

Université Saad Dahlab Blida 1

N° d'ordre :.....



Faculté des Sciences

Département d'Informatique

Mémoire présenté par :

Mossab Asma Sadok Oussama

En vue d'Obtenir le Diplôme de Master en Informatique

Option : Ingénierie de Logiciels

Thème :

**Un Système de Recommandation Basé sur la Qualité
pour la Sélection de Sources d'Information**

Organisme d'Accueil : CERIST

Soutenu le 09/12/2020 devant le jury composé de:

Mr. Bala Mahfoud	Président
Melle. Mancer Yasmine	Examinatrice
Mme. Aroussi Sana	Promotrice
Mme. Matouk Fatma Zahra	Encadreuse

Promotion 2019 / 2020

Session Décembre 2020

Résumé

Notre Travail se situe dans le contexte de la Recherche d'Information Distribuée (RID) ou multi sources, dont le processus de recherche passe par trois phases importantes : la description des sources, la sélection des meilleures sources à interroger et la fusion des résultats renvoyés par les sources choisies. Notre contribution porte sur la sélection et la recommandation des sources d'information qui coopèrent pour répondre à une requête utilisateur. Dans un premier temps, nous avons proposé une sélection basée sur la qualité de la source en considérant les critères de QoS suivants : la disponibilité, le temps de réponse et la fréquence de mise à jour. La qualité d'une source est calculée comme une somme pondérée de ses critères en utilisant les méthodes d'agrégations multicritères SAW, AHP et TOPSIS. Ainsi, les meilleures sources ont les meilleurs scores de qualité. Dans un second temps, en exploitant les données de l'historique de recherche des utilisateurs (les feedback), nous avons proposé aussi des recommandations pertinentes générées sur la base des notations des utilisateurs sur la pertinence des sources en utilisant le filtrage collaboratif social.

Mots clés : recherche d'information distribuée, sélection des sources, critère de qualité, méthode d'agrégation, TOPSIS, AHP, SAW, système de recommandation, recommandation sociale basé item, filtrage collaboratif social.

Abstract

This work considers the context of information retrieval in distributed / multi-source search systems. Three important phases are known in distributed information retrieval system, which are source description, source selection and result merging. Our contribution focuses on the source selection problem; we propose an approach to select and recommend the sources with high quality. First, we propose a source selection based the following set of quality of service (QoS) attributes: availability, response time and update frequency. The quality of a source is calculated as a weighted sum of its criteria using the SAW, AHP and TOPSIS multi-criteria aggregation methods. The best sources therefore have the best quality scores. Second, using data from users' search history (comments/feedback), relevant recommendations are generated based on user ratings on the relevance of sources using collaborative social filtering.

Key words : distributed information retrieval , source selection , quality criteria , aggregation methods , SAW, TOPSIS , AHP , recommendation system , collaborative social filtering , based item filtering

ملخص

يدخل هذا المشروع في سياق استرجاع المعلومات وبشكل أكثر تحديداً استرجاع المعلومات الموزعة. تمر عملية استرجاع المعلومات الموزعة بثلاث مراحل أساسية يديرها الوسيط و هي : أوصاف مصادر المعلومات ، اختيار أفضلها ودمج النتائج التي تعرضها المصادر التي تم اختيارها من بين مجموع المصادر المشتركة . الهدف من المشروع يصب في المرحلة الثانية "اختيار مصادر المعلومات و اقتراحها للمستخدم. يتم ذلك وفقاً لجودة المصدر بإتباع طريقة الاقتراحات المبنية أساساً على المعلومات الاجتماعية باستخدام واحدة من طرق دمج عدة معايير في نتيجة واحدة التالية SAW, AHP, TOPSIS . و باستعمال المشاهدات الفائتة يتم اقتراح مجموعة من مصادر المعلومات التي يمكن ان تهم المستخدم و التي تشبه ما أعجبه من قبل .

كلمات مفتاحية : استرجاع المعلومات الموزعة ، اختيار مصادر المعلومات ، طرق التجميع ، معايير جودة مصدر معلومات ، طرق التوصية

Remerciement

*Nous tenons tout d'abord à remercier **Allah**, le tout-puissant et miséricordieux de nous avoir donné le courage et la patience pour accomplir ce modeste travail.*

*Nous remercions tout particulièrement notre promotrice Mme **Aroussi** pour avoir accepté de nous encadrer et codirigé, et la confiance qu'elle nous a accordée, sa disponibilité, ses précieux conseils et son soutien moral.*

*Nous remercions Mme **Matouk**, nous avoir accueilli et de nous avoir proposé ce thème, pour ses orientations et ses remarques fructueuses.*

Nos remerciements s'adressent également à l'ensemble des enseignants du Département d'informatique.

*Nos vifs remerciements vont à Madame **Boumahdi** et Madame **Rezoug** pour le temps qu'elle nous a consacré pour répondre à toutes nos interrogations et d'avoir mis à notre disposition tous les moyens pour que ce travail puisse aboutir.*

Un grand merci à nos familles surtout nos parents, qui nous ont aidé à suivre nos études dans les meilleures conditions et qui nous ont toujours soutenues et encouragées sans limite et particulièrement,

Nous tenons également à remercier les membres du jury d'avoir accordé de leurs temps précieux pour expertiser notre travail, nous espérons qu'ils en soient satisfaits.

Enfin, nous remercierons tous ceux qui ont participé de loin ou de près à la réalisation de ce travail

Dédicace

*À **ma mère**, qui m'a portée neuf mois dans son ventre et m'a imaginée un être fragile et minuscule à l'abri de tous soucis, tu as été la première à m'initier à la vraie vie, aucun mot ne peut d'écrire mon amour pour toi, tu as joué le rôle du père et la mère à la fois, tu es mon symbole du courage extrême, tu fus mon bouclier dans le moment le plus difficile et tu me donnas toujours du goût à la vie voici le fruit de mille peines et misères.*

*À tous **les informaticiens** du monde, vous êtes des étoiles qui brillent dans le monde technologique.*

*À mes cinq frères et mes quatre sœurs, mes neveux et nièces et à mes beaux-frères en particulier **Abed** paix à son âme,*

*Un spécial remerciement à mes professeurs durant les dernières 19 années d'études de Mme **Setouch** à ma promotrice Mme. **Aroussi**, vous êtes tous magnifiques.*

*À la chère Mme. **Rezoug** et la merveilleuse Mme. **Boumahdi** qui m'a soutenu pendant toutes mes années au niveau du département d'informatique.*

*A mon binôme de Licence , mon frère , mon ami **Ramoul Abdelkader** qui ma toujours soutenu
À tous mes amis, au club **CSCC** qui m'a formé et à toutes les personnes qui ont fait partie de ma vie, vous l'avez rendue spéciale et grâce à vous j'ai acquis de l'expérience.*

Asma

Je dédie ce mémoire

A ma mère, qui a œuvré pour ma réussite, de par son amour, son soutien, tous les sacrifices consentis et ses précieux conseils, pour toute son assistance et sa présence dans ma vie, reçois à travers ce travail aussi modeste soit-il, l'expression de mes sentiments et de mon éternelle gratitude.

A mon père, la personne la plus digne de mon estime et de mon respect, sans vous je n'aurais jamais regardé si haut, merci pour tout papa.

A mes chers et adorables frères et sœurs, El Batoul, Soheib et Choumaïssa. En témoignage de mon affection fraternelle, de ma profonde tendresse et reconnaissance, je vous souhaite une vie pleine de bonheur et de succès et que Dieu, le tout puissant, vous protège et vous garde.

À ne pas oublier mes grands-mères Turkia, et Zohra pour ses douceurs et ses gentillesses.

A tous mes amis , Yacine, Salah ,Hichem, Djamel, Mohamed et Seifeddine à tous ceux qui m'ont donné confiance en l'avenir.

A toutes les personnes qui ont participé à l'élaboration de ce travail, à tous ceux que j'ai omis de citer.

Oussama

Sommaire

Introduction générale	16
Chapitre I : La recherche d'Information Multi-Sources	19
I.1 Introduction	19
I.2. Recherche d'information	19
I.2.1. Définitions	19
I.2.2. Principales phases du processus de recherche d'information	20
I.2.3. Les modèles de recherche d'information	21
I.3. Types des systèmes de recherche d'information	22
I.3.1. Système de Recherche d'Information Centralisée	23
I.3.2. Système de Recherche d'Information Distribuée	23
I.4. Processus de recherche d'information distribuée	24
I.4.1. Module de Gestion (représentations des sources)	25
I.4.2. Module de sélection des sources	26
I.4.3. Module de fusion des résultats	27
I.5. Conclusion	28
Chapitre II : La Qualité des Source de Données	30
II.1. Introduction	30
II.2. définitions	30
II.2.1. Principaux Métriques /Critères.....	31
II.3. Méthodes d'analyse Multicritères	33
II.4. Les Approches d'agrégation	34
II.4.1. Agrégation complète transitive (Approche du critère unique)	34
II.4.2. Agrégation partielle (Approche du sur classement de synthèse)	34

II.4.3. Agrégation locale (Approche du jugement local interactif avec itérations) ..	35
II.4.4. comparaison des methodes d'agrégations	35
II.5. principale Méthodes d'Agrégation Complète.....	36
II.5.1. La methode SAW	36
II.5.2. La methode AHP	37
II.5.3. La methode TOPSIS	40
II.5.4. Comparaison entre les méthodes SAW, AHP, TOPSIS	41
II.6. Conclusion	42
Chapitre III : Systèmes de Recommandation	44
III.1. Introduction	44
III.2. Définitions	44
III.2.1. Construction d'un système de recommandation	45
III.3. Approches de recommandation	47
III.3.1. Recommandation Objet (Filtrage basé contenu)	47
III.3.2. Recommandation Sociale (filtrage collaboratif)	49
III.3.3.. Recommandation Hybride	51
III.3.4. Comparaison des approches	51
III.4. Méthodes de recommandation social (filtrage collaboratif)	52
III.4.1. Filtrage basé sur la mémoire	53
III.4.2. Filtrage basé sur le modèle	54
III.5. Évaluation des systèmes de recommandation	55
III.6. Conclusion	57
Chapitre IV : Approche Proposée	59
IV.1. Introduction	59

IV.2. Description Générale	59
IV.3. Phase de Prétraitement.....	61
IV.3.1. Critères de qualité choisis	61
IV.3.2. construction des descriptions	62
IV.3.3 Matrice de decision	62
IV.4. Processus de sélection	63
IV.4.1. Algorithme de selection	63
IV.4.2. Application des methodes de calcul de score	64
IV.5.Processus de recommandation	76
IV.5.1. Phase 1 : pertinence des sources	76
IV.5.2. Phase 2 : la recommandation	77
IV.6. Conclusion	80
Chapitre V : Conception et Implémentation	82
V.1. Introduction	82
V.2. Etude conceptuelle de notre application	82
V.3. Outils de développement	90
V.4. Présentation de l'application	91
V.4.1. Espace d'accueil	91
V.4.2. Espace administrateur	92
V.4.3. Espace utilisateur	95
V.4.4. La recommandation	95
V.4.5. Test des methodes de sélection	97
IV.5. Conclusion	97
Conclusion et perspectives	99

Liste des Figures

Figure 01	Le processus de recherche d'information	20
Figure 02	Etapes de processus d'indexation	21
Figure 03	Les différents modèles de la recherche d'information	22
Figure 04	SRIC versus SRID	23
Figure 05	Processus de recherche d'information distribuée	24
Figure 06	Composants de la qualité d'une source de données	31
Figure 07	Diagramme d'Ishikiawa des métriques de qualité d'une source	31
Figure 08	les méthodes d'agrégations	34
Figure 09	Exemple d'un système de recommandation	45
Figure 10	Filtrage basé sur le contenu	48
Figure 11	Les méthodes de recommandation collaborative	50
Figure 12	Les méthodes de filtrage collaboratif	52
Figure 13	Architecture générale de l'approche proposée	60
Figure 14	Les principales phases de l'approche proposée	61
Figure 15	Diagramme d'Ishikiawa des métriques choisies	62
Figure 16	Exemple de la description d'une source	62
Figure 17	Application de la méthode SAW	65
Figure 18	Application de la méthode AHP	67

Figure 19	Décomposition hiérarchique	68
Figure 20	Application de la méthode TOPSIS	72
Figure 21	Etapas de calcul de similarité des sources	78
Figure 22	Diagramme de cas d'utilisation du système	83
Figure 23	Diagramme de cas d'utilisation « gérer les sources »	83
Figure 24	Diagramme de séquence « effectuer recherche »	85
Figure 25	Diagramme de séquence « gérer les sources »	86
Figure 26	Diagramme de séquence « consulter les recommandation »	87
Figure 27	Diagramme de classe	88
Figure 28	Outils de développement de SRQSSI	90
Figure 29	Page d'accueil	91
Figure 30	Interface de gestion des sources	92
Figure 31	Ajouter une nouvelle source	93
Figure 32	Choisir une méthode de sélection	93
Figure 33	Interface de gestion des adhérents	94
Figure 34	Compte d'un utilisateur	95
Figure 35	Recommandations générales	96
Figure 36	Résultats d'une recherche	96

Liste des tableaux

Tableau 1	Les principaux paramètres qui reflètent l'état de la source	32
Tableau 2	comparaison entre les méthodes d'agrégation	34
Tableau 3	matrice de décision	37
Tableau 4	Echelle de Saaty	38
Tableau 5	Les ratios de cohérence acceptables	39
Tableau 6	Les indices de cohérence aléatoires	39
Tableau 7	Comparaison entre les méthodes d'agrégation complète	41
Tableau 8	Les deux types de collecte de données utilisateur	46
Tableau 9	Profil de sources	48
Tableau 10	Profil Préférences/Utilisateur	49
Tableau 11	Notations Source	50
Tableau 12	Tableau comparatif des méthodes de recommandation	51
Tableau 13	Matrice de similarité Item-Item	54
Tableau 14	Matrice de similarité Utilisateur-Utilisateur	54
Tableau 15	Les critères de qualité choisis avec leurs mesures	61
Tableau 16	matrice de décision	63
Tableau 17	Valeurs de critères pour l'ensemble des sources – Exemple d'application -	64
Tableau 18	Résultats obtenus par la méthode SAW	66
Tableau 19	Matrice de critères de décision	68
Tableau 20	Détermination des priorités	69

Tableau 21	Multiplication de la matrice de décision par les priorités	69
Tableau 22	Matrice de jugement des critères –priorité complète	70
Tableau 23	Résultats obtenus par la méthode AHP	71
Tableau 24	Matrice normalisée TOPSIS	73
Tableau 25	Matrice pondérée / calcul A+&A-	74
Tableau 26	Calcul E-&E+ / Résultats obtenus par la méthode TOPSIS	75
Tableau 27	Classement des sources selon les méthodes appliquées	76
Tableau 28	Confiance donnée pour chaque type d'adhérent	77
Tableau 29	Exemple d'une matrice d'évaluation utilisateur/source	79
Tableau 30	Calculs nécessaires pour calculer la similarité entre s7,s10	79
Tableau 31	Calcul de score de recommandation	80
Tableau 32	la description de diagramme de cas d'utilisation	84
Tableau 33	Tableau descriptif des classes, attribut et méthode	88
Tableau 34	Temps d'exécution fait par l'application des trois méthodes sur l'exemple	97

Liste des abréviations

AHP	Analytic Hierarchy Process
CV	Cue Validity
CVV	Cue Validity Variance
DEA	Data Enveloppement Analysis
df	fréquence de document
DTF	The decision-theoretic Framework
ISO	International Organisation for Standardization
J2EE	Java 2 Entreprise Edition
MADM	Multiple Attributs Decision Making
MAUT	Multiple Attribute Utility Theory
MRDD	modeling relevant document distribution
QC	Query Clustering
ReDDE	The relevant document distribution estimation
RI	Recherche d'Information
RID	Recherche d'Information Distribuée
RSV	Retrieval Statut Value
SAW	Simple Additive Weighting
SMART	Salton's Magical Automatic Retriever of Text
SR	Système de recommandation
SRI	Système de Recherche d'Information
SRIC	Systèmes de Recherche d'Information Centralisés
SRID	Recherche d'Information Décentralisés ou Distribués
SRQSSI	Système de Recommandation basé sur la Qualité pour la Sélection de Sources d'Information
STARTS	Stanford protocol proposal for internet recuperation and research
TF	Fréquence de terme
TOPSIS	Technique for Order Preference by Similarity to Ideal Solution
UTA	Utilities Additives
WPM	Weight Product Method

INTRODUCTION GENERALE

Introduction Générale

Le développement considérable qu'a connu Internet ces dernières années a conduit à une croissance exponentielle du nombre d'utilisateurs du réseau et du nombre de documents accessibles aux utilisateurs. Mais la complexité du Web est encore bien supérieure à ce que la taille seule du corpus pourrait laisser supposer. En effet, les documents sur le réseau sont très hétérogènes, ne possèdent aucune véritable structure, sont écrits dans plus de cent langues différentes et utilisent plusieurs encodages incompatibles des alphabets non latins. Il est donc devenu très rapidement impossible de naviguer dans ce joyeux fouillis et de localiser rapidement de l'information pertinente. Pour se retrouver dans cette masse de données, d'innombrables Systèmes de Recherche d'Information (SRI) ont été installés sur Internet afin de guider l'utilisateur vers l'information recherchée. Ces systèmes ont tous le même but, répondre au mieux à la requête de l'utilisateur et ceci en couvrant au mieux toute l'information disponible en ligne.

Les SRI se divisent en deux catégories selon la stratégie utilisée pour indexer les documents qu'ils couvrent. Dans la première catégorie, on trouve les Systèmes de Recherche d'Information Centralisée (SRIC) qui construisent un index unique consulté à chaque réception d'une requête de l'utilisateur. Ce type de système a des limites, la plus importante étant son extensibilité réduite face à la prolifération spectaculaire des données sur Internet. Dans la deuxième catégorie, on retrouve les Systèmes de Recherche d'Information Distribuée (SRID) qui viennent apporter des solutions aux limites des SRICs en distribuant les processus d'indexation et de recherche. Dans un SRID, l'utilisateur formule son besoin sous forme d'une requête et l'envoie au courtier (un pseudo-moteur de recherche) qui va la transférer à un certain nombre de sources jugées aptes à répondre à la requête parmi l'ensemble des sources disponibles. Cette opération est appelée la sélection des sources qui se fait en fonction de plusieurs critères de qualité. Les sources interrogées transmettent leurs réponses au courtier qui se charge alors de fusionner les résultats et les afficher à l'utilisateur. Ce dernier peut chercher de l'aide ou de l'expertise auprès d'autres personnes pour filtrer les résultats retournés. On parle ici de la recommandation qui permet de bénéficier des expériences des autres et de trouver ce qu'on souhaite plus facilement et rapidement.

Dans ce contexte, nous nous intéressons à la sélection et à la recommandation des sources d'information en se basant sur leur qualité. L'objectif principal de notre travail est de concevoir et de développer un système de recommandation basé sur la qualité pour la sélection de sources d'information. Plusieurs critères de qualité existent, le choix des critères dépend du domaine d'application et des objectifs fixés. Nous proposons d'utiliser un ensemble de critères que nous jugeons pertinents pour évaluer la pertinence de la source pour une requête donnée. Sur la base de l'historique de recherche et de notation des sources utilisées, des recommandations sont proposées à l'utilisateur.

Ce présent mémoire est composé de cinq chapitres :

- Le premier chapitre consiste à introduire les SRI distribués.
- Dans le deuxième chapitre, nous abordons le concept de qualité d'une source, les critères de qualité et les méthodes de prise des décisions multicritères.
- Le troisième chapitre est consacré aux systèmes de recommandations
- Dans le quatrième chapitre, nous expliquons notre approche de sélection et de recommandation proposée et qui est basée sur un ensemble de critères de qualité.
- Dans le cinquième chapitre, nous décrivons les étapes de réalisation et de mise en place de notre solution.

Enfin, nous clôturons notre mémoire par une conclusion et quelques perspectives.

CHAPITRE I

LA RECHERCHE

D'INFORMATION

MULTI-SOURCES

Celui qui détient l'information, détient le pouvoir.

Celui qui l'entretien, détient le monde.---

Adam Spires

Chapitre I : La Recherche d'Information Multi-Sources

I.1 Introduction

Le but de la Recherche d'Information (RI) est de renvoyer les informations appropriées liées à la requête de l'utilisateur. Le processus RI démarre lorsque l'utilisateur soumet une requête au système de récupération. Ensuite, le système de récupération recherche l'index des documents qui contient une partie ou la totalité du texte de la requête, calcul un score pour la liste de tous les documents avant d'utiliser leurs scores pour classer la liste des documents. D'autre part, l'objectif de la RI Distribuée (RID) est de bénéficier des emplacements distribués des collections et d'exploiter les réseaux informatiques pour accéder à ces collections. Le processus du RID et de la RI centralisé sont similaires, sauf qu'une requête est répartie entre différents emplacements dans le système RID. Dans ce chapitre, nous discutons brièvement des principales phases de la RID, l'évaluation de la RID, et les méthodes et les techniques utilisées dans la RID.

I.2. Recherche d'information

La recherche d'information est un domaine historiquement lié aux sciences de l'information et à la bibliothéconomie qui ont toujours eu le souci d'établir des représentations des documents dans le but d'en récupérer des informations à travers la construction d'index. L'informatique a permis le développement d'outils pour traiter l'information et établir la représentation des documents au moment de leur indexation, ainsi que pour rechercher l'information. On peut aujourd'hui dire que la recherche d'information est un champ transdisciplinaire qui peut être étudié par plusieurs disciplines utilisant des approches qui devraient permettre de trouver des solutions pour améliorer son efficacité.

I.2.1. Définitions

D'une part, plusieurs définitions de la recherche d'information ont vu le jour dans ces dernières années, nous citons dans ce contexte les trois définitions plus importantes :

- **Définition 1** : La recherche d'information est une activité dont la finalité est de localiser et de délivrer des granules documentaires à un utilisateur en fonction de son besoin en information. [1].
- **Définition 2** : La recherche d'information est une branche de l'informatique qui s'intéresse à l'acquisition, l'organisation, le stockage, la recherche et la sélection d'information [2].
- **Définition 3** : La recherche d'information est une discipline de recherche qui intègre des modèles et des techniques dont le but est de faciliter l'accès à l'information pertinente pour un utilisateur ayant un besoin en information [3].

Toutes ces définitions partagent l'idée que la RI a pour objectif d'extraire d'un document ou d'un ensemble de documents les informations pertinentes qui reflètent un besoin d'information.

D'autre part, nous définissons par la suite les concepts de base de la RI que nous utilisons tout au long du document :

Chapitre I : La Recherche d'Information Multi-Sources

- **Une source d'information** : est un ensemble de documents mis à disposition par un particulier ou une organisation désirant les publier [4] .
- **La requête** : peut être considérée comme une description partielle du besoin d'informations (sous forme de mots-clés ou de phrases en langage naturel) à un instant donné [5].
- **Un Système de Recherche d'Informations (SRI)** : est un système informatique qui permet de retourner à partir d'un ensemble de documents, ceux dont le contenu correspond le mieux à un besoin en informations d'un utilisateur, exprimé à l'aide d'une requête [6].

I.2.2. Principales phases du processus de recherche d'information

La Recherche d'Information (RI) traite la représentation, le stockage, l'organisation et l'accès à l'information [1]. La Figure 1 représente le processus de la RI qui est construit selon les étapes suivantes :

- Un utilisateur formule son besoin d'information sous la forme d'une requête.
- Sur la base de l'index construit par le système, le système calcul le degré de correspondance de la requête avec l'index.
- Les documents pertinents pour la requête utilisateur sont alors identifiés.
- Le système retourne la liste des documents pertinents à l'utilisateur.

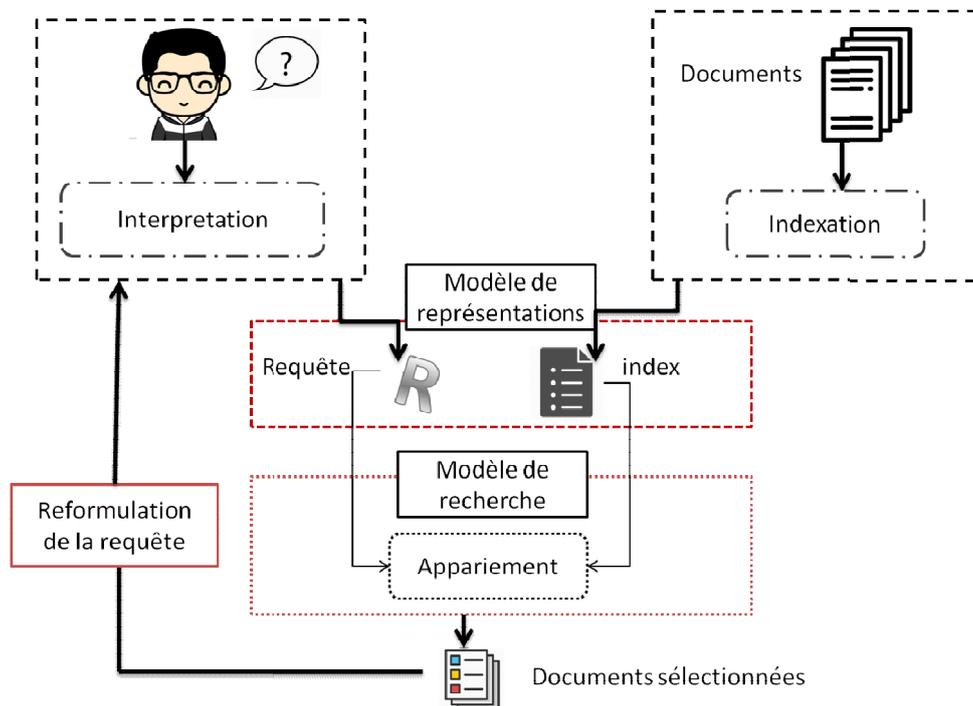


Figure 1: Le processus de recherche d'information [7].

Notons ici que la reformulation de requête est un processus permettant de générer une requête plus adéquate à la recherche d'information dans l'environnement du SRI que celle initialement formulée par l'utilisateur. Son principe est de modifier la requête de l'utilisateur par ajout de termes significatifs et/ou ré estimation de leur poids.

Chapitre I : La Recherche d'Information Multi-Sources

Le processus de recherche d'information induit deux principales phases : l'indexation des documents et l'appariement (ou correspondance) requête/ document.

- L'indexation

L'indexation des documents est une étape très importante dans le processus de RI. Elle consiste à déterminer et à extraire les termes représentatifs (appelés aussi mots clés) du contenu d'un document ou d'une requête. En effet, le processus d'indexation a pour rôle de construire un index, et éventuellement un index inversé, reliant les documents trouvés et les termes d'indexation selon le schéma suivant :

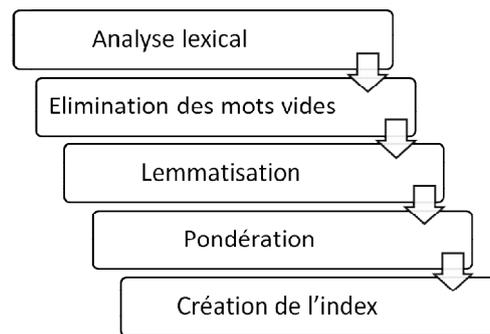


Figure 2: Etapes de processus d'indexation.

Techniquement, l'indexation peut-être manuelle (à l'aide d'un expert du domaine), automatique ou semi-automatique (combinaison des deux approches d'indexation précédentes).

- Fonction d'appariement

La fonction d'appariement (ou de correspondance) document-requête permet de mesurer la valeur de pertinence d'un document pour une requête donnée. Le processus d'appariement nécessite que les documents et la requête soient représentés avec le même formalisme. La comparaison entre les deux représentations (de la requête et du document) est traduite par un score qui détermine la probabilité de pertinence (degré de similarité ou degré de ressemblance) du document à une requête. Cette fonction d'appariement est notée $RSV(d, q)$ (Retrieval Statut Value) [8], où d représente un document de la collection et q la requête. Cette valeur permet ensuite au système de RI d'ordonner les documents renvoyés à l'utilisateur. La méthode de calcul de cette valeur de correspondance est différente d'un modèle à autre.

I.2.3. Les modèles de recherche d'information

Un modèle de RI a pour rôle de fournir une formalisation du processus de RI et un cadre théorique pour la modélisation de la mesure de pertinence. Il existe un grand nombre de modèles de RI textuelle développés dans la littérature (Figure 3) [9]. Ces modèles ont en commun le vocabulaire d'indexation basé sur le formalisme mots clés et diffèrent principalement par le modèle d'appariement requête-document. Parmi ces modèles, nous trouvons :

Chapitre I : La Recherche d'Information Multi-Sources

- **Le modèle Booléen** : qui a été l'un des tous premiers à voir le jour dans le domaine de la recherche d'informations. Il se base sur la théorie des ensembles et sur l'algèbre de Boole pour évaluer la pertinence d'un document à une requête. Bien qu'il soit le modèle le plus critiqué, il a été utilisé par plusieurs SRI commerciale [10] .
- **Le modèle Vectoriel** : fait partie des premiers modèles utilisés en recherche d'information. Il a été proposé au début des années 70 par Salton [11] dans le cadre du système SMART (Salton's Magical Automatic Retriever of Text). Il se base sur l'idée que les documents les plus pertinents sont ceux les plus proches des requêtes (c'est-à-dire qui contiennent les mêmes termes)
- **Le modèle Probabiliste** : consiste à classer les documents dans un ordre décroissant de leurs probabilités de pertinence à un besoin d'information d'un utilisateur. L'objectif est de répondre, pour chaque document et chaque requête, à la question : Quelle est la probabilité que le document soit pertinent pour la requête [12]? Le premier modèle probabiliste de RI a vu le jour vers les débuts des années 60. Il a été proposé par Maron et Kuhn [13].

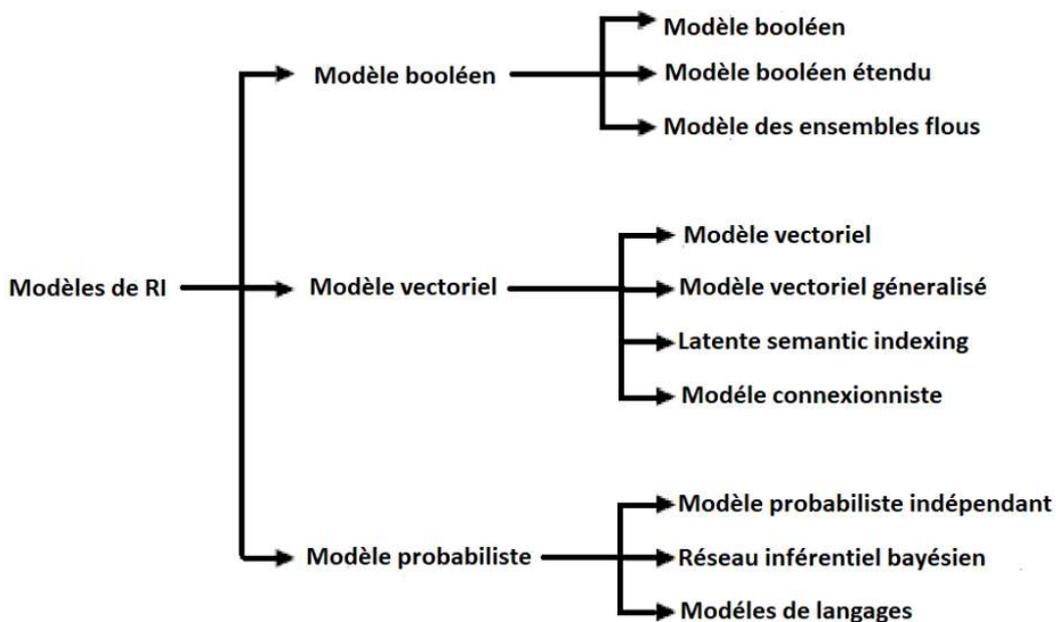


Figure 3 : Les différents modèles de la recherche d'information [9].

I.3. Type des systèmes de recherche d'information

Les Systèmes de Recherche d'Information (SRI) se divisent en deux grandes catégories selon l'architecture sous-jacente. On distingue alors les Systèmes de Recherche d'Information Centralisés (SRIC), où une entité centrale s'occupe de toutes les tâches liées à la collecte, d'indexation, de sauvegarde et de traitement des requêtes, et les Systèmes de Recherche d'Information Décentralisés ou Distribués (SRID), où un ensemble d'entités du même ordre se partagent les différentes fonctionnalités de systèmes. Dans la suite de cette section, nous présentons en détails ces deux types de systèmes.

Chapitre I : La Recherche d'Information Multi-Sources

I.3.1. Système de Recherche d'Information Centralisée

La particularité d'un SRIC est l'unicité de son index, i.e. tous les documents de son corpus sont indexés dans le même fichier ou la même structure qui sert de base pour la recherche. Cependant, vu la quantité phénoménale des données sur Internet, il est impensable d'employer un index unique pour toute cette quantité de données. De plus, certaines sources d'information n'offrent pas l'accès à ces ressources qu'à travers son propre interface. Par conséquent, les SRIC souffrent de certaines limites, y compris la couverture de l'ensemble du Web et des documents indisponibles en raison d'un accès limité.

I.3.2. Système de Recherche d'Information Distribuée

La recherche d'information distribuée (ou multi-sources)¹ est apparue pour résoudre les problèmes des SRIC en offrant aux utilisateurs la possibilité de rechercher simultanément plusieurs sources d'information (moteurs de recherche ou sites Web) via une interface unique [14]. Dans un SRID, les requêtes sont soumises à un sous-ensemble de sources les plus susceptibles de retourner des réponses pertinentes. Les résultats retournés par les sources sélectionnées sont intégrés et fusionnés en une seule liste qui est envoyée à l'utilisateur. La figure 4 montre la différence entre un système de recherche centralisé et un système de recherche distribuée.

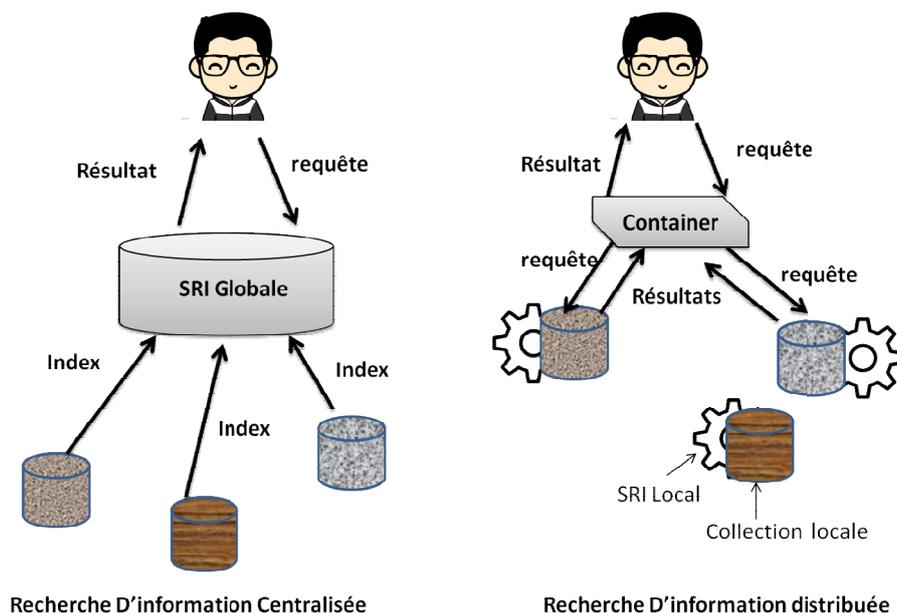


Figure 4: SRIC versus SRID.

La recherche distribuée est préférée par rapport à la recherche centralisée dans de nombreux cas. Par exemple, les moteurs de recherche commerciaux tels que Google ne peuvent pas facilement indexer des collections Web cachées non explorables tandis que les systèmes de

¹ Autres noms pour la RID : La recherche fédérée, la recherche d'information décentralisée, la recherche multi sources.

Chapitre I : La Recherche d'Information Multi-Sources

recherche distribuée peuvent rechercher le contenu des collections Web cachées sans explorer. Un autre exemple dans les environnements d'entreprise, où chaque organisation maintient un moteur de recherche indépendant, les techniques de recherche fédérée peuvent fournir une recherche parallèle sur plusieurs collections [15] [16] [17].

I.4. Processus de recherche d'information distribuée

La figure suivante montre le processus de recherche d'information dans un environnement distribué.

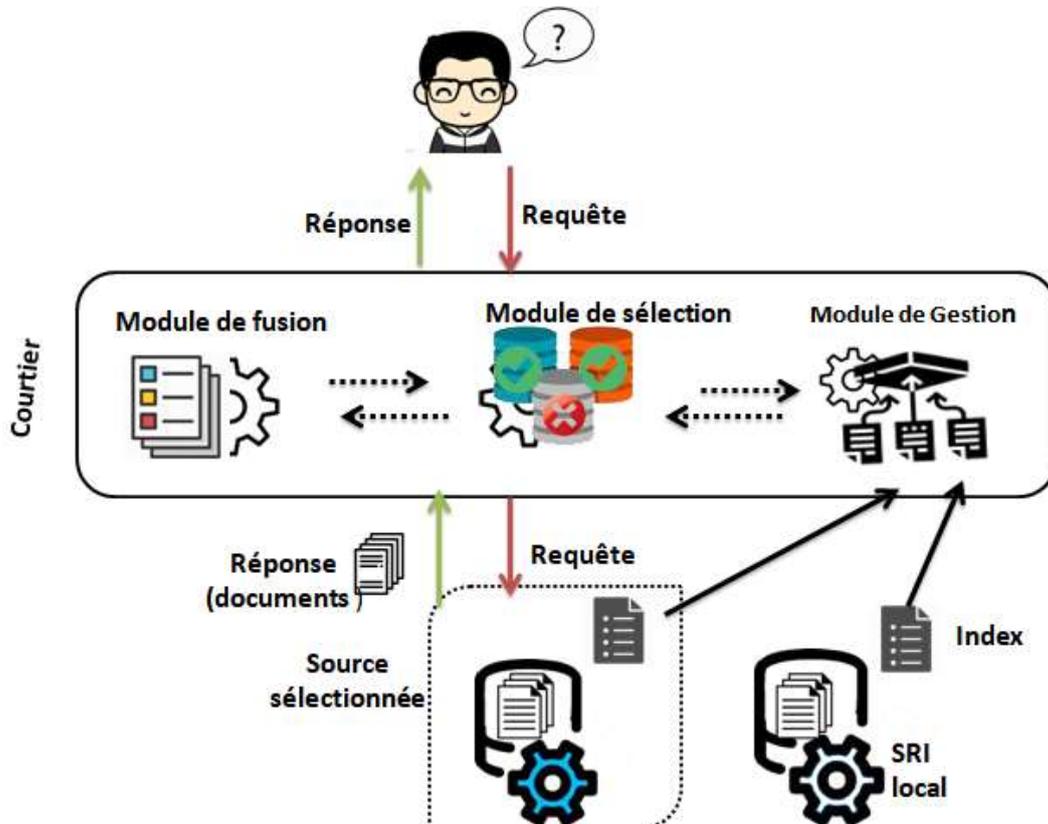


Figure 5: Processus de recherche d'information distribuée

Le courtier est un module qui joue le rôle d'intermédiaire entre l'utilisateur et les différentes sources d'information. Il agit comme un pseudo-moteur de recherche qui reçoit en entrée une requête et fournit en sortie une liste de résultats. Pour ce faire, il détient un ensemble d'informations (appelé représentant de source qui se trouve dans le module de gestion) décrivant certaines caractéristiques de source comme par exemple les termes apparaissant dans la collection de source, leurs fréquences d'apparition, ainsi que des informations utiles à son interrogation (sa localisation, son coût, ses droits d'accès, etc.) [4].

Le courtier comprend trois sous-processus séparés mais entrelacés:

- **Module de gestion:** dans laquelle les descriptions des sources à distance disponibles sont créées [18] [19].
- **Module de sélection :** dans lequel un sous-ensemble de sources d'information les plus pertinentes est choisi pour traiter la requête [19] [20].

Chapitre I : La Recherche d'Information Multi-Sources

- **Module de fusion des résultats** dans lequel les résultats distincts sont combinés en une seule liste des résultats fusionnés qui est renvoyée à l'utilisateur [21] [22] [23].

I.4.1. Module de Gestion (représentations des sources)

En général, la description de la source peut être tout ce qui décrit un contenu d'une ressource :

- Le contenu complet de la ressource
- L'index du contenu complet de la ressource
- Des métadonnées
- Des statistiques: tf (fréquence de terme), df (fréquence de document), longueur moyenne des documents, etc.
- Une combinaison de deux ou plusieurs descripteurs

Au fait, il existe deux types d'environnements qui déterminent la manière de décrire les sources d'information dans la recherche multi-sources, à savoir les environnements coopératifs et les environnements non coopératifs.

- **L'environnement coopératif** : Dans lequel chaque serveur de recherche envoie sa description, ou au moins quelques-uns de ses composants, au méta chercheur. Ce cas est souvent associé à l'initiative STARTS [24] (Stanford protocol proposal for internet récupération and research) qui est un protocole spécifiant comment les serveurs doivent exporter leurs descriptions au méta chercheur. Le protocole STARTS est une excellente solution quand les serveurs de recherche acceptent de coopérer. Toutefois, il échoue quand les serveurs ne peuvent pas coopérer, ne veulent pas coopérer, ou ne présentent pas correctement leur contenu.
- **L'environnement non coopératif** : Dans cet environnement, une partie ou la totalité des serveurs n'exportent pas leurs descriptions vers le méta chercheur, ce dernier trouve des difficultés pour la sélection et la reformulation de requête. Dans ce cas, le méta chercheur peut utiliser les fonctionnalités de base de réponse aux requêtes et faire une approximation des modèles du serveur en envoyant des requêtes préparées spécialement pour les sonder et en analysant leurs réponses. Parmi les serveurs non coopératifs nous distinguons: Google, Altavista, Excite, Yahoo [25].

La construction des descriptions des sources [26] [27] [28] dans les environnements non coopératifs est plus difficile car le système de RID n'a pas un accès aux métadonnées et aux résumés des sources. Dans ce cas, des techniques d'échantillonnage [18] peuvent être utilisées afin de construire une description représentative et précise de chacune des sources qu'il gère.

Chapitre I : La Recherche d'Information Multi-Sources

I.4.2. Module de sélection des sources

Compte tenu d'une requête d'un utilisateur, le système de RID sélectionne uniquement les sources susceptibles de contenir les documents pertinents. Plusieurs approches de sélection des sources existent, les plus importants. Citant [29] :

- **Approches basées sur le lexique** : utilisées pour les environnements coopératifs, le système de RID calcule la similitude de la requête avec la description de la source en utilisant les statistiques détaillées du lexique de la source [30] [31] [32] [33] [34] [35].

Parmi les méthodes les plus connues dans cette approche :

- **CORI** : *Collection Retrieval Inference network* introduit par Callan et al [14] [20] dont le principe est de considérer une collection comme un immense document virtuel et d'effectuer le classement des sources de manière similaire au classement des documents. Chaque source S_i est représentée par la fréquence document de chaque terme t_j de sa collection.
- **GLOSS** : *Glossary of servers servers*, introduit par Gravano et al [36] où dans la première version, ils utilisent la fréquence documents et la taille des collections pour calculer le score d'une source. Ensuite, GLOSS a été généralisé (et nommé vGLOSS [28]) où le score d'une source est la somme des scores de ses documents supérieurs à un seuil donné. Ces scores n'étant pas connus par le courtier, ils sont estimés.
- **CVV** : *Cue Validity Variance* , présenté par Yuwono et al [32] dans un SRID appelé D-WISE . pour une collection donnée, le CV (Cue Validity) d'un terme indique sa capacité à distinguer les documents entre eux .le CVV est la variance de CV à travers toutes les collections du système. Le score d'une source est obtenu en combinant le CVV des termes de la requête et leur fréquences –documents.

Autres stratégies basées sur le lexique sont présentées dans [30] [35] [37] [38] [39] [40].

- **Approches de document de substitution** : utilisées pour les environnements non coopératifs, le courtier utilise les documents échantillonnés de chaque source pour calculer la similitude des sources à une requête donnée. Les méthodes les plus connues de cette approche sont :
 - **ReDDE** : *The relevant document distribution estimation* [41] conçu pour sélectionner un petit nombre de collections avec le plus grand nombre de documents pertinents. Pour cela, ReDDE estime explicitement la distribution des documents pertinente dans toutes les collections. Cette méthode a été variée dans plusieurs travaux : UUM [42], RUM [43], CRCS [44], SUCHI [45].
 - **DTF** : *The decision-theoretic Framework* [46], dont le but est de minimiser les facteurs comme le coût et le temps et maximiser le nombre de documents pertinents récupérés et l'efficacité de recherche des collections peut être apprise en utilisant un ensemble de requêtes de test [47] [48].

Chapitre I : La Recherche d'Information Multi-Sources

- **Approches basées sur la classification (clustering)** : utilisé pour les environnements non coopératifs, le courtier classe les ressources dans une hiérarchie de sujets et prend des décisions de sélection en fonction de la correspondance entre la requête et les catégories de la hiérarchie [49]. Parmi les méthodes de cette approche, nous citons :
 - **MRDD** : modeling relevant document distribution, où les utilisateurs apprennent la pertinence des collections en leur envoyant un certain nombre de requêtes de test et en analysant le nombre de documents pertinents retournés par chaque collection.
 - **Qc** : Query clustering où les requêtes de test sont regroupées en fonction du nombre de documents communs qu'elles renvoient des collections. Le poids final de chaque collection est calculé en fonction de ses performances sur les requêtes de test passées pour les clusters les mieux classés.

Les approches citées au dessus sont généralement basées sur l'information textuelle (le contenu). Elles permettent le classement des sources selon le classement de leurs documents. Ce dernier est calculé en fonction de leur degré de pertinence mesuré par des méthodes comme: " similitude terme " ou " probabilité terme ". Cependant, d'autres sources d'évidence peuvent être considérées pour rechercher des informations pertinentes dans les documents. Leur évaluation se fait selon des critères de qualité en négligeant tout le contenu de la source.

Par ailleurs, la sélection des sources peut s'effectuer selon des scénarios :

- la sélection fixe : consiste à sélectionner un nombre prédéfini de sources;
- la sélection par seuil : consiste à sélectionner toutes les sources dont le score dépasse un certain seuil.
- la sélection économique : consiste à choisir les sources qui satisfont certains critères de façon à ce que l'on obtienne le maximum de documents pertinents. Fuhr [50] propose des métriques basées par exemple sur le coût de traitement d'une requête par une source et le temps de réponse de cette source.

I.4.3. Module de fusion des résultats

Ce module permet de fusionner des listes des documents les mieux classés retournés par les différentes sources sélectionnées (Dans [26], deux techniques sont proposées pour avoir le nombre de documents qui vont être retournés à partir de chaque source pour une requête et renvoyer une seule liste classée à l'utilisateur final. À ne pas confondre avec la fusion de données, dont les résultats proviennent une seule ressource et sont ensuite classés par plusieurs modèles de récupération.

Le processus de fusion des résultats implique un certain nombre de problèmes:

- Détection et suppression des doublons.
- Normalisation et fusion des scores de pertinence.

Chapitre I : La Recherche d'Information Multi-Sources

Différentes solutions ont été proposées pour ce problème [51] [52] [53] [31] [54] [55], en fonction de l'environnement RID.

- **Fusion des résultats dans des environnements coopératifs :** [56] c'est la plus simple et qui propose différentes solutions:
 - Récupérer des documents de chaque source, réindexer et classer selon le modèle RI du courtier.
 - Obtenir des informations sur la façon dont le score du document a été calculé et normaliser le score.

Au plus haut niveau de collaboration, il est possible de demander aux sources pour adopter le même modèle de recherche.

- **Fusion des résultats dans des environnements non coopératifs:** [54] [57] dans les environnements non coopératifs, les sources peuvent fournir des scores, mais le système de RID n'a aucune information sur la façon dont le score a été calculé par les sources. La normalisation des scores nécessite une certaine manière de comparer les scores.

Alternativement, les sources peuvent fournir uniquement des postes de rang, mais le système de RID n'a aucune information sur la pertinence de chaque document dans les listes de classement. La fusion des rangs nécessite une certaine manière de comparer les rangs.

I.5. Conclusion :

Ce chapitre était consacré aux concepts clés de la recherche d'information distribuée. Dans notre travail, nous nous intéressons au module de la sélection des sources d'information selon des critères de qualité. Donc, nous allons présenter, au niveau du chapitre suivant, cette notion de qualité et ses différents aspects.

CHAPITRE II

LA QUALITE DES

SOURCES

D'INFORMATION

*"There's too many Too many people making too many
problems and not much love to go round Can't you see
This a land of confusion" --- Genesis,*

Chapitre II : La Qualité d'une Source d'Information

II.1. Introduction

Les sources d'information sont considérées comme des composants de modules logiciels qui fournissent des fonctionnalités spécifiques sur Internet. Lorsque plusieurs sources offrent la même fonctionnalité, la qualité joue un rôle essentiel dans la distinction entre ces sources. Dans la première partie de ce chapitre, nous allons décrire les concepts clés liés à la qualité d'une source d'information. Dans la deuxième partie de ce chapitre, nous allons nous intéresser aux différentes techniques de mesure de qualité existantes. Vu comme étant un problème de décision multicritères, la sélection des sources en fonction leur qualité offerte peut être résolue par des méthodes d'analyse multicritères, notamment les méthodes d'agrégation que nous allons présenter et les comparer pour choisir la meilleure méthode pour notre problématique.

II.2. Définitions

Le terme qualité est un terme éminemment ambigu qui comporte plusieurs significations et implique des sens différents selon le contexte. La norme de référence ISO 9000 [58] donne une définition commune et très générale du terme qualité. Elle l'a défini comme étant : « *L'ensemble des caractéristiques d'une entité qui lui confèrent l'aptitude à satisfaire des besoins exprimés et implicites* ».

La qualité dans les SRIs dépend essentiellement de la qualité du résultat de la recherche retourné. On parle ici de la qualité des données qui peut être définie en fonction de l'usage attendu par son utilisateur. Cependant, nous n'avons pas trouvé une définition formelle de la qualité de la source d'une information, en dehors des notions générales d'aptitude à l'emploi. La qualité est plutôt conçue comme une valeur agrégée de multiples critères subjectifs et objectifs. Ces critères ont généralement des unités différentes ou pas du tout et dépendent souvent de l'utilisateur.

De ce fait et dans le contexte de notre travail, nous adoptons la définition suivante à la qualité d'une source d'information : Etant donné que le rôle principal d'une source d'information est de répondre aux requêtes envoyées, sa qualité dépend de (Figure 6) :

- la qualité du réseau qui permet d'accéder à la source et de transmettre les requêtes. On parle ici de la performance du réseau ou de la qualité de service qui désigne « *la manière dont le service de livraison de paquets est fourni et qui est décrite par des paramètres tels que la bande passante, le délai de paquet et les taux de perte de paquets* » [59].
- la qualité du serveur qui traite les requêtes. On parle ici des caractéristiques de la machine physique intégrant le serveur (espace de stockage, taille du mémoire, puissances processeurs, puissances des périphériques d'entrée/sortie, ...) qui est décrite par les paramètres suivants : définir la qualité de traitement : le nombre des requêtes traitées par seconde, la vitesse de la machine, les pannes ..etc [60].
- la qualité des données qui représentent le résultat de la requête. On parle ici de la valeur des données qui est fondée sur une appréciation de leur exactitude, de leur

Chapitre II : La Qualité d'une Source d'Information

actualité, de leur précision, de leur exhaustivité, de leur pertinence et de leur accessibilité, en vue de leur utilisation [61]

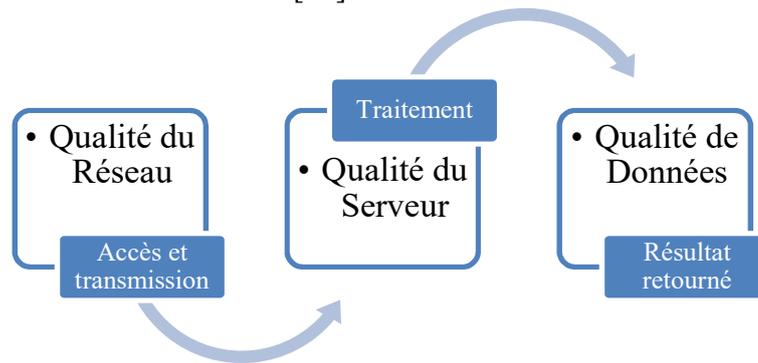


Figure 6. Composants de la qualité d'une source de données

II.2.1. Principaux Métriques/Critères

Les principaux paramètres (métriques ou critères) qui reflètent l'état de la source, et permettant de mesurer la qualité de la source sont décrits dans le tableau 1. Ces paramètres peuvent être classifiés en trois catégories en fonction de leur dépendance au réseau, au serveur ou aux données (Figure 7).

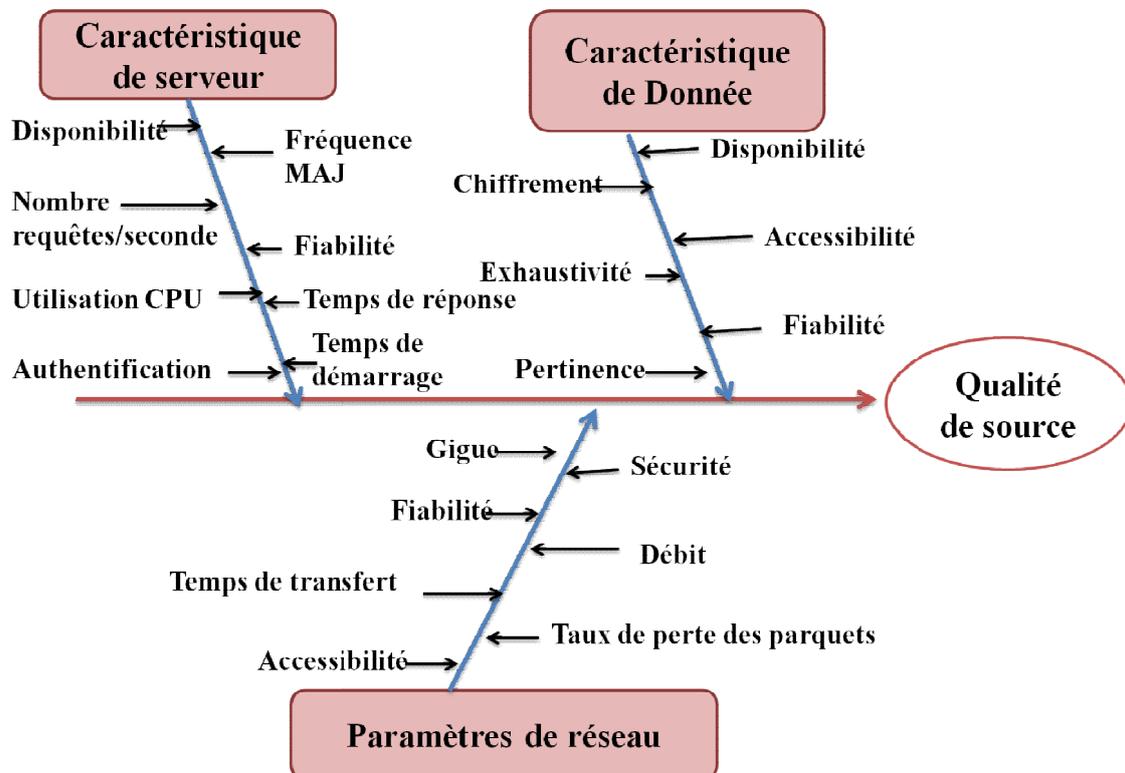


Figure7 : Diagramme d'Ishikiawa des métriques de qualité d'une source.

Chapitre II : La Qualité d'une Source d'Information

Tableau 1 : Les principaux paramètres qui reflètent l'état de la source.

Classes	Métriques	Description
Paramètres du réseau	Connectivité/Accessibilité	L'absence d'interruption de telle sorte qu'il existe plusieurs itinéraires alternatifs.
	Fiabilité	fonctionnement sans défaillances pendant une durée donnée en contrôlant l'état de l'infrastructure pour savoir si elle fonctionne bien et supporte la charge du réseau.
	Bande passante ou débit	C'est la vitesse à laquelle les données peuvent être transférées.
	Temps de transfert ou délai de bout en bout	Il désigne le temps nécessaire à un paquet de données pour passer de la source à la destination.
	Gigue (variation de délai, temps de réponse)	C'est la différence de délai de transmission de bout en bout entre des paquets choisis dans un même flux de paquets, sans prendre en compte les paquets éventuellement perdus.
	Taux de perte des paquets	C'est le pourcentage de paquets perdus lors de la transmission de données. Le taux est une moyenne de toutes les mesures effectuées au cours d'une période donnée.
	Sécurité	Elle comprend les politiques et les pratiques adoptées pour empêcher et surveiller l'accès non autorisé, l'utilisation abusive, la modification ou le refus d'un réseau informatique et des ressources accessibles sur le réseau.
Caractéristiques du serveur	Disponibilité	L'absence d'interruption et que le serveur est toujours prêt à l'emploi malgré la variation de volume.
	Fiabilité	fonctionnement sans défaillances pendant une durée donnée.
	Utilisation CPU	C'est l'intensité avec laquelle le processeur traite les programmes et les processus en cours d'exécution
	Temps de réponse	c'est la durée d'exécution entre le moment où la demande est reçue et le moment où les résultats sont retournés.
	Nombre des requêtes par secondes	C'est le nombre maximal de requêtes traitées par un serveur dans une seconde.
	Fréquence de mise à jour	Les vérifications de matériels et le bon fonctionnement de serveur dans une durée dite un mois.
	Sécurité	C'est la capacité du système informatique à résister à des agressions externes physiques comme incendie ou logiques comme le piratage.
Caractéristiques des données	Disponibilité	la propriété d'une information d'être accessible lorsqu'un utilisateur autorisé en a besoin.
	Fiabilité	Donnée correcte, crédible et actualisée.

Chapitre II : La Qualité d'une Source d'Information

	exhaustivité	Les données doivent permettre l'identification des principaux groupes de risques homogènes de l'engagement évalué.
	pertinence	Données intéressantes et utilisables
	accessibilité	le degré avec lequel une demande de donnée est servie. Le degré élevé d'accessibilité signifie qu'une donnée est disponible pour un grand nombre d'utilisateurs et que les utilisateurs peuvent consulter la donnée relativement et facilement.
	Sécurité (chiffrement)	le <i>chiffrement</i> ou cryptage, qui empêche la lecture des données par des utilisateurs non autorisés.

En résumé, la qualité de la source peut être affectée par un ou plusieurs paramètres. Dans notre travail, nous nous intéressons aux métriques les plus importantes suivant : le temps de réponse, la disponibilité la fréquence de mise à jour. La question qui se pose maintenant est comment calculer ou évaluer ou mesurer la qualité en fonction de ces paramètres afin de choisir la meilleure source d'information. Ce problème de « sélection des sources » peut être ramené au problème de décisions multicritères qui s'intéresse aux problèmes de prise de décision en présence de critères multiples caractérisant la qualité de la source. Cela fait l'objet de la partie suivante.

II.3. Méthodes d'Analyse Multicritères

Si nous sommes dans n'importe quel contexte qui a besoin de décision, nous pouvons distinguer plusieurs types de décision selon les résultats attendus :

- **Problématique de choix** : sélectionner la ou les meilleures solutions.
- **Problématique de tri** segmenter l'ensemble des solutions en les affectant à des catégories préalablement définies.
- **Problématique de rangement** : ranger les solutions en classes d'équivalence de la meilleure à la moins bonne.
- **Problématique de description** : décrire dans un langage approprié les solutions et leurs conséquences

Dans ce contexte, notre problématique se résume à la problématique de choix où l'objectif est de choisir la ou les meilleures sources d'information en fonction de plusieurs critères de qualité.

Pour résoudre ce problème de décision multicritère, plusieurs approches existent : agrégation, machine Learning, l'optimisation notamment les algorithmes génétiques, les réseaux de neurones artificiels, etc. Dans notre travail, nous nous intéressons aux approches d'agrégation qui seront décrites dans la section suivante.

Chapitre II : La Qualité d'une Source d'Information

II. 4. Approches d'agrégation

Elles consistent à agréger les différents critères en une fonction unique qui retourne une seule valeur. Les travaux relatifs à cette famille étudient les conditions mathématiques d'agrégation, les formes particulières de la fonction et les méthodes de construction. La figure suivante montre les méthodes d'agrégation les plus connues.

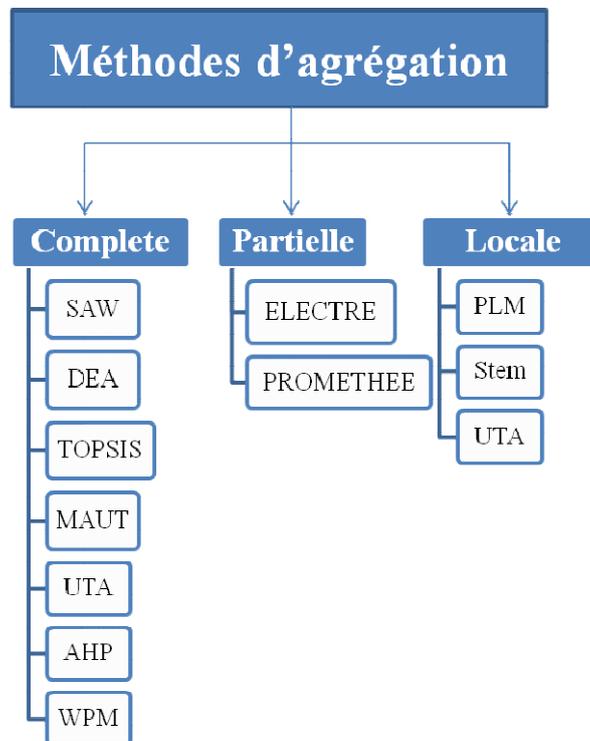


Figure8 : les méthodes d'agrégations

II.4.1. Agrégation complète transitive (Approche du critère unique)

Il s'agit d'évacuer toute situation d'incomparabilité et introduire toutes les performances dans une seule fonction d'agrégation ou d'utilité en leur attribuant d'éventuels poids. La difficulté liée au choix de la fonction d'agrégation est considérée, entre autres, comme inconvenient [62]. Parmi les méthodes les plus connues de cette approche, on cite [63] :

- SAW ((Simple Additive Weighting)
- DEA (Data Enveloppement Analysis)
- TOPSIS (Technique for Order Preference by Similarity to Ideal Solution)
- MAUT (Multiple Attribute Utility Theory)
- UTA (Utilities Additives),
- AHP (Analytic Hierarchy Process),
- WPM (Weight Product Method)

II.4.2. Agrégation partielle (Approche du surclassement de synthèse)

Cette approche repose sur la comparaison des actions deux à deux, puis une synthèse des résultats de ces comparaisons est établie, et c'est la façon de synthétiser qui différencie les

Chapitre II : La Qualité d'une Source d'Information

méthodes de cette famille. Cette approche respecte l'incomparabilité, mais au détriment de la clarté des résultats, et est privilégiée si un critère au moins est qualitatif (elle travaille à la fois avec les critères qualitatifs et quantitatifs). Les critères sont très hétérogènes et leur codage en une échelle commune est difficile [64]. Des seuils de préférence ou de véto doivent être pris en considération. Parmi les méthodes les plus connues de cette approche, on cite [63] :

- ELECTRE (Elimination Et Choix Traduisant la Réalité)
- PROMETHEE (Preference Ranking Organisation METHOD For Enrichement Evaluation)

II.4.3. Agrégation locale (Approche du jugement local interactif avec itérations)

Cette technique consiste à partir d'une solution de départ, supposée aussi bonne que possible, et voir dans le voisinage, s'il existe une solution mieux qu'elle. En d'autres termes, une solution de départ est choisie, ensuite, on sélectionne un groupe de variantes relativement proche à la solution de départ, par la suite, on vérifie s'il n'existe pas de meilleure variante par rapport à celle sélectionnée, ce nouveau choix constitue la solution de départ pour une nouvelle itération. On pratique donc une exploration locale et répétitive [65]. Cette approche trouve toute son utilité dans les situations où il existe un nombre quasi infini de variantes. Les principales méthodes de cette catégorie sont [63] :

- PLM (Programmation Linéaire Multicritères)
- Stem (Pop)
- UTA interactive

II.4.4. Comparaison entre les méthodes d'agrégation

Chacune des approches décrites précédemment possède des avantages et souffre de certaines limites (Tableau 2) [66] [67] [62] et chaque méthode adoptant une de ces approches tente de palier à ses limites et apporter une nouvelle contribution [68].

Tableau 2 comparaison entre les méthodes d'agrégation.

Caractéristiques	Agrégation Complete	Agrégation Partielle	Agrégation Locale
Type de problème	Choix	Classement /choix	Choix
Type de critères	Qualitative	Qualitatif Quantitatif	Qualitatif Quantitatif
Points forts	- Mise en œuvre Simple - Permettre la transitivité - Transformer n-tuples en une seule fonction de sortie	- Eviter la compensation - Accepter l'incomparabilité	- Accepter un nombre infini des actions
Points faibles	- Compensation des mauvais critères	- Nombre finid'actions	- Beaucoup d'itérations

Chapitre II : La Qualité d'une Source d'Information

	- Pas de situation d'incomparabilités	- Complexe à mettre en oeuvre - Difficile à assurer la transitivité	- Complexe à mettre en oeuvre
--	---------------------------------------	--	-------------------------------

Cependant, nous constatons qu'un choix judicieux de la méthode d'agrégation est d'une importance majeure pour un bon déroulement du processus décisionnel. Pour cela, il est important de connaître la nature des entrées de notre système (critères, actions, etc.), la problématique à traiter (choix, tri, rangement, etc.) et le type d'agrégation à effectuer (totale, partielle, locale). Dans notre processus décisionnel de sélection de sources, les critères représentent les critères de qualité de chaque source et l'action est de choisir une ou plusieurs sources les plus adéquates (Problématique de choix). Etant donné leurs points forts et faibles, nous optons pour les méthodes d'agrégation complètes qui font l'objet de la section suivante.

II.5. Principales Méthodes d'Agrégation Complète

Nous avons choisi pour résoudre le problème de sélection de sources d'utiliser les trois méthodes d'agrégation complète, à savoir SAW, AHP et TOPSIS, parce qu'elles sont les plus simples à implémenter et permettent de fournir les résultats attendus pour choisir les sources de qualité élevée. Néanmoins, elles sont différentes par rapport à leurs principes : SAW basée sur la somme pondérée, AHP basée sur la hiérarchie, TOPSIS basée sur la distance. Ces trois méthodes sont décrites en détail dans les sous sections suivantes.

II.5.1. Méthode Saw :

La méthode SAW (Simple Additive Weighting) est la plus ancienne et la plus connue, très simple et très utilisée en pratique. Elle est basée sur la somme pondérée. Un score d'évaluation est calculée pour chaque critère/attribut en multipliant la valeur à l'échelle donnée à l'alternative de cet attribut avec les poids d'importance relative directement attribué par le décideur puis en additionnant des produits pour tous les critères. L'avantage de cette méthode est qu'elle fait une transformation linéaire proportionnelle des données brutes ce qui signifie que l'ordre relatif de grandeur des scores normalisés reste égal [69].

Formellement, l'algorithme de calcul de score dans la méthode SAW se décompose en trois étapes :

- **Étape 1 : création de la matrice de décision** qui représente les données disponibles et les critères comme illustré dans le tableau 3 où A_i est la i ème alternative, C_j est le j ème critère et d_{ij} est la mesure de la performance de la i ème alternative pour le j ème critère.

Chapitre II : La Qualité d'une Source d'Information

Tableau 3 : Matrice de décision [70].

Alternatives	C ₁	C ₂	...	C _n
Poids relatifs	W ₁	W ₂	...	W _n
A ₁	d ₁₁	d ₁₂	...	d _{1n}
A ₂	d ₂₁	d ₂₂	...	d _{2n}
...
A _m	d _{m1}	d _{m2}	...	d _{mn}

- **Étape 2 : création de la matrice de décision normalisée** qui consiste à normaliser les poids et les mesures de performance afin d'obtenir des valeurs entre 0 et 1. La normalisation se fait selon les formules suivantes :

$$\text{Pour le poids : } W_j = \frac{(\sum_{i=1}^m \frac{1}{k_i})}{m} \quad m \text{ est le nombre de critères (F1)}$$

$$\text{Pour les mesures : } v_{ij} = \begin{cases} \frac{x_{ij}}{\max x_{ij}} & \text{pour les critères à maximiser (F2)} \\ \frac{\min x_{ij}}{x_{ij}} & \text{pour les critères à minimiser (F3).} \end{cases}$$

- **Étape 3 : calcul du score de chaque alternative** En utilisant cette formule :

$$\text{score}(A_i) = \sum w_j v_{ij} \quad (F4) \quad 0 < \text{score}(A_i) \leq 1$$

II.5.2. Méthode AHP :

La méthode AHP repose foncièrement sur des calculs ayant recours à l'algèbre matricielle [71] [72]. Elle a été utilisée dans plusieurs domaines: la planification des transports combinés, le rationnement de l'énergie, la gestion des risques de projet, l'analyse comparative des opérations logistiques, la gestion de la qualité des services dans le domaine hospitalier, le domaine du management des opérations, l'allocation des ressources en matière de gestion de portefeuille de produits, ainsi que plusieurs applications concernant les entreprises etc. Elle a été développée par Thomas Saaty en 1970 et permet de décomposer un problème complexe en un système hiérarchique, dans lequel sont établies des combinaisons binaires à chacun des niveaux de la hiérarchie. Le classement hiérarchique s'effectue au moins à trois niveaux (environnement, objectifs de l'entreprise, cours des activités) auxquels sont associés différents critères. Ainsi, il est possible de déterminer l'alternative la plus pertinente, en fonction de la priorité accordée à chacun des critères pris en considération. L'application de la méthode AHP passe par les cinq étapes suivantes :

Chapitre II : La Qualité d'une Source d'Information

- **Etape 1 : Etablir la structure hiérarchique** qui consiste à décomposer le problème aussi complexe soit en une structure hiérarchique à travers ces niveaux :
 - **Niveau 0** : Définir l'objectif cible.
 - **Niveau 1** : Définir les critères de jugements principaux. Il peut y avoir plusieurs niveaux intermédiaires pour déterminer les sous critères.
 - **Niveau 2** : Définir les alternatives ou les actions possible.
- **Etape 2 : Effectuer les pondérations binaires** qui consiste à étudier les points suivant :
 - Comparer l'importance relative de tous les éléments appartenant à un même niveau de la hiérarchie pris deux par deux, par rapport à l'élément du niveau inter médiatement supérieur en utilisant l'échelle de T.L Saaty présenté dans le tableau 4.

Tableau 4 : Echelle de Saaty [40].

Poids ou intensité de la comparaison	Jugement verbal de la préférence
1	Même importance
3	Importance modérée
5	Forte importance
7	Très forte importance
9	Extrême importance ou importance absolue
2,4,6,8	Utilisés pour les jugements intermédiaires par rapport à ceux listés ci-dessus

- Former une matrice carrée réciproque $N \times N$ composée par les ratios issus des comparaisons. N étant le nombre d'éléments comparés. Un Exemple de matrice est présenté dans le tableau et par la formule suivante.

$$A = [a_{ij}] \text{ avec } \begin{cases} a_{ij} = 1 \text{ pour } i = 1..k \text{ et} \\ a_{ij} = \frac{1}{a_{ji}} \text{ (valeur réciproque)} \end{cases}$$

- **Etape 3 : Calcul des priorités:** dans le but de trouver un ensemble de poids $w_1 \dots w_n$ tel que w_i/w_j correspond aux comparaisons a_{ij} dans la matrice de comparaison par pair. Le calcul se fait comme suit :

- a. Effectuer la somme des éléments de chaque colonne j :

$$\sum_{i=1}^n a_{ij} \quad \forall i, j \quad F5$$

- b. Diviser chaque élément de la matrice par la somme des éléments de sa colonne (Normaliser)

$$a_{ij} = a_{ij} / \sum_{i=1}^n a_{ij} \quad \forall i, j \quad F6$$

Chapitre II : La Qualité d'une Source d'Information

c. Calcul de la moyenne de chaque ligne i :

$$w_i = \sum_{j=1}^n a'_{ij} / n \quad F7$$

- **Etape 4 : calculer le ratio de cohérence (RC) :**

Les priorités calculées ont un sens seulement si les matrices de comparaison par pair sont cohérentes (respect de la transitivité), un contrôle de cohérence doit être appliqué. Saaty [73] a proposé un index de cohérence (IC) qui se calcule en se basant sur le calcul des valeurs propres :

$$IC = (\lambda_{max} - n) / (n - 1) \quad F8$$

Tel que : n est la dimension de la matrice de comparaison, λ_{max} est la valeur propre maximale de la matrice de comparaison.

Le ratio de consistance est calculé par la formule : $RC = IC/IA \quad F9$

Tel que : RC ration de cohérence, IA indice aléatoire, IC indice de cohérence

Si $RC < 10\%$ alors la matrice est consistante et acceptable, Sinon revoir la comparaison

Le tableau 5 illustre l'indice accepté par Saaty selon les démentions de la matrice

Tableau 5 : les ratios de cohérence acceptable [73].

Taille de la matrice (n)	Ratio de cohérence acceptable
3	0.05
4	0.08
5 +	0.10

IA est un indice déterminé par T.L Saaty [73] d'une manière empirique, cet indice est présenté dans le tableau 6.

Tableau 6 : les indices de cohérence aléatoires [73].

N=nombre de critères	2	3	4	5	6	7	8	9	10	11
IA	0	0.58	0.90	1.12	1.24	1.32	1.41	1.45	1.49	1.51

- **Etape 5 : Etablir les priorités finales**

La dernière étape doit synthétiser les priorités locales à travers tous les critères pour déterminer la priorité globale. En utilisant le mode « distributif », on adopte une fonction additive avec normalisation de la somme des priorités locales à l'unité :

$$p_i = \sum_j w_j \cdot l_{ij} \quad F10$$

Tel que : p_i priorité globale de l'alternative i ; L_{ij} : priorité locale ; W_j : poids du critère j

Chapitre II : La Qualité d'une Source d'Information

II.5.3. Méthode TOPSIS

TOPSIS (Technique for Order Preference by Similarity to Ideal Solution) est une méthode dont le but est de pouvoir classer par ordre de choix un certain nombre d'alternatives sur la base d'un ensemble de critères favorables ou défavorables. Elle a été développée par Hwang et Yoon en 1981 [74]. Son principe consiste à déterminer pour chaque alternative un coefficient compris entre 0 et 1 sur la base des distances (euclidiennes) entre chaque alternative d'une part et les solutions idéales favorable et défavorable. Une alternative est dite idéale favorable si elle est la plus loin de la pire alternative et la plus proche de la meilleure alternative. Une alternative est dite idéale défavorable si elle est la plus proche de la pire alternative et la plus loin de la meilleure alternative.

Les principales étapes de la méthode sont :

- **Etape 1: Construction et normalisation de la matrice de decision :**

- On attribue pour chaque critère une pondération (un poids qui reflète l'importance du critère dans notre choix final). Les pondérations doivent être définies de sorte que leur somme soit égale à 1. Généralement elles sont définies en pourcentage (%). Même si les poids ne sont pas compris entre 0 et 1, on peut toujours les ramener à l'intervalle [0, 1] en divisant tout simplement chaque poids par la somme de tous les poids.
- On normalise tous les scores de la matrice des niveaux attribués aux critères. Pour cela on applique la formule indiquée ci-contre pour obtenir les nouvelles entrées v_{ij} de la matrice. :

$$v_{ij} = \frac{x_{ij}}{\sqrt{\sum_{i=1}^n x_{ij}^2}} \quad F11$$

- **Etape 2 : Calcul de la matrice pondérée :**

Dans cette étape, on multiplie simplement toutes les entrées (r_{ij}) de la matrice normalisée par la pondération associée à chaque critère .

- **Etape 3 : calcul de pire et meilleure solution A- & A+**

- **Calcul de A+ :** Pour chaque critère (attribut) on calcule la valeur associée la plus favorable A+ en fonction de la nature du critère (à maximiser ou à minimiser).
Si le critère est à maximiser on choisit la plus grande valeur de chaque colonne. Si le critère est à minimiser, on sélectionne la plus petite valeur de chaque colonne.
- **Calcul de A- :** Pour chaque critère (attribut) on calcule la valeur associée la moins favorable A- en fonction de la nature du critère (à minimiser ou à maximiser). Si le critère est à maximiser, on choisit la plus petite valeur de chaque colonne. Si le critère est à minimiser, on sélectionne la plus grande valeur de chaque colonne.

- **Etape 4 : Calcul d'E+ & E- :**

- **Calcul d'E+ :** Dans cette étape, nous calculons pour chaque alternative son écart par rapport à la valeur la plus favorable déjà évaluée à l'étape précédente.

L'ensemble des écarts sont exprimés par le vecteur E+. Chaque écart est exprimé en tant que distance euclidienne entre la valeur de chaque critère associé et la valeur associée d'A+.

Chapitre II : La Qualité d'une Source d'Information

- **Calcul d'A-** : Cette étape est analogue à l'étape précédente où il suffit d'utiliser plutôt les composants du vecteur A- à la place de A+

Les calculs sont faits en suivant la formule suivante :

$$E_{\pm} = \sum \sum_j^n ((A_{\pm})_j - x_{ij})^2 \quad F12$$

- **Etape 5 : Rangement des résultats** : consiste à déterminer le score en utilisant une autre métrique exprimant les deux critères E- et E+ calculés dans l'étape précédente à la fois qu'on appelle « Facteur de proximité » défini par la formule mathématique :

$$Score = \frac{E -}{(E +) + (E -)} \quad F13$$

- Il s'agit de choisir celle qui est la plus loin possible de la solution idéale défavorable A- et la plus proche de la solution idéale favorable A+.

II.5.4. Comparaison entre les méthodes SAW, AHP, TOPSIS :

Dans l'approche d'agrégation complète, la qualité des sources peut être exprimée par une affectation de poids aux métriques de la qualité en reposant essentiellement sur le calcul de score de la qualité à partir des scores des métriques. En se basant sur la comparaison de Neumann [75] et l'étude de complexité de Burgess [76], les trois méthodes sont comparées en fonction des propriétés suivantes (tableau 3).

- **interaction** : l'utilisateur peut exprimer des préférences sur les critères (affectation d'un poids numérique aux critères de qualité).
- **dominance** : une source d'information domine l'autre si elle est égale ou meilleure pour tous les critères et meilleure au moins pour un critère.
- **type de résultat** : on distingue deux types de résultats : ordonnancement (score de qualité) ou classement.
- **critères positifs et négatifs** : la méthode distingue entre les critères négatifs (prix, début réseau, temps de réponse) et les critères positifs (disponibilité, complétude ...).
- **complexité** : la complexité de l'algorithme de calcul de scores de qualité.

Tableau 7: comparaison des méthodes d'agrégation complète [70].

Propriété	SAW	AHP	TOPSIS
Interaction	Poids	Décisions	Décisions
Dominance	Oui	Oui	Oui
Type de résultat	Ordonnancement	Ordonnancement	Ordonnancement
Critères positifs/négatifs	Oui	Non	Non
Complexité	$O(n.m)^2$	$O(2^n)$	$O(n^2)$

² n: nombre de critères ; m : nombres des alternatives

Chapitre II : La Qualité d'une Source d'Information

D'après l'étude comparative des méthodes d'analyses multicritères (tableau 7), nous concluons que la méthode SAW nous permet de travailler avec des critères négatifs et positifs avec une complexité minimum. C'est une méthode de combinaison ou notation linéaire pondérée la plus souvent utilisée. C'est pour cela, que nous l'avons choisi comme une des méthodes pour la sélection des sources et nous allons la détailler dans le chapitre suivant.

Nous avons choisi également d'utiliser une méthode d'agrégation totale AHP. AHP compte parmi les plus simples à mettre en œuvre. Elle permet de calculer une note synthétique (valeur comprise entre 0 et 1) agrégée sur la base d'une hiérarchisation et d'une pondération de l'ensemble des critères (qualitatifs ou quantitatifs) entrant en compte dans la décision. C'est à dire structurer un problème complexe, multicritère, multi personne et multi période de façon hiérarchique. La méthode TOPSIS permet de travailler avec des critères qualitatifs et quantitatifs au même temps pour arriver à la solution la plus proche de la solution idéale et plus loin de la pire solution. C'est une méthode prouvée mathématiquement, facile à mettre en œuvre.

II.6. Conclusion :

Ce chapitre a pour but d'exposer d'une part, la qualité notamment la qualité de la source et ses différents critères. D'autre part, un aperçu sur les méthodes d'agrégation est donné afin de justifier notre choix pour les méthodes SAW, AHP et TOPSIS qui nous permettent de résoudre le problème de décision multicritère du choix d'une ou de plusieurs sources d'information parmi un ensemble des sources disponibles. En plus la sélection des sources de bonne qualité pour une requête donnée, l'objectif de notre travail est de générer les recommandations pour les sources à interroger,. Le chapitre suivant est consacré aux systèmes de recommandation.

CHAPITRE III

LES SYSTEMES DE

RECOMMANDATION

“La suggestion consiste à faire dans l'esprit des autres une petite incision où l'on met une idée à soi.”—Victor Hugo

Les Systèmes de Recommandation

III.1. Introduction

Fréquemment, nous sommes confrontés à faire des choix. Comment se vêtir ? Quel film regarder ? Quel article acheter ? Que visiter lorsque l'on est en voyage ? La taille de ces domaines de décision est très souvent grande. Par exemple, Netflix disposait en 2007 de plus de 17,000 films dans sa base de données [77], et ce nombre ne cesse de croître au fil des années. La liste des possibilités qui s'offrent à nous est donc en général de très grande taille, l'évaluation de ces possibilités pour trouver ce qui nous convient le plus est une tâche difficile et peut consommer beaucoup de notre temps.

Les systèmes de recommandation sont apparus dans le début des années 1990 pour répondre à ce problème de surcharge d'information et de choix. C'est un besoin similaire à celui des moteurs de recherche (Exemple Google) mais différent dans sa conception. Un moteur de recherche reçoit une requête de la part de l'utilisateur, en général sous forme de texte, et fournit une liste ordonnée d'éléments (pages web, images, vidéos...) dans le but de permettre à l'utilisateur d'accéder rapidement à un contenu considéré comme pertinent par le système par rapport à sa recherche parmi le très grand nombre d'informations disponibles sur Internet. À l'inverse, un système de recommandation ne reçoit pas de requête directe de la part de l'utilisateur, mais il doit lui proposer de nouvelles possibilités en apprenant ses préférences à partir de son comportement passé ou le comportement des utilisateurs similaires. Un système de recommandation doit donc avoir accès à un historique des données qui peut être sous plusieurs formes : des notes, des achats, des clics sur des pages web, des historiques de navigation, etc. À partir de ces informations, le système de recommandation sera en mesure d'adapter la réponse à l'utilisateur.

Dans ce chapitre, nous allons étudier les systèmes de recommandation. Nous commençons par le principe général de la recommandation. Ensuite nous allons présenter les différentes approches de recommandation et pour finir, nous présentons les méthodes d'évaluation de la pertinence d'un système de recommandation.

III.2. Définitions

Les systèmes de recommandation ont été définis de plusieurs façons. La définition la plus populaire et la plus générale que nous citons ici est celle de Robin [78] que nous avons traduite ainsi : *Un système de recommandation est un système capable de fournir des recommandations personnalisées ou permettant de guider l'utilisateur vers des ressources intéressantes ou utiles au sein d'un espace de données important.*

Les systèmes de recommandation (SR) sont des outils logiciels et des techniques fournissant des suggestions pour les articles qui peuvent être utiles à un utilisateur. Les suggestions concernent divers processus de décision, tels que les articles à acheter, la musique à écouter ou les nouvelles en ligne à lire [79]. Les systèmes de recommandation peuvent être aussi vus comme une réponse donnée aux utilisateurs ayant des difficultés à prendre une décision dans le cadre d'utilisation d'un système de recherche d'information. Ils ont pour objectif de fournir à un utilisateur des ressources pertinentes en fonction de ses préférences.

Les Systèmes de Recommandation

En d'autres termes, ils cherchent à prédire la valorisation ou préférence qu'un utilisateur attribuerait à un objet (livre, musique, film, etc.) ou à un élément social (personne, groupe, communauté) qu'il n'avait pas encore considéré. Afin de pouvoir fournir des recommandations personnalisées, le SR doit connaître les préférences de chaque utilisateur. Il tente, alors, d'acquérir les informations nécessaires pour construire des profils d'utilisateurs. En particulier, il exploite les traces laissées par les utilisateurs eux-mêmes. Il collecte les traces laissées explicitement ou implicitement [80]. Les traces explicites sont fournies volontairement par l'utilisateur [79]. Les traces implicites sont collectées en traçant les actions spontanées de l'utilisateur pendant la navigation [80].

La Figure 9 montre un exemple d'un système de recommandation des ressources d'articles.

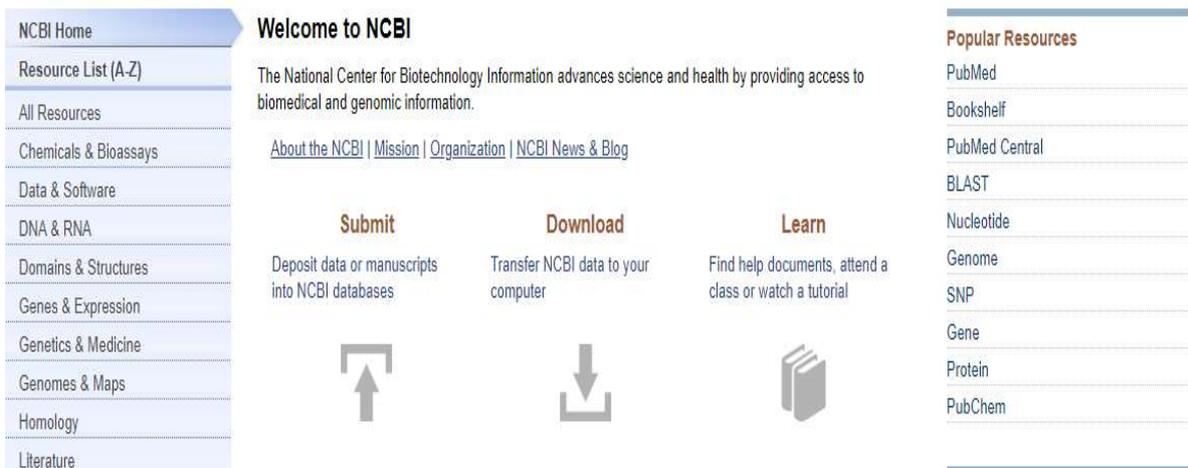


Figure 9 : Exemple d'un système de recommandation [81].

III.2.1. Construction d'un système de recommandation

La construction d'un système de recommandation passe généralement par 3 étapes [82].

- **Etape 1** : consiste à recueillir de l'information sur l'utilisateur,
- **Etape 2** : consiste à bâtir une matrice ou un modèle utilisateur contenant l'information recueillie.
- **Etape 3** : consiste à extraire à partir de cette matrice une liste de recommandations.

a. Collecte d'information sur l'utilisateur :

Pour être pertinent, un système de recommandation doit pouvoir faire des prédictions sur les intérêts des utilisateurs. Il faut donc pouvoir collecter un certain nombre de données sur ceux-ci afin d'être capable de construire un profil pour chaque utilisateur. On distingue deux formes de collecte de données, à savoir explicite et implicite. Le tableau suivant présente les deux types de collecte de données utilisateur ainsi que leurs avantages et leurs inconvénients.

Les Systèmes de Recommandation

Tableau 8 : les deux types de collecte de données utilisateur

Forme	Description	Exemple	Avantages	Inconvénients
Explicite (Filtrage actif)	repose sur le fait que l'utilisateur indique explicitement au système ses intérêts.	demander à un utilisateur de commenter, noter, aimer des contenus qui l'intéressent. On utilise souvent une échelle de <i>ratings</i> (d'évaluation)	capacité à reconstruire l'historique d'un individu et capacité à éviter d'agréger une information qui ne correspond pas à ce dernier	les informations recueillies peuvent contenir un biais dit de déclaration.
Implicite (Filtrage passif)	repose sur une observation et une analyse des comportements de l'utilisateur effectué de façon implicite dans l'application qui embarque le SR	Analyser la fréquence de consultation d'un contenu par un utilisateur	Aucune information n'est demandée aux utilisateurs, toutes les informations sont collectées automatiquement	Les données récupérées sont plus difficilement attribuables à un utilisateur et peuvent donc contenir des biais d'attribution

b. Modèle Utilisateur :

Le modèle utilisateur se présente généralement sous forme de matrice. On peut le représenter comme un tableau qui contient des données recueillies sur l'utilisateur associées aux produits disponibles sur le site web.

Un autre point important est comment le temps influence le profil de l'utilisateur. Les intérêts des utilisateurs, généralement, évoluent au cours du temps. Les données du modèle utilisateur devraient donc constamment être réajustées pour rester conformes aux nouveaux centres d'intérêts de l'utilisateur.

c. Liste de recommandations :

Pour extraire une liste de suggestions à partir d'un modèle utilisateur, les algorithmes utilisent la notion de mesure de similarité entre objets ou personnes décrits par le modèle utilisateur [83]. La similarité a pour but de donner une valeur ou un nombre (au sens mathématique du terme) à la ressemblance entre deux choses. Plus la ressemblance est forte, plus la valeur de la similarité sera grande. A l'inverse, plus la ressemblance est faible, et plus la valeur de la similarité sera petite.

Les Systèmes de Recommandation

III.3. Approches de recommandation :

Les systèmes de recommandation fonctionnent avec deux types d'information :

- **Informations caractéristiques** : Il s'agit d'informations sur les éléments (mots-clés, catégories, etc.) et les utilisateurs (préférences, profils, etc.).
- **Interactions utilisateur-élément** : Il s'agit d'informations telles que les notes, le nombre d'achats, les likes, etc.

Selon l'utilisation de ces deux types d'information, nous distinguons trois approches possibles pour un système de recommandation:

- **Recommandation Objet (Filtrage basé contenu)** : Système basé sur le contenu qui utilise des informations caractéristiques
- **Recommandation Sociale (Filtrage collaboratif)** : Système basé sur des interactions utilisateur-article.
- **Recommandation Hybride** qui combinent les deux types de systèmes dans le but d'éviter les problèmes générés lorsque l'on travaille avec un seul type.

Dans ce qui suit, nous approfondirons un peu plus dans ces trois types et verrons en quoi ils sont différents.

III.3.1. Recommandation Objet (Filtrage basé contenu) :

Les systèmes de recommandation [79] basés sur le contenu (content-based) fonctionnent en analysant les caractéristiques des objets à recommander (produits, etc.) puis en les regroupant. Par la suite, le système va suggérer aux utilisateurs ayant consulté un item quelconque par le passé, les objets estimés similaires. Les systèmes de recommandation basés sur le contenu s'appuient sur des évaluations effectuées par un utilisateur sur un ensemble des items. L'objectif est alors de comprendre les motivations l'ayant conduit à juger comme pertinent ou non un item donné. Le système peut alors proposer à l'utilisateur un choix parmi de nouveaux items jugés proches des items qu'il a précédemment appréciés.

Les items qui peuvent être recommandés aux utilisateurs sont représentés par un ensemble de caractéristiques, aussi appelées attributs, variables ou propriétés dans la littérature. Un item est représenté dans le système au moyen d'une donnée structurée. Cette donnée structurée est un vecteur. Dans la plupart des systèmes de recommandations basés sur le contenu, les descriptions des items sont des caractéristiques textuelles extraites à partir de pages Web, emails, articles de news ou descriptions de produits. Les items correspondent souvent à des mots-clés.

Les Systèmes de Recommendation

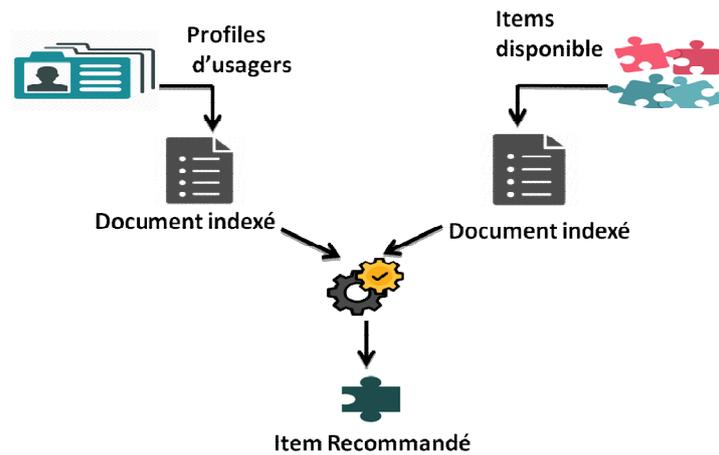


Figure 10 : Filtrage basé sur le contenu.

Exemple : Nous considérons le cas de recommandation de sources d'information pour une recherche donnée. Les caractéristiques des sources, comme par exemple le type d'accès, le domaine, le type de document souhaité, etc., sont nécessaires pour générer les recommandations. Parmi un ensemble de sources, le système va choisir, par exemple 3 sources, à savoir ResearchGate, I3EE et GoogleScholar pour l'utilisateur. Le système pourra donc accomplir cette tâche seulement s'il a à disposition deux types d'information :

- 1) la description des caractéristiques de la source ou le profil de la source (Tableau 9)
- 2) un profil utilisateur qui décrit les intérêts (passés) de celui-ci en termes de préférences (Tableau 10)

La tâche de recommandation consiste donc à déterminer les sources qui correspondent le mieux aux préférences de l'utilisateur.

Tableau 9: Profils des sources

Sources	Accès	Domaines	Type de document	Mots clés
ResearchGate	Public	Informatique Biologie	These Rapport Article	Algorithme, Cellule , Génétique , Base de données, Developement, Programmation

Les Systèmes de Recommandation

I3EEE	Privé	Electronique Informatique Mechanique	Article Rapport Compte- rendu	Robotique , IA,Circuit, Programmation, Algorithme, Turbo , Moteur , combustion
GoogleScholar	Privé	Biologie Informatique Electronique	Livre , Rapport	Molécule , Programmation, Robotique, IA, Génitique , Biochimie

Tableau 10 : Profil / Préférences utilisateur.

utilisateur	Accé	Domaine	Type de document	Mots clés
id 1	Public	Electronique	Article	IA , Robotique, Programmation
id 2			These	

III.3.2. Recommandation Sociale (filtrage collaboratif)

Le filtrage collaboratif est parmi les technologies les plus populaires dans le domaine des systèmes de recommandation [7]. Il fonctionne en recommandant des items en fonction du comportement passé des utilisateurs similaires, en effectuant une corrélation entre des utilisateurs ayant des préférences et intérêts similaires. Cette technologie nécessite des méthodes qui collectent et analysent des données sur le comportement, les activités et les préférences des utilisateurs.

Des algorithmes de filtrage collaboratif tentent de prédire ce que l'utilisateur aimera en cherchant d'autres utilisateurs qui ont les mêmes comportements que l'utilisateur à qui nous souhaitons faire des recommandations (Figure 11). L'idée de base est donc de dire que si des utilisateurs ont partagé des mêmes intérêts dans le passé, il y a de fortes chances qu'ils partagent aussi les mêmes goûts dans le futur.

Ces approches utilisent essentiellement les données contenues dans une matrice dite matrice d'usage. La matrice peut être construite en « sondant » les comportements des utilisateurs ou en proposant à ces derniers de déclarer eux-mêmes leurs mesures d'intérêts sur les items.

Les Systèmes de Recommandation

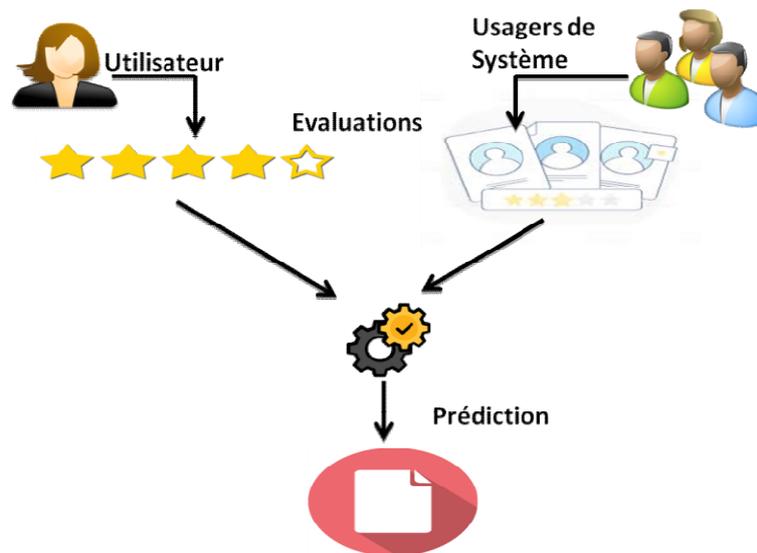


Figure 11 : Les méthodes de recommandation collaborative.

Exemple : Comme illustré dans le Tableau 11, une matrice de *notation* (rating) des sources, avec sur un axe les utilisateurs et sur un autre les sources. Chaque cellule de la matrice contient le rating donné par un utilisateur pour une source. Un signe “+” indique que l'utilisateur a aimé la source, un signe “-” qu'il ne l'a pas aimé, pas de signe signifie qu'il n'a pas d'avis particulier sur cette source. Pour pouvoir prédire si Ali apprécierait la source “ResearchGate” et éventuellement lui recommander cette source, on compare les ratings d’Ali à ceux des autres utilisateurs sélectionnés. On peut alors voir qu’Ali et Mohammed ont des ratings identiques, et que Mohammed a aimé la source ResearchGate, on pourrait alors prédire que Ali aimera aussi cette source et lui faire cette suggestion.

Tableau 11: Notations des sources.

	Aicha	Bilal	Mohammed	Nadjia	Ali
I3EEE	-	-	+		+
GoogleScholar	-	+	+	-	+
GitHub	+		+	+	-
ResearchGate	-	+	+	-	?

Les Systèmes de Recommandation

III.3.3. Recommandation Hybride

Cette approche est en fait une combinaison des deux types d'approches précédemment citées [84]. Les systèmes hybrides tirent parti des avantages des deux approches citées précédemment. Cette solution permet de combler les lacunes de l'une des approches sur des cas d'utilisation précis. Il existe différentes possibilités de combinaison [85], Adomavicius et Tuzhilin [84] les ont classifiées en quatre catégories :

- Implémenter la méthode collaborative et la méthode basée sur le contenu séparément puis combiner leurs prédictions.
- Incorporer quelques caractéristiques de la méthode basée sur le contenu dans l'approche collaborative.
- Incorporer quelques caractéristiques de la méthode collaborative dans l'approche à base de contenu.
- Construction d'un modèle général unifié qui incorpore les caractéristiques des deux modèles.

III.3.4. Comparaison entre les approches:

Le tableau suivant permet de comparer les trois approches de recommandation décrites ci-dessus en citant les avantages et les inconvénients de chaque méthode.

Tableau 12 : Tableau comparatif des méthodes de recommandation [78] [83] [79].

Type de recommandation	Avantages	Inconvénients
Recommandation Objet	<ul style="list-style-type: none"> - Pas besoin d'une large communauté d'utilisateurs pour pouvoir effectuer des recommandations. - Une liste de recommandations peut être générée même s'il n'y a qu'un seul utilisateur. 	<p>Les caractéristiques et propriétés sont fournies par les créateurs du livre ou du film.</p> <p>La problématique, c'est l'acquisition de caractéristiques subjectives et qualitatives des propriétés, comme le style et le design, qui peuvent difficilement être acquises automatiquement, et devront plutôt être introduites manuellement avec tout ce que ça implique, comme le coût, les éventuelles erreurs.</p>
	<ul style="list-style-type: none"> - Cette approche n'exploite pas ou ne demande aucune connaissance sur les contenus eux-mêmes. - Pas besoin de se baser sur l'analyse des propriétés intrinsèques d'un livre ou d'un contenu, la recommandation 	<ul style="list-style-type: none"> - Démarrage à froid : Le lancement d'un service de recommandation peut souffrir au début du manque d'utilisateurs et d'informations sur ceux-ci [86]. - Rareté : le nombre de produits ou

Les Systèmes de Recommandation

Recommandation Sociale	sociale est capable de recommander des contenus sans avoir besoin de comprendre le sens ou la sémantique du contenu lui-même.	contenus est énormes sur certaines plates-formes, et même les utilisateurs les plus actifs auront noté ou valorisé qu'un tout petit sous-ensemble de toute la base de données. Donc, même l'article le plus populaire n'aura que très peu de bonnes notes.
Recommandation Hybride	Le but de cette approche est de réunir le meilleur des deux approches objet et sociale explicitées ci-dessus. L'idée est de compenser le <i>démarrage à froid</i> et la <i>rareté</i> .	Une nouvelle approche, qui se montre difficile à mettre en œuvre

Dans notre travail, nous nous intéressons au filtrage collaboratif qui est largement utilisé avec les réseaux sociaux. Cette méthode prend en considération les relations de confiance entre les utilisateurs, ce qui nous a conduit à l'appliquer dans le domaine de sélection des sources. Nous proposons un système de recommandation sociale qui prend en compte les avis des utilisateurs sur les différentes sources coopérantes sur le système. Le principe c'est d'intégrer les appréciations données par les utilisateurs dans la recommandation des sources pour un utilisateur donné.

III.4. Méthodes de recommandation sociale (filtrage collaboratif)

Les techniques de filtrage collaboratif sont classées en 2 sous-types [87] [88] (Figure 12) :

- filtrage basé sur la mémoire
- filtrage basé sur un modèle

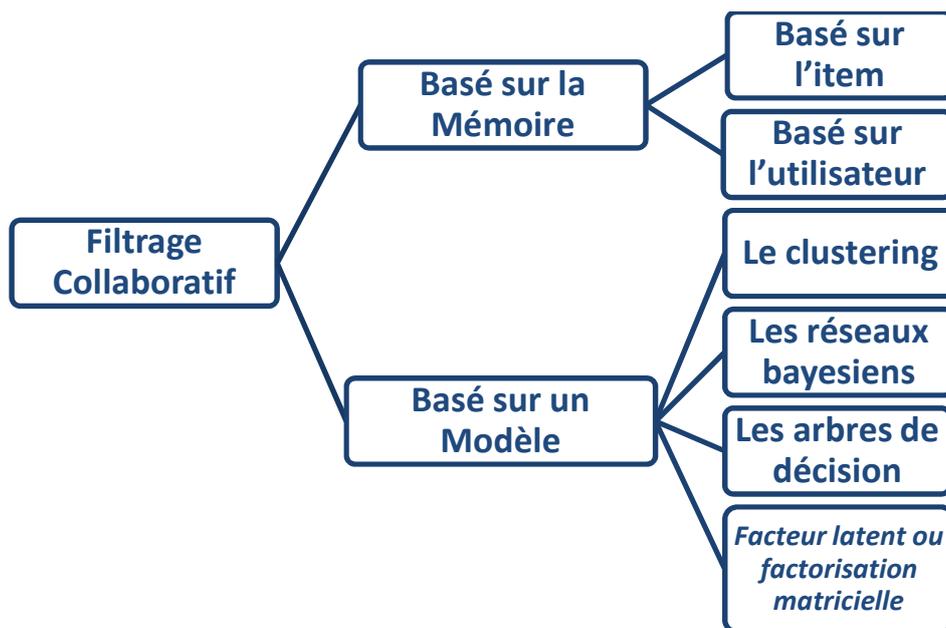


Figure 12 : Les méthodes de filtrage collaboratif.

Les Systèmes de Recommandation

III.4.1. Filtrage basé sur la mémoire :

Les approches basées sur la mémoire, en d'autre part, font des prédictions d'évaluation basées sur l'ensemble de la collection d'évaluations [89] dans la matrice d'usage qui peut être exploitée de deux façons différentes [90] :

- 1- L'orientation utilisateurs (user-Based) : consistant à calculer les similarités entre utilisateurs à l'aide de leurs profils ;
- 2- L'orientation items (item-Based) : consistant à calculer les similarités entre items selon les mesures attribuées par les utilisateurs.

▪ User-Based :

Le principe sur lequel repose le filtrage collaboratif est le suivant : si des utilisateurs ont eu des comportements et goûts de consultations similaires par le passé, alors, ils devraient également en avoir dans le futur. L'avantage du filtrage collaboratif est qu'il est agnostique au contenu. Il se base uniquement sur le profil de consultation des utilisateurs pour faire des recommandations personnalisées.

▪ Item-Based

Cet algorithme fonctionne un peu comme l'user-Based, mais utilise cette fois les similarités entre les profils des items pour proposer les recommandations aux utilisateurs. Pour identifier les similarités entre items (objets), l'item-Based utilise un profil d'item qui est constitué de la liste des utilisateurs ayant consulté/aimé cet item. Les items identifiés comme similaires à un item donné parce que les mêmes gens les ont aimés sont alors recommandés aux gens qui ont consulté cet item.

▪ Item-Based vs User-Based, quelles différences ?

Pour faire simple, les deux algorithmes ont un fonctionnement globalement similaire. Leurs différences résident dans le fait que l'Item-Based et l'User-Based ont un point de départ inversé pour le calcul des recommandations personnalisées. L'un commence par regarder les profils utilisateurs (User-Based), l'autre débute le calcul en utilisant les items (Item-Based). Leur efficacité dépend des applications, typiquement le user-Based est souvent plus pertinent dans la presse, l'item-Based est utilisé en particulier pour les utilisateurs qui ont encore un profil trop petit pour bénéficier du user-Based.

▪ Mesure de similarité:

Le calcul des similarités consiste à mesurer la similitude entre les lignes ou les colonnes de la matrice d'usage. Le choix de la mesure utilisée dépend généralement de la nature des vecteurs. Par exemple, si les vecteurs contiennent uniquement des données binaires, la distance de Jaccard peut être utilisée. Considérons deux vecteurs a et b , la mesure de similarité S utilisant la distance de Jaccard peut être définie comme suit [79]:

$$S_{Jaccard}(a, b) = \frac{|E_a \cap E_b|}{|E_a \cup E_b|} \quad \mathbf{F14}$$

Il s'agit du rapport entre la cardinalité de l'intersection des ensembles considérés et la cardinalité de l'union des ensembles. En revanche, si les données sont des notes les deux

Les Systèmes de Recommandation

mesures les plus couramment utilisées sont la similarité Cosinus et la similarité de Pearson définies comme suit pour deux produits a et b [79] (et symétriquement dans le cas de deux items a et b) :

$$S_{Cosinus}(a, b) = \frac{\sum_{\{x \in E_a \cap E_b\}} u(a, x) \times u(b, x)}{\sqrt{\sum_{\{x \in E_a \cap E_b\}} u(a, x)^2 \sum_{\{x \in E_a \cap E_b\}} u(b, x)^2}} \quad \mathbf{F15}$$

$$S_{Pearson}(a, b) = \frac{\sum_{\{x \in E_a \cap E_b\}} (u(a, x) - \bar{u}_a) \times (u(b, x) - \bar{u}_b)}{\sqrt{\sum_{\{x \in E_a \cap E_b\}} (u(a, x) - \bar{u}_a)^2 \sum_{\{x \in E_a \cap E_b\}} (u(b, x) - \bar{u}_b)^2}} \quad \mathbf{F16}$$

- \bar{u}_a (respectivement \bar{u}_b) représente la moyenne des valeurs contenus dans le vecteur a (respectivement b)
- Un ensemble C de n utilisateurs $\{c1, c2, \dots, cn\}$;
- Un ensemble P de m items $\{p1, p2, \dots, pm\}$;
- Un ensemble U de mesures $u(p, c)$ de l'utilisateur $c \in C$ pour l'item $p \in P$;
- $E_c \subseteq P$ l'ensemble des items mesurés par l'utilisateur c ;
- $E_p \subseteq C$ l'ensemble des utilisateurs ayant mesuré l'item p

Une matrice de similarités Items-Items (tableau 13) ou Utilisateurs-Utilisateurs (tableau 14) est alors établie et utilisée pour prédire les valeurs manquantes, i.e les non votes ou les non achats.

Tableau 13 : Matrice de similarités Items-Items.

	P1	P2	P3
P1		S(P1,P2)	S(P1,P3)
P2	S(P2,P1)		S(P2,P3)
P3	S(P3,P1)	S(P3,P2)	

Tableau 14 : Matrice de similarités Utilisateurs-Utilisateurs.

	C1	C2	C3
C1		S(C1,C2)	S(C1,C3)
C2	S(C2,C1)		S(C2,C3)
C3	S(C3,C1)	S(C3,C2)	

III.4.2. Filtrage basé sur le modèle :

Les approches basées sur des modèles construisent des modèles statistiques de modèles d'évaluation utilisateur / élément afin de fournir des prédictions d'évaluation automatiques. Les algorithmes de filtrage collaboratif basés sur un modèle fournissent des recommandations d'éléments en développant d'abord un modèle d'évaluation d'audience.

Les algorithmes de cette catégorie adoptent une approche probabiliste et envisagent le processus de filtrage collaboratif comme le calcul de la valeur attendue d'une prévision de l'utilisateur, compte tenu de ses évaluations d'autres éléments. Le processus de création de

Les Systèmes de Recommandation

modèle est effectué par différents algorithmes d'apprentissage machine tels que le réseau bayésien, la mise en cluster et les approches basées sur des règles. Le modèle de réseau bayésien [88] formule un modèle probabiliste pour le problème du filtrage collaboratif. Le modèle de classification traite le filtrage collaboratif comme un problème de classification [91] et fonctionne en regroupant des utilisateurs similaires dans la même classe et en estimant la probabilité qu'un utilisateur particulier se trouve dans une classe particulière C , puis à partir de là, calcule la probabilité conditionnelle d'évaluation.

III.5. Évaluation des systèmes de recommandation

Shani et Gunawardana [92] expliquent qu'initialement la plupart des systèmes de recommandation ont été évalués en fonction de leur capacité à prédire avec précision les choix de l'utilisateur. Maintenant, il est largement admis que la précision des prédictions est cruciale mais insuffisante pour déployer un bon système de recommandation.

Selon [93], les évaluations des systèmes de recommandations peuvent être effectuées en utilisant une analyse hors ligne (offline analysis) ou une expérimentation avec des utilisateurs réels (live user experiment). Il existe une autre classification des méthodes d'évaluation des systèmes de recommandation. Comme expliqué dans [94], ces méthodes d'évaluation sont classées en trois types : expérimentations offline, études avec des utilisateurs (user studies) et tests réels (real life testing). Ce dernier type est nommé expérimentations en ligne (Online experiments). Ces trois méthodes sont expliquées par la suite.

III.5.1. Évaluation offline

Une grande partie du travail d'évaluation des algorithmes des systèmes de recommandation s'est concentrée sur l'analyse hors ligne de la précision des prédictions que peuvent faire ces systèmes [93]. Les évaluations offline utilisent des ensembles de données (dataset) constitués d'actions des utilisateurs (principalement des évaluations de ressources). Les évaluations offline simulent le processus de recommandation où un sous ensemble des actions utilisateurs du dataset est caché et le système de recommandation prédit ces actions cachées. Le système de recommandation est évalué en fonction de sa capacité à prédire ces interactions cachées. Les résultats de ces prédictions sont analysés en utilisant une ou plusieurs métriques.

Deux types d'ensembles de données sont souvent utilisés dans ces évaluations :

- **Ensembles de données naturels** : ils sont constitués de données issues de l'historique des interactions d'utilisateurs réels dans un système donné sur une période donnée. De nombreux dataset sont disponibles pour mener des évaluations sur des algorithmes de recommandation.
- **Ensembles de données de synthèse** : ils sont construits de données artificielles. Ce type de dataset est habituellement utilisé pour tester comment les algorithmes de recommandation fonctionnent dans certaines conditions.

Les Systèmes de Recommandation

Les évaluations offline ont l'avantage d'être rapide et économique et peuvent être réalisées sur plusieurs ensembles de données ou plusieurs algorithmes différents à la fois. Ce type d'évaluation est une évaluation objective des résultats de la prédiction. Aucune analyse hors ligne ne peut déterminer si les utilisateurs préfèrent un système particulier, soit en raison de ses prédictions, soit en raison d'autres critères moins objectifs tels que l'esthétique ou l'ergonomie de l'interface utilisateur.

III.5.2. Étude sur les utilisateurs

Les auteurs dans [95] et [96] expliquent que l'objectif d'un système de recommandation va au-delà des métriques de précision. Les évaluations se basant sur les métriques de précision ne répondent pas à la question de savoir si les utilisateurs sont réellement satisfaits des recommandations proposées par le système. Comme mentionné dans [97], près de 50% du succès commercial d'un système de recommandation est lié aux interactions entre les utilisateurs et ce système alors que l'algorithme de recommandation ne compte que pour 5%. Le point de vue de l'utilisateur sur le système de recommandation est encore plus important dans le domaine de l'éducation [98] [94]. En plus de la précision, les systèmes de recommandation appartenant à ce domaine doivent prendre en compte des mesures de qualité telles que l'utilité, la nouveauté ou la diversité des recommandations. Le point de vue de l'utilisateur peut être recueilli via une étude utilisateurs. Il s'agit d'une méthode utilisée pour découvrir comment un système de recommandation influence l'expérience, la perception et les interactions d'un utilisateur avec un système [96]. Une étude sur les utilisateurs est menée en recrutant un ensemble d'utilisateurs, et en leur demandant d'effectuer certaines tâches dans un environnement contrôlé pendant une courte période de temps. L'interaction entre les utilisateurs et le système de recommandation est observée et des informations sont enregistrées telles que le temps nécessaire pour terminer la tâche ou la qualité des résultats de la tâche. En plus de l'observation du comportement de l'utilisateur, il est possible de faire passer des questionnaires aux utilisateurs pour recueillir des données qui ne sont pas directement observables telles que l'appréciation de l'interface utilisateur ou de la pertinence des recommandations.

III.5.3. Évaluation online :

L'évaluation online peut aussi recueillir le point de vue de l'utilisateur concernant le système de recommandation. Dans ce type d'évaluation, des utilisateurs réels utilisent le système dans des conditions réelles sur une longue période [99]. Selon [100], ce type d'évaluation peut montrer les usages et les habitudes d'utilisation des utilisateurs, les problèmes et les besoins non satisfaits, et les problèmes que les chercheurs n'ont peut-être pas envisagés dans une étude utilisateurs. Avec ces tests réels sur le terrain, la plupart des objectifs centrés sur l'utilisateur peuvent être efficacement évalués, comme l'évaluation de l'expérience utilisateur, la satisfaction des utilisateurs ou la rétention des utilisateurs [100] [92].

Les Systèmes de Recommandation

III.6. Conclusion

Ce chapitre avait pour objectif de faire un tour d'horizon, non exhaustif, sur les systèmes de recommandation. Nous avons présenté les trois principales approches de systèmes de recommandation : celles basées sur le contenu, celles basées sur un filtrage collaboratif, et finalement les approches hybrides. En comparant entre ces approches, nous avons opté pour l'approche de recommandation sociale basée sur la mémoire, plus particulièrement sur les items. En effet, cette approche est la mieux adaptée à notre problématique de recommandation des sources où les items représentent les sources qui seront notées par les utilisateurs. Cette notation représente la pertinence de la source qui est un critère de qualité important à considérer dans le processus de recommandation.

CHAPITRE IV

APROCHE PROPOSEE

“ On a les qualités qu'on veut avoir “

Bernard Lazar

Approche proposée

IV.1. Introduction

Dans ce chapitre, nous allons présenter notre proposition qui consiste en une approche de recommandation sociale des sources basée sur la qualité. L'approche proposée permet d'améliorer les performances des systèmes de recherche d'information multi-sources en ne sélectionnant que des sources qui présentent un niveau de qualité de service (QoS) élevé pour une requête donnée. La QoS est utilisée pour mesurer la pertinence du service fourni par une source donnée. Plusieurs critères de QoS peuvent être utilisés, tels que le temps de réponse, la pertinence, la couverture, etc. Nous identifions trois critères importants de QoS pour mesurer la pertinence d'une source pour une requête, à savoir le temps de réponse, la disponibilité et la fréquence de mise à jour. Pour la recommandation, l'historique de l'utilisation des sources est exploité pour fournir des recommandations plus précises.

Nous commençons ce chapitre par une description générale de notre solution. Ensuite, nous expliquons le processus de sélection et le processus de recommandation des sources d'information. Enfin, nous terminons par une étude conceptuelle de notre application.

IV.2. Description Générale

Dans un système de recherche multi-sources, choisir un sous-ensemble de sources pour effectuer la recherche et ne pas interroger toutes les sources disponibles réduit le coût d'accès aux sources et diminue le temps de réponse total. Nous proposons une approche de recommandation de sources d'information basée sur l'historique d'utilisation des sources et les relations entre les utilisateurs et les sources. Les commentaires des utilisateurs sur la qualité des sources utilisées sont exploités pour recommander des sources qui correspondent aux exigences des utilisateurs, exprimées de manière implicite sur la base de leurs expériences de recherche passées. Par conséquent, les sources les mieux notées sont déterminées en utilisant une technique de filtrage collaboratif.

Les niveaux réels de QoS des sources sont dérivés de l'historique d'utilisateurs similaires et / ou de sources similaires. Pour distinguer le grand nombre de sources, la qualité de service (QoS) est considérée comme un facteur clé dans le processus de sélection et de recommandation de sources. Les attributs de QoS que nous avons considérés dans notre approche sont le temps de réponse, la disponibilité, la fréquence de mise à jour.

Pour la sélection des sources, nous avons proposé d'utiliser les méthodes de prise de décision multicritères permettant de classer les alternatives sur la base de différents critères caractérisant ces alternatives pour sélectionner la ou les meilleures alternatives.

La figure 13 présente l'architecture générale de l'approche proposée, où nous distinguons les entités suivantes :

- **Utilisateur** : qui introduit la requête de recherche pour déclencher le processus de sélection.
- **Un ensemble (ou pool) des sources** : où chaque source d'information est caractérisée par son nom, son adresse, la liste des contenus disponibles, les dernières valeurs des métriques de qualités. La liste des sources est maintenue par le courtier.

Approche proposée

- **Le courtier** : qui joue le rôle d'intermédiaire entre l'utilisateur et les différentes sources d'information. Le courtier agit comme un pseudo-moteur de recherche qui reçoit en entrée une requête et fournit en sortie une liste de résultats.

Une fois l'utilisateur lance une requête de recherche, le processus de sélection se déclenche en se basant sur un ensemble de critères de qualité pour choisir les meilleures sources. Ensuite, le courtier va transmettre la requête aux sources sélectionnées pour recevoir les résultats qui vont être transmis à l'utilisateur. Enfin, ce dernier peut exprimer son appréciation ou sa satisfaction des résultats de recherche retournés. Les données des commentaires des utilisateurs sont utilisées pour générer les recommandations à un nouvel utilisateur.

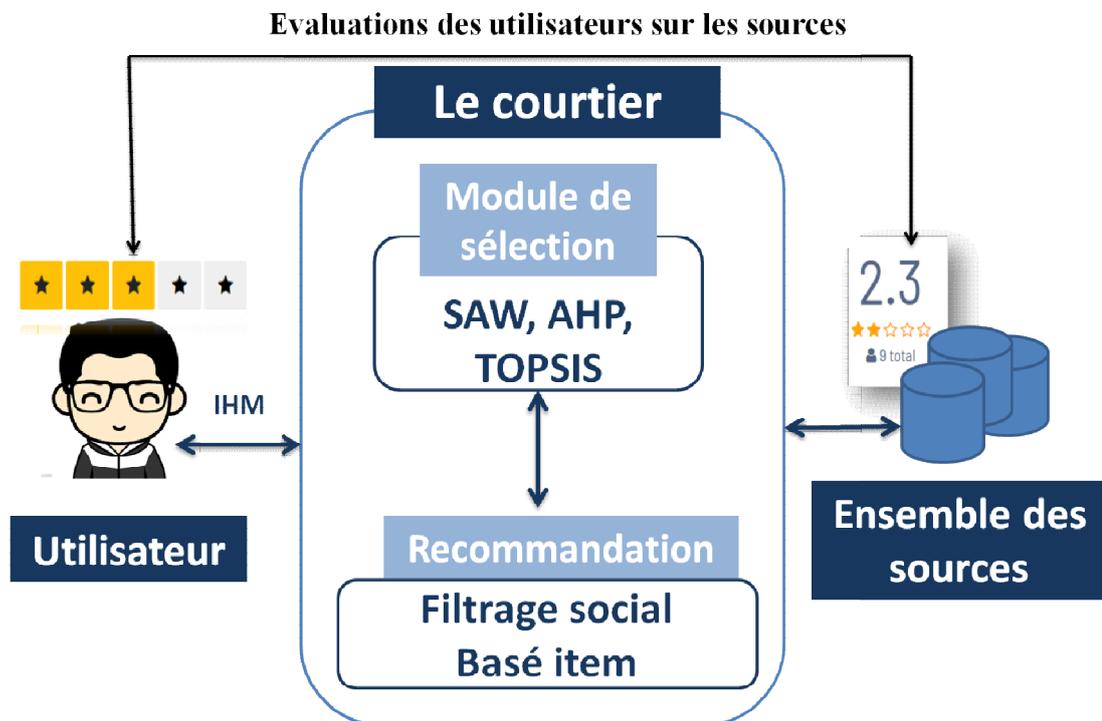


Figure 13 : Architecture générale de l'approche proposée.

La figure 14 montre les principales phases de l'approche proposée, à savoir la phase de prétraitement, le processus de sélection et le processus de recommandation. Ces trois phases sont décrites en détail dans les sous sections suivantes.

Approche proposée

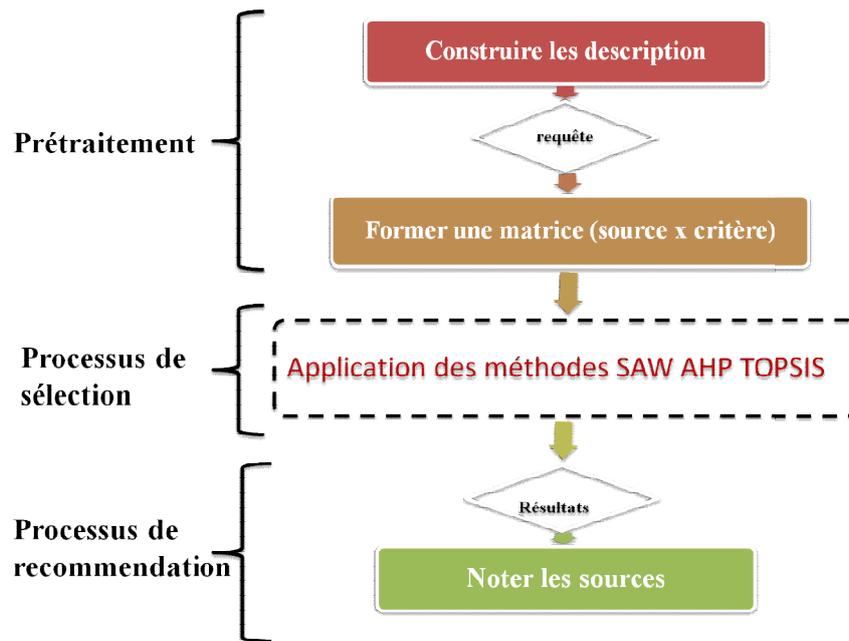


Figure 14 : Les principales phases de l'approche proposée.

IV. 3. Phase de Prétraitement

La phase de prétraitement est considérée une étape importante pour la sélection et la recommandation des sources. Elle consiste à identifier les critères de qualité de service pertinents pour la sélection et la recommandation des sources, et à construire une description précise des sources disponibles.

IV.3.1. Critères de la qualité choisis

Nous définissons la qualité de la source comme la combinaison des trois critères de QoS présentés dans le tableau suivant:

Tableau 15 : Les critères de QoS choisis avec leurs mesures.

Critère	Description	Formule de mesure
Temps de Réponse (millisecondes)	Est le temps qu'une source a besoin pour répondre à une demande de l'utilisateur. Il inclut le temps du transfert du réseau et le temps du démarrage du serveur ainsi que le temps de réponse du serveur.	La différence entre le temps (instant) de la réception de la réponse et le temps (instant) d'envoi de la requête
Disponibilité	Est le fait que cette source soit accessible et opérationnelle. Elle est influencée par la disponibilité du réseau, la disponibilité du serveur ainsi que la disponibilité des données	Le rapport du nombre de requêtes réussies sur le nombre total des requêtes
Fréquence des Mises à Jour	Représente la fraîcheur de contenu de la source	Nombre des mises à jour faites pour un mois

Approche proposée

Ces critères sont classés selon le diagramme Ishikawa suivant:

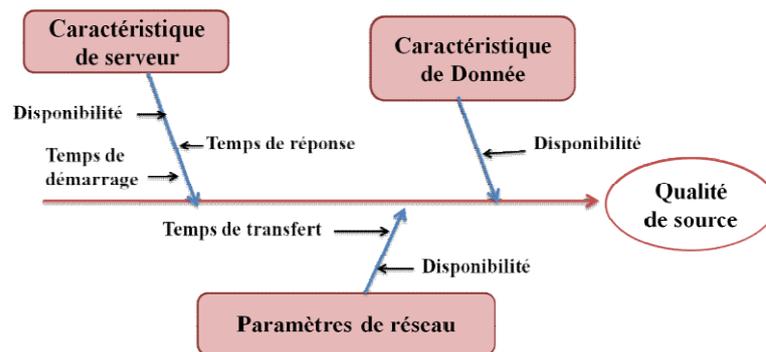


Figure 15 : Diagramme d'Ishikawa des métriques choisies.

Nous considérons une source de bonne qualité qui présente un temps de réponse faible et une fréquence de mise à jour et un taux de disponibilité élevés.

IV.3.2. Construction des descriptions

La sélection et la recommandation des sources sont basées entre autres sur des informations décrivant les sources disponibles. Le courtier construit une description précise de chaque source qui le gère. En plus de son nom, son adresse et une brève description de son contenu, la description de la source doit contenir des informations sur les paramètres de QoS. La figure 16 montre un exemple de description d'une source au format XML.

This XML file does not appear to have any style information associated with it. The document tree is shown below.

```
<source>
  <nom>S2</nom>
  <mot-clé>java</mot-clé>
  <adresse>http://localhost:8001</adresse>
  <désignation>La désignation de la source S2</désignation>
  <temps-réponse>0.125s</temps-réponse>
  <disponibilité>20/40</disponibilité>
  <fréquence-mise-à-jour>10 par mois</fréquence-mise-à-jour>
</source>
```

Figure 16: exemple de description d'une source.

Nous considérons un environnement de recherche coopératif, où les sources coopèrent pour fournir les informations nécessaires qui décrivent son contenu et son état. Le courtier peut estimer le taux de disponibilité de chaque source sur la base du nombre des requêtes envoyées aux sources et les réponses retournées. Pour le temps de réponse, il est calculé en fonction du moment où une requête est envoyée et du moment où une réponse est reçue.

IV.3.3. Matrice de décision

Les vecteurs de description des sources sur les attributs des critères de QoS utilisés peuvent être combinés pour former la matrice de décision qui est une matrice $n \times m$ (n sources et m critères de QoS). Dans notre cas $m=3$ (trois critères sont considérés)

Où chaque x_{ij} est l'évaluation de la source i par rapport au critère j .

Le tableau 16 présente la matrice de décision, où :

Approche proposée

- Si est la ième source, C_j est le jème critère
- x_{ij} est la mesure de l'ième source pour le jème critère

Un classement de priorité k est donné d'une manière arbitraire comme suit

Temps de réponse (K1) → Disponibilité (K2) → Fréquence de mise à jour (K3)

Tableau 16 : La matrice de décision.

Sources	C1 (TR)	C2 (Disp)	C4 (Fmaj)
Poids relatifs	W_1	W_2	W_3
S_1	x_{11}	x_{12}	x_{13}
S_2	x_{21}	d_{22}	x_{23}
...
S_n	x_{n1}	x_{n2}	x_{n4}

IV. 4. Processus de sélection

La sélection des sources est formalisée comme un problème de prise de décision multicritères, qui consiste à trouver la meilleure alternative qui représente une valeur de QoS élevée. Nous choisissons trois critères de QoS pour la sélection des sources, à savoir le temps de réponse, la disponibilité et la fréquence de mise à jour. Nous proposons d'utiliser trois méthodes différentes de prise de décision multicritères pour la sélection de la meilleure alternative. Nous utilisons les méthodes d'agrégation SAW, AHP, et TOPSIS.

IV.4.1 Algorithme de sélection

Dans notre système, nous allons suivre le scénario de la sélection fixe (section I.4.2) pour sélectionner un nombre prédéfini (K) de sources qui seront classées selon la meilleure valeur de QoS. Cette qualité est calculée en utilisant l'une des méthodes d'agrégation, à savoir SAW, AHP, TOPSIS. L'algorithme de sélection peut être donné comme suit :

Algorithme 1 : sélection des sources basée QoS.

Entrées : un ensemble de sources S , $S = \{s_1, s_2, \dots, s_n\}$, un ensemble de critères C , $C = \{\text{temps de réponse, disponibilité, fréquence de mise à jour}\}$ et le nombre de sources à sélectionner K .

Sorties : Les « K » meilleures sources.

Début

Pour chaque source S_i appartenant à l'ensemble S faire
début

Calculer son score de qualité en appliquant la fonction de score relative à l'une des méthodes suivantes : SAW, AHP ou TOPSIS

Classifier les sources par ordre décroissant (plus grand au plus petit) de leur score

Sélectionner les « K » sources qui ont les scores les plus élevés

fin

Algorithme 1 : Processus décisionnel de sélection

Approche proposée

IV.4.2. Application des méthodes de calcul de score

Nous avons proposé d'utiliser trois méthodes de calcul de score, à savoir SAW, AHP et TOPSIS afin de pouvoir comparer les résultats obtenus par chaque méthode.

Nous allons décrire ces trois méthodes par l'exemple illustratif suivant.

Exemple :

Soit dix (10) sources numérotées de 1 jusqu'à 10. Le tableau suivant résume les valeurs des critères de QoS, (temps de réponse (TR), disponibilité (Disp) , fréquence de mise à jour (FMaj)) de chacune des sources.

Tableau 17 : valeurs des critères de QoS pour l'ensemble des sources disponibles.

Source	TR	Disp	F Maj
S1	150	3/5	4
S2	200	4/7	3
S3	210	5/7	5
S4	167	1/4	2
S5	243	2/6	3
S6	120	3/5	4
S7	124	4/5	5
S8	265	5/7	4
S9	291	3/5	3
S10	97	2/5	2

IV.4.2.1. Application de la méthode SAW

Le principe de la technique SAW consiste à affecter un vecteur de qualité à chaque source candidate dans la sélection. Ce vecteur contient les valeurs des critères de qualité qui caractérisent chaque source en formant une matrice de décision. La figure suivante montre les étapes de l'application de la méthode SAW dans le processus de sélection.

Approche proposée

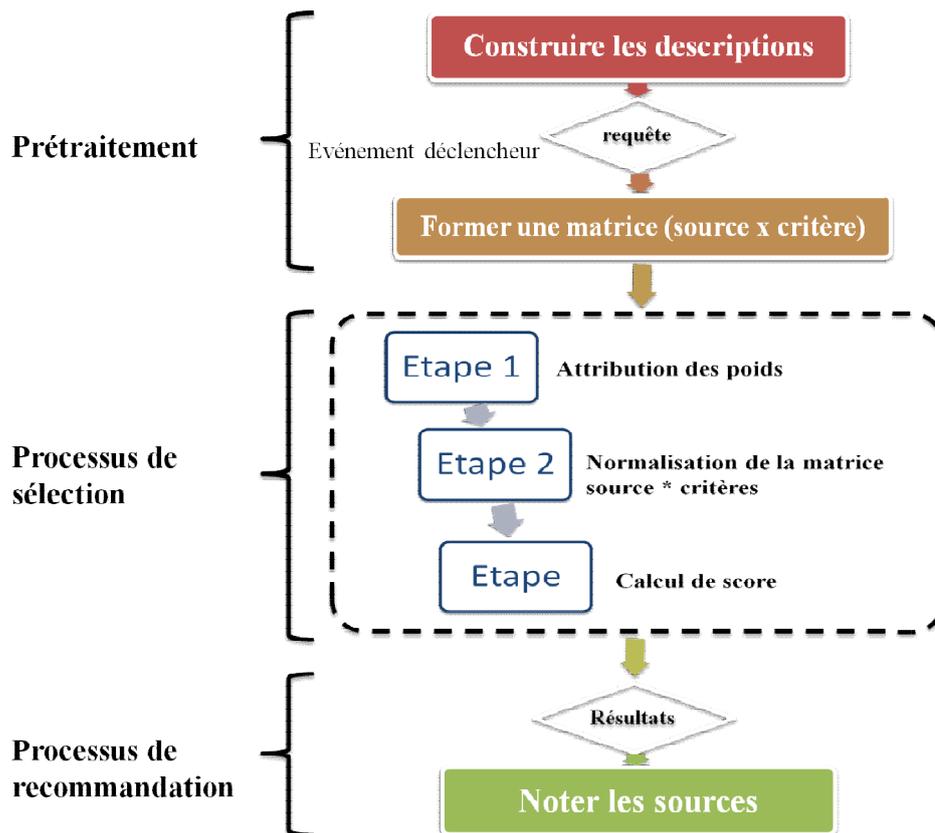


Figure 17 : Application de la méthode SAW.

Étape 1: Attribution des poids W_j

- Les poids W_j sont définis d'une manière arbitraire selon un classement de priorité k , calculés en utilisant la règle (F1) suivante :

$$W_j = \frac{(\sum_{i=k}^m \frac{1}{i})}{m} \quad \mathbf{m \text{ est le nombre de critères} \quad (F1)}$$

$$\text{Poid } W \text{ (TR)} = \frac{1}{3} = 0.33 \quad \text{Poid } W \text{ (Disp)} = \frac{1}{2} = 0.17 \quad \text{Poid } W \text{ (F Maj)} = \frac{1}{3} = 0.11$$

Étape 2: Normalisation de la matrice

- Vu que les attributs (critères) n'ont pas les mêmes unités de mesures, il faut donc les mettre tous dans une échelle commune, d'où le terme normalisation s'est inspiré. Cette étape consiste à normaliser les mesures afin d'obtenir des valeurs entre 0 et 1. La normalisation se fait selon les formules suivantes :

$$v_{ij} = \begin{cases} \frac{x_{ij}}{\max x_{ij}} & \text{pour les critères à maximiser} \quad (F2); \\ \frac{\min x_{ij}}{x_{ij}} & \text{pour les critères à minimiser} \quad (F3). \end{cases}$$

Où x_{ij} représente la valeur non normalisée de l'alternative i et le critère j

Approche proposée

- Noter que le temps de réponse est un critère à minimiser, la disponibilité et la fréquence de mise à jour à maximiser.

Les calculs nécessaires pour la source (S1)

$$V_{\text{temps de réponse}} = \frac{97}{150} = 0.65 \quad \text{en suivant la formule F3}$$

$$V_{\text{disponibilité}} = \frac{3/5}{4/5} = 0.75 \quad \text{en suivant la formule F2}$$

$$V_{\text{fréquence maj}} = \frac{4}{5} = 0.8 \quad \text{en suivant la formule F2}$$

Étape 3: calcul du score

Le score de qualité de chaque source est calculé par cette formule :

$$\text{score}(S_i) = \sum w_j v_{ij} \quad (\text{F4}) \quad 0 < \text{score}(s_i) \leq 1$$

$$\begin{aligned} \text{Score (source S1)} &= W_{\text{temps de réponse}} * V_{\text{temps de réponse}}(S1) + W_{\text{disponibilité}} * V_{\text{disponibilité}}(S1) \\ &+ W_{\text{fréquence de maj}} * V_{\text{fréquence de maj}}(S1) = 0.33 * 0.65 + 0.17 * 0.85 + 0.11 * 0.8 = 0.45 \end{aligned}$$

Le tableau suivant montre les différents calculs :

Tableau 18 : résultats obtenus par la méthode Saw.

Source	TR	Disp	F Maj	Score
Poids	0.33	0.17	0.11	
S1	$\frac{97}{150} = 0.65$	$\frac{3/5}{4/5} = 0.75$	$\frac{4}{5} = 0.8$	0.43
S2	$\frac{97}{200} = 0.49$	$\frac{4/7}{4/5} = 0.71$	$\frac{3}{5} = 0.6$	0.35
S3	$\frac{97}{210} = 0.46$	$\frac{5/7}{4/5} = 0.89$	$\frac{5}{5} = 1$	0.41
S4	$\frac{97}{167} = 0.58$	$\frac{1/4}{4/5} = 0.31$	$\frac{2}{5} = 0.4$	0.28
S5	$\frac{97}{243} = 0.4$	$\frac{2/6}{4/5} = 0.42$	$\frac{3}{5} = 0.6$	0.27
S6	$\frac{97}{120} = 0.81$	$\frac{3/5}{4/5} = 0.75$	$\frac{4}{5} = 0.8$	0.49
S7	$\frac{97}{124} = 0.78$	$\frac{4/5}{4/5} = 1$	$\frac{5}{5} = 1$	0.54
S8	$\frac{97}{265} = 0.37$	$\frac{5/7}{4/5} = 0.89$	$\frac{4}{5} = 0.8$	0.36
S9	$\frac{97}{291} = 0.33$	$\frac{3/5}{4/5} = 0.75$	$\frac{3}{5} = 0.6$	0.31

Approche proposée

S10	$\frac{97}{97}=1$	$\frac{2/5}{4/5}=0.5$	$\frac{2}{5}=0.4$	0.46
-----	-------------------	-----------------------	-------------------	------

Le classement des sources selon la méthode SAW est : S6- S7- S10- S1- S3- S8- S2- S9- S4- S5.

IV.4.2.2. Application la méthode AHP

Depuis son invention [71], AHP est très utilisée par les chercheurs et les praticiens, c'est une méthodologie systématique, flexible et simple pour l'aide à la décision multicritères. L'application de la méthode AHP est montrée dans la figure 18.

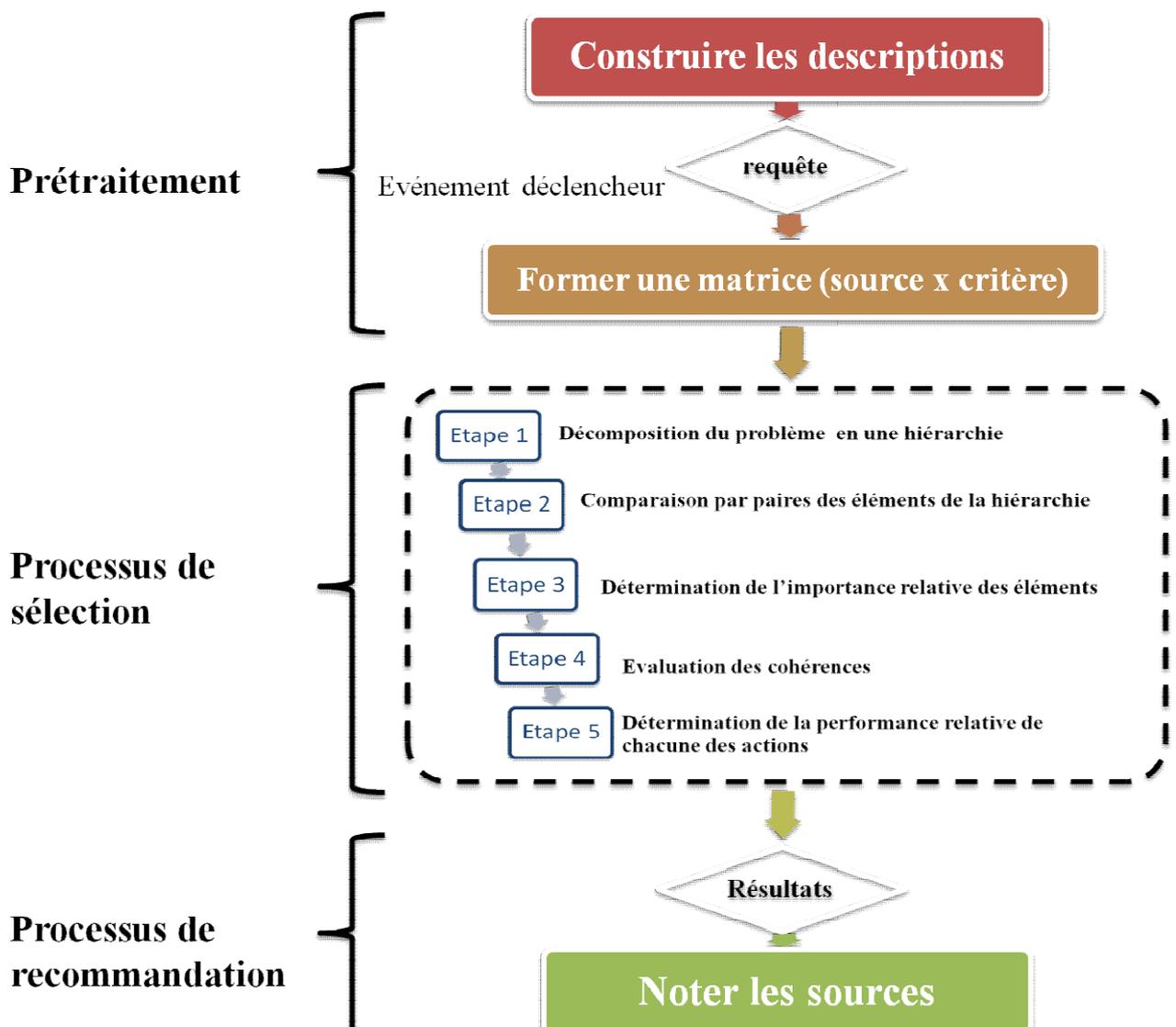


Figure 18 : Application de la méthode AHP.

Approche proposée

Étape 1: décomposition du problème en une hiérarchie

Pour notre exemple, la décomposition du problème donne l'hiérarchie suivante (Figure 19). L'objectif est de sélectionner des sources dont la qualité est élevée. Les critères de sélection retenus sont les critères choisis pour la qualité d'une source, dix (10) sources sont définies comme des alternatives.

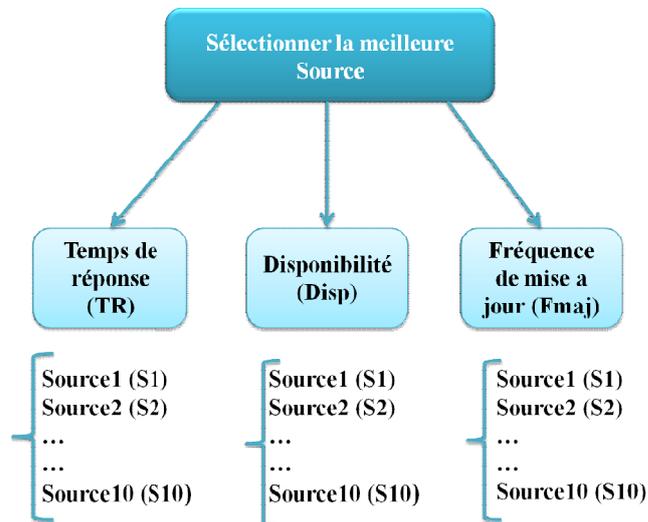


Figure 19 : décomposition hiérarchique.

Étape 2 : comparaison par paires des éléments de la hiérarchie

Il s'agit ici de calculer l'importance relative de chacun des éléments de la hiérarchie à partir des évaluations obtenues de l'étape précédente.

Tableau 19 : matrice de critères de décision

	TR	Disp	F Maj
TR	1	3	5
Disp	1/3	1	3
F Maj	1/5	1/3	1
Somme	1.53	4.33	9

Étape 3: Calcul du vecteur de priorité

Effectuer les sommes de colonne, et puis diviser les éléments de la matrice par la somme de la colonne

Approche proposée

Tableau 20 : détermination des priorités.

	TR	Disp	FMaj	Moyenne
TR	$\frac{1}{1.53} = 0.65$	$\frac{3}{4.33} = 0.69$	$\frac{5}{9} = 0.56$	0.63
Disp	$\frac{0.33}{1.53} = 0.22$	$\frac{1}{4.33} = 0.23$	$\frac{3}{9} = 0.33$	0.26
FMaj	$\frac{0.2}{1.53} = 0.13$	$\frac{0.33}{4.33} = 0.08$	$\frac{1}{9} = 0.11$	0.11
Somme \cong	1	1	1	1

Etape 4 : Evaluation des cohérences

- Calcul de la valeur propre λ_{\max}

Nous allons multiplier la matrice de décision par les éléments de vecteur de priorité. Nous calculons la moyenne des valeurs trouvées. Le résultat est appelé valeur λ_{\max} .

Tableau 21 : Multiplication de la matrice de décision par les priorités.

	TR	Disp	F Maj	Somme
TR	1	3	5	1.96
Disp	$\frac{1}{3}$	1	3	0.8
F Maj	$\frac{1}{5}$	$\frac{1}{3}$	1	0.11

0.63^* (multiplying factor for TR row)
 0.26^* (multiplying factor for Disp row)
 0.11^* (multiplying factor for F Maj row)

En divisant les éléments du vecteur de la somme pondérée par la priorité correspondant à chaque critère comme suit :

$$\begin{aligned} \text{TR} &= 1.96/0.63=1.23 \\ \text{Disp} &= 0.8/0.26=3.08 \\ \text{F Maj} &= 0.11/0.11=1 \\ \lambda_{\text{Max}} &= 5.31/3=1.7 \end{aligned}$$

- Détermination de la valeur d'Indice Aléatoire (IA)

Saaty a élaboré une échelle où les indices aléatoires IA furent établis en effectuant des jugements aléatoires pour un nombre élevé de réplification. Ce nombre IA représente la moyenne des indices

Approche proposée

calculés à chaque réplique pour différente grandeur de matrice carrée (N). La lecture de la valeur d'IA est indiquée par le tableau d'indice aléatoire.

Nous avons $N = 3$, la valeur correspondante est égale à $IA = 0,58$

▪ Calcul de l'Indice de Cohérence :

IC Pendant l'étape d'explication des jugements, nous faisons des comparaisons redondantes pour améliorer la validité des réponses, vu que les décideurs peuvent être incertains ou peuvent faire des jugements pauvres pour la comparaison de quelques éléments. L'indice de cohérence (IC) est le rapport entre la différence de la valeur propre λ max moins le nombre de comparaison sur ce dernier moins un.

Nous avons : $IC = (1.77 - 3) / (3 - 1) = IC = -0.615$ (Section II.5.2.)

▪ Calcul du Ratio de Cohérence (RC)

Le Ratio de Cohérence (RC) est le rapport de l'indice de cohérence calculé sur la matrice correspondant aux jugements du décideur et de l'indice aléatoire IA d'une matrice de même dimension. $RC = IC / IA$ (Section II.5.2)

Si $RC \leq 0,1$ la matrice est considérée comme suffisamment cohérente, dans le cas où cette valeur dépasse 10%, les appréciations peuvent exiger certaines révisions.

Nous avons $IC = -0.615$ et $IA = 0,58$ alors $RC = IC/IA =$ soit -1.06 $RC < 0.10$

Le degré de cohérence de comparaison est limite acceptable.

▪ Établissement de tableau de comparaison complète de critères

Nous récapitulons dans un tableau appelé « matrice de jugement des critères : priorité complète » les résultats de calculs tels que : la matrice de jugement, la priorité, la valeur propre λ max, l'indice de cohérence, et le ratio de cohérence.

Tableau 22: matrice de jugement des critères : priorité complète.

	TR	Disp	F Maj	Priorité
TR	1	3	6	0.63
Disp	1/3	1	3	0.26
F Maj	1/6	1/3	1	0.11
$\Lambda_{max} = 1.77$		$IC = -1.06$	$RC = -1.06$	

Etape 5 : Détermination de la performance relative de chacune des actions

Le calcul de score de chaque source est fait en suivant la formule F4 (Section II.5.1).

Approche proposée

$$\text{score } (S_i) = \sum w_j v_{ij}$$

Le tableau suivant résume les calculs nécessaires pour le calcul du score de chaque source.

Tableau 23: résultats obtenus par la méthode AHP.

Source	TR	Disp	F Maj	Score
Poids	0.63	0.26	0.11	
S1	$\frac{97}{150}=0.65$	$\frac{3/5}{4/5}=0.75$	$\frac{4}{5}=0.8$	0.70
S2	$\frac{97}{200}=0.49$	$\frac{4/7}{4/5}=0.71$	$\frac{3}{5}=0.6$	0.57
S3	$\frac{97}{210}=0.46$	$\frac{5/7}{4/5}=0.89$	$\frac{5}{5}=1$	0.63
S4	$\frac{97}{167}=0.58$	$\frac{1/4}{4/5}=0.31$	$\frac{2}{5}=0.4$	0.49
S5	$\frac{97}{243}=0.4$	$\frac{2/6}{4/5}=0.42$	$\frac{3}{5}=0.6$	0.43
S6	$\frac{97}{120}=0.8$	$\frac{3/5}{4/5}=0.75$	$\frac{4}{5}=0.8$	0.79
S7	$\frac{97}{124}=0.72$	$\frac{4/5}{4/5}=1$	$\frac{5}{5}=1$	0.86
S8	$\frac{97}{265}=0.37$	$\frac{5/7}{4/5}=0.89$	$\frac{4}{5}=0.8$	0.55
S9	$\frac{97}{291}=0.33$	$\frac{3/5}{4/5}=0.75$	$\frac{3}{5}=0.6$	0.47
S10	$\frac{97}{97}=1$	$\frac{2/5}{4/5}=0.5$	$\frac{2}{5}=0.4$	0.80

Le classement des sources selon la méthode AHP est : S10- S6- S7- S1- S3- S8- S2- S4- S9- S5

IV.4.2.3. Application de la méthode TOPSIS :

Le principe de la méthode TOPSIS est de sélectionner les sources les plus proche de la source idéale favorable et plus loin de la source défavorable.

L'application de la méthode TOPSIS est montrée dans la figure 20.

Approche proposée

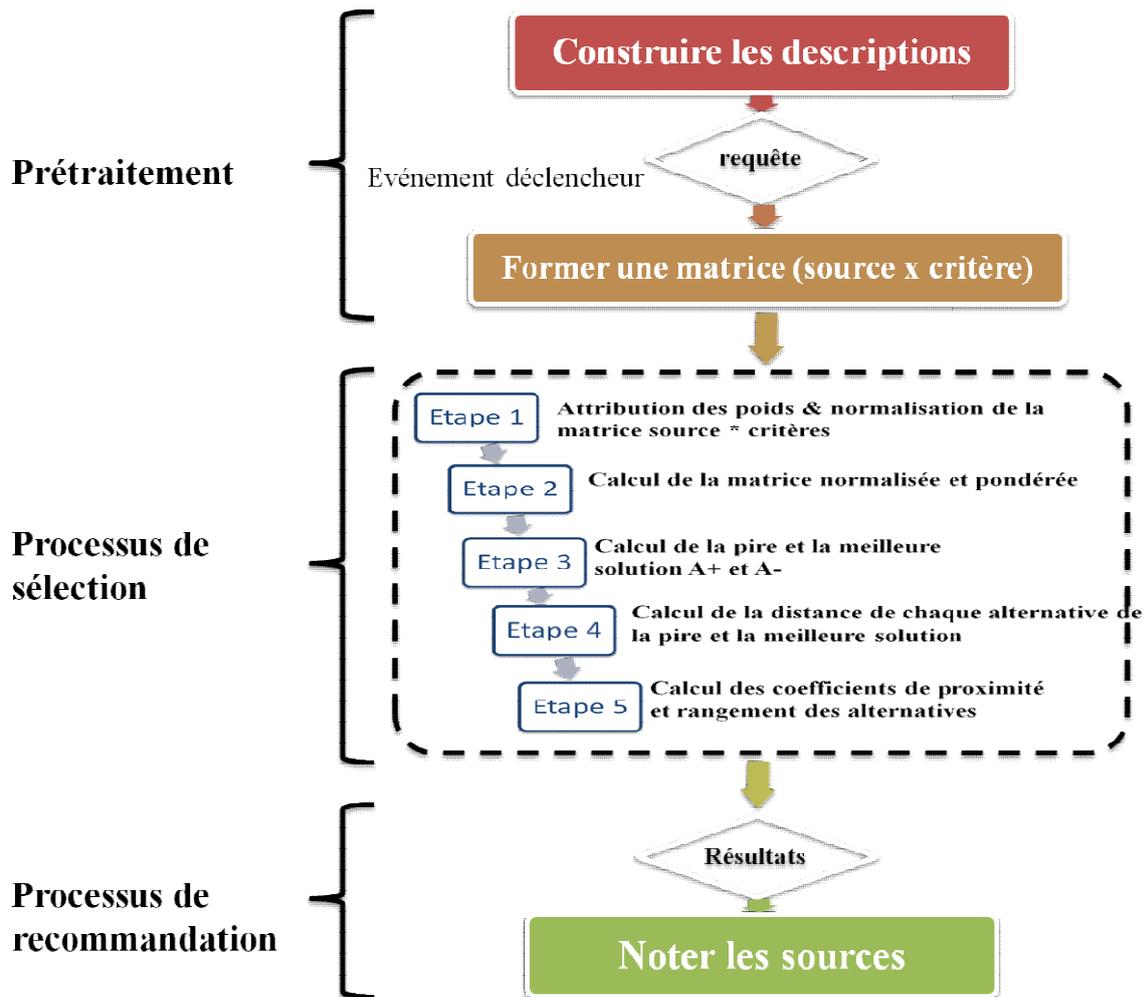


Figure 20 : Application de la méthode TOPSIS.

Etape 1: Normalisation de la matrice

- Nous avons utilisé pour la normalisation de la matrice la formule F11 (section II.5.3) :

$$v_{ij} = \frac{x_{ij}}{\sqrt{\sum_{i=1}^n x_{ij}^2}}$$

$$v_{TR}(s1) = \frac{150}{\sqrt{150^2 + 200^2 + \dots + 97^2}} = \frac{150}{\sqrt{387629}} = 0.24$$

Notez que nous avons opté pour une normalisation euclidienne, car les métriques qui seront calculées dans la suite sont basées sur la distance euclidienne ce qui garantit des résultats cohérents.

- Les poids W_j sont définis d'une manière arbitraire selon un classement de priorité k , On associe à chaque critère une valeur d'importance de 1 à 10 telque le critère favorable prends la valeur la plus grande et le critère défavorable prend la valeur la plus petite.
 - Les importances associées pour les critères choisis sont comme suit :

Importance (TR) = 3 ; Importance (Disp) = 2 ; Importance (F Maj) = 1

Approche proposée

Ensuite, nous allons ramener les poids à l'intervalle [0, 1] en divisant tout simplement chaque poids par la somme de tous les poids. Les pondérations suivantes sont attribuées aux 4 critères dans l'ordre :

$$\text{Poid } W(\text{TR}) = \frac{3}{3 + 2 + 1} = 0.5 \quad ; \quad \text{Poid } W(\text{Disp}) = \frac{2}{3 + 2 + 1} = 0.33 \quad ;$$

$$\text{Poid } W(\text{F Maj}) = \frac{1}{3 + 2 + 1} = 0.17$$

Tableau 24: matrice normalisée –TOPSIS.

Source	TR	Disp	F Maj
Poids	0.5	0.33	0.17
S1	0.24	0.33	0.35
S2	0.32	0.31	0.26
S3	0.34	0.39	0.43
S4	0.27	0.14	0.17
S5	0.39	0.18	0.26
S6	0.19	0.33	0.35
S7	0.20	0.43	0.43
S8	0.43	0.39	0.35
S9	0.47	0.33	0.26
S10	0.16	0.22	0.17

Etape 3 : Calcul de la matrice pondérée

- Multiplication des valeurs de la matrice par les poids associés aux critères (voir le tableau 25).

Etape 4 : calcul de pire et meilleure solution A- & A+ :

On calcule la valeur associée la plus favorable A+ en fonction de la nature du critère (à minimiser /à maximiser). Les calculs sont montrés dans le tableau suivant.

- Si le critère est à maximiser, on choisit la plus grande valeur de chaque colonne.
- Si le critère est à minimiser on sélectionne la plus petite valeur de chaque colonne.

Tableau 25: Matrice pondérée / calcul de A+ et A-.

Approche proposée

Source	TR	Disp	F Maj
Poids	0.5	0.33	0.17
S1	0.12	0.11	0.06
S2	0.16	0.10	0.04
S3	0.17	0.13	0.07
S4	0.14	0.05	0.03
S5	0.20	0.06	0.04
S6	0.10	0.11	0.06
S7	0.1	0.14	0.07
S8	0.22	0.13	0.06
S9	0.24	0.11	0.04
S10	0.08	0.07	0.03
A+	0.08	0.14	0.07
A-	0.24	0.05	0.03

Étape 5 : Calcul de E+ & E- :

- les vecteurs E+ , E- expriment la distance de chaque alternative de la (meilleure solution A+/ pire solution A-)

Nous calculons pour chaque alternative son écart par rapport à la valeur la plus favorable A+ (respectivement la plus pire A-) déjà évaluées par la formule F12 (Section II.5.3):

$$E_{\pm} = \sqrt{\sum \sum_j^n ((A_{\pm})_j - x_{ij})^2}$$

Prenons la source S1 : $E+(S1) = \sqrt{(0.08 - 0.12)^2 + (0.14 - 0.11)^2 + (0.07 - 0.06)^2}$

$$E-(S) = \sqrt{(0.24 - 0.12)^2 + (0.05 - 0.11)^2 + (0.03 - 0.06)^2}$$

- le coefficient de proximité de chaque alternative est considéré comme score de la source. il est obtenu par la formule F13 (section II. 5.3) :

$$Score = \frac{E-}{(E+) + (E-)}$$

Approche proposée

Tableau 26: Calcul E+ & E-/ Résultats obtenus par la méthode TOPSIS.

Source	E+	E-	Score
S1	0.05	0.13	0.72
S2	0.09	0.09	0.50
S3	0.09	0.11	0.55
S4	0.12	0.10	0.45
S5	0.14	0.04	0.22
S6	0.04	0.16	0.80
S7	0.02	0.17	0.89
S8	0.14	0.09	0.39
S9	0.17	0.06	0.26
S10	0.08	0.18	0.69

Le classement des sources selon la méthode TOPSIS est : S7- S6- S1-S10-S3- S2- S4-S8-S9- S5.

IV.4.2.4. Comparaison des résultats obtenus par l'application des trois méthodes

Le tableau suivant présente une récapitulation sur les scores obtenus par les 3 méthodes :

Tableau 27: Classement des sources selon les 3 méthodes appliquées.

Classement des sources	SAW	AHP	TOPSIS
1.	S7	S7	S7
2.	S6	S10	S6
3.	S10	S6	S1
4.	S1	S1	S10

Approche proposée

5.	S3	S3	S3
6.	S8	S2	S2
7.	S2	S8	S4
8.	S9	S4	S8
9.	S4	S9	S9
10.	S5	S5	S5

D'après le tableau 27, nous remarquons que les résultats obtenus par les trois méthodes sont presque identiques. Si par exemple on prend les $k=5$ meilleures sources on obtient les sources S6, S7, S10, S1, S3 avec un peu de désordre.

Cette différence peut être due à la pondération des critères de QoS. Chaque méthode utilise une méthode différente pour définir les poids utilisés dans le calcul du score de pertinence.

Dans le chapitre suivant, des tests seront effectués pour comparer les trois méthodes en fonction le temps d'exécution.

IV.5. Processus de Recommandation

Nous proposons une approche de recommandation qui utilise le filtrage collaboratif basé item. L'approche proposée exploite le comportement des utilisateurs (évaluations) afin de recommander les sources de bonne qualité à un utilisateur donné. Le score de pertinence de chaque source est estimé sur la base de la similarité des sources. La similarité entre deux sources est calculée en fonction des notes données par les utilisateurs pour exprimer leurs opinions sur les sources consultées. Les sources ayant un score le plus élevé (k sources) sont recommandées à l'utilisateur.

Le processus de recommandation passe par deux phases :

IV.5.1. Phase 1 « la pertinence de la source » :

Cette phase consiste à construire une matrice de pertinence des sources pour tous les utilisateurs en fonction des étapes suivantes :

- La collecte des appréciations des utilisateurs sur les sources. Pour cela, nous demandons à l'utilisateur de noter lui-même les résultats de recherche et ainsi leur source à partir d'une échelle de note fixée de 1 à 5 (Rating stars)
- Calculer la pertinence de chaque source en suivant la formule suivante :

$$Pert = \frac{(p - 1) * Pert + (Note \times conf)}{p} \quad F17$$

où, p : présente le nombre des notes données par un utilisateur

Approche proposée

Conf : est une valeur de confiance donnée pour un type d'adhérent (utilisateur) en fonction de son niveau de formation (professeur, doctorant, ingénieur etc.).

Par exemple, nous attribuons des valeurs de confiance à ces types d'adhérent :

Tableau 28 : confiance donnée pour chaque type d'adhérent

Type d'adhérent	Professeur	doctorant	FE Master	Stagiaire2Cycle	FE Licence	Stagiaire1Cycle
confiance	0.9	0.8	0.7	0.6	0.5	0.4

Cette confiance est attribuée d'une manière arbitraire afin de filtrer les évaluations données par les utilisateurs et pour éliminer les fausses évaluations (données par des utilisateurs non compétents).

Notons que la formule F17 est une formule de récurrence qui permet de calculer la moyenne en connaissant seulement la dernière valeur enregistrée de la pertinence et le nombre des notes données. Ce qui nous permet de ne pas sauvegarder toutes les notes données pour calculer la moyenne comme dans la formule itérative.

En conclusion, nous obtenons une matrice d'évaluation utilisateur/source, cette matrice est creuse car y'aura des utilisateurs qui n'ont pas donné d'avis sur toutes les sources.

IV.5.2. Phase 2 « La recommandation » :

Cette phase permet d'abord de trouver les sources similaires aux sources sélectionnées par la phase de sélection et notées par l'utilisateur. La formule de Pearson F16 (Section III.4.1) est utilisée pour calculer la similarité entre deux sources.

$$S_{Pearson}(a, b) = \frac{\sum_{\{x \in E_a \cap E_b\}} (u(a, x) - \bar{u}_a) \times (u(b, x) - \bar{u}_b)}{\sqrt{\sum_{\{x \in E_a \cap E_b\}} (u(a, x) - \bar{u}_a)^2 \sum_{\{x \in E_a \cap E_b\}} (u(b, x) - \bar{u}_b)^2}}$$

Les étapes à suivre sont indiquées dans la figure 22. : Nous commençons par sélectionner deux sources, une de ces sources doit être déjà notée par l'utilisateur. Puis, nous recherchons tous les utilisateurs qui ont évalué ces sources pour construire leur vecteur d'évaluation. Ensuite, nous calculons la similarité entre les deux vecteurs en utilisant l'algorithme 2. Enfin, un score de recommandation est calculé pour chaque source sur la base des similarités des sources et les notes données par les utilisateurs. Les sources qui ont une valeur de score élevée sont recommandées à l'utilisateur.

Notons ici que pour les nouveaux utilisateurs qui n'ont pas d'historique de recherche, nous recommandons les sources ayant une pertinence élevée étant donné que notre recommandation s'appuie sur la popularité auprès des utilisateurs.

Approche proposée

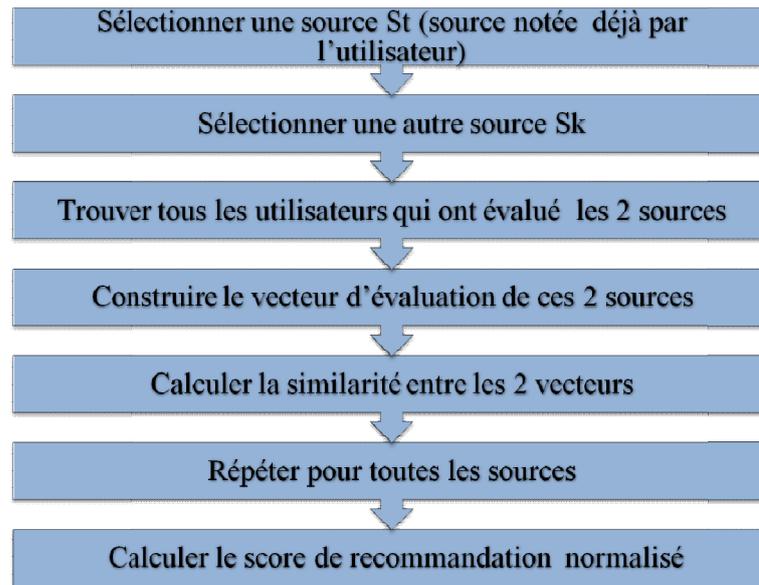


Figure 21: Etapes de calcul de la similarité des sources.

L'algorithme 2 décrit le calcul de la similarité des sources.

Algorithme 2 : calcul la similarité entre deux sources.

Entrée : deux sources s_1 et s_2 et leurs évaluations

Sortie : $\text{Sim}(s_1, s_2)$, la similarité entre s_1 et s_2 .

Début

1 : Identifier les évaluations en commun entre deux sources (s_1 et s_2)

2 : Construire le vecteur d'évaluation des utilisateur sur la source s_1 : $S_1=(a_1, b_1, \dots, z_1)$ et celui de source s_2 : $S_2=(a_2, b_2, \dots, z_2)$

3 : Calculer les deux sommes : $S_1=a_1+ \dots +z_1$ et $S_2=a_2+ \dots +z_2$ F18

4 : Calculer la sommes des carrés : $Sq_1=a_1^2+ \dots +z_1^2$ et $Sq_2=a_2^2+ \dots +z_2^2$ F19

5 : Calculer la somme des produits $P_{tot}=a_1* a_2 + \dots + z_1* z_2$ F20

6 : Calculer la similarité de Pearson (la formule F21) entre les deux sources s_1 et s_2 :

$$\text{sim}(s_1, s_2) = \text{Pearson}(S_1, S_2) = \frac{P_{tot} - \frac{S_1 * S_2}{n}}{\sqrt{\left[Sq_1 - \frac{S_1^2}{n}\right] * \left[Sq_2 - \frac{S_2^2}{n}\right]}} \quad \text{F21} \quad / (n \text{ est la longueur des vecteurs})$$

Écrire $\text{sim}(s_1, s_2)$

Fin

Approche proposée

Notons que la formule F21 et la formule F16 (Section III.) sont la même. La première est la formule de calcul de Pearson, alors que la deuxième est la formule conceptuelle de la même formule.

Notons aussi que le calcul du score de recommandation est donné par les étapes suivantes :

1. Multiplication de la similarité (sim) par l'évaluation (eval), $sim * eval$ (pour toutes les sources)

la similarité de Pearson peut être négative, donc nous prenons les valeurs positives des similarités.

2. La somme des valeurs de similarité positives $\sum sim$
3. La somme des valeurs positive $\sum sim * eval$
4. Score normalisé = $\frac{\sum sim * eval}{\sum sim}$

Exemple

La matrice d'évaluation utilisateur/source de 3 utilisateurs qui ont donné leurs évaluation au 4 sources S7, S3, S6, S10 (voir l'exemple dans le tableau 29).

On peut voir que c'est une matrice creuse. l'utilisateur Aicha a noté la source S7 et S6 et n'as pas donner son avis sur les sources S3 et S10.

Nous allons chercher si la source S10 est similaire a la source S7 déjà noté

Tableau 29: Exemple d'une matrice d'évaluation utilisateur/source

source Utilisateur	S7	S3	S6	S10
Aicha	3		1	
Bilal		2.5	2	4.5
Ali	2		2	4

a. Calcul de similarité

Le calcul de la similarité entre les deux sources S7 et S10 de l'exemple présenté dans le tableau 28 est montré dans le tableau 30.

Tableau 30: calculs nécessaires pour calculer la similarité entre S7 et S10.

	S7	S10
	3	1
	2	4
Somme	3+2=5	1+4=5
Somme des carrés	3 ² +2 ² =13	1 ² +4 ² =17
Somme des produits	3*1+2*4=11	

Approche proposée

En utilisant les résultats de tableau 30, la similarité entre S7 et S10 est donnée comme suit :

$$Pearson (S7, S10) = \frac{11 - \frac{5 * 5}{2}}{\sqrt{\left[13 - \frac{5^2}{2}\right] * \left[17 - \frac{5^2}{2}\right]}}$$

$$Pearson (S7, S10) = \frac{-1.5}{\sqrt{[0.5]*[4.5]}} = -1 < 0 \text{ (non similaires)}$$

b. Calcul de score de recommandation

Exemple : Nous considérons deux sources non consultées S7 et S4. Le but est de recommander une de ces deux sources à un utilisateur donné. Pour choisir la source à recommander, on doit calculer le score de recommandation (normalisé) pour chacune des deux sources.

Le score de recommandation est calculé pour les sources S7 et S4 comme indiqué dans le tableau suivant :

Tableau 31 : calcul du score de recommandation

	Evaluation	S7(sim)	S7(Sim*Eval)	S4(Sim)	S4(Sim*Eval)
S1	4	0.7	2.8	0.34	1.37
S8	3.5	0.4	1.4	0.25	0.87
S10	1	-1	-1	-0.6	-0.6
Total		1.1	4.2	0.59	2.24
Score Normalisé		4.2/1.1=3.82		2.24/0.59=3.80	

La source avec le score le plus élevé sera recommandée à l'utilisateur. À partir du tableau 30, score (S7)= 3.82 et score (S4)= 3.80, alors la source S7 est recommandée à l'utilisateur.

IV.6. Conclusion

Dans ce chapitre, nous avons décrit la solution proposée pour la sélection et la recommandation des sources basée sur la QoS lorsque l'environnement de recherche est composé de plusieurs sources d'information. Dans la suite, nous présenterons la conception et l'implémentation de l'approche proposée.

CHAPITRE V
CONCEPTION ET
IMPLEMENTATION

Conception et Implémentation

V.1. Introduction :

Dans ce chapitre, nous allons présenter la partie de réalisation de notre solution qui a pour objectif de mettre en œuvre la solution décrite dans le chapitre précédent. Nous allons commencer par une description de l'environnement de développement. Ensuite, nous allons présenter les différentes interfaces de notre application.

V.2. Etude Conceptuelle de notre application:

Nous abordons dans cette partie la conception du notre système, dans laquelle, nous détaillons les différents éléments de conception, Nous construisons une vue dynamique de la solution sous la forme d'un diagramme de cas d'utilisation et aussi une vue statique sous forme de diagramme de classe.

V.2.1. Diagramme de cas d'utilisation:

Le diagramme de cas d'utilisation décrit les utilisations requises d'un système ou ce qu'un système est supposé faire. Les principaux concepts de ce diagramme sont les acteurs et les cas d'utilisation. Nous mettons en évidence, dans cette section, les différentes fonctionnalités du système avec l'utilisateur.

Dans notre système, nous pouvons distinguer les acteurs suivants :

- **Utilisateur:** c'est lui qui déclenche les différentes fonctionnalités du système, il entre en contact avec le système par : la recherche de l'information. On a deux types d'utilisateur : **un visiteur** qui peut faire une recherche ou s'inscrire, et **un adhérent** ayant un compte qui possède des fonctionnalités avancées tel que la recommandation et la consultation de l'historique
- **Administrateur:** le responsable de système, la gestion des sources, la gestion des adhérents.

Conception et Implémentation

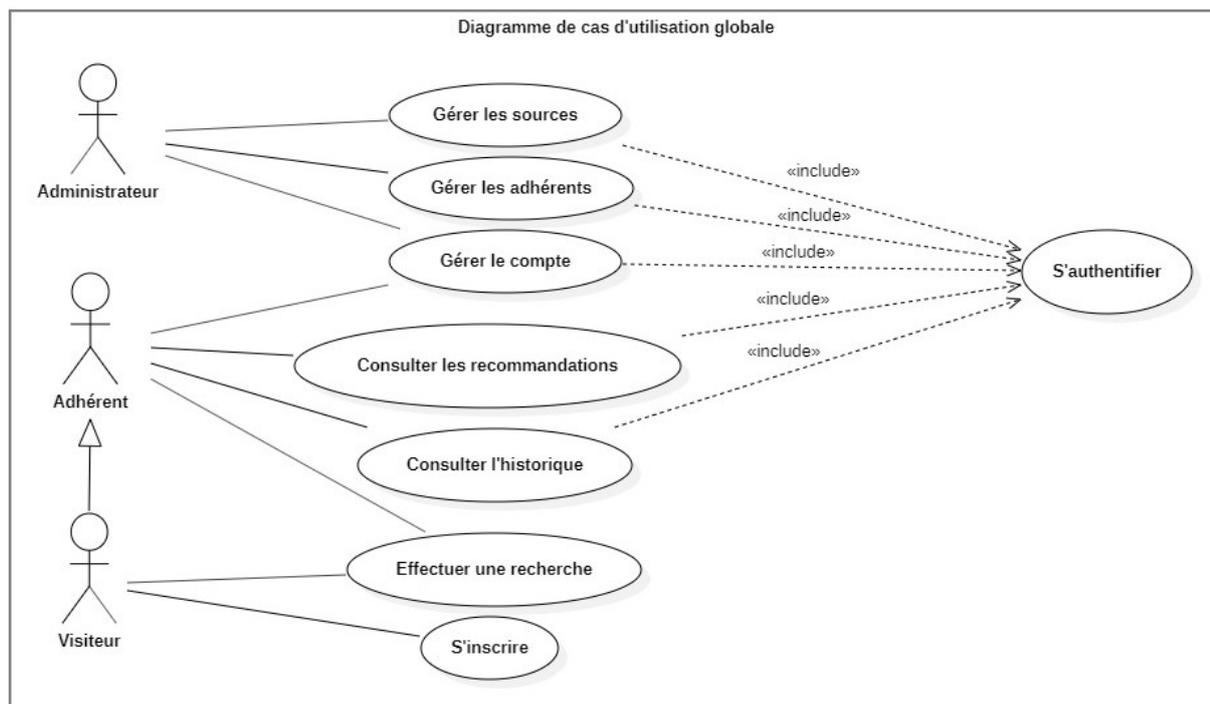


Figure 22: Diagramme de cas d'utilisation du système.

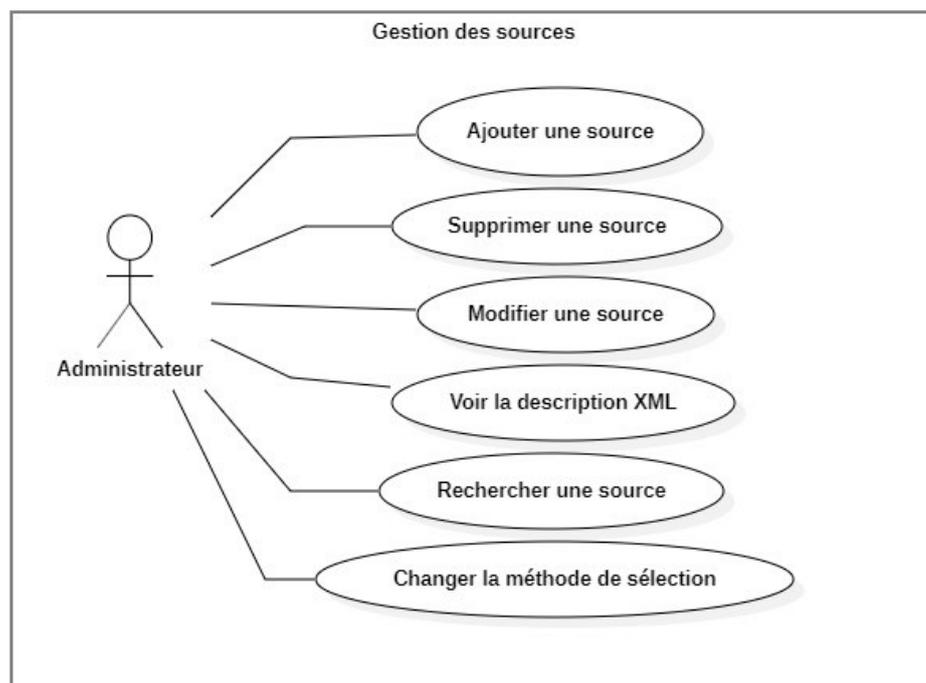


Figure 23: Diagramme de cas d'utilisation « Gérer les sources ».

Conception et Implémentation

Tableau 32 : la description de diagramme de cas d'utilisation

Cas d'utilisation	Acteur	Description
S'inscrire	Visiteur	Le visiteur peut s'inscrire en introduisant les informations essentielles dans un formulaire fourni.
Effectuer une recherche	Visiteur Adhérent	Le visiteur peut faire une recherche, mais il ne peut pas recevoir des recommandations et il ne peut pas donner son avis (il doit être inscrit d'abord).
Gérer le compte	Adhérent Administrateur	Un adhérent/ administrateur peut gérer son compte (modifier les informations de son compte).
Consulter l'historique	Adhérent	L'adhérent peut voir les traces de ces recherches.
Consulter la recommandation	Adhérent	L'adhérent a la possibilité de voir les sources recommandées.
Gérer les sources	Administrateur	L'administrateur peut ajouter, modifier et supprimer les sources. Il peut aussi consulter la description des sources. Changer la méthode de sélection.
Gérer l'adhérent	Administrateur	L'administrateur peut accepter ou refuser une inscription, rechercher ou supprimer un compte.

V.2.2. Diagrammes de Séquence

Le diagramme de séquence permet de montrer les interactions d'objets dans le cadre d'un scénario d'un diagramme des cas d'utilisation.

- **Diagramme de séquence « effectuer une recherche »** (figure 24) : Après avoir saisi la requête par l'utilisateur, le processus de sélection des sources est déclenché comme décrit dans la section IV.4. L'utilisateur reçoit ensuite le résultat trié selon le classement des sources. Une fois connecté, Adhérent peut donner son avis sur les sources.

Conception et Implémentation

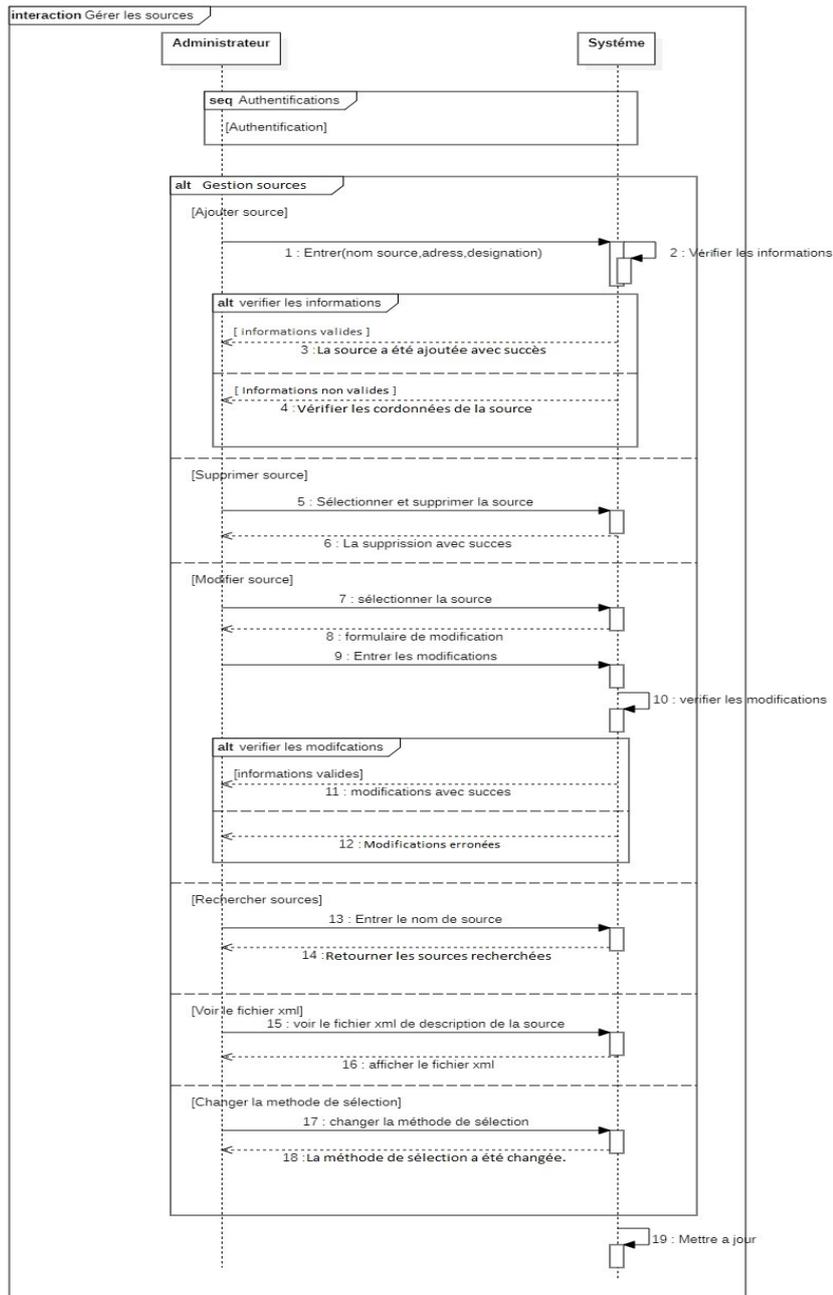


Figure 24 : Diagramme de séquence « Effectuer une recherche ».

- **Diagramme de séquence « gerer les sources »** (figure 25) : Une fois connecté, l'administrateur peut gérer le pool des sources en ajoutant une nouvelle source, modifiant/supprimant une source ou bien en changeant la méthode de sélection

Conception et Implémentation

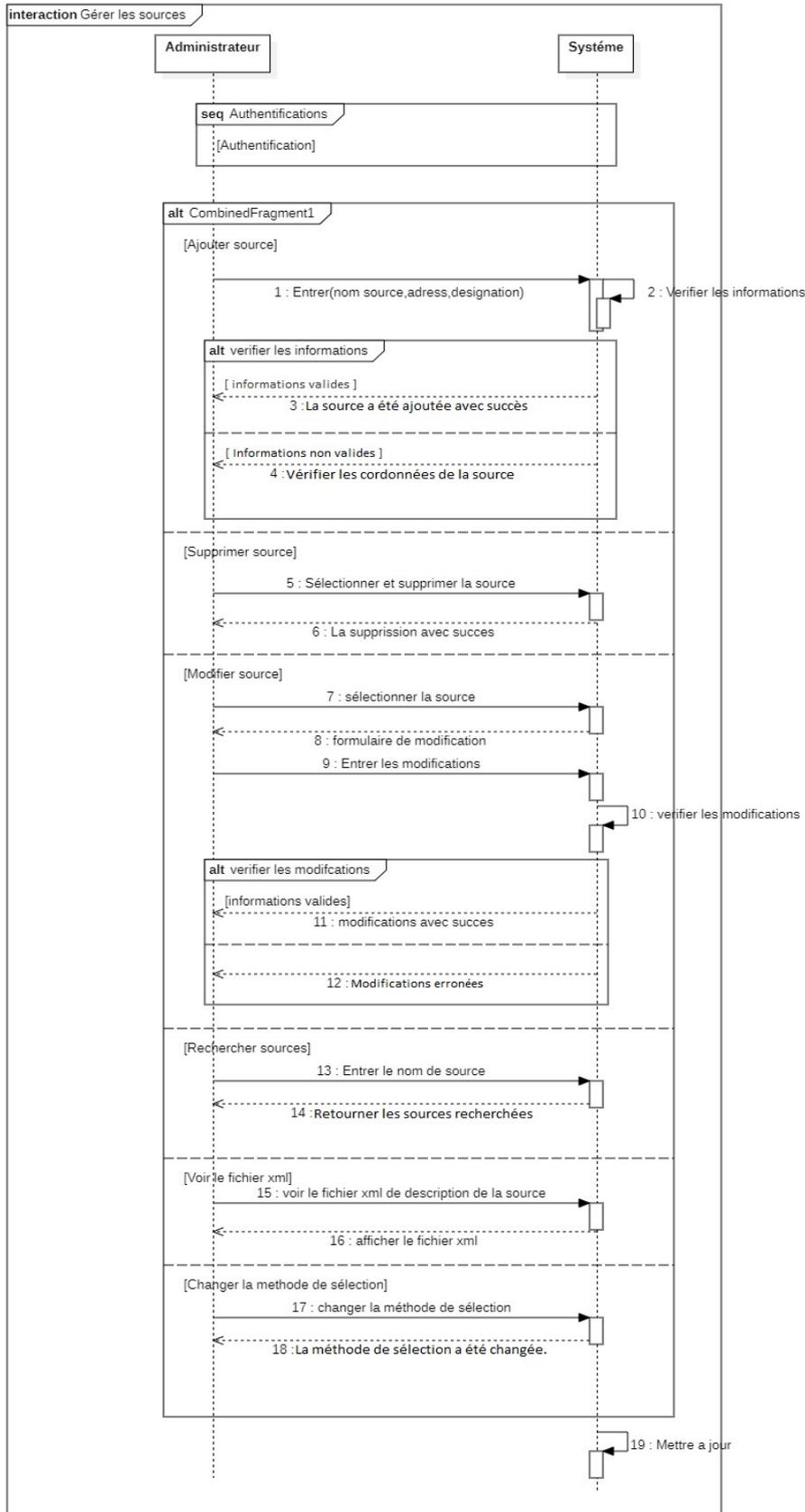


Figure 25: Diagramme de séquence « Gerer les sources ».

Conception et Implémentation

- **Diagramme de séquence « consulter les recommandations »** (figure 26) : L'adhérent peut demander de consulter des recommandations, ainsi le système recherche les sources déjà notées et les sources non notées puis il calcule la similarité afin de classer les sources qui seront affichées aux utilisateurs.

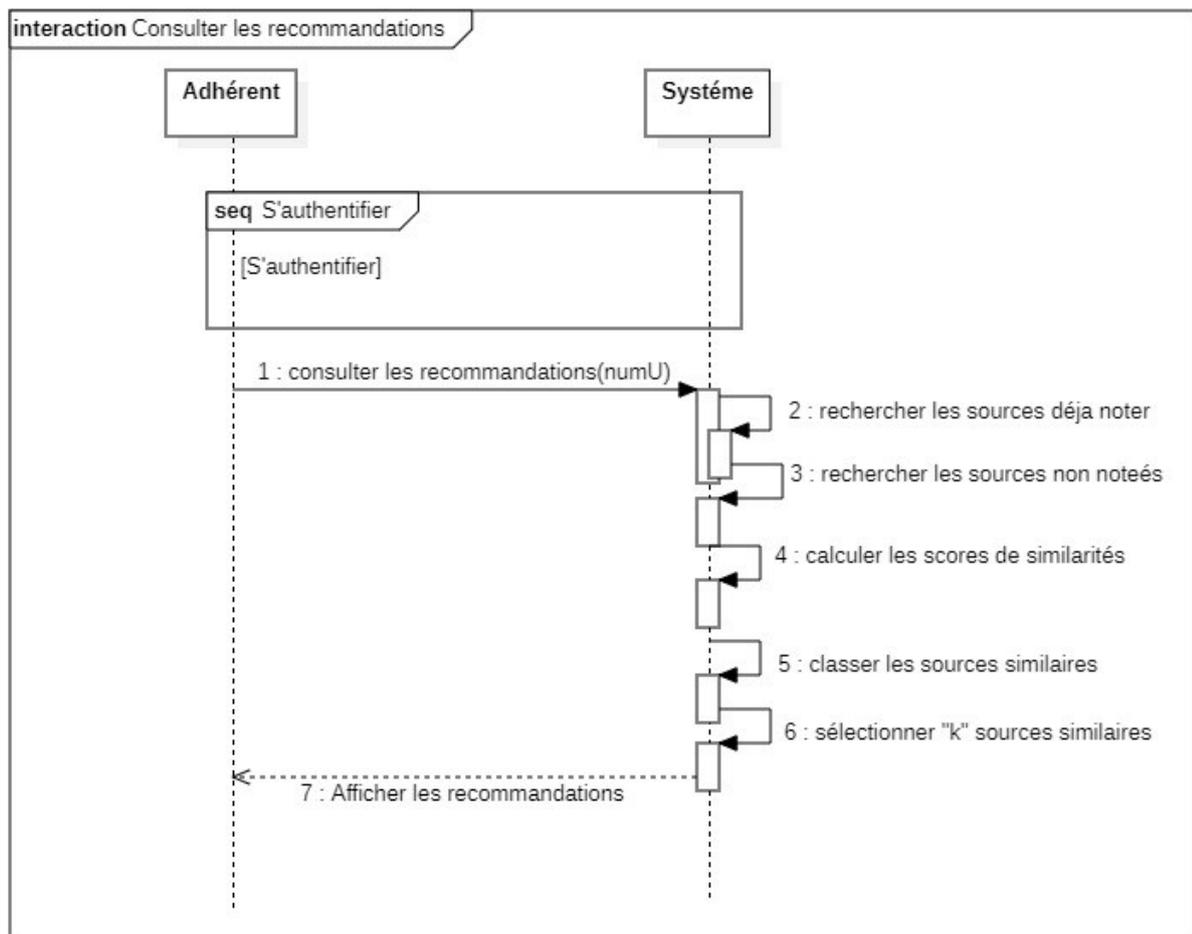


Figure 26: Diagramme de séquence « consulter les recommandations ».

V.2.3. Diagramme de classes :

Le diagramme de classes est une modélisation statique du système en termes de classes et de relations entre ces classes. Son intérêt réside dans la modélisation des entités du système. Nous présentons dans cette partie les différentes classes du système et la relation entre eux.

Conception et Implémentation

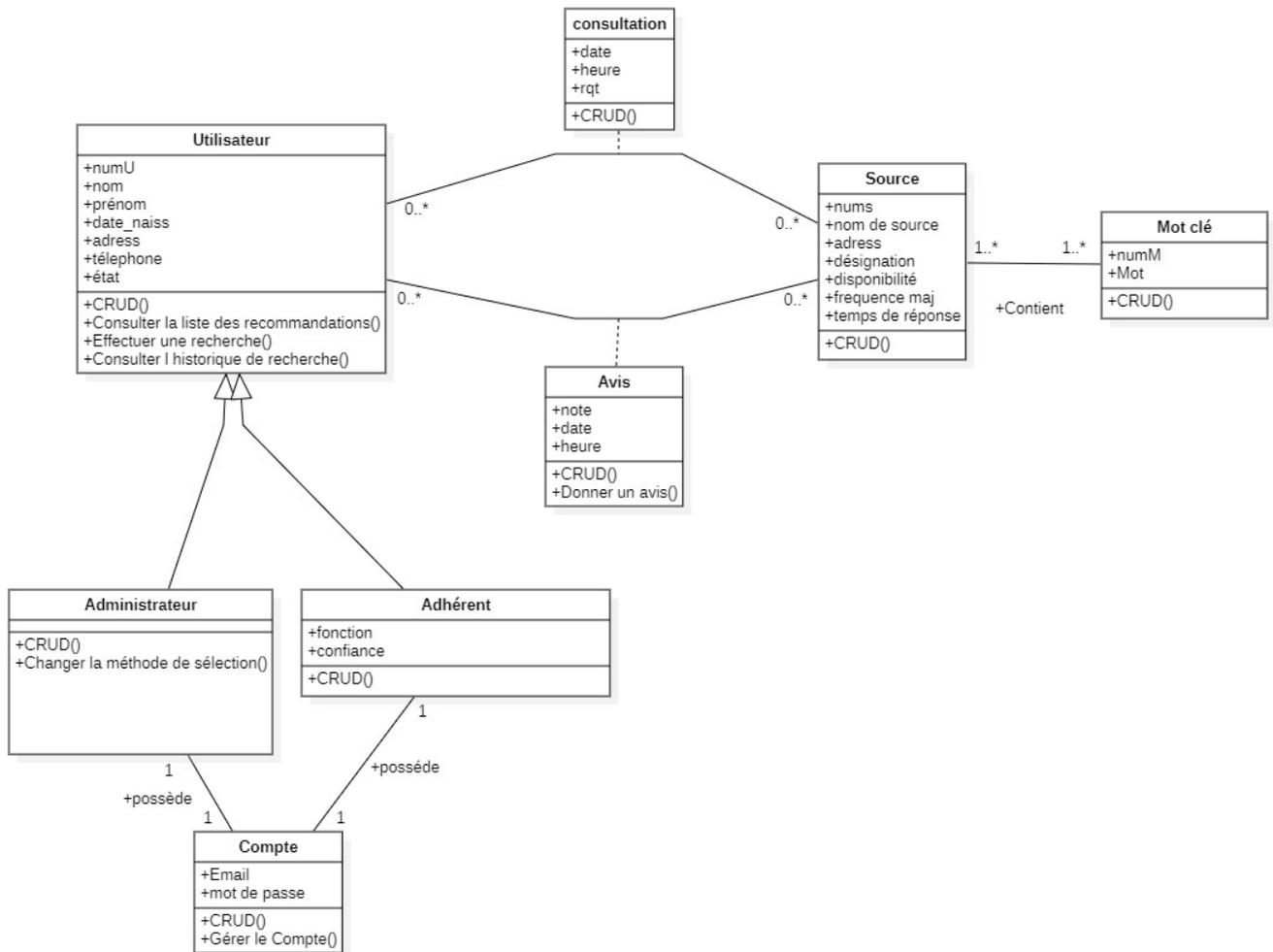


Figure 27: Diagramme de classe.

Tableau 33: Tableau descriptif des classes, attribut et méthode.

Classe	Attribut	Type	Désignation	Méthode
Utilisateur	NumU	Integer	Identifiant d'utilisateur	- CRUD() - Consulter l'historique () - Effectuer recherche () - Consulter recommandation ()
	Nom	String	Nom de l'utilisateur	
	Prenom	String	Prenom de l'utilisateur	
	Date_naiss	Date	Date de naissance	
	Adress	String	Adress de l'utilisateur	
	Téléphone	Integer	Téléphone de l'utilisateur	
	Etat	Integer	Accepté ou en attente	
Adherent	Fonction	String	Profession d'adhérent	- CRUD()
	Confiance	Float	La valeur de confiance de chaque adhérent	
Administrateur				- CRUD() - Changer méthode

Conception et Implémentation

				de selection()
Source	numS	Integer	Identifiant de la source	CRUD()
	Nom de source	String	Nom de la source	
	Adress	String	Adress ip de la source	
	Designation	String	Description de source	
	Disponibilité	Float	Valeur de disponibilité de la source	
	Fréquence maj	Integer	Valeur de fréquence MAJ de la source	
	Temps de Réponse	Integer	Valeur de temps de réponse de la source	
Compte	Email	String	Identifiant de compte	CRUD()
	Mot de passe	String	Mot de passé de compte	
Consultation	Date	Date	Date de consultation	CRUD()
	Heure	Time	L'heure de consultation	
	Rqt	String	Les mot de requête	
Avis	Note	Integer	Note	- CRUD() - Donner avis ()
	Date	Date	Date de notation	
	Heure	Time	L'heure de notation	
Mots cles	numM	integer	Identifiant de mot clé	CRUD()
	Mot	String	un mot	

Conception et Implémentation

V.3. Outils de développement :

La figure 28 montre les principaux outils utilisés pour mettre en œuvre notre système SRQSSI (J2EE, NetBeans, Tomcat, Bootstrap, SQL server).

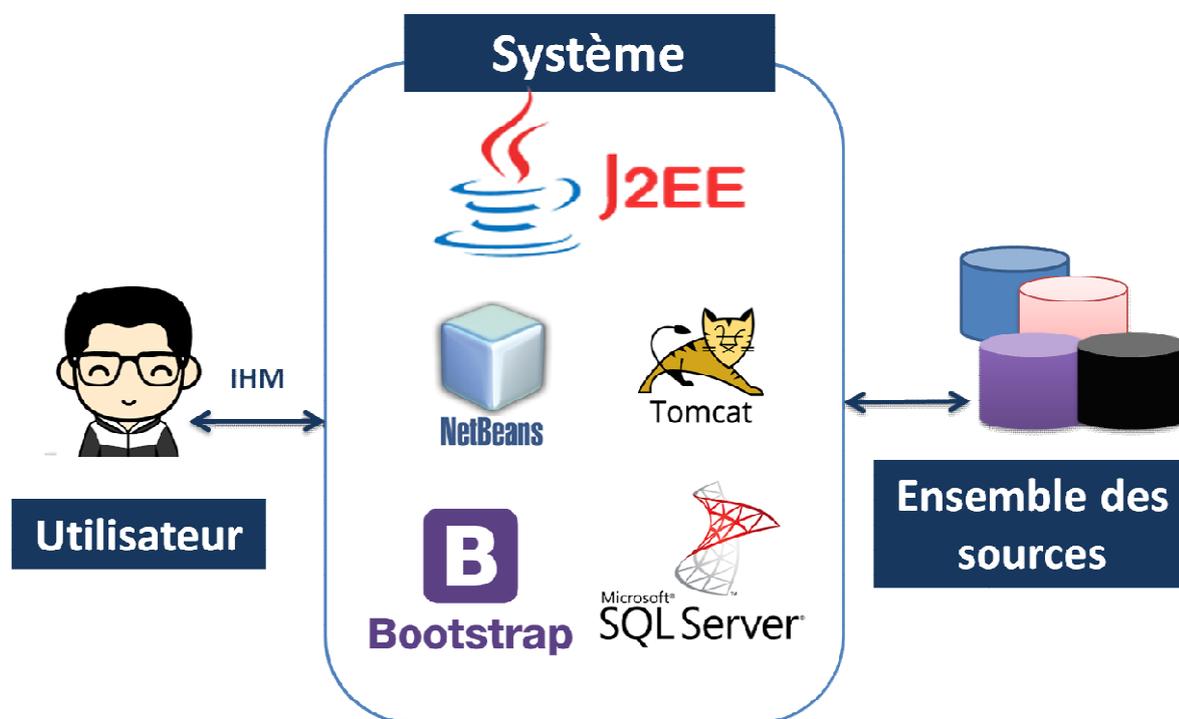


Figure 28 : Outils de développement de SRQSSI

L'interface **Java EE** (Entreprise Edition) permet de créer des sites Web dynamiques avec une technologie Java. La mise en place d'un environnement Java EE nécessite l'utilisation d'un serveur d'applications capable d'exécuter le code et de répondre aux requêtes des clients. **Apache-Tomcat** est notre choix comme serveur d'applications Java du projet Jakarta de la fondation Apache. Ce serveur est libre, sous licence Apache permet d'exécuter des applications Web développées avec les technologies Java (Servlets, JSP...).

Il est également nécessaire d'utiliser un environnement de développement (EDI) évolué Java. Nous avons choisi d'utiliser **Netbeans** qui permet aux développeurs de créer toute une gamme d'applications de bureau, Web et mobiles avec Java 8. L'EDI prend également en charge un large éventail de technologies – HTML, CSS, JavaScript, PHP et C / C ++ – en plus à Java. Les éditeurs, les analystes et les convertisseurs fournis par NetBeans facilitent encore la tâche des programmeurs pour mettre à niveau leurs applications héritées vers la dernière version de Java.

Pour accélérer grandement notre projet web, nous avons utilisé le **Bootstrap** qui est un Framework frontend (HTML5, CSS et JavaScript) spécialement conçu pour le développement d'application web "responsive", c'est-à-dire qui s'adapte automatiquement à différents dispositifs et tailles d'écran (tablettes, Smartphones, desktop...etc.). Il fournit des outils avec des styles déjà en place pour des typographies, des boutons, des interfaces de navigation et bien d'autres encore.. Pour gerer notre base de données nous avons utilisé SQL server développé par Microsoft, un SGBD relationnel originellement multibase et multishéma.

Conception et Implémentation

V.4. Présentation de l'application :

L'application est composée de trois espaces :

- Espace d'accueil destiné à tous les utilisateurs de notre application à savoir le visiteur, l'adhérent et l'administrateur. Il permet d'effectuer une recherche, de s'inscrire ou de se connecter. .
- Espace d'administrateur permet de gérer les adhérents et les sources.
- Espace d'adhérent permet de consulter l'historique de recherche et les recommandations.

V.4.1. Espace d'accueil

En plus de la barre de recherche permettant à un utilisateur d'introduire sa requête, cette page contient aussi (Figure 29):

- Un lien vers la page de connexion pour un administrateur ou un utilisateur déjà inscrits.
- Un lien vers la page d'inscription pour un nouvel utilisateur. l'utilisateur doit remplir un formulaire contenant toutes les informations nécessaires (nom, prénom, date de naissance, adresse, email, numéro de téléphone, mot de passe et la fonction).

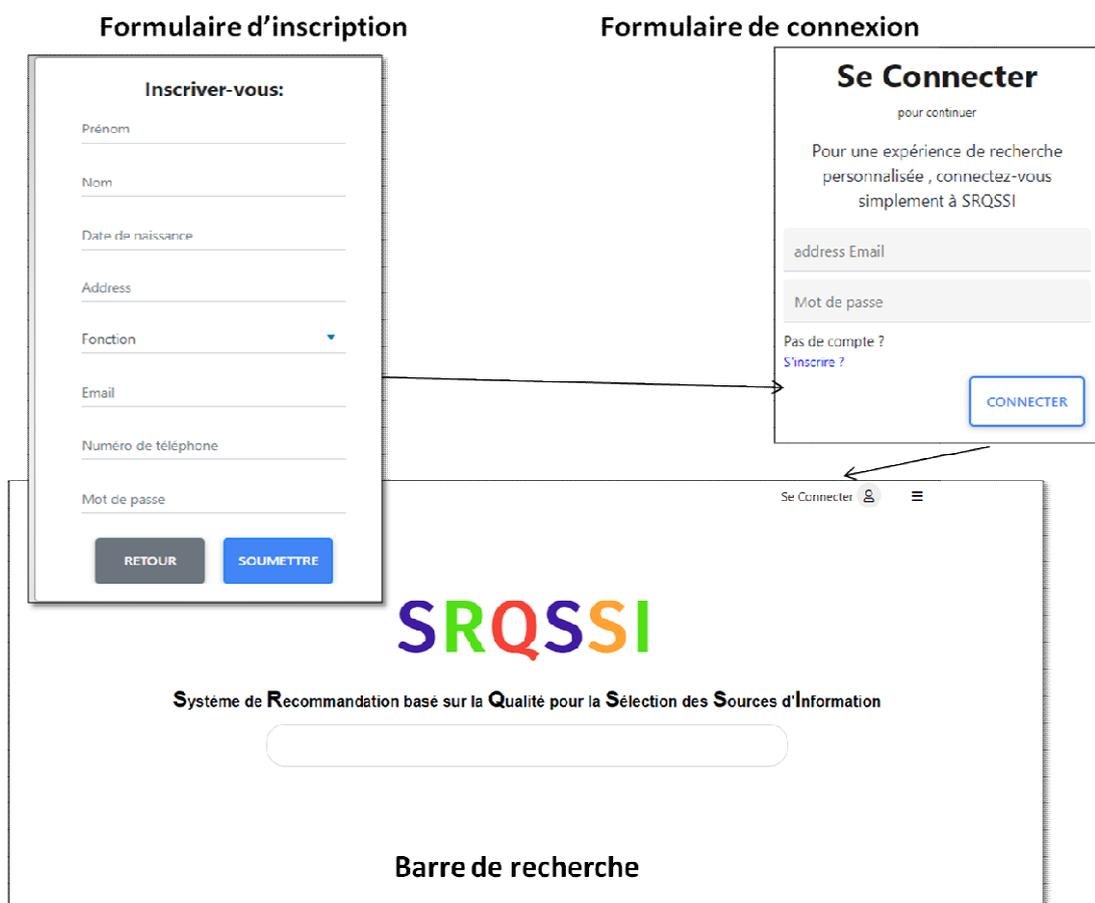


Figure 29: Page d'accueil.

Conception et Implémentation

V.4.2. Espace administrateur :

L'administrateur est le responsable de la gestion des sources et des adhérents.

a. Gestion des sources

L'administrateur peut consulter la liste des sources coopérantes dans le système (figure 30), et il peut effectuer également les tâches suivantes :

- La suppression d'une source.
- La modification d'une source, il peut changer les informations de la source comme la désignation, les mots clés qui réfèrent au domaine de la source.
- Consulter la description de la source sous format XML, cette description contient les informations essentielles sur la source.

Description XML de la source

```
<source>
  <nom>S2</nom>
  <mot-clé>java</mot-clé>
  <adresse>http://localhost:8001/</adresse>
  <désignation>La désignation de la source S2</désignation>
  <temps-réponse>0.125s</temps-réponse>
  <disponibilité>20/40</disponibilité>
  <fréquence-mise-à-jour>10 par mois</fréquence-mise-à-jour>
</source>
```

Les tâches de l'administrateur

The interface shows a sidebar with 'Fonctions' (Gestion Des Sources, Gestion Des Adhérents, Mon Compte) and a main area with 'Liste Sources'. The table below is a representation of the data shown in the screenshot.

#	Nom source	Adress source	Disponibilité	Temps de réponse	Fréquence maj	Description	Modifier	Supprimer
1	S1	http://localhost:8000/	21/40	0.15 s	11 par mois	👁	✎	🗑
2	S2	http://localhost:8001/	20/40	0.125 s	10 par mois	👁	✎	🗑
3	S3	http://localhost:8002/	30/40	0.1 s	10 par mois	👁	✎	🗑
4	S4	http://localhost:8003/	40/40	0.05 s	8 par mois	👁	✎	🗑
5	S5	http://localhost:8004/	10/40	0.025 s	4 par mois	👁	✎	🗑

Modifier Source

S1

http://localhost:8000/

Désignation source
La désignation de la source S1

java

Modifier

Êtes-vous sûr?

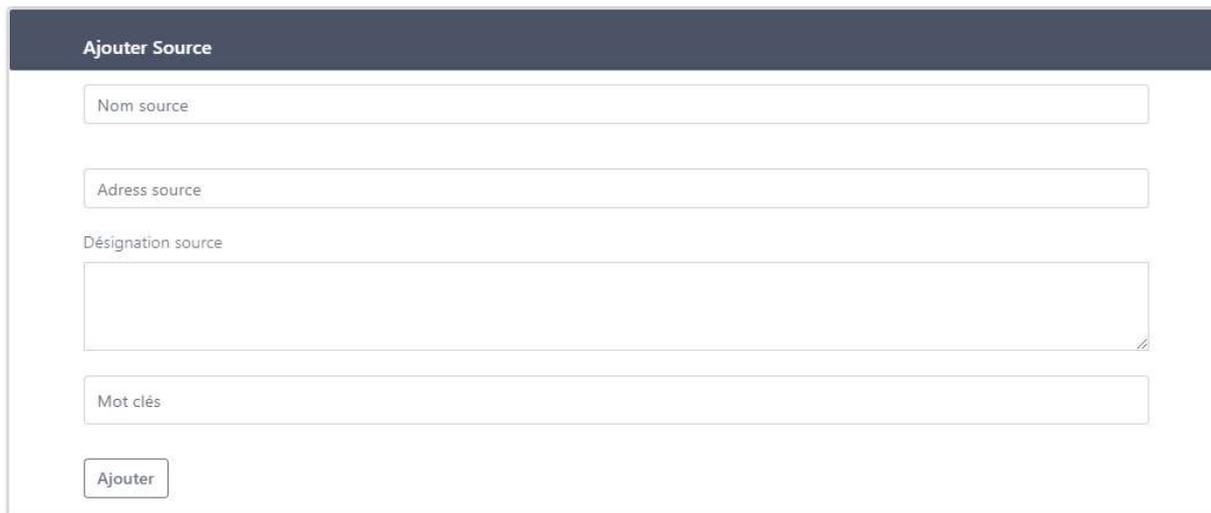
Voulez-vous vraiment supprimer cette source?

Annuler **Supprimer**

Figure 30: Interface de gestion des sources.

Conception et Implémentation

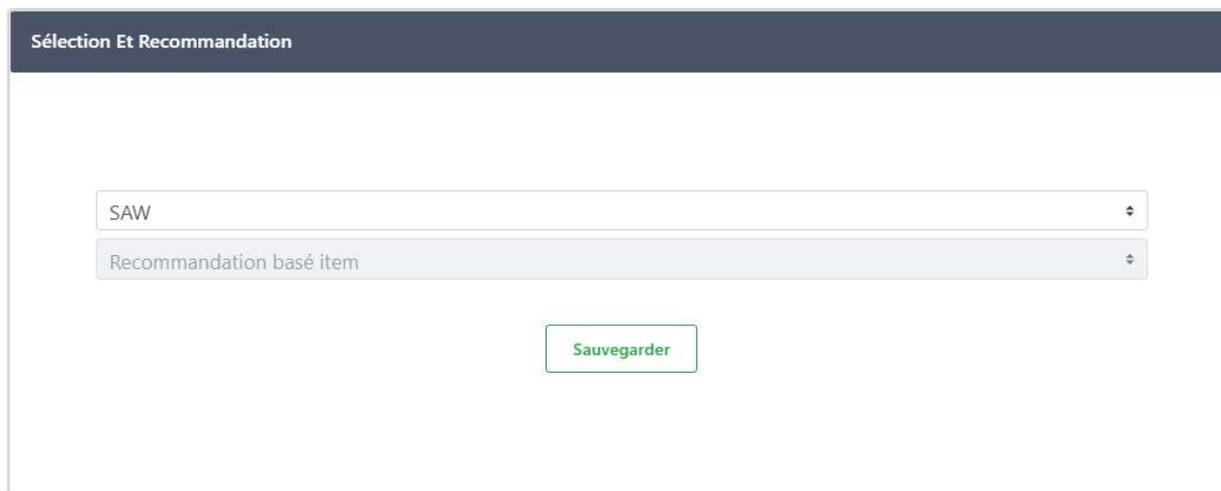
La liste des sources contient toutes les informations sur la source: son nom, son adresse, les valeurs des attributs de QoS, etc. L'administrateur peut aussi ajouter une nouvelle source au système en remplissant un simple formulaire (figure 31). Les mots clés désignent le domaine auquel appartient chaque source.



The screenshot shows a web form titled "Ajouter Source". It contains four input fields: "Nom source", "Adress source", "Désignation source", and "Mot clés". Below these fields is a button labeled "Ajouter".

Figure 31: Ajouter une nouvelle source.

Une autre fonctionnalité essentielle de notre système est le choix de la méthode à appliquer pour la sélection des sources. L'administrateur a la possibilité de changer l'algorithme de sélection comme indiqué dans la figure 32.



The screenshot shows a web form titled "Sélection Et Recommandation". It features two dropdown menus. The first dropdown menu is currently set to "SAW". The second dropdown menu is currently set to "Recommandation basé item". Below these dropdowns is a green button labeled "Sauvegarder".

Figure 32: Choisir une méthode de sélection de sources.

Conception et Implémentation

b. Gestion des adhérents :

Pour avoir un compte, l'utilisateur effectue une demande d'inscription et attends la confirmation de l'administrateur. Ce dernier a la possibilité d'accepter ou de refuser une demande (figure 33) en fonction des caractéristiques de chaque utilisateur.

L'acceptation ou le refus de chaque nouvelle inscription se fait à l'aide d'une base de données EXCEL contenant la mesure de confiance donnée pour chaque adhérent en vérifiant leur fonction.

Il est possible également de supprimer un utilisateur ayant déjà un compte. La suppression se fait lorsqu'un des adhérents change de fonction, il sera considéré comme un nouvel utilisateur.

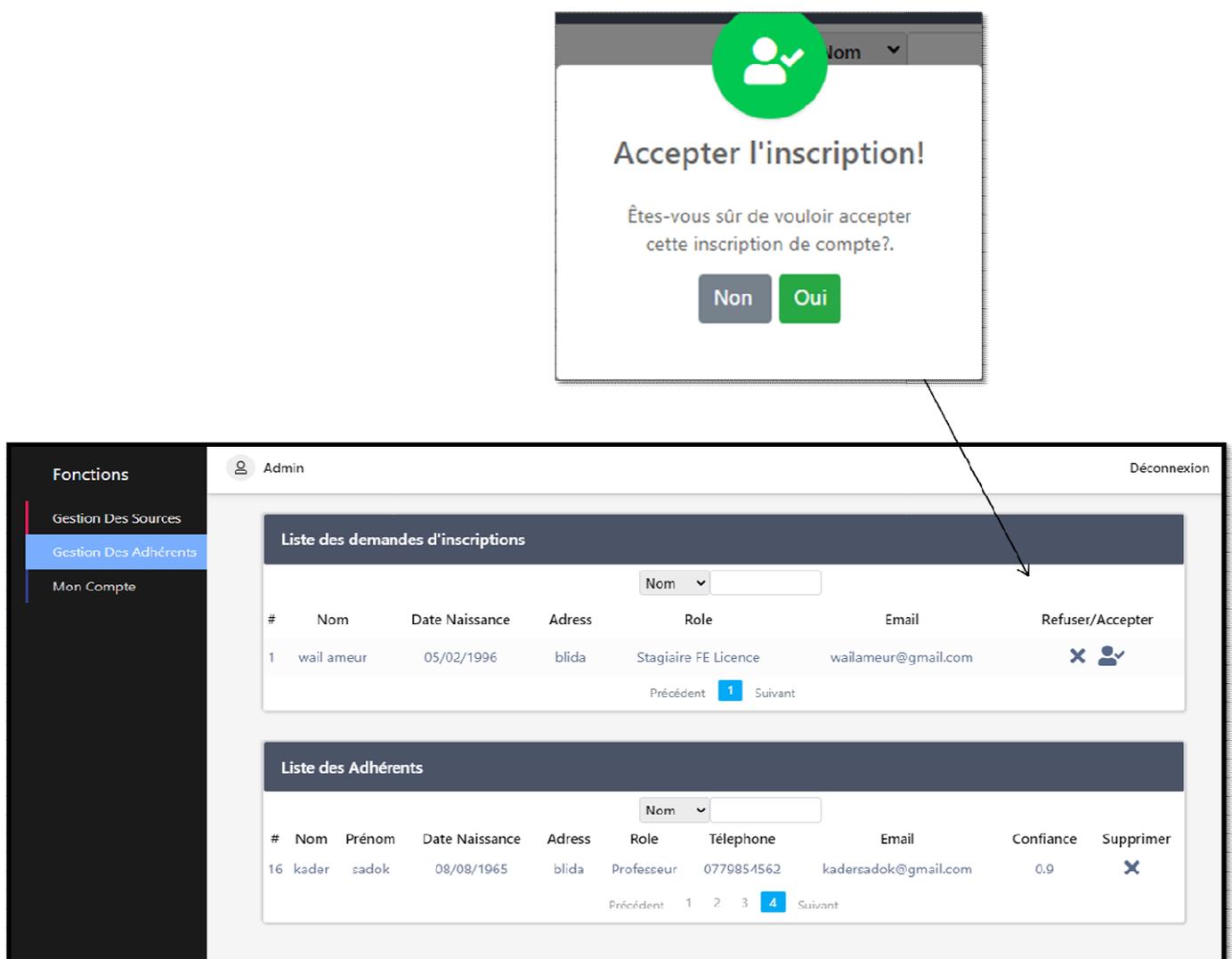


Figure 33: Interface de gestion des adhérents.

Conception et Implémentation

V.4.3. Espace utilisateur :

L'utilisateur ayant un compte peut mettre à jour ses informations personnelles et/ou changer son mot de passe.

Il peut également consulter l'historique de sa recherche, les sources utilisées et les avis des utilisateurs sur ces sources (figure 34).

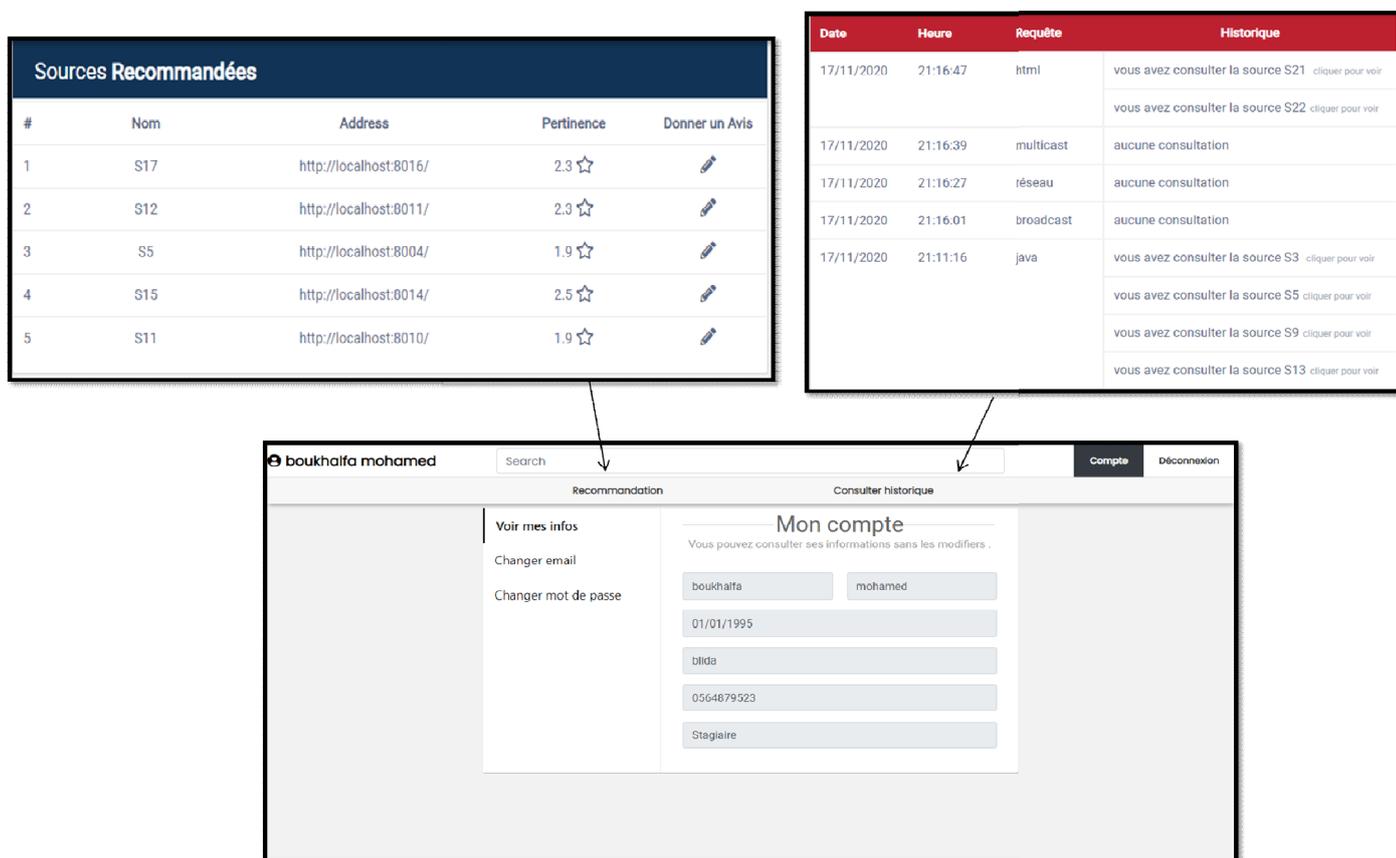


Figure 34 : Compte d'un utilisateur.

V.4.4. La recommandation

Par ailleurs, un utilisateur inscrit au système peut recevoir les recommandations concernant les sources d'information. Ces recommandations sont soit générales ou personnalisées.

- **Les recommandations générales** concernent les sources proches aux intérêts de l'utilisateur. elles sont générées avant que l'utilisateur introduit sa requête (figure 35).
- **Les recommandations personnalisées** concernent les sources similaires à une source que l'utilisateur vient de choisir après une recherche (figure 36).

L'utilisateur peut donner son avis sur une source en suivant l'échelle de rating (de 1 à 5 étoiles) en lui affichant une boîte de dialogue (Figure 36). Il peut également consulter les statistiques des évaluations (des pourcentages) données par tous les utilisateurs à une source donnée (figure 36). En choisissant une source donnée à consulter, l'utilisateur reçoit des recommandations

Conception et Implémentation

personnalisées concernant la source choisie, par exemple dans la figure 36 les sources similaires à la source S5 sont recommandées à l'utilisateur (les sources S7, S18 ...).

Un utilisateur non inscrit dans le système peut seulement voir les résultats de ses recherches. Il ne peut pas donner son avis ou avoir des recommandations.

Sources Recommandées				
#	Nom	Address	Pertinence	Donner un Avis
1	S17	http://localhost:8016/	2.3 ☆	
2	S12	http://localhost:8011/	2.3 ☆	
3	S5	http://localhost:8004/	1.9 ☆	
4	S15	http://localhost:8014/	2.5 ☆	
5	S11	http://localhost:8010/	1.9 ☆	

Figure 35 : recommandations générales.

Consultation de la source

#	Nom	Address	Pertinence	Donner un avis
1	S5	http://localhost:8004/	2.1 ☆	
2	S4	http://localhost:8003/	2.4 ☆	
3	S30	http://localhost:8029/	2.0 ☆	
4	S3	http://localhost:8002/	2.6 ☆	
5	S7	http://localhost:8006/	2.5 ☆	
6	S6	http://localhost:8005/	2.4 ☆	
7	S9	http://localhost:8008/	2.6 ☆	
8	S1	http://localhost:8000/	2.3 ☆	

Page 1 sur 3 [Precedent](#) [1](#) [2](#) [3](#) [Suivant](#)

Recommandation personnalisée a une source

SOURCES SIMILAIRES

Nom: S7
Adress: http://localhost:8006/
☆☆☆☆☆
[DONNER UN AVIS](#)

Nom: S10
Adress: http://localhost:8017/
☆☆☆☆☆
[DONNER UN AVIS](#)

[Sauvegarder](#)

Evaluation de la source

Donner une note : ☆☆☆☆☆

statistic de notation

2.3

☆☆☆☆☆
10 total

1 ☆ 10%

2 ☆ 30%

3 ☆ 20%

4 ☆ 20%

5 ☆ 20%

Figure 36 : résultats d'une recherche.

Conception et Implémentation

V.4.5. Test des méthodes de sélection

Pratiquement, après avoir implémenté les trois méthodes d'agrégation SAW, AHP et TOPSIS, nous avons trouvé que la méthode AHP est la plus compliquée par rapport aux deux autres méthodes SAW et TOPSIS qui se basent sur des formules mathématiques simples.

Nous avons aussi mesuré temps d'exécution de chaque méthode pour l'exemple déjà cité dans (section IV.4.2) où $n = 3$ et $m = 10$, nous avons obtenus les résultats présentés dans le tableau 33. Le temps d'exécution correspond bien à la complexité temporelle calculée théoriquement.

Tableau 34 : Temps d'exécution fait par l'application des trois méthodes sur l'exemple

Méthode	Temps d'exécution (ms)	Complexité temporelle
SAW	2007ms	$O(n*m)$
AHP	2325ms	$O(2^n)$
TOPSIS	2019ms	$O(n^2)$

IV.5. Conclusion

Dans ce chapitre, nous avons décrit le processus de réalisation de notre système en respectant la conception élaborée, en spécifiant les outils de développement et le déroulement du processus de sélection et de recommandation des sources d'information. Nous avons ensuite présenté les différentes interfaces de notre application à travers lesquelles l'utilisateur peut lancer une recherche et recevoir les recommandations sur les sources de qualité élevée.

CONCLUSION ET PERSPECTIVES

*Le projet est le brouillon de l'avenir.
Parfois, il faut à l'avenir des centaines de
brouillons. - - Jules Renard*

Conclusion et Perspectives

L'objectif de notre projet était de concevoir et d'implémenter un Système de Recommandation basé la Qualité pour la Sélection de Sources d'Information (SRQSSI). Pour ce faire, nous avons présenté dans premier temps la recherche d'information en général et la recherche d'information distribuée en particulier en présentant ses majeurs défis. Ensuite, nous avons présenté les notions clés de la qualité qui est définie en fonction de plusieurs paramètres. Nous avons cité aussi les principales méthodes de décision multicritères permettant de mesurer ou d'évaluer cette qualité. Dans le troisième chapitre, nous avons décrit et comparé les approches de recommandation existantes pour choisir entre elles.

A partir de cette étude bibliographique, nous nous sommes ressorties avec une contribution qui porte sur la sélection et la recommandation des sources d'information et cela en se basant sur l'ensemble de critères de qualité suivants : la disponibilité, le temps de réponse, la fréquence de mise à jour. En fonction de ces critères, la qualité était mesurée par trois méthodes : SAW, AHP, TOPSIS qui servent à résoudre notre problème de sélection multicritère. Ainsi, les meilleures sources choisies sont celles ayant les meilleurs scores de qualité. Notre solution prend aussi en considération les appréciations des utilisateurs sur les sources utilisées pour générer des recommandations plus précises en utilisant le filtrage collaboratif social. La prédiction de la valeur de qualité d'une source est basée sur la similarité des sources en considérant les évaluations (notes) données par les utilisateurs lors de leurs recherches antérieures. Comme présenté dans le dernier chapitre, nous avons mis en œuvre notre système (SRQSSI) en utilisant un ensemble des outils comme J2EE, Bootstrap et d'autres langages web comme CSS, JavaScript.

Notre application permet de rassembler un ensemble de sources de données et de sélectionner un sous ensemble pour répondre à une requête de l'utilisateur. Les résultats fournis par les sources choisies sont fusionnés et renvoyés à l'utilisateur. Ce dernier après avoir eu les résultats peut noter les sources utilisées, lui permettant d'exprimer son opinion sur la qualité de la source. Le système et sur la base des évaluations de tous les utilisateurs génère les recommandations les plus pertinentes pour un utilisateur actif. La recommandation des sources considère également la confiance des utilisateurs, les données des évaluations sont filtrées pour considérer que les évaluations données par des utilisateurs de confiance. Pour simplifier, nous considérons dans ce travail la fonction d'un utilisateur (développeur, chercheur...) pour estimer son degré de confiance. D'autres méthodes peuvent être envisageables.

Toutefois plusieurs améliorations qui peuvent y être apportées à notre système, notamment:

- **Automatisation du calcul des poids** : Dans notre application, les poids sont affectés par l'administrateur par défaut, il serait judicieux d'automatiser le calcul de poids en prenant en considération l'avis des utilisateurs et en se basant sur leurs préférences.
- **Automatisation de la création des descriptions de sources** : Au lieu que l'administrateur crée les descriptions manuellement. Cette tâche peut être automatique. les valeurs de QoS des sources peuvent être estimées d'une manière

automatique à travers des statistiques faites par le système sur les recherches passées. Par exemple, dans le domaine du réseau, on peut mesurer facilement le temps de réponse défini comme la différence entre l'instant d'envoi de la requête et l'instant de réception de la réponse.

- **Considération d'autres critères de qualité** : nous avons considéré que trois paramètres de QoS (temps de réponse, la disponibilité, fréquence de mise à jour), néanmoins notre solution peut être étendue en intégrant d'autres paramètres. Nous pensons à ajouter la réputation qui est considérée un facteur de QoS important pour améliorer la sélection et la recommandation des sources.
- **Considération d'autres sources d'information**: nous avons créé des sources fictives pour faire les tests nécessaires et pour montrer les résultats obtenus. mais il y a toujours une possibilité de travailler avec d'autres sources réelles et distantes.
- **Comparer l'approche proposée avec d'autres approches de recommandation de l'état de l'art** : il serait intéressant de comparer l'approche proposée avec les approches de recommandation basées sur le filtrage collaboratif de l'état de l'art pour montrer la performance de notre solution.
- **Amélioration de la fusion des résultats** : notre approche se base essentiellement sur la sélection des sources. la fusion de résultats est également une étape importante permettant d'assurer la satisfaction de l'utilisateur, l'affichage des top-meilleurs résultats qui correspond aux préférences et les intérêts de l'utilisateur améliore nettement la recherche d'information.
- **Automatisation de l'acceptation et le refus des nouvelles inscriptions** : la vérification des nouveaux adhérents va être faite automatiquement et non pas manuellement.
- **Avoir la possibilité de modifier la fonction de l'adhérent** : quand il passe vers un notre niveau de formation, sa fonction sera modifié sans création d'un nouveau compte.

Bibliographie

- [1] **N. Hernandez**, « Ontologie de Domaine pour la Modélisation du Contexte en Recherche d'Information », **These**, Université Paul Sabatier, 2006.
- [2] **F. Boubekeur**, « Contribution a la Définition de Modèles de Recherche d'Information Flexibles basés sur les CP-Nets », **These**, Université Paul, 2008.
- [3] **M. Daoud**, « Accès Personnalisé à l'information : Approche Basée sur l'utilisation d'un profil Utilisateur Sémantique Dérivé d'une Ontologie de Domaines à Travers l'historique des Sessions de Recherche » , **These**, Université Paul Sabatier, 2009.
- [4] **F. Abbaci**, « Methodes de Selection de Collections dans un Environnement de Recherche d'Information Itribuée » , **These**, Ecole Nationale Supérieure des Mines, Saint-etienne, 2003.
- [5] **A. Sriti**, « La Recherche d'Informations Semantique d'Informations dans le Cadre du Web Semantique » , **These**, Université Mohamed Khider , Biskra, 2012.
- [6] **W. Zemirli**, « Vers le Developpement d'un Systeme de Recherche d'Information Personnalisé Integrant le Profil Utilisateur », **These** Université Paul Sabatier de Toulouse III, 2004.
- [7] **A. Hanane** et **C. Fadia**, « Realisation d'un Systeme de Recherche d'Information », **Mémoire** , Université Abou Bekr Belkaid, Tlemcen, 2015.
- [8] **C. Yaël**, « Un Modele de Recherche d'Information Basé sur les Graphes et les Similarités Structurelles pour l'Amelioration du Processus de Recherche d'Information », **These**, Université Paul Sabatier de Toulouse 3, 2009.
- [9] **R. Baeza Yates** et **R. N. Berthier**, “Modern Information Retrieval”, **Livre**, the ACM Press, New York 1999.
- [10] **F. Wiesman**, **A. Hasman** et **H. Van den Herik**, “ Information Retrieval : an Overview of System Characteristics ”, *International Journal of Medical Informatics*”, n°147, pp. 5-26, 1997.
- [11] **G. Salton**, “The SMART Retrieval System—Experiments in Automatic Document Processing ”, **Livre**, Prentice-Hall, NJ, USA ,Inc, 1971.
- [12] **K. Jones**, **K. Walker** et **S. Robertson**, “Aprobabilistic Model of Information Retrieval : Development and Comparative experiments ” , *Information Processing & Management*, vol. 6, n° 136, pp. 809-840, 2000.

- [13] **M. Maron** et **J. Kuhns**, “ On Relevance, Probabilistic Indexing and Information Retrieval ”, *Journal of the ACM*, vol. 3, n°17, pp. 216-244, 1960.
- [14] **J. Callan**, “Distributed Information Retrieval ”, *Advances in Information Retrieval*, Kluwer Academic Publishers, pp. 127-150 ,2000.
- [15] **M. K. Bergman**, “The Deep Web : Surfacing Hidden Value ”, septembre 2001.[Enligne].Available:
<https://quod.lib.umich.edu/j/jep/3336451.0007.104?view=text;rgn=main>.
- [16] **C. Sherman** et **G. Price**, “The Invisible Web : Uncovering Information Sources Search, Information Today”, **Livre**, Inc./CyberAge Books, 2001.
- [17] **J. Madhavan**, **D. Ko**, **V. Ganapathy**, **A. Rasmussen**, **A. Halevy** et **L. Kot**, “Google’s Deep-Web Crawl”, *Proceedings of the VLDB Endowment* , pp 1241-1252, 2008.
- [18] **J. Callan** et **c. Margaret**, “Query Based Sampling of Text Database” *ACM Transactions on Information Systems* , pp. 97-130, 2001.
- [19] **Luo.Si** et **J. Callan**, “Relevant Document Distribution Estimation Method for Resource Selection ”, *SIGIR03: The 26th ACM/SIGIR International Symposium on Information Retrieval*, New York, 2003.
- [20] **J. P. Callan**, **Z. Lu** et **W. Croft**, “Searching Distributed Collections with Inference Networks”, *18th annual international ACM SIGIR conference on Research and developement in information retrieval* , New York , 1995.
- [21] **N. Eric.Craswell**, “Methodes for Distributed Information Retrieval ”, **These**, The Australian National University, 2000.
- [22] **J. Callan** et **L. Si**, “A Semisupervised Learning Method to Merge Search Engine Results”, *ACM Transactions on Information Systems*, pp. 457-491, 2003.
- [23] **G. Paltoglou**, **M. .Satzrize** et **M. Salampanis**, “Hybrid Results Merging ”, *CIKM '07: Proceedings of the sixteenth ACM conference on Conference on information and knowledge management*, New York, 2007.
- [24] **L.Gravano**, **C.K. Chang**, **H.Gracia-Morina** et **A. Paepacke**, “ STARTS : Stanford Proposal for Internet Meta-searching ”, *Proceeding of the 1997 ACM SIGMOD conference on Management Data*, New York, 1997.
- [25] **C. Dwork**, **R. Kumar**, **M. Naor** et **D. Sivakuma**, “Rank Aggregation Methods for the Web”, *Proceedings of the 10th international conference on World Wide Web*, New York, 2001.

- [26] **E. Voorhees, N. Gupta et B. Johnson-Laird**, “Learning Collection Fusion Strategies”, *Proceeding of the 18 Annual international ACM SIGIR conference on Research and Development in Information retrieval*, Seattle, 1995.
- [27] **G. K. Zipf**, “Humain Behavior and the Principle of Least Effort : an Introduction to Humain Ecology ”, **Livre**, Martino Fine Books ,2012
- [28] **L.Gravano et H.Gracia-Morina**, “Generalizing GLOSS to Vector-space Databases and Broker Hierarchies ”, *Proceeding of the 21st International Conference on Very Large Databases (VLDB)*,1995
- [29] **M. Shokouhi et L. Si**, “Federated Search, Journal Foundations and Trends in Information Retrieval” ,*Now Publishers inc*, Hanover USA, Vol. 5: No. 1, pp 1-102 2011.
- [30] **J. Zobel**, “Collection Selection Via Lexicon Inspection”, *Proceeding Of The Australian Document Computing Symposium*, Melbourne Australie , 1997.
- [31] **J. Xu et J. Callan**, “Effective Retrieval of Distributed Collections”, chez *Proceeding of the 21st annual international ACM SIGIR conference on research and development in information retrieval*, Melbourne, 1998.
- [32] **B. Yuwono et D. Lee**, “Server Ranking for Distributed Text Retrieval Systems on the Internet”, *Proceeding of the fifth international conference on database systems for advanced applications*, Melbourne ,Australia, 1997.
- [33] **D. D'Souza, J. Thom et J. Zobel**, “A Comparison of Techniques for Selection Text Collections”,*Proceedings 11th Australasian Database Conference*, Canberra Australia, 2000.
- [34] **D. D'Souza, J. Thom et J. Zobel**, "Collection Selection of Managed Distributed Document Databases", *Proceeding of the australian database conference*, ISSN, Melbourne, Australia, pp 527–546 , 2004.
- [35] **C. Baumgarten**, "A Probabilistic Model for Distributed Information Retrieval " , *ACM SIGIR*, vol. 31, pp. 258-266, july 1997.
- [36] **L. Gravano, H. Garcia-Molina et T. Anthony**, "Gloss Text-source Discovery Over the Internet," *ACM transactions on information systems* , vol. 3, n° 124, pp. 229-264, 1999.
- [37] **S. Deerwester, S. Dumais, T. Landauer, G. Furnas et R. Harshman**, «Indexing by Latesnt Semantic Analysis,» *The american society for information sciences* , vol. 6, n°141, pp. 391-4.7, 1990.

- [38] **L. Si et J. Callan**, “Using Sampled Data and Regression to Merge Search Engine Results”, *Proceedings of the 25th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, ISBN, 2002, pp. 19-26.
- [39] **M. Sogrine, T. Kechadi et N. Kushmerick**, “Latent Semantic Indexing for Text Database Selection”, *Proceeding of the SIGIR Workshop on Hetrogeneous and distributed information retrieval* , pp. 12-19, 2005.
- [40] **G. Salton et M. McGill**, Introduction to Modern Information Retrieval, **Livre**, M. Hill, Éd., New York , 1986.
- [41] **L. Si et J. Callan**, “Relevant Document Distribution Estimation Method for Resource Selection“, *SIGIR '03: Proceedings of the 26th annual international ACM SIGIR conference on Research and development in information retrieval*, Canada, pp. 298-305, 2003.
- [42] **L. Si et J. Callan**, “Unified Utility Maximization Framework for Resource Selection”, *Proceeding of the 13th ACM international conference on information and knowledge management* , pp 32-41, 2004.
- [43] **L. Si et J. Callan**, “Modeling Search Engine Effectiveness for Federated Search”, *Proceedings of the 28th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, Salvador , Brazil, pp 83-90, 2005.
- [44] **M. Shokuhi et J. Zobel**, “Federated Text Retrieval from Uncooperative Overlapped Collections”, *Proceedings of the 30th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*,, Amsterdam ,pp. 495-502 2007.
- [45] **P. Thomas et M. Shokouhi**, “SUSHI: Scoring Scaled Samples for Server Selection”, *Conference: Proceedings of the 32nd Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, Boston ,MA,USA, pp. 419-426, 2009.
- [46] **N. Fuhr**, «Optimum Database Selection in Networked IR,» chez *Proceeding of the SIGIR 96 workshop on networked information retrieval* , Zurich , Switzerland , 1996.
- [47] **N. Fuhr**, «A decision-theoretic Approach to Database Selection in Networked IR,» *ACM Transactions on information systems* , vol. 3, n° 117, pp. 229-249, 1999.
- [48] **N. Fuhr**, «Resource Discovery in Distributed Digital Libraries,» *Proceeding of digital libraries advanced methods and technologies*, pp. 35-45, 1999.

- [49] **F. Abbaci, S. J et M. Beigbeder**, “A methodologie for collection selection in Heterogeneous Contexts”, *IEEE international conference on information technology : coding and computing* ,pp.529-535, 2002.
- [50] **N. Fuhr**, “A Decision-theoretic Approach to Database Selection in Networked ir”, *ACM Transactions on Information Systems*, pp. 229-249, 1999.
- [51] **L.Kwok, L.Grunfeld et D. Lewis**, “TREC-3 ad-hoc Routing Retrieval and Thresholding Experiments Using PIRCS,” *The third text retrieval conference (TREC-3)*, pp. 247-255, 1995.
- [52] **P. J. Callan, Z.Luet W.Bruce Croft**, “Searching Distributed Collections with Inference Networks ”, *Proceeding ACM SIGIR Internation conference on research and developement in information retrieval*, pp. 21-28, 1995
- [53] **C. Viles et J. French**, “Dessemination of Collection Wide Infomration in a Distributed Information Retrieval System”, *Proceedings of the 18th annual international ACM SIGIR conferece on Research and development in information retrieval*, Seattle, pp.12-20, 1995.
- [54] **Z.Lu; J.Callan ,W.Croft**, “Applying Inference Networks to Multiple Collection Searching”, **Memoire** ,Université de Massachusetts 1996.
- [55] **S. Dumais**, “Latent Semantic Indexing (LSI) and TREC2”, chez *The second text retrieval conference(TREC2)*,pp. 105-115, 1994.
- [56] **S. Kirsch**, “Document Retrieval Over Networks where in Ranking and Relevance Scores are Computed at the Client for Multiple Database Documents”, **Brevet** , US, 1997.
- [57] **C. Jamie, Distributed Information Retrieval**, *Advances in Information Retrieval* , edit **W.Bruce Croft**, New York , pp. 127-150 2002.
- [58] «ISO,»2015.[Enligne].Available: <https://www.iso.org/fr/standard/29280.html>. [Accès le juin 2020].
- [59] **Y. Snir, Y. Ramberg, J. Strassner, R. Cohen et B. Moore**, “Policy Quality of Service (QOS) Information Model,” RFC Editor , USA, 2003.
- [60] **F. Rafamantanantsoa, P. Laurencot et A. Aussem**, “Analyse des Performances et Modelistion d'un Serveur Web”, *5th International Conference: Sciences of Electronic,Technologies of Information and Telecommunications* , Tunisie , 2009.
- [61] «Le Grand Dictionnaire Terminologique (GDT),» [En ligne]. Available: <http://gdt.oqlf.gouv.qc.ca/>. [Accès le 20 juin 2020].

- [62] **S. Ben Mena**, Introduction aux méthodes multicritère d'aide à l'aide à la décision , *journal of Biology Biotechnologie Agronomie Société et Environnement* ,pp.83-93, 2000.
- [63] **F.Younsi , D.Hamdadou** , Vers un Système d'Aide à la Décision en Aménagement du Territoire : SIG, Méthodes Multicritères et Réseaux de Neurones, International *Conference on Computer Science and its Applications* , université d'Oran ES-Senia, ORAN, 2009
- [64] **S. Ben Mena**, «Introduction aux Methodes Multicriteres d'aide a la Decision,» *Biotechnologie , Agronomie , Société et Environnement* , vol. 4, pp. 83-93, 2000.
- [65] **L. Maystre, J. Pictet et J. Simos**, “Methodes Multicriteres ELECTRE”, **Livre** ,Presse Polytechniques et Universitaire , 1994.
- [66] **A. Le Gall**, “Panorama des Methodes d'Analyse Multicritere comme Outils d'aide a la Decision, **RAPPORT D'ÉTUDE**” , ONEMA-INERIS France , 2009.
- [67] **A. Nafi et C. Werey**, “Aide a la Decision Multicritere: Introduction aux methodes d'Analyse Multicritere de Type ELECTRE, ENGEES Ingenierie Financière”, **Mémoire**, Strasbourg, 2010 .
- [68] **B. Roy et D. Bouyssou**, “Aide a la Decision : Methodes et Cas”, **Livre**, Paris: Economica, 1993.
- [69] **A.Memariani, A.Amini, A.Alinezhad** , “Sensitivity Analysis of Simple Additive Weighting Method SAW : The Results of Change in the Weight of One Attribute on the Final Ranking of Alternatives”, *Journal of Industrial Engineering* 4, vol. 13, n°118, 2009.
- [70] **R. Harrathi**, “ Facteurs de Qualité et Personnalisation de l'Information”, **Mémoire**, Lyon, 2005.
- [71] **T. L. Saaty**, “The Analytic Hierarchy Process McGraw-Hill”, *journal: Agricultural Economics Review*, Vol 70,New York , 1980.
- [72] **T. L. Saaty**, “Fundamentals of Decision Making and Priority Theory with Analytic Hierarchy Process”, **Livre**, RWS Publications , 2000.
- [73] **T. L. Saaty**, “A Scaling Method for Priorities in Hierarchical Structures”, *Journal of Mathematical Psychology* , vol. 15, n° 13, pp. 234-281, 1977.
- [74] **C. L. Hwang et K. Yoon**, “Multiple Attribute Decision Making: Methods and Applications”, **Livre**, New York: Springer-Verlag., 1981.

- [75] **F. Nauman, D.Linden** , “Data Fusion and Data Quality ”, chez *Seminar on New Techniques et Technologies for Statistic* , Berlin ,2001.
- [76] **M. Burgess**, ”Using Multiple Quality Criteria to Focus Information Search Results”, **These** , UK, 2003.
- [77] **J. Bennett** et **S. Lanning**, «The Netflix Prize,» chez *Proceedings of KDD Cup and Workshop*, 2007.
- [78] **R. Burke**, “Hybrid Recommender Systems: a Survey and Experiments” , *user modeling and user-adapted interaction*, vol 12 n° 112, USA, pp. 331-370, 2002.
- [79] **F. Ricci, L. Rokach** et **B. Shapira**, “Introduction to Recommender Systems Handbook”, *Recommender Systems Handbook*, F. Ricci, L. Rokach, B. Shapira et P. B. Kantor, Éd.s., US, Spring , 2011, pp. 1-35.
- [80] **O. Douglas** et **K. Jinmook**, “Modeling Information Content Using Observable Behavior,» *Proceedings of the annual meeting of the american society for information science* , vol. 38, 2001.
- [81] «Welcome to NCBI,»[Enligne].Available : <https://www.ncbi.nlm.nih.gov/> . [Accès le septembre 2020].
- [82] **S.-A. Fayrouz**, “ CAMLearn: Une Architecture de Système de Recommandation Sémantique Sensible au Contexte. Application au Domaine du M-Learning”, **These** , Université de Bourgogne ,2015.
- [83] **Mathieu**, Les Algorithmes de Recommandation,» [En ligne]. Available: <https://www.podcastscience.fm/dossiers/2012/04/25/les-algorithmes-de-recommandation/>. [Accès le avril 2020].
- [84] **G. Adomavicius** et **A. Tuzhilin**, “ Toward the Next Generation of Recommender Systems: A Survey of the State-of-the-art and Possible Extensions”, *Knowledge and data engineering IEEE Transactions on* , vol. 17, pp. 734-749, 2005.
- [85] **I. Esslimani**, “ Vers une Approche Comportementale de Recommandation : apport de l'analyse des Usages dans un Processus de Personnalisation Interface Homme-machine ” , **These**, Université Nancy 2, 2010.
- [86] **E. Zitzler, M. Laumanns** et **T. Lothar**, “ SPEA2: Improving the Strength Pareto Evolutionary Algorithm”, *TIK Report journal*, vol. 103, Zurich , 2001.

- [87] **D. Goldberg, D. Nichols, B. M. Oki et D. Terry**, “ Using Collaborative Filtering to Weave an Information Topstry ”, *Communication of the ACM*, vol. 35, n° 112, pp. 61-70, 1992.
- [88] **J. S. Breese, D. Heckerman et C. Kadie**, “ Empirical Analysis of Predictive Algorithms for Collaborative Filtering ”, *Proceedings of the Fourteenth Conference on Uncertainty in Artificial Intelligence*, San Francisco, CA, USA, Morgan Kaufmann Publishers Inc, pp. 43-52, 1998.
- [89] **C. Desrosiers et K. George**, “ A Comprehensive Survey of Neighborhood-based Recommendation Methods ”, chez *Recommender systems handbook*, Boston , Spring , 2011, pp. 107-144.
- [90] **D. Jannach, A. Zanker, A. Felfernig et G. Friedrich**, “ Recommender Systems An Introduction ”, *Livre*, Cambridge University Press , 2010.
- [91] **L. Ungar et D. Foster**, “ Clustering Methods for Collaborative Filtering” , chez *Workshop on recommender systems at the 15th national conference on artificial intelligence* , 1998.
- [92] **G. Shani et A. Gunawardana**, “Evaluating Recommendation Systems”, *Recommender Systems Handbook*, vol. 12, 2011, pp. 257-297.
- [93] **J. L. Herlocker, J. A. Konstan et J. Riedl**, “ Explaining Collaborative Filtering Recommendation”, chez *Proceedings of the 2000 ACM conference on computer supportef cooperative work* , New York , USA, 2000.
- [94] **M. Erdt nee Anjorin, A. Fernandez et C. Rensing**, “Evaluating Recommender Systems for Technology Enhanced Learning : a Quantitative Survey ”, *IEEE transactions on learning Technologies* , vol. 8, pp. 1-1, 2015.
- [95] **S.M.McNee, J.Riedl ,J.A.Konstan** , “Being Accurate is not Enough: How Accuracy Metrics have Hurt Recommender Systems “, chez *Extended Abstracts Proceedings of the 2006 Conference on Human Factors in Computing Systems, CHI 2006*, Montreal , Québec, Canada, 2006.
- [96] **B.P.Knijnenburg, M.C.Willemsen, Z.Gantner, H.Soncu et C.Newell** , “Explaining the User Experience of Recommender Systems”, *User Modeling and User-Adapted Interaction* , vol. 22, pp. 441-504, 2012.
- [97] **S. Azeli, H. Drachsler, M. Bitter-Rijkema, F. Brouns, W. Vegt et P. Sloep**, “User-centric Evaluation of Recommender Systems in Social Learning Platforms: Accuracy is Just the Tip of the Iceberg”, *IEEE Transactions on Learning Technologies*, vol. PP, pp. 1-1, 2017.

- [98] **M. Chatti, S. Dakova, H. Thus et U. Schroeder**, “Tag-Based Collaborative Filtering Recommendation in Personal Learning Environments”, *Learning Technologies ,IEEE Transactions on Learning Technologies*, vol. 6, pp. 337-349, 2013.
- [99] **N. Manouselis, H. Drachsler, K. Verbert et e. duval**, “Recommender Systems for Learning”, *Livre*, Springer-verlag, New York, 2012.
- [100] **J. Herlocker, J. Konstan, L. Terveen, J. C. Lui et T. Riedl**, “Evaluating Collaborative Filtering Recommender Systems”, *ACM Transactions on Information Systems* , vol. 22, pp. 5-53, 2004.

