyser ces données et extraire les informations pertinentes.

La fouille de données d'usage du Web (Web Usage Mining (WUM), en anglais) est l'utilisation de techniques d'exploration de données (Data Mining) pour découvrir et extraire automatiquement des informations à partir de données Web [120]. L'objectif principal de la technique WUM est de découvrir des modèles d'utilisation pour comprendre les intérêts des utilisateurs [72]. En effet, les intérêts de l'utilisateur peuvent être extraits de plusieurs manière, à partir de son propre profil (par exemple, les attributs d'intérêt) ou de son comportement social (par exemple, le comportement de tagging) [10, 97], de son réseau social (par exemple, des amis) [162], ou à partir des données de journal [35]. Nous proposons dans ce chapitre l'utilisation des techniques de WUM pour analyser les fichiers journaux du système de recherche multi-source, afin d'en extraire les données pertinentes nécessaires à la modélisation du profil utilisateur. Le profil de l'utilisateur peut alors être considéré pour améliorer la recherche d'information, personnaliser les résultats de recherche pour chaque utilisateur, filtrer les informations en fonction de ses centres d'intérêt, ou recommander des sources ou des documents appropriés à un utilisateur spécifique.

2 Fouille de l'usage du web

2.1 Définition

La fouille de l'usage du web (Web Usage Mining ou Web log mining) est défini comme l'application de techniques de data mining pour l'extraction de connaissance des données de journaux Web (web log data) [92], afin de connaître le comportement des utilisateurs d'après leurs activités effectuées sur le Web.

L'objectif de Web Usage Mining (WUM) est de découvrir des patrons d'utilisation à partir des données web dans le but majeur est de mieux comprendre et servir les besoins des applications web.

2.2 Processus de la Fouille de l'usage du web

Le processus de WUM (Figure 7.1) se décompose généralement en trois étapes interdépendantes, qui sont [147] :

— Prétraitement des données.

— Découverte des motifs.
— Analyse des motifs.

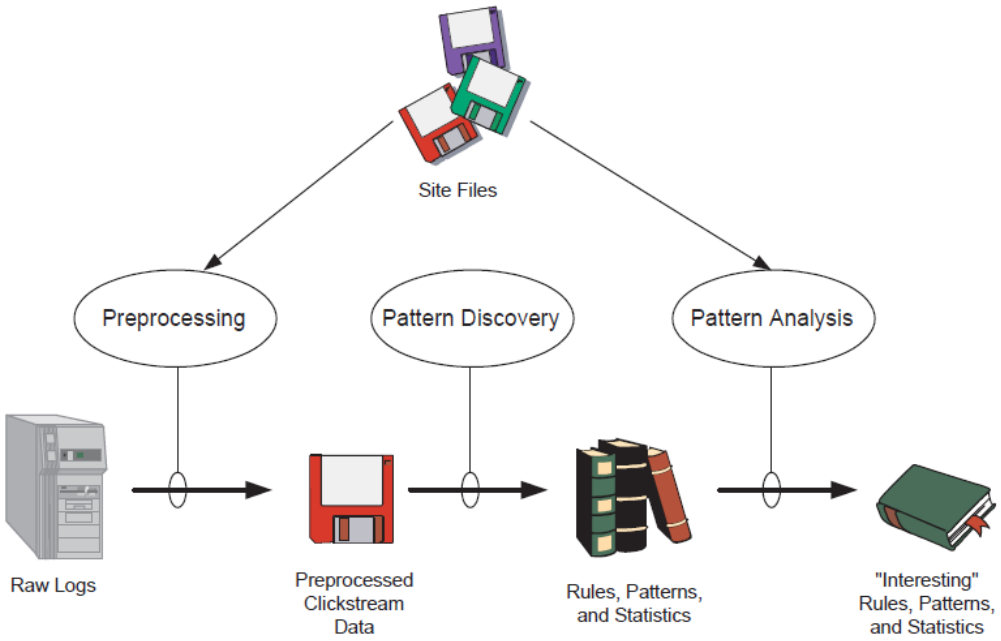


Figure 7.1 – Le processus de WUM [147].

La phase de prétraitement consiste à convertir les informations d'utilisation, de contenu et de structure, contenues dans les différentes sources de données disponibles en abstraction des données nécessaires à la découverte de modèles. La découverte de motifs repose sur des méthodes et des algorithmes développés dans plusieurs domaines tels que les statistiques, l'exploration de données, l'apprentissage automatique et la reconnaissance de formes. L'analyse des motifs est la dernière étape du processus global d'exploration des données d'utilisation.

2.2.1 Prétraitement des données

Les informations disponibles sur le Web sont hétérogènes et non structurées. Par conséquent, la phase de prétraitement est une condition préalable à la découverte des motifs [135]. Le but du prétraitement est de transformer les données brutes du flux de clics en un ensemble de profils d'utilisateurs. le prétraitement des données est devenu la tâche la plus difficile dans l'exploration de l'utilisation du Web, qui repose sur l'utilisation d'une variété d'algorithmes et de techniques heuristiques. Le prétraitement des données du journal Web comprend les étapes suivantes : nettoyage des données, identification de l'utilisateur et de la session, achèvement du chemin et identification des actions [135].

1. **Nettoyage des données :** Le nettoyage des données est un processus de suppression des éléments non pertinents tels que les fichiers jpeg, gif ou audio et les références dues aux navigations d'araignée.
2. **Identifier les utilisateurs :** L'identification des utilisateurs individuels qui accèdent à un site Web est une étape importante dans l'exploration de l'utilisation du Web. Diverses méthodes doivent être suivies pour identifier les utilisateurs. La

méthode la plus simple consiste à attribuer différents identifiants utilisateurs à différentes adresses IP.

3. **Identifier les sessions :** Cette étape consiste à segmenter les activités des utilisateurs par session. Une session utilisateur peut être définie comme un ensemble de pages visitées par le même utilisateur pendant la durée d'une visite particulière sur un site Web. Un utilisateur peut avoir une ou plusieurs sessions pendant une période. Une fois qu'un utilisateur a été identifié, le flux de clics de chaque utilisateur est divisé en clusters logiques.
4. **Achèvement de chemin :** La mise en cache côté client ou côté proxy peut souvent entraîner des références d'accès manquantes aux pages ou aux objets qui ont été mis en cache. Par exemple,si un utilisateur revient sur une page A au cours de la même session, le deuxième accès à A entraînera probablement la visualisation de la version précédemment téléchargée de A qui était mise en cache côté client, et donc, aucune demande ne sera faite au serveur. Par conséquent, la deuxième référence à A n'est pas enregistrée dans les journaux du serveur. Le processus d'achèvement du chemin consiste à reconstruire le chemin de navigation de l'utilisateur en ajoutant des demandes de page non enregistrées aux journaux du serveur. Ceci est accompli par un processus d'identification de chemin. Si la page demandée n'est pas liée à la page précédente consultée par l'utilisateur unique, la page à l'origine de la demande est identifiée à l'aide du fichier journal référent (referrer). Si la page est disponible dans l'historique de l'utilisateur, on suppose que l'utilisateur a appuyé sur le bouton de retour. Par conséquent, chaque session reflète le chemin d'accès complet, y compris les pages Web qui ont été annulées.
5. **identification des transactions :** Afin de regrouper les références de pages Web individuelles en transactions significatives pour découvrir des modèles tels que des règles d'association, un modèle sous-jacent du comportement de navigation de l'utilisateur est nécessaire [135]. Par conséquence, une étape de prétraitement supplémentaire est nécessaire, à savoir l'identification des transactions. L'identification des transactions est utilisée pour préparer les données dans le format approprié pour l'algorithme d'exploration de données spécifique à utiliser.

   Chaque session utilisateur dans un fichier de sessions utilisateur peut être visualisée de deux manières ; soit comme une transaction unique de plusieurs références de page, soit comme un ensemble de plusieurs transactions consistant chacune en une seule référence de page. Le but de l'identification des transactions est de créer des groupes de références significatifs pour chaque utilisateur. Par conséquent, la tâche d'identification des transactions consiste à diviser une grande transaction en plusieurs transactions plus petites ou à fusionner de petites transactions en moins de transactions plus importantes. Ce processus peut être étendu en plusieurs étapes de fusion ou de fractionnement pour créer des transactions appropriées pour une tâche donnée d'exploration de données [40].

2.2.2 Découverte des motifs

Une fois que les transactions des utilisateurs ont été identifiées, diverses techniques d'exploration de données sont exécutées pour la découverte des motifs. Ces méthodes représentent les approches qui apparaissent souvent dans la littérature sur la fouille de données telles que la découverte de règles d'association et de modèles séquentiels et le regroupement (clustering) et la classification, etc.

2.2.3 Analyse des motifs

L'analyse des motifs est la dernière étape de l'exploration de l'utilisation du Web. Les motifs minés ne conviennent pas aux interprétations et aux jugements. Il est donc important de filtrer les règles ou les motifs non pertinents de l'ensemble trouvé lors de la phase de découverte des motifs. À cette étape, des outils sont fournis pour faciliter la transformation de l'information en connaissance. La méthode d'analyse exacte est généralement régie par l'application pour laquelle l'exploration Web est effectuée. Le mécanisme de requête de connaissances tel que SQL est la méthode d'analyse de motifs la plus courante. Une autre méthode consiste à charger les données d'utilisation dans un cube de données pour effectuer des opérations OLAP [135]. Les techniques de visualisation, telles que les schémas graphiques peuvent souvent mettre en évidence des motifs ou la tendance générale des données.

2.2.4 Logiciels d'analyse de fichiers journaux

Malgré le type de format simple d'un fichier journal, il est très difficile d'interpréter immédiatement les informations contenues à l'état brut. Cependant, il existe de plus en plus d'outils pour faciliter l'analyse des fichiers journaux afin de générer des informations pertinentes et utiles. Plusieurs logiciels d'analyse de fichiers journaux existent, certains sont payants et d'autres open source. Parmi ces logiciels nous citons Webalizer [1], Webalizer Xtended [2], AWStats [3] et Advanced Log Analyzer [4]. La plupart de ces outils fournissent des fonctionnalités de base pour analyser les fichiers journaux, ils fournissent également des statistiques, comme le nombre d'accès à une ressource, le nombre de visiteurs, la quantité des données téléchargées, etc. Dans certains cas, une analyse approfondie est nécessaire pour extraire des informations très précises.

3 Approche proposée pour l'analyse des fichiers journaux du système de recherche multisource basée sur un nettoyage en profondeur

La démarche d'analyse proposée permet de découvrir des informations utiles qui caractérisent le profil et les intérêts des utilisateurs d'un système de recherche multi-sources. Les informations que nous considérons importantes à découvrir lors de l'analyse des fichiers journaux de ce système sont les suivantes :

— Le nom d'utilisateur
— La session
— La date et l'heure d'accès
— La requête de l'utilisateur (recherche et téléchargement)
— Les sources consultées par un utilisateur

1. http ://www.webalizer.org/
2. https ://www.patrickfrei.ch/webalizer/index.html
3. https ://awstats.sourceforge.io/
4. https ://www.abacre.com/ala/

L'approche d'analyse proposée comprend deux étapes principales, à savoir : le prétraitement des fichiers journaux et le traitement des fichiers journaux, comme le montre la Figure 7.2.

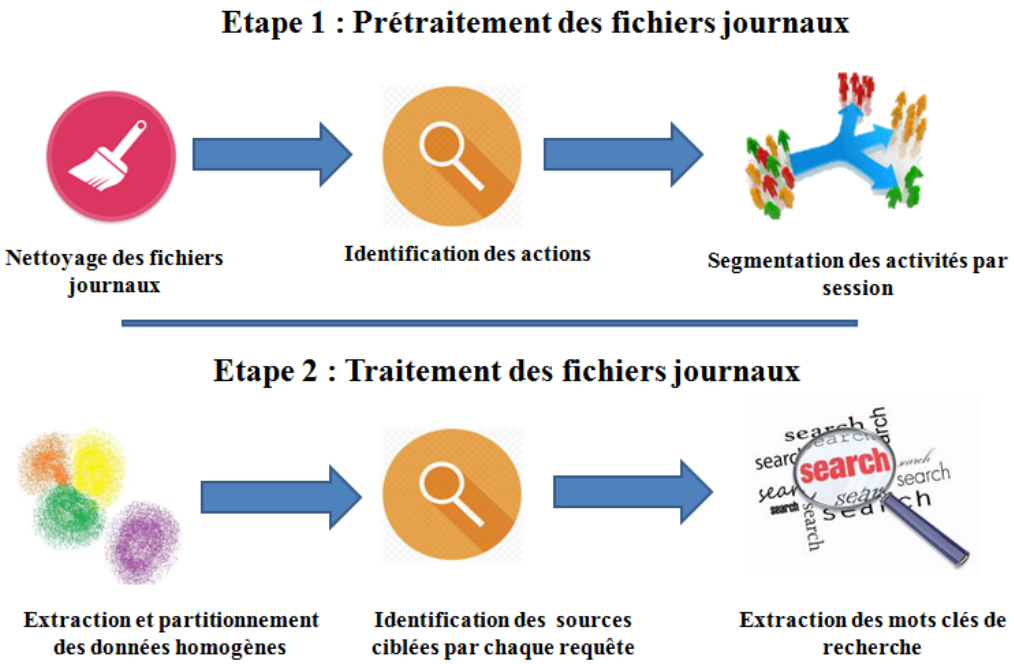


Figure 7.2 – Schéma général de l'approche d'analyse proposée.

i) La première étape, c'est-à-dire le prétraitement des fichiers journaux, prépare le fichier journal pour un traitement ultérieur. Il comprend les trois sous-étapes suivantes :

— Nettoyage des fichiers journaux

— Identification des actions

— Segmentation des activités par session

ii) La deuxième étape, à savoir le traitement des fichiers journaux, permet d'effectuer une analyse approfondie à travers les sous-étapes suivantes :

— Extraction et partitionnement des données homogènes

— Identification des sources ciblées par chaque requête

— Extraction des mots clés de recherche

Ces différentes étapes de l'approche proposée sont décrites en détail dans les sous-sections suivantes.

3.1 Prétraitement des fichiers journaux

3.1.1 Nettoyage des fichiers journaux

Les données des fichiers journaux sont généralement bruyantes et peu claires, le nettoyage des données est donc un processus essentiel pour un processus d'exploration de données efficace. Le nettoyage des données supprime les éléments non pertinents stockés dans les fichiers journaux qui peuvent ne pas être utiles pour l'analyse [40], tels que l'accès aux fichiers JPEG, GIF, Java Scripts, autres fichiers audio / fichiers vidéo, les accès non

humains (accès effectués par des robots d'indexation Web), des accès avec des codes d'état HTTP erronés, etc.

Nous considérons que les données intéressantes que l'on retrouve dans les fichiers journaux sont les requêtes soumises par les utilisateurs aux différentes sources et les documents téléchargés à partir de ces sources. Le processus de nettoyage proposé élimine toutes les données non pertinentes générées par le système et ne conserve que les données concernant les requêtes et les documents téléchargés.

Le processus de nettoyage du fichier journal est divisé en deux étapes différentes, à savoir le nettoyage conventionnel et le nettoyage en profondeur, où :

— **Nettoyage conventionnel :** est similaire aux méthodes de nettoyage utilisées dans la plupart des techniques d'exploration de données qui impliquent la suppression des fichiers multimédias, des fichiers de style, des fichiers de script java, des codes d'erreur et des requêtes du robot d'indexation.

— **Nettoyage en profondeur :** analyse la structure des requêtes et élimine celles qui ne sont pas pertinentes, c'est à dire qu'elles ne contiennent pas d'informations sur les requêtes des utilisateurs et les documents téléchargés.

Avant de décrire ces deux étapes de nettoyage, les formulations mathématiques suivantes sont données.

La description formelle du processus de nettoyage. Afin de mieux comprendre le processus de nettoyage, nous utiliserons les formulations mathématiques suivantes.
Soit F un fichier journal au format standard, comme illustré à la Figure 7.3.

IP	Session	Utilisateur	Date	Méthode	Requête	Protocole	Statut	Taille

Figure 7.3 – Le format standard d'un fichier journal.

Soit E l'ensemble des lignes (ou événements) de F.
$E = \{e_1, e_2, e_3, \ldots, e_m\}$, $e_k$un événement de F tel que $k \in [1, m]$.
Soit e_k^t un événement de F en cours de traitement.
Nous définissons un ensemble de règles de la forme :
si (*condition*) **alors** (*action*)
Tel que, la condition sera liée à un événement dans le fichier journal et l'action consiste à supprimer l'événement s'il satisfait à la condition.
Nous formulons le nettoyage des données par la règle générale suivante :
si (e_k^t vérifie la condition c) **alors** ($E = E - \{e_k^t\}$)

Nettoyage conventionnel. Cette première étape de nettoyage consiste à éliminer les lignes du fichier journal qui sont généralement inutiles dans presque tous les processus d'exploration de données car elles ne contiennent aucune information pertinente. Le nettoyage proposé du fichier journal est basé sur des règles de nettoyage que nous avons définies. Ces règles sont basées sur la structure d'un événement de fichier journal et sur un ensemble de fonctions prédéfinies. Selon le format standard du fichier journal étudié (voir Figure 7.3), nous avons défini des fonctions nécessaires au nettoyage conventionnel, ces fonctions sont décrites ci-dessous.

Statut : $e_k \longmapsto Statut(e_k)$: cette fonction renvoie la valeur de l'attribut « Statut » de l'événement e_k.

Requête : $e_k \longmapsto Requête(e_k)$: cette fonction renvoie la valeur de l'attribut « Requête » de l'événement e_k.

Taille : $e_k \longmapsto Taille(e_k)$: cette fonction renvoie la valeur de l'attribut « Taille » de l'événement e_k.

Utilisateur : $e_k \longmapsto Utilisateur(e_k)$: cette fonction renvoie la valeur de l'attribut « Utilisateur » de l'événement e_k.

Extension : $Requête(e_k) \longmapsto Extension((Requête(e_k))$: cette fonction renvoie le type du fichier ciblé par la requête associée à e_k.

Fichier : $Requête(ek) \longmapsto Fichier(Requête(ek))$: cette fonction renvoie le nom du fichier ciblé par la requête associée à e_k.

Soit D un ensemble d'extensions de fichiers, telles que :

D = {jpg, jpeg, gif, png, bmp, ico, mp3, wma, wav, ogg, mp4, mkv, avi, WebM, CSS, js}

Les règles présentées dans Table 7.1 sont définies pour un nettoyage conventionnel, qui élimine les lignes non pertinentes dans E.

Table 7.1 – Règles de nettoyage conventionnel.

Numéro de règle	Règle	Description de la règle
1	$(Statut(e_k^t) < 200 \vee Statut(e_k^t) > 399) \Rightarrow E = E - \{e_k^t\}$	Supprimer l'événement avec un code d'état supérieur à 299 ou inférieurs à 200 (code d'état HTTP ayant échoué).
2	$Taille(e_k^t) = 0 \Rightarrow E = E - \{e_k^t\}$	Supprimer l'événement qui ne contient pas d'informations pertinentes (pages utilisées pour attribuer une session, ouvrir un compte, rediriger vers une autre page, etc.).
3	$Extension(e_k^t) \subset D \Rightarrow E = E - \{e_k^t\}$	Supprimer l'événement avec l'extension $gif, jpeg, css$, etc. (fichiers multimédia).
4	$Fichier(e_k^t) = robots.txt \Rightarrow E = E - \{e_k^t\}$	Supprimer l'événement où la requête est générée par les robots d'indexation (requête de fichier $robots.txt$).
5	$Utilisateur(e_k^t) =' -' \Rightarrow E = E - \{e_k^t\}$	Supprimer l'événement où l'utilisateur n'est pas identifié.

L'algorithme 7 décrit le processus de nettoyage conventionnel des fichiers journaux du système SNDL.

Nettoyage en profondeur. Cette étape est réalisée après un nettoyage conventionnel, afin d'éliminer d'autres lignes (ou événements) qui n'ont pas d'intérêt pour cette analyse. Les événements qui ne contiennent aucune information sur les requêtes des utilisateurs et les documents téléchargés sont supprimés. Cela nécessite une analyse des requêtes dans les fichiers journaux. Le système de recherche multi-sources gèrent plusieurs sources d'information, qui utilisent différents formats de requête. Chaque source a son propre format de requête. Cette étape nécessite de comprendre la

Algorithm 7 Nettoyage Conventionnel

Input: Le fichier journal F
Output: Le fichier journal après le nettoyage F_c

while (il reste des lignes dans F) **do**
  $e_k^t = lire(F)$ {ligne courante}
  if $(Statut(e_k^t) < 200 \vee Statut(e_k^t) > 399)$ **then**
    $E = E - \{e_k^t\}$
  else if $Taille(e_k^t) = 0$ **then**
    $E = E - \{e_k^t\}$
  else if $(Extension(e_k^t) \subset D)$ **then**
    $E = E - \{e_k^t\}$
  else if $(Fichier(e_k^t) = robots.txt)$ **then**
    $E = E \cup \{e_k^t\}$
  else if $(Utilisateur(e_k^t) = \text{`} - \text{`})$ **then**
    $E = E - \{e_k^t\}$
  end if
end while
Supprimer (IP, Méthode, Statut, Protocole, Taille) de F
$F_c = F$

structure de requête de chaque source dans le système de recherche pour identifier les requêtes pertinentes et celles qui ne le sont pas et qui doivent être ignorées.

L'étude de la structure de la requête est basée sur le découpage principal d'une requête *http* , en effet, une requête *http* se décompose en plusieurs termes [5], à savoir :

- Protocole utilisé (*http* dans notre cas).
- Nom de domaine (indique quel serveur Web ou source est demandé (e)).
- Chemin d'accès au fichier (est le chemin d'accès à la ressource sur le serveur Web).
- Paramètres (les paramètres fournis au serveur Web ou à la ressource renvoyés par le serveur Web. Chaque serveur Web a ses propres règles concernant les paramètres).

La Figure 7.4 montre la structure d'une requête *http*.

Protocole	Domaine	Chemin	Paramètres

Figure 7.4 – La structure d'une requête *http*.

On distingue deux types de requêtes, à savoir :

- **Requête de téléchargement :** utilisée pour télécharger un document à partir d'une source d'information.
- **Requête de recherche :** contenant des mots clés de recherche entrés par l'utilisateur et envoyés à une source d'information.

5. *https* : $//fr.wikipedia.org/wiki/Uniform_Resource_Locator$

Chaque type de requête est représenté par un modèle qui le caractérise. Nous distinguons ainsi deux types de modèles, un modèle de recherche et un modèle de téléchargement.

1) **Le modèle de recherche :** décrit une requête de recherche contenant les attributs suivants :
   - URL du domaine (source concernée).
   - Nom de la source.
   - Liste des chemins vers les pages dédiées à la recherche.
   - Liste des paramètres de recherche (contenant les mots clés entrés par l'utilisateur).

2) **Le modèle de téléchargement :** décrit une requête de téléchargement contenant les attributs suivants :
   - URL du domaine.
   - Nom de la source.
   - Liste des paramètres de téléchargement (les marqueurs de téléchargement).

Le nettoyage en profondeur utilise également des règles de nettoyage basées sur un certain nombre de fonctions définies selon la structure de la requête *http* (voir la Figure 7.4).

Soient les fonctions prédéfinies suivantes :

Domaine : $Requête(e_k) \Rightarrow Domaine(Requête(e_k))$: cette fonction retourne la valeur de l'attribut « Domaine » de la requête associée à l'événement e_k.

Chemin : $Requête(e_k) \Rightarrow Chemin(Requête(e_k))$: cette fonction retourne la valeur de l'attribut « Chemin » de la requête associée à l'événement e_k.

Params : $Requête(e_k) \Rightarrow Params(Requête(e_k))$: cette fonction retourne la valeur de l'attribut « Paramètres » de la requête associée à l'événement e_k.

Soit D_R l'ensemble des domaines dans le modèle de recherche.

Soit D_T l'ensemble des domaines dans le modèle de téléchargement.

Soit M l'ensemble des marqueurs de téléchargement dans le modèle de téléchargement.

Soit N l'ensemble des chemins dans le modèle de recherche.

Soit P l'ensemble des paramètres du modèle de recherche.

Nous avons défini des règles de nettoyage en profondeur présentées dans Table 7.2.

L'algorithme 8 décrit le processus de nettoyage en profondeur des fichiers journaux du système SNDL.

3.1.2 Identification des actions

Cette étape consiste, après avoir conservé uniquement les événements pertinents, à identifier le type de chaque événement. Un événement pouvant être, après le processus de nettoyage, soit un événement de recherche, soit un événement de téléchargement. Cette étape est importante pour la prochaine étape.

Le nettoyage en profondeur permet, en plus d'éliminer les requêtes inutiles, de reconnaître si la requête de l'événement en cours de traitement est une requête de recherche ou de téléchargement ou autre.

Table 7.2 – Règles de nettoyage en profondeur.

Numéro de règle	**Règle**	**Description de la règle**
1	$(Domaine(Requête(e_k^t)) \notin D_R \cup D_T) \Rightarrow E = E - \{e_k^t\}$	Supprimer l'événement où la requête n'est associée à aucune source.
2	$Chemin(Requête(e_k^t)) = \emptyset \wedge Params(Requête(e_k^t) = \emptyset)) \Rightarrow E = E - \{e_k^t\}$	Supprimer l'événement où la requête ne contient ni chemin ni paramètre.
3	$(Domaine(Requête(e_k^t)) \subset D_T \wedge \forall x \in M, x \notin Chemin(Requête(e_k^t))) \wedge (Domaine(Requête(e_k^t)) \subset D_R \wedge (Chemin(Requête(e_k^t)) \notin N \vee \forall y \in P, y \notin Params(Requête(e_k^t)))) \Rightarrow E = E - \{e_k^t\}$	Supprimer l'événement où la requête n'est ni une requête de téléchargement ni une requête de recherche.

Algorithm 8 Nettoyage en profondeur

Input: Le fichier journal après le nettoyage conventionnel F_c
Output: Le fichier journal après le nettoyage F_n

```
while (il reste des lignes dans F_c) do
    e_k^t = lire(F_c) {la ligne en cours}
    if règle1 = vrai then
        E = E - {e_k^t}
    else if règle2 = vrai then
        E = E - {e_k^t}
    else if règle3 = vrai then
        E = E - {e_k^t}
    end if
end while
F_n = F_c
```

Dans la *règle* 3 de nettoyage en profondeur, nous prenons la proposition suivante : $(Domaine(Requête(e_k^t)) \subset D_T \wedge \forall x \in M, x \notin Chemin(Requête(e_k^t)))$.

Cette proposition peut se décomposer en deux sous propositions :

1 : $Domaine(Requête(e_k^t)) \subset D_T$.

2 : $\forall x \in M, x \notin Chemin(Requête(e_k^t))$.

Nous en déduirons donc que e_k^t est un événement de téléchargement si et seulement si $(1 = Vrai)$ et $(2 = Faux)$.

De même pour une requête de recherche, la proposition suivante : $(Domaine(Requête(e_k^t)) \subset D_R \wedge (Chemin(Requête(e_k^t)) \notin N \vee \forall y \in P, y \notin Params(Requête(e_k^t))))$ peut être décomposé en trois sous-propositions : **1 :** $Domaine(Requête(e_k^t)) \subset D_R$.

2 : $Chemin(Requête(e_k^t)) \notin N$.

3 : $\forall y \in P, y \notin Params(Requête(e_k^t))$.

Nous en déduirons donc que e_k^t est un événement de recherche si et seulement si $(1 = Vrai)$ et $(2 = Faux)$ et $(3 = Faux)$.

3.1.3 Segmentation des activités par session

Cette étape consiste à segmenter les activités des utilisateurs par identifiant de session (ID de session) avant d'extraire les données, afin de distinguer les différentes visites de chaque utilisateur.

À l'instant t, une session est attribuée à un utilisateur connecté au système. L'utilisateur conservera cet identifiant tout au long de son activité, lorsque celle-ci est terminée, ou en fermant le navigateur, une nouvelle session lui est attribuée. La probabilité que deux utilisateurs aient le même ID de session tend vers 0. En raison de la complexité de la génération d'identifiant, on suppose ci-dessous que si l' ID de session est connu, alors le nom de l'utilisateur est également connu.

3.2 Traitement des fichiers journaux

3.2.1 Extraction et partitionnement des données homogènes

Dans cette étape, les données utiles sont extraites et partitionnées en deux ensembles, à savoir un ensemble d'événements de recherche et un ensemble d'événements de téléchargement, tels que :

Un événement de recherche se caractérise par :

— ID de session
— Non d'utilisateur
— Nom de la source
— La date et l'heure auxquelles l'utilisateur a accédé au système
— Requête de recherche
— Liste des mots clés saisis par l'utilisateur

Un événement de téléchargement se caractérise par :

— ID de session
— Nom d'utilisateur
— La date et l'heure auxquelles l'utilisateur a accédé au système
— Nom de la source
— Requête de téléchargement

3.2.2 Identification des sources ciblées par chaque requête

Cette étape consiste à identifier la source visée par chaque requête. Pour cela, nous avons utilisé à la fois des modèles de recherche et de téléchargement pour identifier le nom de la source en fonction du nom de domaine dans la requête URL.

3.2.3 Extraction de mots clés de recherche

Cette dernière étape consiste à extraire les mots clés saisis par l'utilisateur, à partir de sa requête de recherche. Étant donné que le format de requête est différent pour chaque source, le modèle de recherche est utilisé pour extraire les mots-clés de la requête de recherche de l'utilisateur.

La requête utilisateur peut contenir des caractères spéciaux, tels que " : ", "/", ".", qui sont encodés au format URL. Pour ce faire, un décodage de l'URL est effectué pour convertir ces caractères au format d'origine.

4 Implémentation

L'approche proposée est implémentée dans une architecture client-serveur, comme le montre la Figure 7.5. Le client (utilisateur, administrateur ou mainteneur) peut accéder à l'application via un compte utilisateur. Un serveur distant héberge une base de données qui peut contenir une très grande quantité de données.

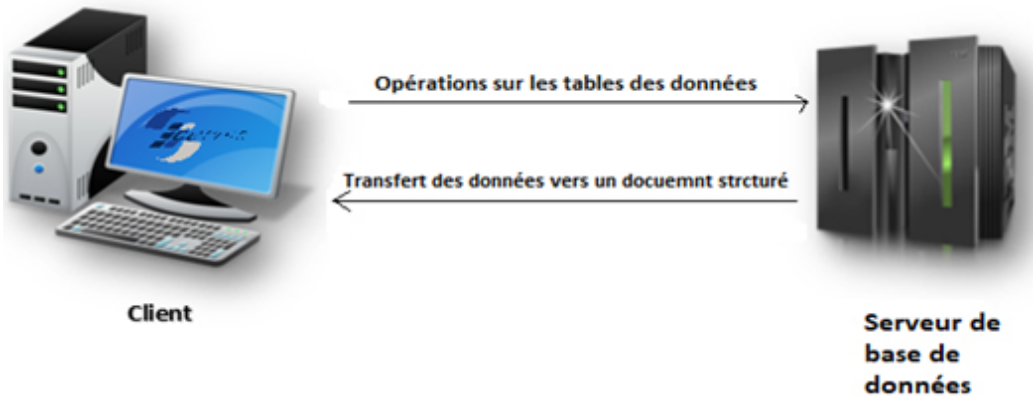


Figure 7.5 – L'architecture du système proposé.

Nous avons identifié trois acteurs principaux qui interagissent avec le système, à savoir :

— **Utilisateur** : peut uniquement télécharger les données extraites des fichiers journaux et consulter les résultats.

— **Administrateur** : responsable de l'analyse des fichiers journaux et la génération des rapports.

— **Mainteneur** : responsable de la gestion et de la maintenance de la base de données (par exemple, l'ajout d'une nouvelle source d'information).

4.1 Données utilisées dans les tests

Pour nos expérimentations, nous avons utilisé les fichiers journaux de la plateforme de documentation en ligne SNDL[6] du CERIST. L'analyse de ces fichiers permet de générer les données qui caractérisent les utilisateurs de la plateforme.

Les fichiers journaux du système SNDL sont tous au format standard. Un événement représente une ligne dans un fichier journal. Nous montrons un exemple d'événement dans un fichier journal SNDL :

41.98.29.193 ilWvTPtsaK6yDwC sahnounefoudil [20/Feb/2013 :00 :00 :18 +0100] "GET *http* : *//www.springermaterials.com* : 80/*docs*/*pdf*/10000866_118.*html*HTTP/1.1" 200 15413

Cet événement est composé des champs suivants :

— **41.98.29.193** : l'adresse IP de l'utilisateur.

— **ilWvTPtsaK6yDwC** : l'identifiant de session.

— **sahnounefoudil** : le nom de l'utilisateur.

6. *https* : *//www.sndl.cerist.dz/*

— **[20/Feb/2013 :00 :00 :18 +0100]** : la date et l'heure d'envoi de la requête.
— **GET** : la méthode du protocole utilisé pour la requête.
— **http** : //**www.springermaterials.com** : **80/docs/pdf/10000866_118**. **html** : la requête de l'utilisateur destinée à une source SNDL.
— **HTTP/1.1** : le protocole utilisé et sa version.
— **200** : le statut de la recherche ou le code de retour du serveur.
— **15413** : la taille de la page demandée en octets.

Un exemple de requête *http* destinée à la source Springer Link est illustré ci-dessous :

$$http : //link.springer.com : 80/search?facet - discipline = \\ \%22Physics\%22$$

Cette requête est une requête de recherche composée des champs suivants :

— **Protocole** : *http*.
— **Domaine** : link.springer.com (contient le non de la source)
— **Chemin** : /search.
— **Paramètres de recherche** : facet-discipline (contenant les mots clés de la requête).

Prenons un exemple de requête de téléchargement destinée à la source CAIRN :

$$http : //www.cairn.info : 80/load\_pdf.php?ID\_ARTICLE = \\ DRS\_056\_0151$$

Le marqueur de téléchargement est le paramètre « *load_pdf*).

Deux dictionnaires de motifs sont créés, qui contiendront tous les motifs (modèles) de requête des sources SNDL (recherche et téléchargement).

— Un dictionnaire des motifs de recherche (Table A.4 de l'annexe).
— Un dictionnaire des motifs de téléchargement (Table A.5 de l'annexe).

Deux bases de données (DB) sont utilisées, l'une pour stocker les données à analyser (DB de prétraitement) et l'autre pour stocker les données extraites des fichiers journaux (DB de traitement).

Trois fichiers journaux de tailles différentes sont analysés (Table 7.3).

Table 7.3 – Fichiers journaux analysés.

Description	**Fichier journal 1**	**Fichier journal 2**	**Fichier journal 3**
Taille (Mo)	62.2	140	150

4.2 Résultats d'analyse

La Table 7.4 montre la quantité de données (nombre d'événements) avant et après le processus de nettoyage, ainsi que les données restantes dans chaque fichier. Diverses autres statistiques peuvent être générées, telles que le nombre d'événements détectés concernant l'erreur 404, les fichiers image et les fichiers JavaScript.

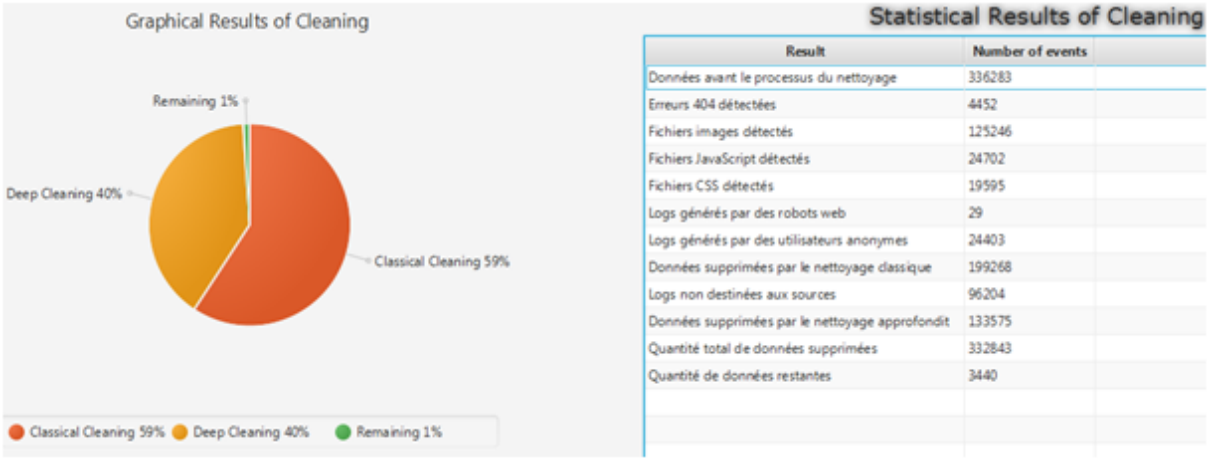
Table 7.4 – Description des trois fichiers journaux avant et après le processus de nettoyage.

Description	**Fichier journal 1**	**Fichier journal 2**	**Fichier journal 3**
Quantité de données avant nettoyage	336283	421320	412146
Quantité de données après un nettoyage conventionnel	199268	286446	285412
Quantité de données après un nettoyage en profondeur	133575	119628	122035
Quantité totale de données supprimées	332843	406074	407447
Quantité de données restantes	3440	15246	4699

La Table 7.5 montre les résultats du processus de nettoyage, illustrant la quantité de données supprimées par le nettoyage conventionnel et le nettoyage en profondeur, ainsi que les données restantes après le nettoyage dans chaque fichier journal. Nous pouvons voir que l'approche proposée élimine une grande quantité de données qui ne sont pas importantes pour l'analyse, ce qui accélère le processus d'exploration de données. Un nettoyage en profondeur peut supprimer une partie considérable des données indésirables des fichiers journaux, jusqu'à 40% du fichier journal, ce qui est bien illustré dans la Figure 7.6.

Table 7.5 – Les résultats du processus de nettoyage.

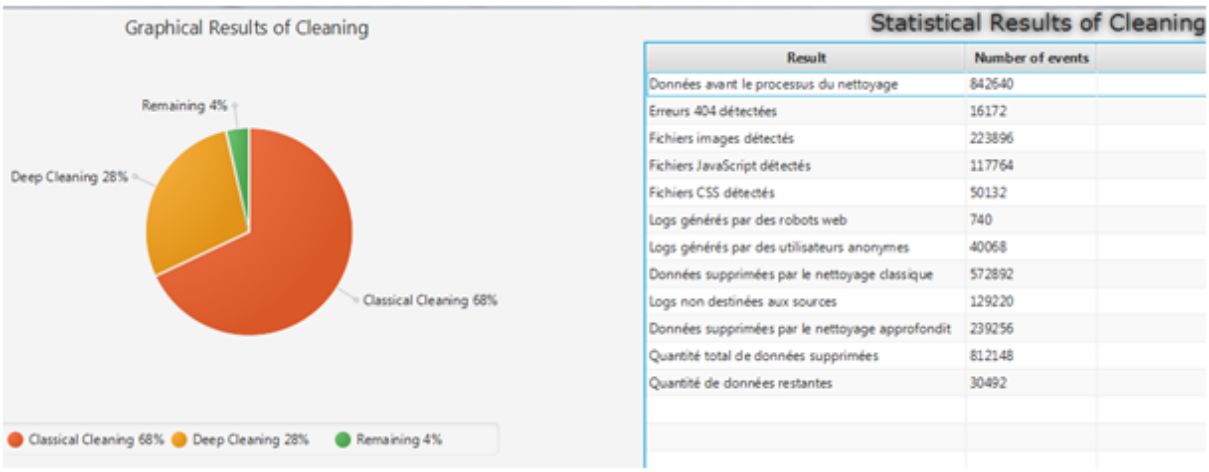
Description	**Fichier journal 1**	**Fichier journal 2**	**Fichier journal 3**
Données supprimées par le nettoyage conventionnel	59 %	68 %	69 %
Données supprimées par le nettoyage en profondeur	**40 %**	28 %	**30 %**
Données restantes	1 %	4 %	1 %



Statistical Results of Cleaning

Result	Number of events
Données avant le processus du nettoyage	336283
Erreurs 404 détectées	4452
Fichiers images détectés	125246
Fichiers JavaScript détectés	24702
Fichiers CSS détectés	19595
Logs générés par des robots web	29
Logs générés par des utilisateurs anonymes	24403
Données supprimées par le nettoyage classique	199268
Logs non destinées aux sources	96204
Données supprimées par le nettoyage approfondit	133575
Quantité total de données supprimées	332843
Quantité de données restantes	3440

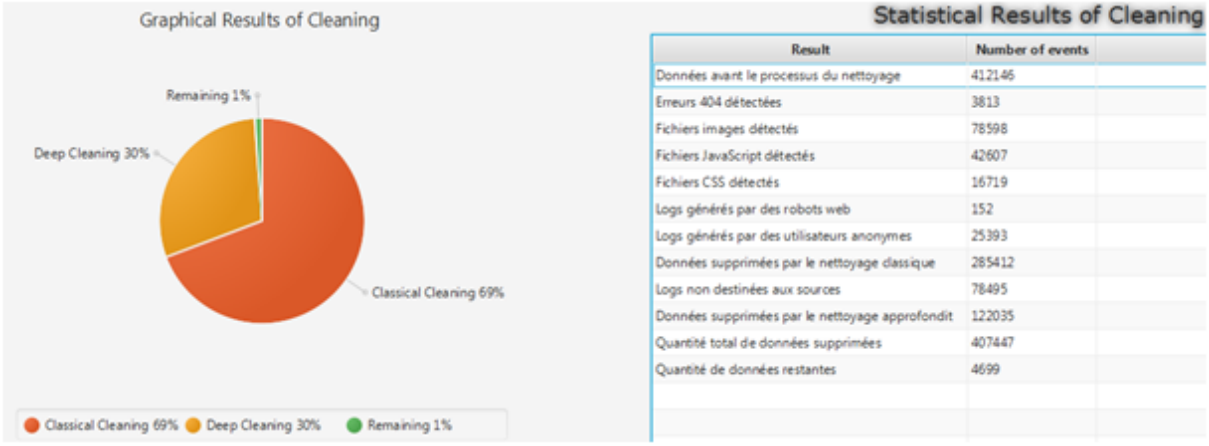
(a) Fichier 1.



Statistical Results of Cleaning

Result	Number of events
Données avant le processus du nettoyage	842640
Erreurs 404 détectées	16172
Fichiers images détectés	223896
Fichiers JavaScript détectés	117764
Fichiers CSS détectés	50132
Logs générés par des robots web	740
Logs générés par des utilisateurs anonymes	40068
Données supprimées par le nettoyage classique	572892
Logs non destinées aux sources	129220
Données supprimées par le nettoyage approfondit	239256
Quantité total de données supprimées	812148
Quantité de données restantes	30492

(b) Fichier 2.



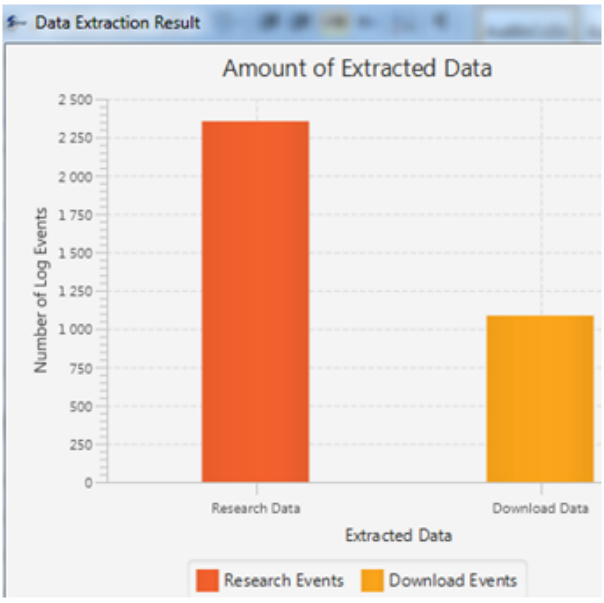
Statistical Results of Cleaning

Result	Number of events
Données avant le processus du nettoyage	412146
Erreurs 404 détectées	3813
Fichiers images détectés	78598
Fichiers JavaScript détectés	42607
Fichiers CSS détectés	16719
Logs générés par des robots web	152
Logs générés par des utilisateurs anonymes	25393
Données supprimées par le nettoyage classique	285412
Logs non destinées aux sources	78495
Données supprimées par le nettoyage approfondit	122035
Quantité total de données supprimées	407447
Quantité de données restantes	4699

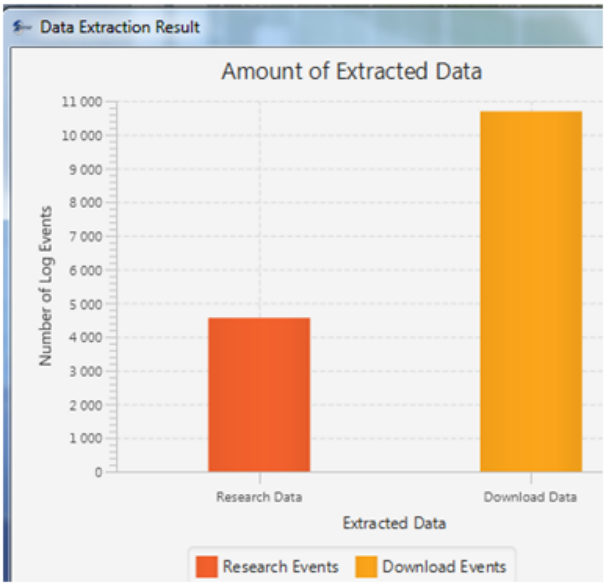
(c) Fichier 3.

Figure 7.6 – Résultats graphiques du processus de nettoyage.

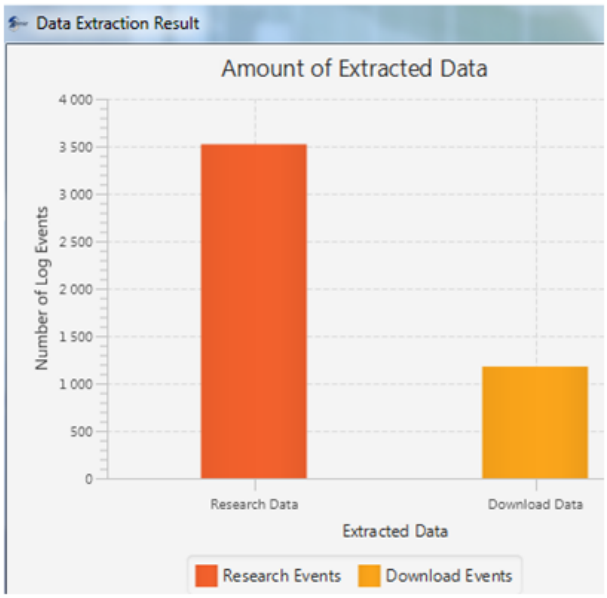
La Figure 7.7 montre la quantité de données extraites des trois fichiers, à savoir les données de recherche et les données de téléchargement. Les données extraites sont structurées sous forme des données Mysql qui peuvent être téléchargées sous forme de fichier Excel pour une utilisation ultérieure ou directement exploitées à l'aide du langage SQL.



(a) Fichier 1.



(b) Fichier 2.



(c) Fichier 3.

Figure 7.7 – Résultats des données extraites des trois fichiers journaux.

L'analyse des fichiers journaux permet également de préparer des statistiques pertinentes fournissant des informations sur les utilisateurs de la plateforme SNDL, tels que les mots clés les plus utilisés, leurs activités sur les sources SNDL, les documents les plus téléchargés et les documents ou sources les plus demandés. Nous présentons dans la figure 7.8 les résultats statistiques des 15 premiers mots clés utilisés par les utilisateurs de la plateforme SNDL.

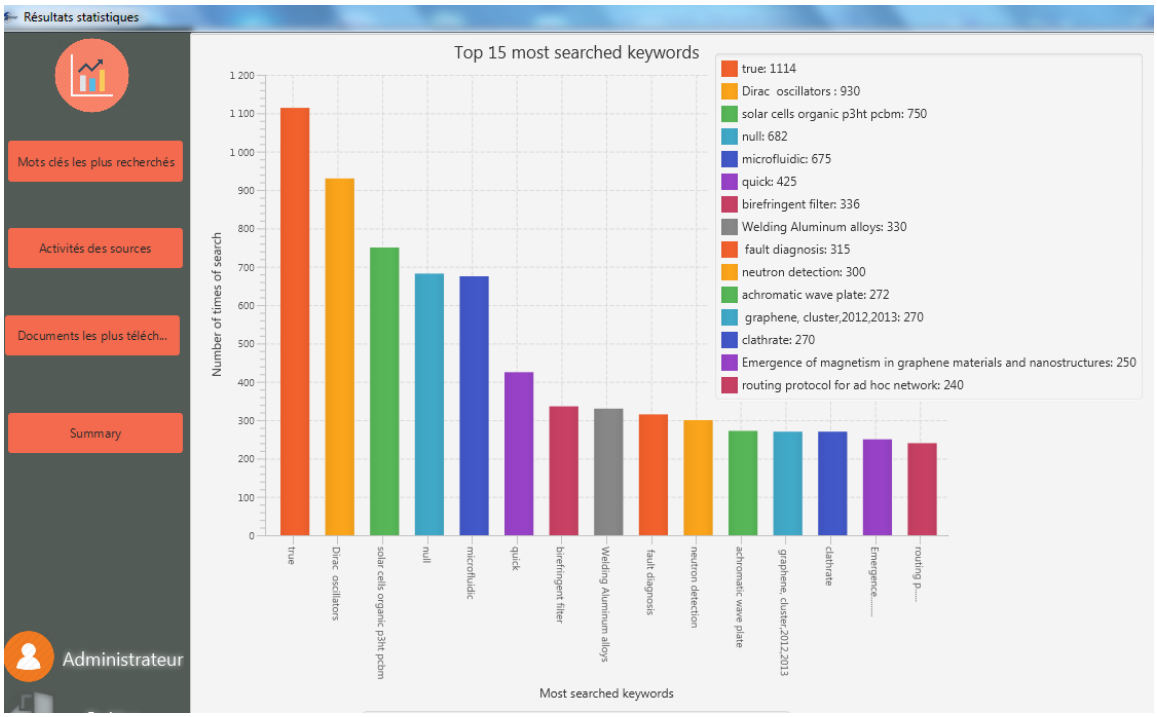


Figure 7.8 – Statistiques sur le Top 15 des mots clés les plus recherchés.

La Figure 7.9 montre la fréquence d'accès aux sources, en calculant le nombre d'utilisateurs visitant chacune des sources SNDL.

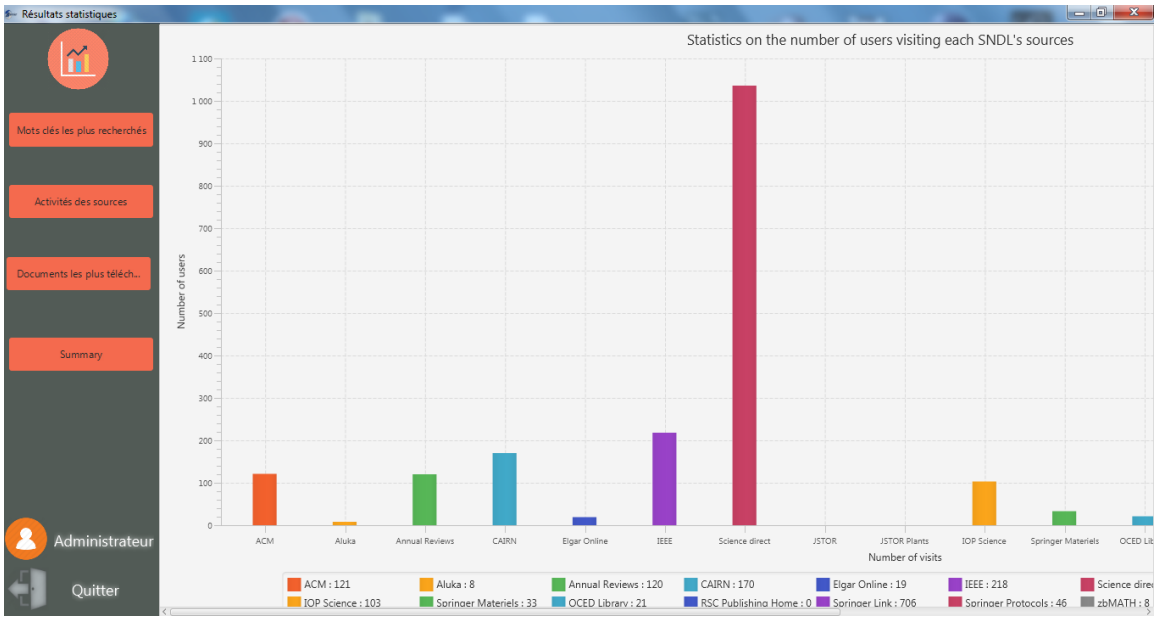


Figure 7.9 – Fréquence d'accès aux sources.

5 Conclusion

Les fichiers journaux sont une source de données pertinente qui peut être exploitée dans diverses applications telles que la personnalisation d'un système de recherche. Dans ce chapitre, nous avons proposé une approche basée sur les techniques de fouille de données

Web pour analyser les comportements des utilisateurs en explorant les fichiers journaux d'un système de recherche multi-source. L'approche proposée consiste en deux étapes, l'étape de prétraitement des données de journal qui supprime les données indésirables et l'étape de traitement des données du journal qui extrait les données pertinentes décrivant les intérêts de l'utilisateur. Dans la phase de prétraitement, nous proposons d'effectuer un nettoyage en profondeur pour éliminer un maximum de données non pertinentes et ainsi réduire le temps nécessaire à la phase de traitement et obtenir des résultats plus précis. Les données extraites sont utilisées pour créer un profil utilisateur et pour comprendre les activités des utilisateurs grâce à des rapports générés.

Nous considérons la gestion des différentes sources d'information du système de recherche comme une limitation de notre approche car elle nécessite une intervention humaine pour gérer les sources et mettre à jour les modèles de requête correspondants dans les dictionnaires de recherche et de téléchargement. Pour cela, nous prévoyons d'automatiser la gestion des sources grâce à l'utilisation de méthodes d'apprentissage.

Les résultats d'analyse générés peuvent être utilisés ultérieurement pour former des clusters, faciliter la coopération, faire émerger des relations qui n'existaient pas auparavant, personnaliser l'interaction avec le système ou générer des recommandations.

Conclusion générale et perspectives

1 Conclusion générale

Nous avons abordé dans cette thèse le problème de la sélection des sources d'information dans un environnement de recherche multi-sources où le nombre de sources ne cesse de croître de jour en jour. En raison du grand nombre de sources disponibles, la sélection des sources est devenue une étape cruciale voire critique du processus de recherche multi-sources.

Les approches de sélection des sources reposent généralement sur la correspondance des termes de la requête de l'utilisateur avec la description de la source. Outre la description de la source, qui est un facteur important pour évaluer la pertinence d'une source pour une requête donnée, il est également important d'intégrer les informations qui caractérisent les utilisateurs (intérêts, préférences, relations sociales, etc.) dans le processus de sélection des sources pour améliorer les performances de la recherche et pour satisfaire l'utilisateur en fournissant des résultats de recherche personnalisés.

Les réseaux sociaux sont désormais devenus une source importante de collecte d'informations sur les utilisateurs, une énorme masse de données est générée chaque jour. Ces données sont généralement brutes et non structurées et nécessitent des techniques d'analyse de données efficaces et robustes pour en extraire des informations utiles. Les données extraites des réseaux sociaux peuvent être utilisées pour construire un modèle d'utilisateur plus précis nécessaire pour personnaliser la recherche d'information.

Dans cette thèse, nous proposons de nouvelles solutions aux problèmes de recherche multi-sources mentionnés ci-dessus et qui sont décrits dans ce qui suit.

Solution 1 : Approche intelligente pour la sélection des sources d'information : Nous considérons le problème de sélection des sources dans un espace de recherche élevé défini par le nombre de sources disponibles. Nous avons proposé une nouvelle approche qui explore intelligemment cet espace afin de trouver la sélection optimale, ou à défaut, de bonne qualité. L'approche proposée utilise une méthode d'intelligence artificielle pour résoudre le problème d'optimisation combinatoire qui consiste à identifier la meilleure combinaison parmi un grand nombre de combinaisons possibles. Nous définissons une solution au problème de sélection des sources comme **une sélection** ou une combinaison représentée par un vecteur de k sources. Nous avons abordé le problème avec les algorithmes génétiques, qui ont été les plus répandus dans le monde en raison de leur simplicité de mise en œuvre, nous pouvons également envisager d'utiliser des algorithmes beaucoup plus récents. Des expériences ont été menées sur des bases de données réelles d'articles de recherche scientifique couvrant différents domaines tels que l'informatique, les mathématiques et les sciences humaines et sociales. Les résultats basés sur la mesure de précision sont très encourageants.

Solution 2 : Approche basée sur les données sociale pour améliorer la sélection des sources : La sélection des sources est basée sur la description des sources et d'autres informations importantes pour identifier les sources susceptibles de contenir des documents pertinents pour une requête d'utilisateur donnée. La description d'une source d'information est généralement construite à partir d'informations locales fournies par la source elle-même, telles que les statistiques des termes contenus dans la source d'information. Une description précise de la source est nécessaire pour une sélection efficace de la source. Nous proposons d'améliorer la qualité de la description des sources afin d'augmenter l'efficacité de la méthode proposée basée sur l'algorithme génétique (solution 1). Pour ce faire, nous exploitons le comportement des utilisateurs lors de l'utilisation des sources pour enrichir la description des sources par des balises (tags) qui sont attribuées par les utilisateurs aux sources disponibles. Ces balises offrent des informations complémentaires et utiles qui sont intégrées dans l'estimation de la pertinence d'une source pour une requête donnée. L'approche proposée basée sur les données de marquage social a fourni une bonne précision par rapport aux approches de sélection des sources de l'état de l'art sur des ensembles de données réels extraits d'un réseau de marquage social.

Solution 3 : une approche multidimensionnelle pour adapter la sélection des sources aux thèmes d'intérêt de l'utilisateur : Nous nous concentrons dans cette étude sur la manière d'impliquer l'utilisateur dans le processus de sélection des sources pour une recherche multi-sources optimale et personnalisée en combinant les méthodes d'intelligence artificielle et d'apprentissage automatique. Nous avons proposé une approche multidimensionnelle de sélection des sources, qui considère à la fois la dimension intelligence et la dimension sociale. Nous avons montré dans des contributions précédentes que les méthodes intelligentes peuvent offrir une meilleure solution au problème de sélection des sources lorsque l'on considère des environnements à grande échelle. D'autre part, la recherche d'information basée sur l'aspect social de l'utilisateur contribue de manière significative à personnaliser les résultats de recherche en tenant compte du profil social des utilisateurs.

Dans l'approche proposée, nous avons d'abord utilisé des méthodes de modélisation de sujets LDA pour analyser de grandes quantités de données collectées à partir de réseaux sociaux de marquage afin de découvrir les thèmes d'intérêt latents des utilisateurs. Les thèmes d'intérêt générés par LDA sont ensuite intégrés au processus de sélection des sources pour générer intelligemment des résultats plus personnalisés en utilisant un algorithme génétique. Les résultats des expérimentations sur des ensembles de données réels extraits d'un système de recherche multi-sources et de réseaux de marquage social ont montré l'efficacité de notre proposition en terme de precision par rapport aux approches de sélection des sources personnalisées et non personnalisées de l'état de l'art.

Solution 4 : Découverte de connaissances à partir de l'analyse des fichiers journaux d'un système de recherche multi-sources : En cas d'absence de données utilisateur, nous avons proposé une méthode basée sur des techniques d'exploration de données (Data Mining) pour analyser les fichiers journaux du système de recherche multi-sources afin d'extraire des informations pertinentes à partir d'une grande quantité de données contenues dans les fichiers journaux. L'analyse de ces données permet d'extraire de nouvelles connaissances permettant de mieux comprendre les centres d'intérêts des utilisateurs, ce qui contribue à l'amélioration de la recherche d'information multi-sources.

La méthode d'analyse proposée consiste en deux étapes, l'étape de prétraitement des

données qui supprime les données indésirables des fichiers journaux et l'étape de traitement des données qui extrait les données pertinentes des fichiers journaux nettoyés. Dans la phase de pré-traitement, nous avons proposé d'effectuer un nettoyage en profondeur basé sur l'étude des structures des requêtes de chaque source d'information. Le nettoyage en profondeur supprime la plus grande quantité de données non pertinentes, ce qui réduit le temps nécessaire à la phase de traitement et permet de générer des résultats plus précis. Les données extraites de ces fichiers journaux aident à créer un profil utilisateur précis et à comprendre les activités des utilisateurs grâce à des rapports générés.

2 Perspectives

Plusieurs perspectives sont possibles suite à ce travail, nous citons ci-dessous les plus importantes.

— Les mégadonnées générées par les réseaux sociaux nécessitent des méthodes d'analyse de données robustes. Dans un premier temps, nous avons utilisé la méthode de modélisation des sujets LDA. Bien que le modèle LDA fournisse un outil puissant pour découvrir et exploiter la structure de sujets cachés. Une limitation importante de ce modèle est que des sujets de mauvaise qualité dont les significations prêtent à confusion peuvent être générés [101]. Nous envisageons dans le future d'améliorer cette méthode et de l'adapter à nos besoins pour extraire des sujets cohérents et significatifs.

— L'apprentissage en profondeur a largement réussi à résoudre des tâches complexes telles que la reconnaissance d'images (ImageNet), la reconnaissance vocale et la traduction automatique. Ces dernières années, l'apprentissage en profondeur a commencé à montrer des avancées prometteuses dans le domaine des systèmes de sélection et de recommandation personnalisés, en raison de sa capacité à apprendre les représentations des utilisateurs et des éléments. Cependant, à notre connaissance, l'apprentissage en profondeur n'est pas encore bien exploré dans les systèmes de recherche distribués et personnalisés, bien qu'un certain nombre de travaux de recherche aient utilisé l'apprentissage automatique pour la sélection des sources que nous avons cités dans les références. Une perspective importante de la continuité de ce travail est d'exploiter les techniques récentes de l'apprentissage en profondeur (par exemple, les réseaux de neurones CNN ou Convolutional Neural Network et les réseaux de neurones RNN ou Recurrent Neural Network) pour relever certains défis de l'approche proposée, par exemple pour remédier au problème de rareté des données en apprenant une meilleure représentation des données des sources et des profils des utilisateurs dans les réseaux sociaux.

— La prise en compte de la confiance (Trust, en anglais) dans la sélection et la recommandation des sources d'information aide à identifier les sources pertinentes et dignes de confiance pour répondre à une requête avec des résultats plus pertinentes, corrects et fiables. La mesure de la fiabilité d'une source ne devrait dépendre d'aucune information que la source fournit par elle-même. La fiabilité d'une source particulière peut s'exprimer par les relations entre les sources elles-mêmes en analysant, par exemple, les résultats renvoyés par les sources pour une même requête, ou par les interactions entre utilisateurs et sources en analysant les commentaires et avis renvoyés par les utilisateurs sur les sources utilisées. Dans ce dernier cas, les utilisateurs malveillants qui envoient de faux commentaires doivent être pris

en considération. Étudier la fiabilité des sources, c'est aussi considérer la fiabilité des utilisateurs (définie par la réputation). Les relations dans les réseaux sociaux représentent une bonne piste à explorer pour estimer la fiabilité des sources et la réputation des utilisateurs.

— L'un des plus gros problèmes de la recherche personnalisée est l'exigence de données utilisateur, qui présente de réels problèmes de confidentialité. Afin de mieux démontrer les performances et l'évolutivité de l'approche proposée, de grands ensembles de données réelles et publiques associées aux données des utilisateurs sont nécessaires.

— Une autre perspective importante de ce travail est de formuler la solution au problème de sélection des sources par une approche orientée services (SOA). Et pourquoi ne pas déployer le service de sélection des sources en tant que service cloud.

Annexe A

Annexe 1

TABLE A.1 – La liste des requêtes de test.

Numéro de requête	**Requête**
1	danger of smoking
2	developing prostate cancer
3	oxygen respiration and pollution of environment
4	computational chemistry research
5	Java programming language initiation
6	Ontology and semantic data structure
7	corporatism economy
8	danger climate change and protection of environment
9	tumor necrosis
10	algorithm genetic and heuristic
11	skin diseases
12	nutrition preventing diseases promoting health
13	protein absorption
14	gene regulatory networks
15	cardiovascular diseases
16	antioxidant food natural
17	aids HIV-1
18	genome analysis
19	mammography women risk factor breast cancer
20	intelligent bio inspired algorithms

TABLE A.2 – Exemple de balises associées aux sources d'information de test.

Numéro de source	**Source d'information**	**Balises**
1	ACM	mining web usable extracting structures collaborativeFiltering evaluation recommender evaluation measure clustering kmeans bisec text hac ontology
2	Elgar	Econometrics imported IP_theory dissertation governance institutional_economics ipr political_economy political_science software___patent medgov media_economics
3	IEEE	modelmerging artificialintelligence theoremproving artificialchemistry agentbasedsystem economic network distributedmodeling
4	IOP	musynthesis optical_flow robustcontrol rotorcraft prev imported imported community detection overview SturmLiouville inverseproblem
5	JSTOR	papert constructionism history computers microworlds education socialnetworks indologica vedica
6	RSC	sys :relevantfor :deibel.lab P3HT PTB7 thickness adhesion cell functionalization nextgen thiolene ml functionalization microfluidics nextgen thiolene
7	Sciencedirect	graph model cognition cwa model modeling concept empirical cognition ontology personal framework modeling process seminal
8	Springerlink	mathematics people Japan education imported bagging classifier combination ner imported Semantic Web imported Australian

TABLE A.3 – La liste des requêtes de test.

Numéro de requête	**Requête**
1	evolutionary programming to improve information retrieval
2	interface adaptation of a mobile application
3	user similarity model and collaborative filtering
4	topic modeling for tags clustering
5	a recommender system in web research
6	complex graph visualization tools
7	social web survey
8	probabilistic method for image segmentation
9	an improved community detection algorithm
10	ontology network
11	distributed information retrieval based Multiagent technology
12	Learning to rank in information retrieval

Table A.4 – Dictionnaire des motifs de recherche.

Source	Domaine	Chemin(s)	Paramètres
ACM	dl.acm.org	/results.cfm	Query
Aluka	www.aluka.org	/heritage/search ; /struggles/search	Query
Annual Reviews	www.annualreviews.org	/action/doSearch	AllField ;text#
CAIRN	www.cairn.info	/resultats_recherche.php	searchTerm ; Word#
Elgar Online	www.elgaronline.com	/noresults ; /search	q# ;q_#
IEEE Science direct	ieeexplore.ieee.org www.sciencedirect.com	/search/searchresult.jsp /search ; /science/related-books	queryText ; searchWithin qs ; authors ; pub ; cid ; volume ; issue ; page ; Sterm
JSTOR	www.jstor.org	/action/doBasicSearch ; /action/doAdvancedSearch	Query ;q#
JSTOR Plants	plants.jstor.org	/search	Query ;q#
IOP Science	iopscience.iop.org	/search ;nsearch	fieldedquery ; terms
Springer Materiels	materials.springer.com	/textsearch ;/search	searchTerm
OCED Library	www.oecd-ilibrary.org	/search	value# ; form_name ;
RSC Publishing Home	pubs.rsc.org	/en/result ; /en/results/all	Searchtext ; AllText ; ExactText ; AtleastText ; WithoutText
Springer Link	link.springer.com	/search	Query ;term
Springer Protocols	www.springerprotocols.com	/cdp/search/searchResultPage	Text ; text ; abstractText ; title ;author
zbMATH	zbmath.org	/authors/ ;/journals/ ;/classification/ ; /	q ;f
Clinical Key	www.clinicalkey.fr	#!/search/	
Clinical Key	www.clinicalkey.com	#!/search/	

TABLE A.5 – Dictionnaire des motifs de téléchargement.

Domaine	**Source**	**MarqueurPDF**
portalparts.acm.org	ACM	.pdf
delivery.acm.org	ACM	.pdf?
www.annualreviews.org	Annual Reviews	doi/pdf
www.cairn.info	CARIN	load_ pdf
www.elgaronline.com	Elgar Online	downloadpdf
www.clinicalkey.fr	Clinical Key	/pdf/ ; .pdf?
ieeexplore.ieee.org	IEEE	.pdf?
ac.els-cdn.com	Science Direct	.pdf?
www.sciencedirect.com	Science Direct	.pdf ;/pdf/
www.jstor.org	JSTOR	/pdf/ ; .pdf?
plants.jstor.org	JSTOR Plants	/pdf/ ;.pdf?
www.aluka.org	Aluka	/pdf/ ;.pdf?
iopscience.iop.org	IOP science	/pdf; .pdf
www.oecd-ilibrary.org	OCED Library	.pdf
www.pediatricneurology briefs.com	Pediatric Neurology Briefs	/articles ; /download/
pubs.rsc.org	RSC Publishing Home	/articlepdf
link.springer.com	Springer Link	/pdf/ .pdf
www.springerprotocols. com	Springer Protocols	/pdf/
zbmath.org	zbMATH	/pdf/ ; .pdf

Bibliographie

[1] Abel, E., Keane, J., Paton, N. W., Fernandes, A. A. A., Koehler, M., Konstantinou, N., Rios, J. C. C., A.Azuan, N., and M.Embury, S. User driven multi-criteria source selection. *Inf. Sci. 430* (2018), 179–199.

[2] Abu Kausar, M., Nasar, M., and Singh, S. A detailed study on information retrieval using genetic algorithm. *Journal of Industrial and Intelligent Information 1* (01 2013), 122–127.

[3] Agichtein, E., Brill, E., Dumais, S., Brill, E., and Dumais, S. Improving web search ranking by incorporating user behavior. In *Proceedings of SIGIR 2006* (August 2006).

[4] Ai, Q., Zhang, Y., Bi, K., Chen, X., and Croft, W. B. Learning a hierarchical embedding model for personalized product search. In *SIGIR* (2017), pp. 645–654.

[5] Allan, J., Connell, M., and Croft, B. Inquery and trec-9. In *Proceedings of the Ninth Text REtrieval Conference* (2000), p. 551–563.

[6] Aly, R., Hiemstra, D., and Demeester, T. Taily : Shard selection using the tail of score distributions. In *36th international ACM SIGIR conference on research and development in information retrieval* (2013), p. 673–682.

[7] Araujo, L., and Perez-Iglesias, J. Training a classifier for the selection of good query expansion terms with a genetic algorithm. In *IEEE Conference on Evolutionary Computation* (Barcelona 2010), pp. 1–8.

[8] Arguello, J., Callan, J., and Diaz, F. Classification-based resource selection. In *18th International ACM Conference on Information and Knowledge Management* (Hong Kong, China, November 2009), ACM, pp. 1277–1286.

[9] Arguello, J., Diaz, F., Callan, J., and Crespo, J. Sources of evidence for vertical selection. In *In Proceedings of the 32nd International ACM SIGIR Conference on Research and Development in Information Retrieval* (Boston, MA, USA, July 2009), ACM, pp. 315–322.

[10] Astrain, J. J., Cordoba, A., Echarte, F., and Villadangos, J. An algorithm for the improvement of tag-based social interest discovery. In *The Fourth International Conference on Advances in Semantic* (Florence, Italy, October 2010), pp. 49–54.

[11] Back, T., Fogel, D. B., and Michalewicz, Z. *Handbook of Evolutionary Computation*, 1st ed. IOP Publishing Ltd., Bristol, UK, UK, 1997.

[12] Baillie, M., Carman, M., and Crestani, F. A multi-collection latent topic model for federated search. *Information Retrieval 14*, 4 (Aug 2011), 390–412.

[13] Balakrishnan, R., and Kambhampati, S. Factal : Integrating deep web based on trust and relevance. In *20th international conference companion on World wide web* (Hyderabad, India, March 28 - April 01 2011), ACM Press.

[14] BALAKRISHNAN, R., AND KAMBHAMPATI, S. Sourcerank : Relevance and trust assessment for deep web sources based on inter-source agreement. In *19th international Conference on World Wide Web* (Raleigh, NC, United States, 2011), ACM Press, pp. 1055–1056.

[15] BALAKRISHNAN, R., KAMBHAMPATI, S., AND JHA, M. Assessing relevance and trust of the deep web sources and results based on inter-source agreement. *ACM Trans. Web 7*, 2 (May 2013), 11–32.

[16] BAOLI, H., LING, C., AND XIAOXUE, T. Knowledge based collection selection for distributed information retrieval. *Inf. Process. Manage. 54*, 1 (2018), 116–128.

[17] BENDER, M., CRECELIUS, T., KACIMI, M., MICHEL, S., NEUMANN, T., PARREIRA, J. X., SCHENKEL, R., AND WEIKUM, G. Exploiting social relations for query expansion and result ranking. In *2008 IEEE 24th International Conference on Data Engineering Workshop* (April 2008), pp. 501–506.

[18] BERGMAN, M. K. The deep web : surfacing hidden value. *Journal of Electronic Publishing 7*, 1 (August 2001).

[19] BHATNAGAR, P., AND PAREEK, N. A combined matching function based evolutionary approach for development of adaptive information retrieval system. *International Journal of Emerging Technology and Advanced Engineering 2*, 6 (2012), 249–256.

[20] BLEI, D. M. Introduction to probabilistic topic models. *Communications of the ACM 55* (01 2011).

[21] BLEI, D. M. Probabilistic topic models. *Commun.ACM 55*, 4 (April 2012), 77–84.

[22] BLEI, D. M., NG, A. Y., AND JORDAN, M. I. Latent dirichlet allocation. *Journal of Machine Learning Research 3* (2003), 993–1022.

[23] BLUM, C., AND ROLI, A. Metaheuristics in combinatorial optimization : Overview and conceptual comparison. *ACM Comput. Surv. 35* (01 2001), 268–308.

[24] BOUADJENEK, M. R., HACID, H., AND BOUZEGHOUB, M. Social networks and information retrieval, how are they converging ? a survey, a taxonomy and an analysis of social information retrieval approaches and platforms. *Information Systems 56* (03 2016).

[25] BOUADJENEK, M. R., HACID, H., BOUZEGHOUB, M., AND VAKALI, A. Using social annotations to enhance document representation for personalized search. In *SIGIR* (2013), p. 1049–1052.

[26] BOUCHACHIA, H., LENA, A., AND VANARET, C. Online and interactive self-adaptive learning of user profile using incremental evolutionary algorithms. *Evolving Systems 5* (09 2014), 143–157.

[27] BOUHINI, C., GÉRY, M., AND LARGERON, C. Personalized information retrieval models integrating the user's profile. In *2016 IEEE Tenth International Conference on Research Challenges in Information Science (RCIS)* (2016), pp. 1–9.

[28] BRUSILOVSKY, P., AND MILLÁN, E. *User Models for Adaptive Hypermedia and Adaptive Educational Systems*, brusilovsky p., kobsa a., nejdl w. ed. Springer, Berlin, Heidelberg, 2007, ch. The Adaptive Web.

[29] CALLAN, J. Distributed information retrieval. In *Advances in Information Retrieval* (2000), W. B. Croft, Ed., Springer, Boston, MA, pp. 127–150.

[30] CALLAN, J., AND CONNELL, M. Query-based sampling of text databases. *ACM Transactions on Information Systems 19*, 2 (2001), 97–130.

[31] CALLAN, J., LU, Z., AND CROFT, W. B. Searching distributed collections with inference networks. In *Eighteenth International ACM Conference on Research and Development in Information Retrieval (SIGIR)* (Seattle, WA, July 1995), pp. 21–29.

[32] CAO, D., HE, X., NIE, L., WEI, X., HU, X., WU, S., AND CHUA, T.-S. Cross-platform app recommendation by jointly modeling ratings and texts. *ACM Transactions on Information Systems 35*, 4 (October 2017), 1–27.

[33] CARMAN, M., AND CRESTANI, F. Towards personalized distributed information retrieval. In *Proceedings of the 31st Annual International ACM SIGIR Conference on Research and Development in Information Retrieval* (2008), pp. 719–720.

[34] CARMAN, M. J., BAILLIE, M., AND CRESTANI, F. Tag data and personalized information retrieval. In *the 2008 ACM workshop on Search in social media* (Napa Valley, California, USA, 2008), ACM, pp. 27–34.

[35] CARMAN, M. J., CRESTANI, F., HARVEY, M., AND BAILLIE, M. Towards query log based personalization using topic models. In *19th ACM international conference on Information and knowledge management, CIKM* (Toronto, ON, Canada, 2010), ACM, pp. 1849–1852.

[36] CATANIA, B., GUERRINI, G., AND YAMAN, B. Context-dependent quality-aware source selection for live queries on linked data. In *19th International Conference on Extending Database Technology (EDBT)* (Bordeaux, France, March 2016), pp. 716–717.

[37] CETINTAS, S., SI, L., AND YUAN, H. Learning from past queries for resource selection. In *CIKM '09, Proceeding of the 18th ACM conference on information and knowledge management* (2009), pp. 1867–1870.

[38] CHEN, H., CHUNG, Y., RAMSEY, M., AND YANG, C. A smart itsy bitsy spider for the web. *Journal of the Association for Information Science and Technology 49*, 7 (1998), 604–618.

[39] CHIRITA, P.-A., FIRAN, C. S., AND NEJDL, W. Personalized query expansion for the web. In *Annual International ACM SIGIR Conference on Research and Development in Information Retrieval* (2007), pp. 7–14.

[40] COOLEY, R., MOBASHER, B., AND SRIVASTAVA, J. Data preparation for mining world wide web browsing patterns. *Journal of Knowledge and Information Systems 1* (04 1999).

[41] CROFT, W. B., AND CALLAN, J. Lemur project. Tech. rep., https ://www.lemurproject.org/, 2000.

[42] DAI, Z., KIM, Y., AND CALLAN, J. Learning to rank resources. In *40th International ACM SIGIR Conference on Research and Development in Information Retrieval* (Tokyo, Japan, August 2017).

[43] DAI, Z., XIONG, C., AND CALLAN, J. Query-biased partitioning for selective search. In *CIKM* (2016), pp. 1119–1128.

[44] DARWIN, C. *On the Origin of Species*. John Murray, London, 1859.

[45] DENG, S., WAN, C., AND LIU, X. Selection of multimedia data source based on user feedback. In *International Conference on Management of e-Commerce and e-Government* (Hubei, Chine, 5-6 Nov. 2011), IEEE, pp. 285–289.

[46] DONG, X. L., SAHA, B., AND SRIVASTAVA, D. Less is more : Selecting sources wisely for integration. *PVLDB 6*, 2 (December 2012), 37–48.

[47] DRIAS, H., KHENNAK, I., AND BOUKHEDRA, A. A hybrid genetic algorithm for large scale information retrieval. In *2009 IEEE International Conference on Intelligent Computing and Intelligent Systems, ICIS 2009* (11 2009), vol. 1.

[48] DU, Q., XIE, H., CAI, Y., FUNG LEUNG, H., LI, Q., MIN, H., AND WANG, F. L. Folksonomy-based personalized search by hybrid user profiles in multiple levels. *Neurocomputing 204* (2016), 142–152.

[49] EIBEN, A., AND SMITH, J. *Introduction to Evolutionary Computing*. Springer, 2007.

[50] FAN, H., ZENG, G., AND LI, X. Crawling strategy of focused crawler based on niche genetic algorithm. In *2009 Eighth IEEE International Conference on Dependable, Autonomic and Secure Computing* (Dec 2009), pp. 591–594.

[51] FAN, W., GORDON, M., AND PATHAK, P. A generic ranking function discovery framework by genetic programming for information retrieval. *Information Processing and Management 40* (07 2004), 587–602.

[52] FAN, W., GORDON, M. D., AND PATHAK, P. Personalization of search engine services for effective retrieval and knowledge management. In *Proceedings of the Twenty First International Conference on Information Systems* (2000), pp. 20–34.

[53] FOGEL, D. B. *System Identification Through Simulated Evolution : A Machine Learning Approach to Modeling*. Ginn Press, 1991.

[54] FRENCH, J. C., POWELL, A. L., CALLAN, J., VILES, C. L., EMMITT, T., PREY, K. J., AND MOU, Y. Comparing the performance of database selection algorithms. In *Proceedings of ACM-SIGIR'99* (1999), pp. 38–245.

[55] FUHR, N. A decision-theoretic approach to database selection. *ACM Transactions on Information Systems 17*, 3 (1999), 229–249.

[56] FUJITA, S. Retrieval parameter optimization using genetic algorithms. *Inf. Process. Manage. 45* (11 2009), 664–682.

[57] GARBA, A., KHALID, S., ULLAH, I., KHUSRO, S., AND MUMIN, D. Embedding based learning for collection selection in federated search. *Data Technologies and Applications 54*, 5 (06 2021), 703–717.

[58] GAUCH, S., SPERETTA, M., CHANDRAMOULI, A., AND MICARELLI, A. User profiles for personalized information access. In *The Adaptive Web*, P. Brusilovsky, A. Kobsa, and W. Nejdl, Eds. Springer, Berlin, Heidelberg, 2007, pp. 54–89.

[59] GEN, M., AND R., C. *Genetic Algorithms and Engineering Optimization*, john wiley and sons, new york ed. Wiley-Interscience ; 1st edition (December 28, 1999), 1999.

[60] GENERSCHWEFEL, H.-P. *Evolution and Optimum Seeking*. Sixth Generation Computer Technology Series. New York : Wiley, c1995, 1995.

[61] GIRI, R., CHOI, H., HOO, K. S., AND RAO, B. D. User behavior modeling in a cellular network using latent dirichlet allocation. In *International Conference on Intelligent Data Engineering and Automated Learning* (2014), Springer, pp. 36–44.

[62] GOLDBERG, D. E. *Genetic Algorithms in Search, Optimization, and Machine Learning*. Addison-Wesley, 1989.

[63] GOLDER, S., AND HUBERMAN, B. A. The structure of collaborative tagging systems. *Journal of Information Science 32*, 2 (2006), 198–208.

[64] GORDON, M. Probabilistic and genetic algorithms in document retrieval. *Commun. ACM 31*, 10 (1988), 1208–1218.

[65] GORRAB, A., KBOUBI, F., AND GHÉZALA, H. Social information retrieval and recommendation : state- of-the-art and future research. *Revue Africaine de la Recherche en Informatique et Mathématiques Appliquées, INRIA 27* (2019). hal-01444570v2f.

[66] GRAVANO, L., CHANG, C.-C. K., GARCÍA-MOLINA, H., AND PAEPCKE, A. Starts : Stanford proposal for internet meta-searching. *ACM SIGMOD Records 26*, 2 (1997), 207–218.

[67] GRAVANO, L., AND GARCIA-MOLINA, H. Generalizing gloss to vector-space databases and broker hierarchies. In *Proceedings of the 21st VLDB Conference* (January 1995), pp. 78–89.

[68] GRAVANO, L., GARCIA-MOLINA, H., AND TOMASIC, A. Gloss : text-source discovery over the internet. *ACM Transactionson Information Systems (TOIS) 24*, 2 (June 1999), 229–264.

[69] GRIFFITHS, T. L., AND STEYVERS, M. Finding scientific topics. In *Proceedings of the National Academy of Science* (2004), vol. 101, pp. 5228–5235.

[70] HARVEY, M., CRESTANI, F., AND CARMAN, M. J. Building user profile from topic models for personalised search. In *In Proceedings of the 22nd ACM International Conference on Information and Knowledge Management, CIKM* (2013), pp. 2309–2314.

[71] Hawking, D., and Thomas, P. Server selection methods in hybrid portal search. In *28th SIGIR 2005* (Salvador, Bahia, Brazil, 2005), pp. 75–82.

[72] Hernández, S., Álvarez, P., Fabra, J., and Ezpeleta, J. Analysis of users' behavior in structured e-commerce websites. *Access IEEE* (2017).

[73] Holland, J. H. *Adaptation in natural and artificial systems.* Michigan Press University, Ann Arbor, MI, 1975.

[74] Hong, D., and Si, L. Search result diversification in resource selection for federated search. In *SIRIG 2013, Proceedings of 36th international ACM SIGIR conference on Research and Devopment in information retrieval* (Dublin, Ireland, July 2013), ACM, pp. 613–622.

[75] Hong, D., Si, L., Bracke, P., Witt, M., and Juchcinski, T. A joint probabilistic classification model for resource selection. In *In Proceeding of the 33rd international ACM SIGIR conference on Research and development in information retrieval SIGIR* (Geneva, Switzerland, July 2010), pp. 98–105.

[76] Ibrahim, S., Selamat, A., and Selamat, M. H. Query optimization in relevance feedback using hybrid ga-pso for effective web information retrieval. In *2009 3rd Asia International Conference on Modelling and Simulation, AMS 2009* (01 2009), pp. 91–96.

[77] Ipeirotis, P., and Gravano, L. When one sample is not enough : improving text database selection using shrinkage. In *Proceedings of the ACM SIGMOD International Conference on Management of Data* (Paris, France, 2004), pp. 767–778.

[78] Ipeirotis, P., and Gravano, L. Classification-aware hidden-web text database selection. *ACM Transactions on Information Systems 26*, 2 (2008), 1–66.

[79] Ipeirotis, P. G., and Gravano, L. Distributed search over the hidden web : hierarchical database sampling and selection. In *VLDB '02 Proceedings of the 28th international conference on Very Large Data Bases* (Hong Kong, China, August 2002), pp. 394–405.

[80] Jahromi, M., and Valizadeh, M. A proposed query-sensitive similarity measure for information retrieval. *Iranian Journal of Science and Technology. Transaction B : Engineering 30* (04 2006).

[81] Järvelin, K., and Kekäläinen, J. Cumulated gain-based evaluation of ir techniques. *ACM Transactions on Information Systems 20* (2002), 422–446.

[82] Jelodar, H., Wang, Y., Yuan, C., and Feng, X. Latent dirichlet allocation (lda) and topic modeling : models, applications, a survey. *Multimedia Tools and Applications v1* (2017).

[83] Jin, Y., Li, R., Cai, Y., Li, Q., Daud, A., and Li, Y. Semantic grounding of hybridization for tag recommendation. In *International Conference on Web-Age Information Management (WAIM)* (2010), pp. 139–150.

[84] KANG, C., WANG, X., CHANG, Y., AND TSENG, B. Learning to rank with multi-aspect relevance for vertical search. In *In Proceedings of the 5th ACM international conference on web search and data mining* (Seattle, Washington, USA, February 2012), ACM, pp. 453–462.

[85] KECHID, S., AND DRIAS, H. Personalised distributed information retrieval-based agents. *IJISTA 9*, 1 (2010), 49–74.

[86] KIM, J., AND LEE, J.-H. A novel recommendation approach based on chronological cohesive units in content consuming logs. *Inf. Sci. 470* (2019), 141–155.

[87] KIM, Y., CALLAN, J., CULPEPPER, J., AND MOFFAT, A. Load-balancing in distributed selective search. In *SIGIR'16* (07 2016), pp. 905–908.

[88] KOZA, J. R. *Genetic Programming : On the Programming of Computers by Means of Natural Selection*. MIT Press, Cambridge, MA, USA, 1992.

[89] KRESTEL, R., FANKHAUSER, P., AND NEJDL, W. Latent dirichlet allocation for tag recommendation. In *Proceedings of the third ACM conference on Recommender systems* (October 23 - 25 2009), pp. 61–68.

[90] KULKARNI, A., AND CALLAN, J. Selective search : Efficient and effective search of large textual collections. *ACM Transactions on Information Systems 33* (04 2015), 1–33.

[91] KULKARNI, A., TIGELAAR, A. S., HIEMSTRA, D., AND CALLAN, J. Shard ranking and cutoff estimation for topically partitioned collections. In *CIKM '12 Proceedings of the 21st ACM international conference on Information and knowledge management* (Maui, Hawaii, USA, October 2012), Association for Computing Machinery, pp. 555–564.

[92] KUMARI, P., RANOUT, P., SHARMA, A., AND SHARMA, P. Web mining - concept, classification and major research issues : A review. *Asian Journal of Advanced Basic Sciences 4*, 2 (2016), 41–44.

[93] LEBIB, F. Z., DRIAS, H., AND MELLAH, H. Selection of information sources using a genetic algorithm. In *Recent Advances in Information Systems and Technologies* (Cham, 2017), Á. Rocha, A. M. Correia, H. Adeli, L. P. Reis, and S. Costanzo, Eds., Springer International Publishing, pp. 60–70.

[94] LEBIB, F. Z., MELLAH, H., AND DRIAS, H. Enhancing information source selection using a genetic algorithm and social tagging. *International Journal of Information Management 37*, 6 (2017), 741–749.

[95] LEBIB., F. Z., MELLAH., H., AND MEZIANE., A. Knowledge discovery from log data analysis in a multi-source search system based on deep cleaning. In *Proceedings of the 15th International Conference on Web Information Systems and Technologies - WEBIST,* (2019), INSTICC, SciTePress, pp. 257–264.

[96] LEBIB, F. Z., MELLAH, H., AND MEZIANE, A. A multi-dimensional source selection based on topic modelling. *J. Inf. Sci. Eng. 38*, 3 (2022), 619–644.

[97] Li, X., Guo, L., and Zhao, Y. E. Tag-based social interest discovery. In *In WWW'08 : Proceeding of the 17th international conference on World Wide Web* (Beijing, China, April 2008), New York, NY, USA, ACM, pp. 675–684.

[98] Lin, Y., Hu, X., and Wu, X. Quality of information-based source assessment and selection. *Neurocomputing 133* (2014), 95–102.

[99] Lin, Y., Wang, H., Li, J., and Gao, H. Data source selection for information integration in big data era. *CoRR* (2016).

[100] Lin, Y., Wang, H., Zhang, S., Li, J., and Gao, H. Efficient quality-driven source selection from massive data sources. *Journal of Systems and Software 118* (2016), 221–233.

[101] Liu, Y., Du, F., Sun, J., and Jiang, Y. ilda : An interactive latent dirichlet allocation model to improve topic quality. *Journal of Information Science 46* (2020), 23 – 40.

[102] Lobo, F. G., Goldberg, D. E., and Pelikan, M. Time complexity of genetic algorithms on exponentially scaled problems. In *GECCO'00 : Proceedings of the 2nd Annual Conference on Genetic and Evolutionary Computation* (2000), pp. 151–158.

[103] Lu, J., and Callan, J. User modeling for full-text federated search in peer-to-peer networks. In *In Annual ACM Conference on Research and Development in Information Retrieval Proceedings of the 29th annual international ACM SIGIR conference on Research and development in information retrieval* (Seattle, Washington, USA, August 2006), ACM, pp. 332–339.

[104] Madhavan, J., Ko, D., Kot, L., Ganapathy, V., Rasmussen, A., and Halevy, A. Y. Google's deep web crawl. *Proceedings of the VLDB Endowment 1*, 2 (2008).

[105] Maleki-Dizaji, S., Siddiqi, J., Soltan-Zadeh, Y., and Rahman, F. Adaptive information retrieval system via modelling user behaviour. *Journal of Ambient Intelligence and Humanized Computing 5* (02 2012).

[106] Markov, I., Arampatzis, A., and Crestani, F. Unsupervised linear score normalization revisited. In *SIGIR '12 Proceedings of the 35th international ACM SIGIR conference on Research and development in information retrieval* (Portland, Oregon, USA, August 12 - 16 2012), pp. 1161–1162.

[107] Markov, I., Arampatzis, A., and Crestani, F. On cori results merging. In *ECIR 2013* (2013), P. Serdyukov, P. Braslavski, S. Kuznetsov, J. Kamps, S. Agichtein, I. Segalovich, and E. Yilmaz, Eds., vol. 7814, LNCS. Springer, Heidelberg, pp. 736–739.

[108] Markov, I., Carman, M. J., and Crestani, F. Towards risk-aware resource selection. In *Asia Information Retrieval Symposium (AIRS)* (2014), pp. 148–159.

[109] Markov, I., and Crestani, F. Theoretical, qualitative, and quantitative analyses of small-document approaches to resource selection. *ACM Transactions on Information Systems 32*, 2 (Avril 2014), 1–37.

[110] Mashagba, E. A., Mashagba, F. A., and Nassar, M. O. Query optimization using genetic algorithm in the vector space model. *International Journal of Computer Science 8*, 3 (Sept 2011), 450–457.

[111] Mathes, A. Folksonomies – cooperative classification and communication through shared metadata [online report]. *Journal of Computer-mediated Communication - JCMC 47* (01 2004).

[112] Mezghani, M., Zayani, A. P. C. A., Amous, I., and Sèdes, F. Producing relevant interests from social networks by mining users' tagging behavior : A first step towards adapting social information. *Data Knowl. Eng. 108* (2017).

[113] Micarelli, A., Gasparetti, F., Sciarrone, F., and Gauch, S. Personalized search on the world wide web. In *The adaptive web*, P. Brusilovsky, A. Kobsa, and W. Nejdl, Eds. Springer-Verlag, Berlin, Heidelberg, 2007, pp. 195–230.

[114] Nhan, N., Son, V., Binh, H., and Tran, K. Crawl topical vietnamese web pages using genetic algorithm. In *2nd International Conference on Knowledge and Systems Engineering, KSE 2010* (11 2010), pp. 217–223.

[115] Oliveto, P., he, J., and Yao, X. Time complexity of evolutionary algorithms for combinatorial optimization : A decade of results. *International Journal of Automation and Computing 4* (07 2007), 281–293.

[116] O'Sullivan, D., Smyth, B., and Wilson, D. Explicit vs implicit profiling : A case-study in electronic programme guides. In *Proceedings of the 18th International Joint Conference on Artificial Intelligence. IJCAI'03* (Acapulco, Mexico, 2003), p. 1351– 1353.

[117] Paltoglou, G., Salampasis, M., and Satratzemi, M. Results merging algorithm using multiple regression models. In *Advances in Information Retrieval : 29th European Conference on IR Research, ECIR 2007* (Rome, Italy, April 2-5 2007), pp. 173–184.

[118] Paltoglou, G., Salampasis, M., and Satratzemi, M. Integral based source selection for uncooperative distributed information retrieval environments. In *In Proceeding of workshop on LSDS for IR* (Napa Valley, California, USA, October 2008), ACM, pp. 67–74.

[119] Pasi, G. Issues in personalizing information retrieval. *IEEE Intelligent Informatics Bulletin 11* (01 2010), 3–7.

[120] Patel, K. B., and Patel, A. Process of web usage mining to find interesting patterns from web usage data. *International Journal of Computers & Technology 3*, 1 (August 2012).

[121] Pathak, P., Gordon, M., and Fan, W. Effective information retrieval using genetic algorithms based matching function adaptation. In *33rd Hawaii International Conference on Science (HICS)* (02 2000), pp. 8 pp. vol.1–.

[122] Phan, X.-H., and Nguyen, C.-T. *JGibbLDA*, 2008.

[123] Powell, A., and French, J. Comparing the performance of collection selection algorithms. *ACM Transactions on Information Systems 21*, 4 (2003), 412–456.

[124] Powell, A. L., French, J. C., Callan, J., Connell, M., and Viles, C. L. The impact of database selection on distributed searching. In *In SIGIR'00 : Proceedings of the 23rd Annual International ACM SIGIR Conference on Research and Development in Information Retrieval* (Athens, Greece, July 2000), ACM, New York, NY, USA, pp. 232–239.

[125] Qiu, Z., and Shen, H. User clustering in a dynamic social network topic model for short text streams. *Information Sciences 414* (05 2017).

[126] Raghavan, S., and Garcia-Molina, H. Crawling the hidden web. In *VLDB '01 : Proceedings of the 27th International Conference on Very Large Data Bases* (San Francisco, CA, USA, 2001), Morgan Kaufmann Publishers Inc., p. 129–138.

[127] Rehatsinas, T., Dong, X. L., and Srivastava, D. Characterizing and selecting fresh data sources. In *SIGMOD'14 Proceedings of the 2014 ACM SIGMOD International Conference on Management of Data* (Snowbird, Utah, USA, June 2014), ACM, pp. 919–930.

[128] Rochd, E. M., and Quafafou, M. A topic model-based personalization over time. In *KDD User Engagement Optimization* (2014).

[129] Salton, G., and McGill, M. J. *Introduction to Modern Information Retrieval*, mcgraw-hill, new york, ny. isbn 0070544840 ed. ACM, 1983.

[130] Salton, G., Voorhees, E. M., and Fox, E. A. A comparison of two methods for boolean query relevance feedback. Tech. rep., In Technical report, Cornell University, Ithaca, NY, 1983.

[131] Salton, G., Yang, C. S., and Yu, C. T. A theory of term importance in automatic text analysis. *Journal of the American Society for Information Science 26*, 1 (1975), 33–44.

[132] Sanderson, M. Ambiguous queries : test collections need more sense. In *Proceedings of the 31st annual international ACM SIGIR conference on Research and development in information retrieval* (2008), p. 499–506.

[133] Saoud, Z., and Kechid, S. Integrating social profile to improve the source selection and the result merging process in distributed information retrieval. *Inf. Sci. 336* (2016), 115–128.

[134] Sathya, A. S. S., and Simon, B. P. A document retrieval system with combination terms using genetic algorithm. *International Journal of Computer and Electrical Engineering 2*, 1 (2010), 1–6.

[135] Sharma, A. Web usage mining : Data preprocessing, pattern discovery and pattern analysis on the rit web data. Master's thesis, Department of Computer engineering, Rochester Institute of Technology Rochester, 2008.

[136] SHIN CHEN, Y., AND SHAHABI, C. Automatically improving the accuracy of user profiles with genetic algorithm. In *International Conference on Artificial Imelligence and Soft Computing* (2001), pp. 21–24.

[137] SHOKOUHI, M. Central-rank-based collection selection in uncooperative distributed information retrieval. In *29th european conference on information retrieval* (2007), pp. 160–172.

[138] SHOKOUHI, M., CHUBAK, P., AND RAEESY, Z. Enhancing focused crawling with genetic algorithms. In *Proceedings of the International Conference on Information Technology : Coding and Computing (ITCC'05)* (05 2005), pp. 503–508.

[139] SHOKOUHI, M., AND SI, L. Federated search. *Journal of Foundations and Trends in Information Retrieval 5*, 1 (2011), 1–102.

[140] SHOKOUHI, M., AND ZOBEL, J. Robust result merging using sample-based score estimates. *ACM Transactions of Information Systems 27*, 3 (2009), 1–29.

[141] SI, L., AND CALLAN, J. Using sampled data and regression to merge search engine results. In *Proceedings of the 25th annual international ACM SIGIR conference on Research and development in information retrieval* (Tampere, Finland, August 2002), pp. 11–15.

[142] SI, L., AND CALLAN, J. Relevant document distribution estimation method for resource selection. In *in SIGIR'03 : Proceedings of the 26th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval* (Toronto, Canada, July 28 - August 01 2003), ACM, New York, NY, USA, pp. 298–305.

[143] SI, L., JIN, R., CALLAN, J., AND OGILVIE, P. A language modeling framework for resource selection and results merging. In *International Conference on Information and Knowledge Management* (January 2002), pp. 391–397.

[144] SOGRIN, M., KECHADI, T., AND KUSHMERICK, N. Latent semantic indexing for text database selection. In *Proceedings of the SIGIR 2005 Workshop on Heterogeneous and Distributed Information Retrieval* (2005), pp. 12–19.

[145] SPERETTA, M., AND GAUCH, S. Personalized search based on user search histories. In *The 2005 IEEE/WIC/ACM International Conference on Web Intelligence (WI'05)* (2005), pp. 622–628.

[146] SPINK, A., JANSEN, B., BLAKELY, C., AND KOSHMAN, S. A study of results overlap and uniqueness among major web search engines. *Information Processing and Management 42*, 5 (2006), 1379–1391.

[147] SRIVASTAVA, J., COOLEY, R., DESHPANDE, M., AND TAN, P.-N. Web usage mining : Discovery and applications of usage patterns from web data. *SIGKDD Explorations 1* (01 2000), 12–23.

[148] STROHMAN, T., METZLER, D., TURTLE, H., AND CROFT, W. B. Indri : A language model based search engine for complex queries. In *In Proceedings of the International Conference on Intelligent Analysis* (2005), vol. 2, Citeseer, pp. 2–6.

[149] TABASSUM, M., AND MATHEW, K. A genetic algorithm analysis towards optimization solutions. *International Journal of Digital Information and Wireless Communications (4* (01 2014), 124–142.

[150] THOMAS, P., AND HAWKING, D. Evaluating sampling methods for uncooperative collections. In *SIGIR'07* (2007), pp. 503–510.

[151] THOMAS, P., AND HAWKING, D. Server selection methods in personal metasearch : A comparative empirical study. *Information Retrieval 12*, 5 (2009), 581–604.

[152] THOMAS, P., AND SHOKOUHI, M. Sushi : scoring scaled samples for server selection. In *ACM SIGIR* (Boston, MA, USA, July 2009), ACM, pp. 419–426.

[153] TIGELAAR, A. S., AND HIEMSTRA, D. Query-based sampling using snippets. In *Eighth Workshop on Large-Scale Distributed Systems for Information Retrieval* (Aachen, Germany, July 2010), vol. 630, pp. 9–14.

[154] VALLET, D., CANTADOR, I., AND JOSE, J. M. Personalizing web with folksonomy-based user and document profiles. In *European Conference on Information Retrieval (ECIR)* (2010), pp. 420–431.

[155] VALLET, D., AND CASTELLS, P. Personalized diversification of search results. In *SIGIR '12 Proceedings of the 35th international ACM SIGIR conference on Research and development in information retrieval* (August 12-16 2012), pp. 841–850.

[156] WEI, X., AND CROFT, W. B. Lda-based document models for ad-hoc retrieval. In *SIGIR '06 Proceedings of the 29th annual international ACM SIGIR conference on Research and development in information retrieval* (Seattle, Washington, USA, August 2006), ACM, pp. 178–185.

[157] WU, T., LIU, X., AND DONG, S. *LTRRS : A Learning to Rank Based Algorithm for Resource Selection in Distributed Information Retrieval.* Zhang Q., Liao X., Ren Z. (eds) Information Retrieval, 09 2019, pp. 52–63.

[158] WU, Z., LU, C., ZHAO, Y., XIE, J., ZOU, D., AND SU, X. The protection of user preference privacy in personalized information retrieval : Challenges and overviews. *Libri 71*, 3 (2021), 227–237.

[159] XU, B., LIN, H., LIN, Y., AND GUAN, Y. Integrating social annotations into topic models for personalized document retrieval. *Soft Computing 24* (02 2020).

[160] XU, J., AND CROFT, W. B. Cluster-based language models for distributed retrieval. In *SIGIR '99 : Proceedings of the 22nd annual international ACM SIGIR conference on Research and development in information retrieval* (Berkeley, California, USA, August 15-19 1999), pp. 254–261.

[161] XU, J., AND LI, X. Learning to rank collections. In *SIGIR 2007 : Proceedings of the 30th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval* (2007), pp. 765–766.

[162] YANG, P., SONG, Y., AND JI, Y. Tag-based user interest discovery though keywords extraction in social network. In *BigCom* (August 2015), pp. 363–372.

[163] Yuwono, B., and Lee, D. L. Server ranking for distributed text retrieval systems on the internet. In *In Proceedings of the FifthInternational Conference on Database Systems for Advanced Applications (DASFAA)* (Melbourne, April 1997), pp. 41–49.

[164] Zhao, F., Zhu, Y., Jin, H., and Yangbc, L. T. personalized hashtag recommendation approach using lda-based topic model in microblog environment. *Future Generation Computer Systems 65* (December 2016), 196–206.

[165] Zhao, Z., Feng, S., Wang, Q., Huang, J. Z., Williams, G. J., and Fan, J. Topic oriented community detection through social objects and link analysis in social networks. *Knowl.-Based Syst. 26* (2012), 164–173.

[166] Zhou, D., Lawless, S., and Wade, V. Improving search via personalized query expansion using social media. *Information Retrieval 15*, 3-4 (2012), 218–242.

[167] Zhou, D., Lawless, S., and Wade, V. Web search personalization using social data. In *In Proceedings of the Second International Conference on Theory and Practice of Digital Libraries, TPDL* (Paphos, Cyprus, September 2012), P. Z. B. R. Loizides, Ed., Springer, pp. 298–310.

[168] Zhou, D., Lawless, S., Wu, X., Zhao, W., and Liu, J. Enhanced personalized search using social data. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing (EMNLP)* (Austin, Texas, November 2016), Association for Computational Linguistics, pp. 700–710.

[169] Zhou, D., Lawless, S., Wu, X., Zhao, W., and Liu, J. A study of user profile representation for personalized cross-language information retrieval. *Aslib Journal of Information Management 68*, 4 (2016), 448–477.

[170] Zhou, D., Wu, X., Zhao, W., Lawless, S., and Liu, J. Query expansion with enriched user profiles for personalized search utilizing folksonomy data. *IEEE Transactions on Knowledge and Data Engineering 29*, 7 (2017), 1536–1548.