

الجمهورية الجزائرية الديمقراطية الشعبية
République Algérienne Démocratique et Populaire
وزارة التعليم العالي والبحث العلمي
Ministère de L'enseignement Supérieur et de la Recherche Scientifique
جامعة عين تموشنت بلحاج بوشعيب
Universitaire BELHADJ BOUCHAIB - Ain-Témouchent
Faculté des Sciences et de la Technologie
Département de Mathématiques et Informatique



Projet de Fin d'Etudes

Pour l'obtention du diplôme de Master en Informatique

Domaine : Mathématiques et informatique

Filière : Informatique

Spécialité : Réseaux et Ingénierie des Données

Thème

**Système de recommandation à base de connaissances :
Résolution de problèmes à base de modèle ontologique.**

Travail réalisé par :

M^{elle} AHMED BELKACEM Samia

M^{me} GHERRAS Nassima

Devant le jury composé de :

- *D^r SAIDI Mohammed Reda* UAT.B.B (Ain Temouchent) *Président.*
- *D^r BOUCHAKOUR ERRAHMANI Hichem* UAT.B.B (Ain Temouchent) *Examinateur.*
- *D^r BOUHALOUAN Djamila* UAT.B.B (Ain Temouchent) *Encadrante.*

Année Universitaire 2021-2022



Remerciements

Cout d'abord, nous tenons à remercier Allah, le tout puissant de nous avoir donné la santé, la volonté et la patience pour réaliser ce modeste travail.

Nous adressons nos sincères remerciements, notre profonde gratitude à notre Directrice de mémoire, Madame BOUHALOUEN Djamila, que Dieu la protège et prenne soin d'elle et lui donne une longue vie. Nous la remercions de nous avoir encadré durant toute la durée de ce projet avec beaucoup de tact et de gentillesse. Elle nous a orienté, aidé avec une écoute active, ses précieux conseils ainsi que ses directives nous ont été bénéfiques. Sa disponibilité et ses encouragements ont fait que le travail présent soit de cette qualité.

Nos remerciements s'adressent également à tous les membres de jury Monsieur SAIDI et Monsieur BOUENKOUR , qui ont accepté de nous honorer de leur présence et de juger notre travail.

Nous tenons à remercier tous ceux qui ont participé de près ou de loin à l'élaboration de ce mémoire.



The page is decorated with pink roses and gold floral patterns. The roses are in the top right and bottom left corners. The gold patterns, including leaves and small flowers, are scattered throughout the background.

Dédicaces

Je dédie ce travail

À mes chers parents pour tous leurs efforts depuis que j'ai vu le jour jusqu'à aujourd'hui.

À mes sœurs et mes frères qui m'ont encouragé.

Particulièrement, je dédie ce fruit de ma recherche à mon encadreur Madame BOUHLLOUN Djamila de m'avoir aidé, soutenu et guidé avec ses conseils et tous ses efforts importants

À mon binôme Nassima, qui m'a accompagné tout au long de la réalisation de ce travail avec patience.

À tous mes amis et ma famille

Lamia

A decorative background featuring pink roses and gold leaf patterns. The roses are in various stages of bloom, with some showing detailed petal textures. The gold leaf patterns include stylized leaves and small circular motifs, scattered across the page.

Dédicaces

Je dédie ce travail

À mes très chers parents et mes très chères sœurs qui m'ont guidé durant les moments les plus pénibles de ce long chemin, merci pour le soutien moral, psychologique et matériel.

À mon mari qui m'a soutenu toutes ces années, et qui m'a aidé à franchir tous les obstacles que j'ai rencontrés. Bien évidemment, à mes chers enfants.

Spécial dédicace à vous : Madame BOURBOURN Djamila pour tous vos efforts.

À mon binôme Samia qui a été avant tous une très bonne amie, une coéquipière travailleuse, patiente et sérieuse.

À tous mes amis et ma famille

Nassima

Résumé

Les systèmes de recommandation sont une forme spécifique de filtrage de l'information (SI) visant à présenter les éléments d'information (films, musique, livres, news, images, pages Web, etc..) qui sont susceptibles d'intéresser l'utilisateur. Généralement, un système de recommandation permet de comparer le profil d'un utilisateur à certaines caractéristiques de référence, et cherche à prédire l'« avis » que donnerait un utilisateur. Des défis majeurs contraignent les systèmes de recommandation, tels que, le problème de diversité des systèmes de recommandation, la stabilité par rapport au profil dynamique des utilisateurs, ainsi qu'un problème générique de ces systèmes qui est le démarrage à froid.

Des approches classiques de recommandation comme le filtrage collaboratif, le filtrage à base de contenu, le filtrage hybride...etc. ont montré leurs efficacité dans ce contexte. L'ingénierie des connaissances, et plus particulièrement les systèmes à base de connaissances ont joué un rôle important dans l'amélioration des performances des tels systèmes. Tel est le cadre de notre travail qui propose des pistes d'amélioration des SdR. Une première contribution est axée sur les connaissances où nous avons construit un modèle ontologique permettant le raisonnement à base de cas afin d'extraire des cas similaires à un problème donné, une deuxième contribution est axée sur l'évaluation des recommandations par des métriques de similarité afin de valider le raisonnement.

Mots-clés : Système de recommandation, Filtrage, Ingénierie des connaissances, Bases de connaissances, Raisonnement à base de connaissance, Ontologie.

Abstract

Recommender systems are a specific form of information filtering (IS) aimed at presenting the pieces of information (movies, music, books, news, images, Web pages, etc.) that are likely to be of interest to the user. Generally, a recommendation system makes it possible to compare the profile of a user with certain reference characteristics, and seeks to predict the "opinion" that a user would give. Major challenges constrain recommender systems, such as the problem of diversity of recommender systems, stability with respect to the dynamic profile of users, as well as a generic problem of these systems which is cold start.

Classic recommendation approaches such as collaborative filtering, content-based filtering, hybrid filtering, etc. have shown their effectiveness in this context. Knowledge engineering, and more specifically knowledge-based systems have played an important role in improving the performance of such systems. This is the framework of our work which suggests ways to improve the RS. A first contribution is focused on knowledge where we built an ontological model allowing case-based reasoning in order to extract cases similar to a given problem, a second contribution is focused on the evaluation of recommendations by similarity metrics in order to validate the reasoning.

Keywords : Recommender system, Filtering, Knowledge engineering, Knowledge bases, Knowledge-based reasoning, Ontology.

الملخص

أنظمة التوصية هي شكل محدد من أشكال تصفية المعلومات التي تهدف إلى تقديم أجزاء من المعلومات (أفلام ، موسيقى ، كتب ، أخبار ، صور ، صفحات ويب ، إلخ) التي يحتمل أن تكون ذات فائدة للمستخدم. بشكل عام ، يتيح نظام التوصية إمكانية مقارنة ملف تعريف المستخدم بخصائص مرجعية معينة ، ويسعى إلى التنبؤ بالرأي الذي قد يقدمه المستخدم. التحديات الرئيسية تقيد أنظمة التوصية ، مثل مشكلة تنوع أنظمة التوصية ، والاستقرار فيما يتعلق بالملف الشخصي الديناميكي للمستخدمين ، بالإضافة إلى مشكلة عامة لهذه الأنظمة وهي البداية الباردة.

مناهج التوصية الكلاسيكية مثل التصفية التعاونية والتصفية القائمة على المحتوى والتصفية المختلطة وما إلى ذلك. أظهرت فعاليتها في هذا السياق. لعبت هندسة المعرفة ، وبشكل أكثر تحديداً الأنظمة القائمة على المعرفة دوراً مهماً في تحسين أداء هذه الأنظمة. هذا هو إطار عملنا الذي يقترح طرقاً لتحسين أنظمة التوصية . تركز المساهمة الأولى على المعرفة حيث قمنا ببناء نموذج وجودي يسمح بالتفكير القائم على الحالة من أجل استخراج حالات مشابهة لمشكلة معينة ، بينما تركز المساهمة الثانية على تقييم التوصيات بمقاييس التشابه من أجل التحقق من صحة التفكير.

الكلمات المفتاحية : نظام التوصية ، التصفية ، هندسة المعرفة ، قواعد المعرفة ، التفكير القائم على المعرفة ، علم الوجود.

Table des matières

Introduction générale	2
I Concepts de base des systèmes de recommandation	7
I.1 Introduction	8
I.2 Les systèmes de recommandation	8
I.2.1 Quelques définitions	8
I.2.2 Historique des systèmes de recommandation	9
I.2.3 Recherche d'information versus Filtrage d'information	10
I.3 Concepts de base	11
I.3.1 Utilisateur et Item	12
I.3.2 Evaluation	12
I.3.3 Communauté	12
I.3.4 Prédiction	12
I.3.5 Recommandation	12
I.3.6 Notion du profil	12
I.3.7 Notion de connaissance	13
I.4 Classification des systèmes de recommandation	13
I.5 Techniques de recommandation	14
I.5.1 Filtrage collaboratif	14
I.5.1.1 Filtrage basé modèle	15
I.5.1.2 Filtrage basé mémoire (basé voisinage)	15
I.5.2 Filtrage basé sur le contenu	16
I.5.3 Filtrage démographique	17
I.5.4 Filtrage basé utilité	17
I.5.5 Filtrage basé connaissance	17

I.5.6	Filtrage hybride	18
I.6	Avantage et limites des systèmes de recommandation	19
I.7	Conclusion	21
II	Système à base de connaissance (basé ontologie)	23
II.1	Introduction	25
II.2	Apports de l'ingénierie des connaissances aux Systèmes de Recommandation	25
II.3	Systèmes à base de connaissances	26
II.3.1	Définition	26
II.3.2	Rôle des systèmes à base de connaissances	27
II.4	Raisonnement à base de cas	27
II.4.1	Définition	27
II.4.2	Cycle RàPC	28
II.5	Outils intégrant le RàPC	29
II.6	Systèmes de recommandation à base d'ontologie	30
II.6.1	Apport des ontologies	30
II.6.1.1	Ontologie : Définitions et origine	30
II.6.2	Objectifs de développement des ontologies	31
II.7	Composantes d'ontologies	32
II.7.1	Concept (classe)	32
II.7.2	Relation	33
II.7.3	Fonctions	33
II.7.4	Axiomes	33
II.7.5	Instances	33
II.8	Classification des ontologies	33
II.8.1	Typologie selon le niveau de conceptualisation	34
II.8.2	Typologie selon le niveau de formalisation utilisée	35
II.8.3	Typologie selon le niveau de détail de l'ontologie	35
II.8.4	Typologie selon le niveau de complétude	35
II.8.5	Typologie selon le niveau de complexité de l'ontologie	36
II.9	Cycle de vie d'une ontologie	36
II.10	Processus de construction d'une ontologie	37
II.10.1	Spécification (évaluation des besoins)	38
II.10.2	Conceptualisation	38
II.10.3	Ontologisation	38

II.10.4	Opérationnalisation	38
II.11	Méthodes et méthodologies de construction d'ontologies	39
II.11.1	Démarches de construction d'une ontologie	39
II.11.2	Méthodologies de construction	39
II.12	Langages d'ontologie	40
II.12.1	Langages traditionnels	40
II.12.2	Langage web standards ou basé sur xml	40
II.12.3	Langages d'interrogation des ontologies	42
II.12.3.1	Protocole de requête SPARQL	42
II.12.4	Approches du raisonnement	43
II.13	Outils de développement d'ontologies	44
II.13.1	PROTÉGÉ	44
II.13.2	ODE (ONTOLOGY DESIGN ENVIRONMENT)	44
II.13.3	OntoEdit	44
II.13.4	WebOde	45
II.13.5	DoE	45
II.13.6	Framwork jena	45
II.14	Conclusion	45
III	État de l'art des systèmes de recommandation	48
III.1	Introduction	48
III.2	Les travaux connexes aux systèmes de recommandation	48
III.3	Comparaison	55
III.4	Synthèse	57
III.5	Conclusion	58
IV	Système à base de connaissances pour la recommandation de résolution de problèmes en maintenance informatique	60
IV.1	Introduction	61
IV.2	Présentation de l'application	61
IV.3	Proposition d'une approche basée Ontologie du SdR	62
IV.3.1	Architecture de l'approche	62
IV.3.2	Déroulement de l'approche	63
IV.3.3	Description du modèle ontologique	65
IV.3.4	Protocole d'évaluation des recommandations	66

IV.4	Aspects de modélisation	68
	68
IV.4.1	Les cas d'utilisation	68
IV.4.1.1	Diagramme de cas d'utilisation de l'utilisateur	69
IV.4.1.2	Diagramme de cas d'utilisation de l'expert	70
IV.4.1.3	Diagramme de cas d'utilisation de l'Administrateur	71
IV.4.2	Les diagrammes de séquences	72
IV.4.2.1	Diagramme de séquence d'Authentification	72
IV.4.2.2	Diagramme de séquence d'identification et d'envoi du problème	73
IV.4.2.3	Diagramme de séquence de réception et résolution du problème .	74
IV.4.3	Diagramme de classe	74
IV.5	Mise en œuvre de l'approche	75
IV.5.1	Environnement et outils de programmation	75
IV.5.1.1	IDE Netbeans	76
IV.5.1.2	Java	76
IV.5.1.3	API Jena	76
IV.5.1.4	Protégé	77
IV.5.1.5	WampServer	77
IV.5.1.6	L'API JFreeChart	77
IV.6	Réalisation de l'application(scénarios d'exécution)	77
IV.6.1	Raisonnement à base de cas	77
IV.6.2	Scénarios de la démarche décrite dans la section (IV.3.2)	78
IV.7	Conclusion	92

Conclusion générale et perspectives **94**

Table des figures

I.1	Classification des systèmes de recommandation.	14
II.1	Représentation et manipulation de connaissances dans un système à base d connaissances.	27
II.2	Cycle du RàPC [Rac,04].	28
II.3	Organisation d'une ontologie.	32
II.4	Classification des ontologies.	34
II.5	Cycle de vie des ontologies.	37
II.6	Processus de construction ontologies.	38
II.7	Langage d'exploitation des ontologies[Gom,04].	42
IV.1	Architecture de notre système.	62
IV.2	Système de recommandation basé d'Ontologie.	63
IV.3	Différents sous-modèles ontologiques.	65
IV.4	Diagramme de cas d'utilisation de l'utilisateur.	69
IV.5	Diagramme de cas d'utilisation de l'expert.	70
IV.6	Diagramme de cas d'utilisation de l'Administrateur.	71
IV.7	Diagramme de séquence d'authentification.	72
IV.8	Diagramme de séquence d'identification et d'envoi du problème.	73
IV.9	Diagramme de réception et résolution du problème.	74
IV.10	Diagramme de classe.	75
IV.11	Vue partielle de l'ontologie de tâche.	78
IV.12	Interface principale de l'Admin.	78
IV.13	Interface d'authentification de l'utilisateur.	79
IV.14	Inscription d'un nouvel utilisateur.	79
IV.15	Choix de la catégorie de l'appareil.	80

IV.16 Consultation de l'historique des recommandations.	80
IV.17 Identification du problème.	81
IV.18 Requête envoyée aux experts.	81
IV.19 L'interface de notification de l'espace Expert.	82
IV.20 Chargement de la BC de l'expert et aperçu de son fichier owl.	83
IV.21 Extraction de solutions de cas sources similaires.	83
IV.22 Cas sources récupérés du 1 ^{er} Expert.	84
IV.23 Cas sources récupérés du 2 ^{me} Expert.	84
IV.24 Espace principale de l'admin.	87
IV.25 Évaluation des recommandations.	88
IV.26 Représentation des cas sources trouvés pour chaque expert.	90
IV.27 Précision moyenne des métriques.	91
IV.28 Écart entre les mesures de similarité.	92

Liste des tableaux

I.1	Comparaison entre la recherche et le filtrage d'information.	11
I.2	Avantages et inconvénients des systèmes de recommandation traditionnels.	20
III.1	Comparaison entre les systèmes de recommandation	57
IV.1	Cas sources récupérés.	86
IV.2	Similarité [cas source,cas cible].	90

Liste des Abréviations

SdR	Systeme de recommandation.
FC	Filtrage collaboratif.
RàPC	Raisonnement à base / partir de cas.

Introduction générale

Introduction générale

Contexte du travail

De nos jours, Nous sommes de plus en plus exposés à une multitude d'informations. Avec tous les blogues, les journaux, les magasins et bien d'autres, cela apporte beaucoup de diversités aux utilisateurs. Toutefois avec cette multitude d'information, La surcharge d'information peut devenir problématique. Afin d'y remédier à ce problème, avec l'essor du web ont été développés divers outils et techniques informatiques pour faciliter cette recherche ainsi que pour filtrer les informations avant de les transmettre aux utilisateurs. On note parmi celles-ci, les systèmes de recommandation (Recommender System ou RS) permettant un tel filtrage, Le but principal de ces systèmes est de faciliter la prise de décisions pour les utilisateurs en leur offrant des informations selon leurs préférences. Il existe une panoplie de tels systèmes pour divers produits tels que des films (Netflix), de la musique (last.fm) et des livres (Amazon).

Les systèmes de recommandation ont été étudiés dans de nombreux domaines tel que la recherche d'informations, le web, le e-commerce, l'exploitation des usages du web et bien d'autres. Ils sont considérés comme étant un phénomène prévisible pour contribuer à résoudre les problèmes de surcharge informationnelle, valorisant les contenus et focalisant l'attention de l'usage dans un contexte d'abondance.

En explorant la littérature, de nombreuses approches ont été adoptées pour générer des recommandations, les plus reconnues parmi lesquelles, sont le filtrage collaboratif et le filtrage basé sur le contenu [Ado et Tuz,05]. L'approche basée sur le contenu est mise en œuvre en tenant compte des caractéristiques des produits recommandés et en utilisant des métriques pour créer des groupes de produits en utilisant une mesure de similitude sur leur contenu. Et dans un contexte de filtrage collaboratif, on utilise généralement un voisinage d'utilisateurs similaires et on recommande des produits en fonction de l'historique d'autres utilisateurs au sein de même voisinage.

Problématique

Nos travaux s'orientent vers les systèmes de recommandation qui ont montré leur efficacité afin d'apporter une assistance et un guide aux utilisateurs selon leurs potentiels intérêts. Nous visons principalement les systèmes de recommandation à base de connaissances notamment dans des situations de dépannage et maintenance informatique. Particulièrement, il ne faut pas négliger les problèmes qui y sont rattachés, dont les plus fondamentaux, i.e. ; dans un autre sens, génériques et touchent la plupart des systèmes de recommandation dans les premiers stades de fonctionnement, nous rappelons :

- Le démarrage à froid, ces problèmes se produisent généralement lorsqu'un nouvel utilisateur ou une nouvelle ressource est ajouté à la base de recommandation.
- Un autre problème touche la stabilité de ces systèmes lié au profil dynamique de l'utilisateur, (Stability vs. Dynamicity Problem), lorsqu'un utilisateur s'intéresse à plusieurs items différents en même temps, comme il peut alterner ses préférences au fil du temps, notamment si son profil est créé dans le système, il est difficile de modifier ses préférences et de tenir compte de ses différents choix, puis de lui recommander des items qui ne correspondent pas à ses attentes et intérêts, ce qui conduit à un manque de diversité dans les propositions recommandées.
- La sur-spécialisation, dans ce cas, le système recommande à l'utilisateur que les propositions similaires à sa requête, autrement dit, on ne peut pas lui recommander celles qui sont en relation avec la requête émise en terme de recommandation plus spécialisée (spécialisation) / plus généralisée (généralisation), cela devra être résolu en prenant en charge une certaine hiérarchie des items et intérêts.

Nous abordons dans la suite les atouts de notre approche développée.

Contribution et proposition

Nous avons proposé une approche, présentant quelques atouts de force, son principal, étant l'absence du manque d'informations sur les utilisateurs ou sur les items (le problème de démarrage à froid) du fait qu'elle n'est pas sensible à ce manque.

Au fil de ce projet, notre travail doit répondre à quelques objectifs clés qui peuvent se résumer comme suit :

- Réalisation d'un type spécifique de systèmes de recommandation qui est à base de connaissance (knowledge-based recommender systems), il se repose sur une connaissance explicite de l'assortiment du problème à résoudre, des exigences de l'utilisateur et des critères de recommandation (c'est-à-dire quelle solution de dépannage de matériel informatique doit être recommandée dans quel cas). Notre choix de ce type de système est motivé par la flexibilité offerte, vu que nous nous sommes trouvés contraints par des scénarios où les autres approches alternatives telles que le filtrage collaboratif et le filtrage basé sur le contenu ne peuvent pas être appliquées pour proposer des recommandations de résolution de problèmes liés aux pannes/dysfonctionnements du matériel informatique et/ou électronique satisfaisant ainsi l'utilisateur.
- Notre approche combine quelques dimensions telles qu'une dimension contextuelle, sémantique et a caractère collectif, chacune intervient dans une phase d'évolution du système.
 - La première dimension intervient pour résoudre le problème de démarrage à froid, nous récupérons des informations relatives au contexte abordé à cause des ontologies développées (domaine) et d'autres concernant l'utilisateur, ces dernières sont disponibles dès son authentification.
 - Ensuite contribue la dimension sémantique pour recommander à l'utilisateur des résolutions sémantiquement proches du problème posé en termes d'identification et descripteurs de celui-ci., Ces mesures de similarités sont exprimées aussi à travers l'utilisation des ontologies.
 - La dimension collective concerne l'aspect collaboratif qui consiste à proposer des recommandations issues de différents experts du domaine/internaute qui seront postées à l'utilisateur par la suite.
- La construction de ressources ontologiques destinées aux utilisateurs dans ce domaine.
- Utilisation d'une ontologie d'application de la tâche de résolution et notamment, dans la reformulation des requêtes des utilisateurs à travers des concepts plus représentatifs du problème

et particulièrement où les utilisateurs décrivent leurs situations problématiques qui seront interprétées par la suite par une requête à usage ontologique sans pour autant que ces derniers sachent comment les formuler.

Description du contenu de mémoire

Ce manuscrit est composé de quatre chapitres répondant aux problématiques que nous avons identifiées en ce qui concerne les liens et frontières possibles entre l'ingénierie de connaissances et les systèmes de recommandation.

- Au niveau du **chapitre 1** nous allons présenter, dans un premier lieu, les concepts généraux relatifs à la notion des systèmes de recommandation, leur historique, leurs différentes classifications et nous énumérons quelques atouts et limites de certaines classes de systèmes.
- Le **chapitre 2** a été consacré pour la présentation des principes des systèmes à base de connaissances intégrant les ontologies, nous exposons la définition de ce elles ci, ses composantes, ses différentes classifications, ses niveaux, ainsi, nous abordons un concept qui y est étroitement lié; le principe du raisonnement à base de cas qui représente le moteur de recherche ou le cerveau de systèmes à base de connaissances.
- A travers les approches de la littérature, nous présentons dans le **chapitre 3** les travaux connexes aux systèmes de recommandation.
- Dans le **chapitre 4**, notre approche sera détaillée en mettant l'accent sur la conception du système de recommandation à base de connaissances, intégrant les ontologies, avec présentation des différentes dimensions du système. Par la suite, nous décrivons les choix technologiques effectués concernant l'environnement de développement de notre application et l'architecture matérielle de notre système. Quelques exemples d'application seront illustrés afin de montrer la faisabilité de notre contribution.
- Nous terminons, finalement ce manuscrit par tirer quelques conclusions de ce travail et nous projetons ainsi nos idées futures qui pourront parfaire ce nous avons réalisé.

Chapitre I

Concepts de base des systèmes de recommandation

CHAPITRE I

Concepts de base des systèmes de recommandation

Sommaire

I.1	Introduction	8
I.2	Les systèmes de recommandation	8
I.2.1	Quelques définitions	8
I.2.2	Historique des systèmes de recommandation	9
I.2.3	Recherche d'information versus Filtrage d'information	10
I.3	Concepts de base	11
I.3.1	Utilisateur et Item	12
I.3.2	Evaluation	12
I.3.3	Communauté	12
I.3.4	Prédiction	12
I.3.5	Recommandation	12
I.3.6	Notion du profil	12
I.3.7	Notion de connaissance	13
I.4	Classification des systèmes de recommandation	13
I.5	Techniques de recommandation	14
I.5.1	Filtrage collaboratif	14
I.5.1.1	Filtrage basé modèle	15
I.5.1.2	Filtrage basé mémoire (basé voisinage)	15

I.5.2	Filtrage basé sur le contenu	16
I.5.3	Filtrage démographique	17
I.5.4	Filtrage basé utilité	17
I.5.5	Filtrage basé connaissance	17
I.5.6	Filtrage hybride	18
I.6	Avantage et limites des systèmes de recommandation	19
I.7	Conclusion	21

I.1 Introduction

Les systèmes de recommandation ont été utilisés afin de faire face aux problèmes de surcharge et de richesse d'informations disponibles notamment à travers le Web ou les e-services. Les systèmes de recommandation visent à proposer à un utilisateur actif une ou des recommandations d'items susceptibles de l'intéresser. Ces recommandations peuvent concerner un article à lire, un livre à commander, un film à regarder, un restaurant à choisir, etc.

Nous touchons dans ce chapitre les points essentiels des systèmes de recommandation à savoir, leur évolution, concepts de base et notions, ensuite, les principaux types de technique de recommandation tels que le filtrage collaboratif et le filtrage basé contenu, le filtrage hybride et les problèmes rencontrés dans ce genre de systèmes seront présentés. Et nous abordons ainsi un type spécifique de ces systèmes qui repose sur les connaissances, notre approche se base sur ce dernier type.

I.2 Les systèmes de recommandation

I.2.1 Quelques définitions

Les systèmes de recommandation peuvent être définis de plusieurs façons, vue la diversité des classifications proposées pour ces systèmes. Une définition générique est donnée par la majorité des auteurs, où ces systèmes (SR) se décrivent par « *l'identification automatique des préférences des utilisateurs à travers leurs interactions avec le système en se basant sur des retours d'information* ». [Res et Var, 97],[Hu et al,08],[Sal,13]; [SAL et al,13];[Sha et Sum, 13];[Sal et Kam, 13];[Bob et al, 10]

L'expression du filtrage d'information a été introduite pour décrire une variété de processus dédiés à la fourniture de l'information adéquate aux personnes qui en ont besoin.

Le but de filtrage est de sélectionner et suggérer aux utilisateurs, à partir de larges volumes d'informations générés dynamiquement, les informations jugées pertinentes pour eux. Par conséquent, il peut être vu aussi comme étant le processus d'élimination de données indésirables sur un flux entrant, plutôt que la recherche de données spécifiques sur ce flux [Bel et Cro, 92].

Un certain nombre d'approches de SR basés sur le filtrage a été proposé dans la littérature, tels que : le Filtrage Collaboratif dans [Bur,02], [Bal et Sho, 97], et le Filtrage basé sur le Contenu dans [Moo et Roy, 99], [Paz et Bil, 07]. Chacune présentant des forces et des faiblesses [Bur,02].

J. Bobadilla. définit les systèmes de recommandation comme « *Une forme spécifique de filtrage d'information visant à présenter les éléments d'information qui sont susceptibles d'intéresser l'utilisateur, Ils peuvent être définis comme des programmes qui essayent de recommander les articles les plus appropriés (des produits ou des services) aux utilisateurs particuliers (des individus ou des affaires (activités)) en prévoyant l'intérêt d'un utilisateur dans un article basé sur des informations liées sur les articles, les utilisateurs et les interactions entre des articles et des utilisateurs* » [Bob, 13].

Une autre définition de Burke et Robin a été proposée, « *Un SR est un système capable de fournir des recommandations personnalisées permettant de guider l'utilisateur vers des ressources intéressantes ou utiles au sein d'un espace de données important* » [Bur,02].

Les domaines d'application des systèmes de recommandation prennent de plus en plus de l'ampleur, on peut citer par exemple, le domaine du multimédia (musique, films..etc), culture (livres, articles..), restauration, tourisme...etc [Con et Her,99] [Bre et al,98], [Gol et al,92]), [McN et al,02] [Cos et al,02].

Ces domaines d'application se diffèrent mais partagent quasiment les mêmes problèmes.

I.2.2 Historique des systèmes de recommandation

La capacité des ordinateurs pour faire des recommandations à des utilisateurs a été reconnue assez tôt dans l'histoire de l'informatique. Grundy dans [Ric,79] a introduit la première étape vers des systèmes de recommandation automatiques en proposant un SR bibliothécaire. Ce système était assez primitif, mais constituait une première tentative intéressante dans ce domaine. Cependant, son utilisation est restée très limitée.

Au début des années 1990, le filtrage collaboratif apparaît comme une solution pour faire face à la

surcharge d'information. Un SR de recommandation de documents Tapestry a été proposé par [Gol et al,92], ainsi que la création du laboratoire de recherche GroupLens, qui travaille explicitement sur le problème de la recommandation automatique dans le cadre des forums de news de Usenet.

Quelques années plus tard, un système à base de filtrage collaboratif automatique a été développé par les chercheurs de GroupLens [Res et al,94], parallèlement avec le système Ringo [Upe 94] [Sha et Mae,95a], apparu en 1995, et Bellcore [Hil et al,95].

Le terme « Filtrage Collaboratif » a été utilisé jusqu'à l'année 1997. Cependant l'appellation « Systèmes de recommandation » a été standardisée la même année par Resnick et Varian [Res et Var,97]. Ensuite l'apparition des premiers SR hybrides Balavonic et Shoham [Bal et Sho,97].

Plus tard, en 2001, la notion du FC basé item a été introduite par Sarwar et al [Sar et al, 01] et étendu après par [Lin et al, 03] dans le portail Web d'Amazon.com.

I.2.3 Recherche d'information versus Filtrage d'information

Le filtrage d'information et la recherche partagent les mêmes objectifs, néanmoins, La différence entre les deux notions reste toujours pas claire ; cependant, à travers leurs définitions, on peut tirer cette différence. Généralement, un système de recherche d'information est un système ayant pour fonction d'apporter aux utilisateurs les documents qui leur permettent de satisfaire leurs besoins en information, en se basant sur les requêtes qu'ils fournissent [Bel et Cro,92], tandis qu'un système de filtrage d'information fournit des documents (ou des items) à une personne ou ensemble de personnes, en se basant sur leurs profils à long terme [Cro,93].

Nous dressons dans le tableau ci-dessous, quelques points de distinction entre le Filtrage et la Recherche d'information :

Recherche d'information	Filtrage d'information
Guider les utilisateurs à retrouver les items.	Recommande directement les items intéressants.
L'utilisateur saisie une requête et le système la compare avec les sources d'information disponibles.	Le système compare l'ensemble des informations disponibles aux profils des individus ou groupe d'individus.
Les données sont sélectionnées à partir d'une base de données statique.	Sélection ou élimination d'information à partir d'une source de données dynamique.
Traite une seule requête à la fois.	Peut être utilisé par une ou plusieurs personnes en même temps.
Collecte et organise les documents selon la recherche des individus.	Distribue automatiquement des documents et ou des items.
Utilise le principe de la découverte de l'information dans la base de données.	Utilise un processus d'élimination d'information du flux de données entrant.
Utilise les requêtes fournies par l'utilisateur qui représentent des intérêts à court terme.	Utilise les profils des utilisateurs qui représentent généralement des intérêts à long terme.
Autorise l'utilisateur à interagir avec un document pendant une seule session de recherche.	Permet l'interaction dans différentes sessions.
Résultats non personnalisés.	Résultats personnalisés.
Utilise souvent des mots clés.	Plusieurs dispositifs utilisés.
Non adaptatif.	S'adapte au changement des profils des utilisateurs.

TABLE I.1 – Comparaison entre la recherche et le filtrage d'information.

La notion de recommandation a vu le jour à travers la notion de filtrage. Nous aborderons dans la suite, les notions de base ainsi que les techniques de recommandation.

I.3 Concepts de base

Nous présentons dans cette section quelques concepts de base liés aux systèmes de recommandation :

I.3.1 Utilisateur et Item

- **Utilisateur** : toute personne accédant au système et faisant l'enregistrement, en entrant ses informations démographiques, son centre d'intérêt et d'autres informations personnelles. On désigne par U , l'ensemble des utilisateurs où un utilisateur donné $u \in U$.
- **Item** : l'entité qui représente les éléments composants la liste de recommandations et les besoins des utilisateurs est appelée « item », elle comprend aussi tout élément susceptible d'être vendu (livres, produits, etc. les sites e-commerce comme Amazon.com), vu (les films dans les sites de TV en ligne tel que Netflix), écouté (musique) ou lu (comme des informations dans des journaux en ligne, des revues dans des bibliothèques numériques), des destinations de vacance, des restaurants, etc. L'item peut encore être un individu ou un groupe d'individus.

I.3.2 Evaluation

On peut attribuer une valeur numérique dans une échelle qui représente la préférence ou non d'un item donné par un utilisateur donné[Bur,02].

I.3.3 Communauté

L'ensemble d'utilisateurs similaires et partageant les mêmes centres d'intérêt, préférences et goût, peuvent être regroupés selon un critère donné, est appelé Groupe ou Communauté.[Bou,05].

I.3.4 Prédiction

Un calcul d'une note probable dont l'utilisateur va l'attribuer à un item qui n'est pas encore vu ou évalué, est appelé prédiction, en se basant sur des notes données par les voisins de cet utilisateur.

I.3.5 Recommandation

Le calcul d'une liste d'items les plus aimés / préférés par l'utilisateur est appelée recommandation. Celle-ci est faite en attribuant des scores pour les items selon leurs popularités ou leurs préférences [Wen et al,08], par exemple. Contrairement à la prédiction, le calcul des recommandations ne se base pas strictement sur les évaluations.

I.3.6 Notion du profil

- **Profil utilisateur** : c'est un portail incluant une description des caractéristiques de l'utilisateur, telles que, ses centres d'intérêt, ses données démographiques, ou bien ses préférences

exprimées sous forme d'évaluations, etc. Il existe certaines approches de construction des profils utilisateurs. Les travaux des auteurs [Bur,02], [Cho et al,02] et [Sha et al,97] peuvent constituer un référentiel de ces approches.

- **Profil item** : il s'agit ici, de décrire l'item avec un ensemble de propriétés, appelées aussi attributs [Rij,79].

Outre les attributs d'items, on trouve aussi les méta-données qui présentent des informations (données) sur les caractéristiques elles-mêmes [Ber,02].

I.3.7 Notion de connaissance

- **Connaissance** : « ensemble de notions et des principes qu'une personne acquiert par l'étude, l'observation ou l'expérience et qu'elle peut intégrer à des habiletés »
- **Un cas** : est une expérience représentée par une connaissance. Cette expérience constitue une leçon permettant à un système appelé un système de raisonnement à base de cas ou RàPC, de résoudre des problèmes de différentes natures. Selon le domaine d'application et les objectifs à atteindre, les informations contenues dans le cas varient. On peut définir un cas comme étant la description informatique d'un épisode de résolution de problème.

I.4 Classification des systèmes de recommandation

Nous pouvons classer les techniques de recommandation selon diverses manières. Nous avons fait un constat, que, malgré la diversité des termes utilisés, ils désignent parfois une même méthode ou approche. La classification la plus utilisée repose sur trois types : filtrage basé sur le contenu, filtrage collaboratif et filtrage hybride [Ado et Tuz,05]. Robin Burke [Bur,02] a proposé d'autres approches qui sont, la recommandation basée sur les données démographiques, la recommandation basée sur la connaissance (knowledge-based) et la recommandation basée sur l'utilité (utility-based).

Rajoute encore que ces approches peuvent être considérées comme des cas particuliers des approches classiques.

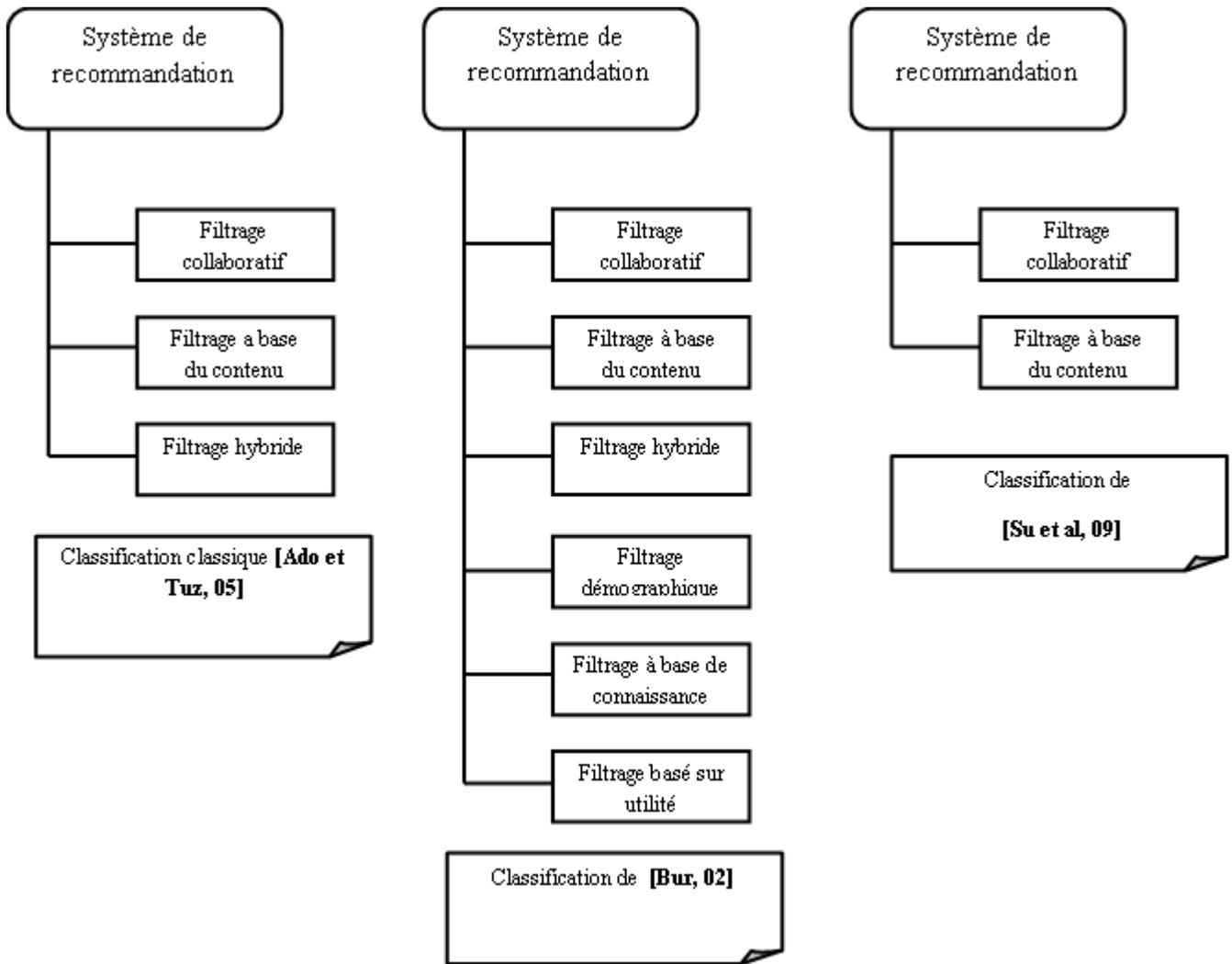


FIGURE I.1 – Classification des systèmes de recommandation.

I.5 Techniques de recommandation

I.5.1 Filtrage collaboratif

Cette technique procède à une comparaison des utilisateurs entre eux en se basant sur leurs évaluations passées pour créer des communautés, les utilisateurs reçoivent les items jugés pertinents par leurs communautés [Bre et al,98].

Des évaluations (notes) constituant les profils des utilisateurs sont fournies afin d'exprimer les préférences de ceux-ci envers les items. Ces évaluations sont comparées ensuite à celles données par les autres utilisateurs, en calculant la similarité entre eux pour sélectionner les plus proches utilisateurs. Ensuite, les prévisions des notes inconnues sont calculées, comme moyenne pondérée des évaluations faites par les plus proches utilisateurs. Autrement dit, les recommandations pour un utilisateur donné se basent sur le comportement et les préférences des utilisateurs similaires [Lin et al,03].

On note dans cette approche deux limites et notamment par rapport au filtrage basé contenu,

qui ont été résorbés, tels que, la possibilité de traitement de n'importe quel type de contenu. Un certain ensemble des plus proches voisins est identifié pour chaque utilisateur, et la décision de proposer ou non un item à un utilisateur dépendra des appréciations des membres de son voisinage.

Usuellement, deux grandes sous-familles du filtrage collaboratif ont été identifiées : les méthodes basées sur la mémoire (Memory-based) et les méthodes basées sur un modèle (model-based). Des algorithmes bien adaptés à chaque sous catégorie ont été proposés, [Ado et Tuz,05], [Des et Kar,11].

I.5.1.1 Filtrage basé modèle

Ici, les informations disponibles sur les utilisateurs et sur les items (évaluations) sont utilisées afin d'élaborer un modèle générant des recommandations. Ces méthodes présentent un atout par rapport à celles basées sur mémoire, ceci réside dans la possibilité d'être conçues hors ligne en utilisant des données d'apprentissage afin d'être utilisées facilement en ligne.

Nous distinguons deux manières différentes utilisées dans ce cadre pour élaborer la prédiction :

- Construction d'un modèle probabiliste pour estimer les valeurs de prédiction fournies par le modèle lui-même.
- Catégorisation (clustering) des utilisateurs items, qui seront évaluées par la suite par les méthodes basées mémoires.

I.5.1.2 Filtrage basé mémoire (basé voisinage)

Cette famille d'algorithmes utilise la totalité ou une partie des profils utilisateur pour générer des nouvelles prédictions. Les méthodes des plus proches voisins ont connu une large popularité et elles sont grandement utilisées dans ce cadre de filtrage.

Le calcul de recommandation passe par deux phases [Ami,09] :

- **Phase de calcul de voisinage** : étant donné un utilisateur u_i , on procède à chercher les utilisateurs u_j ayant des profils similaires. Ces similarités peuvent être calculées en utilisant la similarité vectorielle ou la corrélation de Pearson [Ami,09].

– *Similarité vectorielle* : calculée selon la formule suivante :

$$\cos(A, B) = \sum_{j=1}^{l1} \frac{V_{A,j}}{\sqrt{\sum_{j=1}^{l1} V_{A,j}^2}} \frac{V_{B,j}}{\sqrt{\sum_{j=1}^{l1} V_{B,j}^2}} \quad (\text{I.1})$$

Où :

$l1$: nombre d'items communs entre les utilisateurs A et B voté par v.

$V_{A,j}$: vote de A pour l'item j.

$V_{B,j}$: vote de B pour l'item j.

– *Corrélation de Pearson* : elle se calcule comme suit :

$$\omega(A, B) = \frac{\sum_j (V_{A,j} - \bar{V}_A)(V_{B,j} - \bar{V}_B)}{\sqrt{\sum_j (V_{A,j} - \bar{V}_A)^2 \sum_j (V_{B,j} - \bar{V}_B)^2}} \quad (\text{I.2})$$

Où :

j : nombre d'objets ayant été votés à la fois par les utilisateurs A et B.

$V_{A,j}$: vote de A pour l'item j.

\bar{V}_A : moyenne de votes de A.

- **Phase de prédiction** : après avoir calculé les similarités d'un item j par un utilisateur i, on procède aux calculs des prédictions qui se fera comme suit :

$$P_{A,j} = \bar{V}_{A,j} + \frac{\sum_{i=1}^n \omega(A, i)(V_{i,j} - \bar{V}_i)}{\sum_{i=1}^n |\omega(A, i)|} \quad (\text{I.3})$$

Où :

n : nombre d'utilisateurs présents dans le voisinage de A, ayant déjà voté sur l'objet j.

$V_{i,j}$: vote de l'utilisateur i pour l'objet j.

\bar{V}_i : moyenne de votes de l'utilisateur i.

I.5.2 Filtrage basé sur le contenu

Le principe adopté dans ce type est de recommander des documents similaires à ceux que l'utilisateur a déjà appréciés, [Per et al,04]. La recommandation se fait suite à une comparaison entre le contenu du document et le profil de l'utilisateur. Ce dernier est parallèlement représenté sous forme de termes contenus dans le document que l'utilisateur a jugé intéressant, ainsi que des pondérations de ces termes. Plusieurs algorithmes ont été proposés pour l'analyse du texte des documents et la recherche des régularités qui servent de base dans la recommandation [Paz,99].

I.5.3 Filtrage démographique

Ce filtrage repose sur le groupement (catégorisation) des utilisateurs par rapport à leurs profils démographiques (âge, sexe, profession, localisation, langue, pays..etc.). Ces approches ont été très répandues dans la littérature de marketing, mais elles ont reçu peu d'attention dans le domaine des algorithmes de recommandation vu la limitation de performances qui dépend essentiellement du nombre des plus proches voisins d'un utilisateur donné *i*. [Maa,16].

I.5.4 Filtrage basé utilité

Dans ce type de recommandation, on ne construit pas une généralisation à long terme pour les utilisateurs du système, mais il se base sur une évaluation de la correspondance entre les besoins des utilisateurs et l'ensemble des items disponibles. La notion d'utilité ici repose sur l'idée de calculer une utilité afin de recommander les items les plus utiles pour un utilisateur donné *i*.

Nous optons dans ce type pour quelques techniques telles que les techniques de satisfaction de contraintes pour localiser les meilleures correspondances. Notamment, ce type de recommandation souffre de quelques problèmes à savoir, la fonction d'utilité qui doit être introduite au système, ainsi que le non prise en compte des préférences des usagers [Maa,16].

I.5.5 Filtrage basé connaissance

S'inspirant principalement sur la recherche en raisonnement à base de cas, ces systèmes de recommandation à base de connaissances suggèrent des solutions reposant sur des inférences des besoins et des préférences des utilisateurs. Un constat a été fait, est que la plupart des systèmes de recommandation adoptent un certain type d'inférences.

Comparativement aux autres méthodes de recommandations su-citées, ce type exploite des connaissances du domaines et fait des inférences sur les besoins et préférences des utilisateurs, ce qui le rend plus adaptable dans certaines situations qui souffrent d'un manque d'information (problème de démarrage à froid) [Bur,02].

Cependant, on distingue dans ce type de système, trois classes de connaissances [Bri,98] :

- **Catalogue des connaissances** : représente les connaissances sur les items à recommander et leurs caractéristiques.

- **Connaissance fonctionnelle** : Le système doit être capable de relier entre les besoins de l'utilisateur et l'item qui pourrait satisfaire ces besoins.
- **Connaissance sur l'utilisateur** : le système doit avoir des connaissances sur l'utilisateur qui pourraient prendre la forme d'informations démographiques générales ou des informations spécifiques sur les nécessités, pour lesquels une recommandation est demandée.

Nous abordons en détail dans le chapitre suivant le principe des systèmes de recommandations à base de connaissances.

I.5.6 Filtrage hybride

Lorsque le système de recommandation combine deux ou plusieurs approches différentes, il est dit alors un système hybride. L'idée ici est de travailler sur la complémentarité entre approches, comme par exemple, la recommandation basée sur le contenu et la recommandation collaborative [Ado et Tuz,05].

Plusieurs travaux de recherche ont abordé ce principe d'hybridation pour y remédier aux lacunes et palier aux inconvénients des approches utilisées seules et tirer profit de la combinaison [Bal et Sho,97].

Cependant, les manières d'hybridation ne suivent pas un consensus bien défini par la communauté de chercheur. Toutefois, Burke [Bur,02] a identifié sept manières différentes de faire l'hybridation :

- **Pondérée (Weighted)** : le score ou la prédiction obtenu par chacune des techniques est combiné en un seul résultat.
- **Par sélection (Switching)** : le système bascule entre deux techniques de recommandation, par exemple, en fonction de la situation.
- **Mixte (Mixed)** : ici, les listes des recommandations issues des techniques sont fusionnées en une seule liste.
- **Par combinaison des propriétés (Feature combination)** : les données issues des techniques sont combinées et transmises à un seul algorithme de recommandation.
- **Par augmentation de propriétés (Feature augmentation)** : le résultat d'une technique est utilisé comme entrée de l'autre technique.

- **En cascade** : Dans ce type d'hybridation, une technique de recommandation est utilisée pour produire un premier classement des items candidats et une deuxième technique affine ensuite la liste des recommandations.
- **En définissant un niveau méta** : Cette méthode est analogue à la méthode par augmentation de propriétés mais c'est le modèle appris qui est utilisé en entrée de la deuxième technique et non la liste résultante des recommandations.

I.6 Avantage et limites des systèmes de recommandation

Dans cette section, nous allons faire un récapitulatif sur quelques avantages et inconvénients des systèmes de recommandation selon un certain nombre de critères, à savoir :

- **Adaptabilité** : la précision de la recommandation dépend proportionnellement de l'augmentation de la base de données des évaluations.
- **Nouvel utilisateur** : la pertinence des recommandations est affectée par le nombre d'évaluations accumulées par l'utilisateur.
- **Nouvel item** : dans le processus de recommandation, les items doivent avoir suffisamment d'évaluations pour qu'ils soient pris en considération.
- **Démarrage à froid** : ce type de problème se présente pour les nouveaux utilisateurs intégrant le système, car ce dernier ne dispose pas suffisamment d'informations à leur sujet. Un utilisateur avec un profil vide doit consacrer des efforts à l'aide du système avant d'obtenir la recommandation qui lui est utile. D'un autre côté, la recommandation des nouveaux items (qui viennent d'être ajoutés aux collections) est conditionnée par l'acquisition d'informations sur ces items.
- **Cas du système débutant** : lors du lancement d'un nouveau service de recommandation. Le système ne possède cependant aucune information sur les utilisateurs et les items. et par conséquent, les méthodes de filtrage collaboratif ne peuvent pas opérer sur une matrice d'usages vide. La solution consiste en général à trouver des informations descriptives des items afin d'organiser le catalogue et inciter les utilisateurs à le parcourir jusqu'à ce que la matrice d'usages soit assez remplie et permette de passer en mode collaboratif.

Nous synthétisons dans le tableau ci-dessous les points forts et faibles des méthodes traditionnelles utilisées par les systèmes de recommandation [Lao et Kor,16].

Technique	Points forts	Points faibles
Filtrage à base du contenu	<ul style="list-style-type: none"> <input type="checkbox"/> La présence d'une large communauté d'utilisateurs pour pouvoir effectuer des recommandations n'est pas un besoin nécessaire. <input type="checkbox"/> La génération de listes de recommandation ne dépend pas du nombre des utilisateurs (même avec qu'un seul usager, elle peut être générée). <input type="checkbox"/> Croissance de la qualité croit avec le temps. <input type="checkbox"/> La connaissance des informations sur les autres utilisateurs n'est pas vraiment nécessaire. <input type="checkbox"/> La prise en considération des goûts uniques des utilisateurs. 	<ul style="list-style-type: none"> <input type="checkbox"/> L'analyse du contenu est nécessaire pour faire une recommandation. <input type="checkbox"/> La recommandation des images et de vidéos s'avère problématique en absence de Métadonnées. <input type="checkbox"/> Nécessité du profil d'utilisateur.
Filtrage collaboratif	<ul style="list-style-type: none"> <input type="checkbox"/> Aucune connaissance requise sur le contenu de l'item ni sa sémantique. <input type="checkbox"/> Possibilité d'évaluation de la qualité de recommandation. <input type="checkbox"/> Dépendance proportionnelle entre le nombre d'utilisateurs est la qualité des recommandations. 	<ul style="list-style-type: none"> <input type="checkbox"/> Démarrage à froid. <input type="checkbox"/> Nouvel Item. <input type="checkbox"/> Nouvel utilisateur. <input type="checkbox"/> Problème de confidentialité. <input type="checkbox"/> La complexité notamment dans les systèmes avec un grand nombre d'items et d'utilisateurs le calcul croit linéairement.

TABLE I.2 – Avantages et inconvénients des systèmes de recommandation traditionnels.

I.7 Conclusion

Au long de ce chapitre, nous avons abordé les systèmes de recommandation, ainsi que les concepts qui leurs sont liés. Une classification générale a été présentée décrivant les diverses techniques de filtrage. Ces dernières, comme toute technique, d'ailleurs, peuvent présenter quelques lacunes. Pour y remédier à celles-ci, de nouvelles familles de filtrage d'information ont vu le jour (notamment, ces dix dernières années), telles que les systèmes de filtrage d'information à base d'ontologies, les systèmes d'annotations collaboratives, les systèmes de filtrage d'informations à base de réseaux sociaux, etc.

Ce constat a motivé notre choix qui a porté sur les systèmes de recommandation à base de connaissances intégrant les ontologies comme outils de raisonnement.

Dans le chapitre qui suit, nous présentons en détail les principes des ces systèmes reposant sur les connaissances et le raisonnement.

Chapitre II

Système à base de connaissance (basé ontologie)

CHAPITRE II

Système à base de connaissance (basé ontologie)

Sommaire

II.1 Introduction	25
II.2 Apports de l'ingénierie des connaissances aux Systèmes de Recommandation	25
II.3 Systèmes à base de connaissances	26
II.3.1 Définition	26
II.3.2 Rôle des systèmes à base de connaissances	27
II.4 Raisonnement à base de cas	27
II.4.1 Définition	27
II.4.2 Cycle RàPC	28
II.5 Outils intégrant le RàPC	29
II.6 Systèmes de recommandation à base d'ontologie	30
II.6.1 Apport des ontologies	30
II.6.1.1 Ontologie : Définitions et origine	30
II.6.2 Objectifs de développement des ontologies	31
II.7 Composantes d'ontologies	32
II.7.1 Concept (classe)	32
II.7.2 Relation	33
II.7.3 Fonctions	33

II.7.4	Axiomes	33
II.7.5	Instances	33
II.8	Classification des ontologies	33
II.8.1	Typologie selon le niveau de conceptualisation	34
II.8.2	Typologie selon le niveau de formalisation utilisée	35
II.8.3	Typologie selon le niveau de détail de l'ontologie	35
II.8.4	Typologie selon le niveau de complétude	35
II.8.5	Typologie selon le niveau de complexité de l'ontologie	36
II.9	Cycle de vie d'une ontologie	36
II.10	Processus de construction d'une ontologie	37
II.10.1	Spécification (évaluation des besoins)	38
II.10.2	Conceptualisation	38
II.10.3	Ontologisation	38
II.10.4	Opérationnalisation	38
II.11	Méthodes et méthodologies de construction d'ontologies	39
II.11.1	Démarches de construction d'une ontologie	39
II.11.2	Méthodologies de construction	39
II.12	Langages d'ontologie	40
II.12.1	Langages traditionnels	40
II.12.2	Langage web standards ou basé sur xml	40
II.12.3	Langages d'interrogation des ontologies	42
II.12.3.1	Protocole de requête SPARQL	42
II.12.4	Approches du raisonnement	43
II.13	Outils de développement d'ontologies	44
II.13.1	PROTÉGÉ	44
II.13.2	ODE (ONTOLOGY DESIGN ENVIRONMENT)	44
II.13.3	OntoEdit	44
II.13.4	WebOde	45
II.13.5	DoE	45
II.13.6	Framwork jena	45
II.14	Conclusion	45

II.1 Introduction

Ces dernières années, les domaines de l'ingénierie de connaissance, l'intelligence artificielle et la recherche d'information, sont devenus la tendance actuelle qui a envahi le monde de la recherche.

Les systèmes à base de connaissances intégrant les principes de raisonnement appelés souvent « Raisonnement à base/ partir de Cas – RàPC » en font partie.

Les ontologies ont été amplement utilisées dans ce domaine et ont prouvé leur utilité, de là, nous allons dans un premier lieu, introduire le concept d'ontologie et les éléments qui la constituent. Par la suite, nous présenterons ses différentes classifications. En outre, nous décrivons les principales représentations des connaissances. De plus, nous décrivons brièvement quelques outils de développement d'ontologies.

II.2 Apports de l'ingénierie des connaissances aux Systèmes de Recommandation

L'ingénierie des connaissances constitue une discipline de l'intelligence artificielle qui s'intéresse à la gestion des connaissances et à leur mise en œuvre dans différents emplois, particulièrement dans le volet de recherche d'information, on peut citer par exemple, l'aide à la décision, la conception de systèmes experts. Dans cette optique, elle englobe plus précisément, les processus de conception de modèles de connaissance, c'est-à-dire la représentation des concepts et de leur interconnexion (e.g, ontologies, graphes de connaissances, bases de connaissances, etc.) et les différentes techniques qui permettent d'exploiter ou d'enrichir cette connaissance : extraction de connaissances à partir de textes, mesures de similarité, appariement de ressources, visualisation, etc. Elles disposent Les techniques d'ingénierie de connaissances sont dotées de capacité de raisonnement qui permettent d'inférer de nouvelles connaissances à partir des faits existants et des règles logiques définies.

Un autre apport de l'intégration de connaissances du domaine et d'autres informations fiables permet aux spécialistes et experts du domaine d'affiner l'analyse de leurs données et de les manipuler plus efficacement au travers d'une meilleure représentation, et d'automatiser certains traitements/raisonnements.

Les systèmes de recommandation ont eu leurs parts des bénéfices de l'intégration de L'ingénierie de connaissances qui constitue un atout concret afin de les améliorer en y intégrant une part de

l'expertise du domaine. Ainsi, les systèmes de recommandations ont influencés par la façon avec laquelle sont exploitées ces connaissances.

D'autant plus, on peut garantir quelques points tels que, l'évitement des problèmes de démarrage à froid « rump-up problem », la sensibilité aux changements des préférences, inclusion des attributs divers (non pas seulement de l'item recommandé) et la possibilité de relier les besoins des utilisateurs et les items.

Ce compromis a fait l'objet de divers travaux dans la littérature, nous allons consacrer le prochain chapitre pour faire un survol sur ces œuvres.

II.3 Systèmes à base de connaissances

II.3.1 Définition

Un système à base de connaissances s'appuie sur des connaissances relatives un domaine donné pour résoudre des problèmes se posant dans le domaine. Pour concevoir un tel système, il faut mettre au point des formalismes de représentation des connaissances et de raisonnement qui permettent de prendre en compte les niveaux syntaxique et sémantique des connaissances du domaine considéré.

Il est fondé sur une séparation entre les connaissances nécessaires pour résoudre un problème et les mécanismes de raisonnement exploitant ces connaissances (appelés selon les cas structures de contrôle, interpréteurs, moteurs d'inférence)[Flo,06].

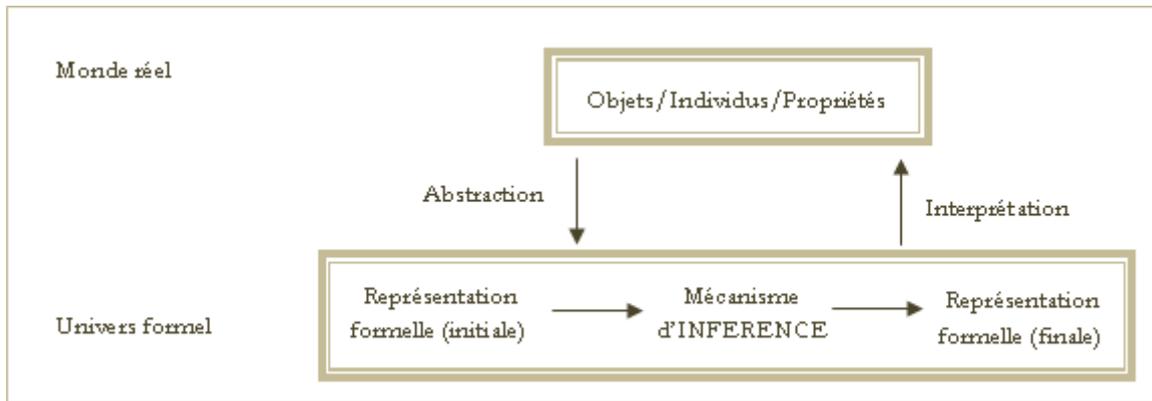


FIGURE II.1 – Représentation et manipulation de connaissances dans un système à base de connaissances.

II.3.2 Rôle des systèmes à base de connaissances

- Inscrit des connaissances issues de l'expertise ou/et de la pratique,
- Est donc spécialisé sur une expertise ou une pratique donnée,
- Fonde le « raisonnement » sur des mécanismes d'inférence logique ou analogique,
- Intègre une représentation symbolique,
- Autorise parfois une certaine prise en compte de l'incertitude,
- Utilisation des heuristiques qui sont des connaissances spécifiques au domaine qui guident la recherche de solutions et orientent les décisions de résolution de problème,
- Conserver des savoir et des savoir faire et la sémantique qui leurs y associée.

II.4 Raisonnement à base de cas

II.4.1 Définition

Traduit par « Case-Based Reasoning » en anglais, Le RàPC est le processus visant la réutilisation des expériences passées. C'est une technique issue du domaine de l'intelligence artificielle. La première fois où il a été conçu c'était à la fin des années 70 par Schank et Abelson [Sch,82]. Il été utilisé au début dans les systèmes experts et les sciences cognitives.

Dans cette optique, l'utilisateur tente de résoudre un nouveau problème en reconnaissant les similarités avec des problèmes préalablement résolus, appelés cas.

Communément, cas est un problème spécifique qui a été identifié, résolu, stocké et indexé dans une mémoire avec sa solution, et éventuellement le processus d'obtention de celle-ci [Cor,06].

De nombreux domaines appliquent Les systèmes de RàPC tels que, la médecine, le commerce, les services de consultation, la maintenance, le contrôle, l'analyse financière.

II.4.2 Cycle RàPC

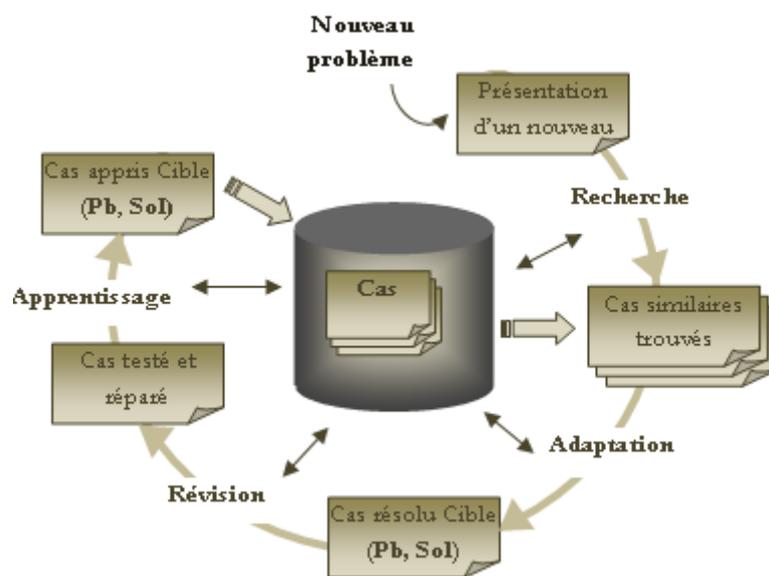


FIGURE II.2 – Cycle du RàPC [Rac,04].

Nous pouvons présenter les étapes du cycle comme suit :[Ros,06]

1. **Phase d'élaboration d'un nouveau problème (cas cible)** : dans cette phase du cycle, on procède à acquérir des informations sur le nouveau problème, afin d'élaborer une description initiale le concernant. Cette description comprend : la structuration, la modélisation et la représentation du cas, qui doit être représenté selon le même formalise (structure) d'un cas source.
2. **Phase de recherche (remémoration) des cas source** : le(s) cas les plus similaires sont recherchés, la remémoration désigne la recherche des correspondances entre les descripteurs des cas

(source) de la base et ceux du cas cible (à résoudre). Les méthodes de recherche utilisées s'inspirent des techniques de calcul d'un degré d'appariement des descripteurs (similarité entre deux cas), la pondération éventuelle des descripteurs, algorithmes d'apprentissage ...etc.

3. **Adaptation des cas** : elle consiste à réutiliser intégralement ou partiellement la solution du cas sélectionné (source) pour résoudre le nouveau problème (cible). L'adaptation peut se faire d'une manière manuelle via l'intervention humaine, ou automatique, il existe une panoplie de méthodes de l'adaptation automatique, on note par exemple :
 - L'adaptation générative ;
 - L'adaptation transformationnelle ;
 - L'adaptation compositionnelle ;
 - L'adaptation hiérarchique.
4. **Phase de révision de la solution trouvée** : la solution proposée à l'issue de la phase d'adaptation, sera évaluée dans le monde réel (bilan d'un cas), ainsi une vérification est faite par introspection de la base de cas en considérant la qualité des cas.
5. **La mémorisation nouveau cas (apprentissage)** : consiste en l'ajout éventuel du cas cible dans la base de cas. Dans cette phase on peut également synthétiser, modifier et apprendre de nouvelles connaissances.

II.5 Outils intégrant le RàPC

Il existe un nombre important d'outils intégrant le raisonnement à base de cas, cependant, nous n'allons pas les exposer toutes, nous allons énumérer quelques unes.

- Fouille de données ;
- Fouille de texte ;
- Graphes de connaissances ;
- Réseaux de neurones ;
- Systèmes experts ;
- Algorithmes évolutionnaires ;
- Logique de description ;

- Règles d'inférences ;
- Techniques des plus proches voisins ;
- Ontologie.
- ...etc.

II.6 Systèmes de recommandation à base d'ontologie

II.6.1 Apport des ontologies

II.6.1.1 Ontologie : Définitions et origine

Le mot ontologie trouve sa racine du grec, il est composé de deux mots : onto (qui signifie l'étude de l'être en tant qu'être) et logos (qui signifie le discours).

Diverses définitions ont été proposées dans la littérature. en 1991, l'auteur Neches et ses collègues furent les premiers qui ont proposé une définition de l'ontologie [Nec et al ,91]. Il la définit comme suit :

«Une ontologie définit les termes et les relations de base du vocabulaire d'un domaine ainsi que les règles qui indiquent comment combiner les termes et les relations de façon à pouvoir étendre le vocabulaire».

La définition la plus célèbre a été proposée par Gruber [Gru,93]

Dans le domaine de l'informatique et plus précisément de l'ingénierie des connaissances et de l'intelligence artificielle, une ontologie peut être définie comme une « *spécification formelle et explicite d'une conceptualisation partagée* » [Gru,93].

Le terme Conceptualisation désigne un modèle abstrait d'un domaine donné qui identifie les concepts représentatifs de ce domaine, et le terme Explicite fait référence aux types de concepts utilisés et les contraintes sur leurs utilisations qui doivent être explicitement définis. Formelle se réfère au fait qu'une ontologie doit être compréhensible par la machine. Partagée indique que l'ontologie représente une connaissance consensuelle, et qu'elle n'est pas restreinte à quelques individus mais acceptée par un large groupe.

L'auteur Borst a modifié légèrement la définition de Gruber en citant qu'une ontologie est définie comme étant [Bor,97] :

«Une ontologie est une spécification formelle d'une conceptualisation partagée».

Dans [Stu et al,98], une définition rassemblant les deux définitions en seule (celles de Gruber et Borst) a été proposée :

«Une ontologie est une spécification formelle et explicite d'une conceptualisation partagée».

En 1997, Guarino [Gua ,97] énonce la définition suivante :

«Les ontologies sont des spécifications partielles et formelles d'une conceptualisation commune».

II.6.2 Objectifs de développement des ontologies

Parmi les objectifs premiers des ontologies est de modéliser un ensemble de connaissances dans un domaine donné.

Diverses domaines d'application des ontologies peuvent être cités tels que, l'intelligence artificielle, le Web sémantique, le génie logiciel, l'informatique biomédicale et l'architecture de l'information pour une représentation de la connaissance d'un domaine. Et delà, les raisons pour lesquelles sont développées les ontologies deviennent multiples, nous citons :

- Partager la compréhension commune de la structure de l'information entre les personnes ou les fabricants de logiciels,
- Permettre la réutilisation du savoir sur un domaine,
- Expliciter ce qui est considéré comme implicite sur un domaine,
- Distinguer le savoir sur un domaine du savoir opérationnel,
- Analyser le savoir sur un domaine [Bel,15].

Nous étendons ces objectifs particulièrement aux **systèmes de recommandation**, dans cette optique, des modèles sémantiques et ontologiques peuvent être exploitée à chaque étape du processus de recommandation : depuis l'inscription de l'utilisateur, la création de son profil initial, la prédiction de ses goûts par rapport aux items, l'ajustement de son profil en fonction des feedbacks et jusqu'à finalement la recommandation proprement parler.

La figure suivante schématise les éléments décrits généralement par une ontologie :

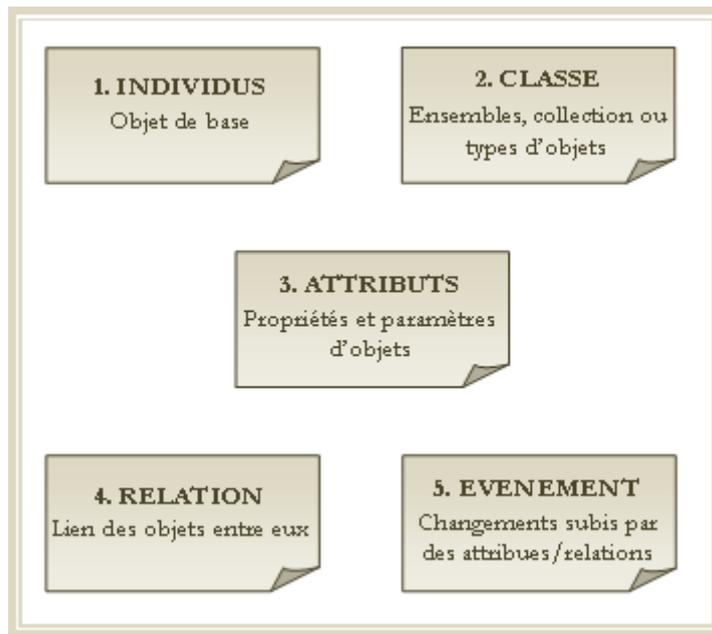


FIGURE II.3 – Organisation d'une ontologie.

II.7 Composantes d'ontologies

A partir du constat que les ontologies offrent un vocabulaire commun d'un domaine et définissent la signification des termes et des relations entre elles. Gomez [Gom,99] a proposé principalement cinq types pour formaliser la connaissance dans les ontologies à savoir : concepts (ou classes), relations (ou propriétés), fonctions, axiomes (ou règles) et instances (ou individus).

II.7.1 Concept (classe)

Appelé aussi termes ou classe d'ontologie, il représente donc les objets ou les classes qui seront relié entre eux.

II.7.2 Relation

Reliant les concepts entre eux et traduisant les associations sémantiquement, Ces relations sont généralement binaires telle que le premier argument appelé domaine et le deuxième appelé co-domaine.

II.7.3 Fonctions

Elles sont au fait, des cas particuliers de relations, dans laquelle un élément de la relation, (le n ème) est défini en fonction des $n-1$ éléments précédents.

II.7.4 Axiomes

Pour modéliser les abstractions vraies du domaine traduites par l'ontologie. Gomez précise que les axiomes formels sont utilisés pour vérifier la consistance de l'ontologie [Gom,99].

II.7.5 Instances

Représentent les éléments singuliers véhiculant les connaissances à propos du domaine.

II.8 Classification des ontologies

Plusieurs aspects sont adoptés pour classifier les ontologies, parmi ceux-ci, on distingue :

- Typologie selon le niveau de conceptualisation ;
- Typologie selon le niveau du formalisme utilisé ;
- Typologie selon le niveau de détail de l'ontologie ;
- Typologie selon le niveau de complétude

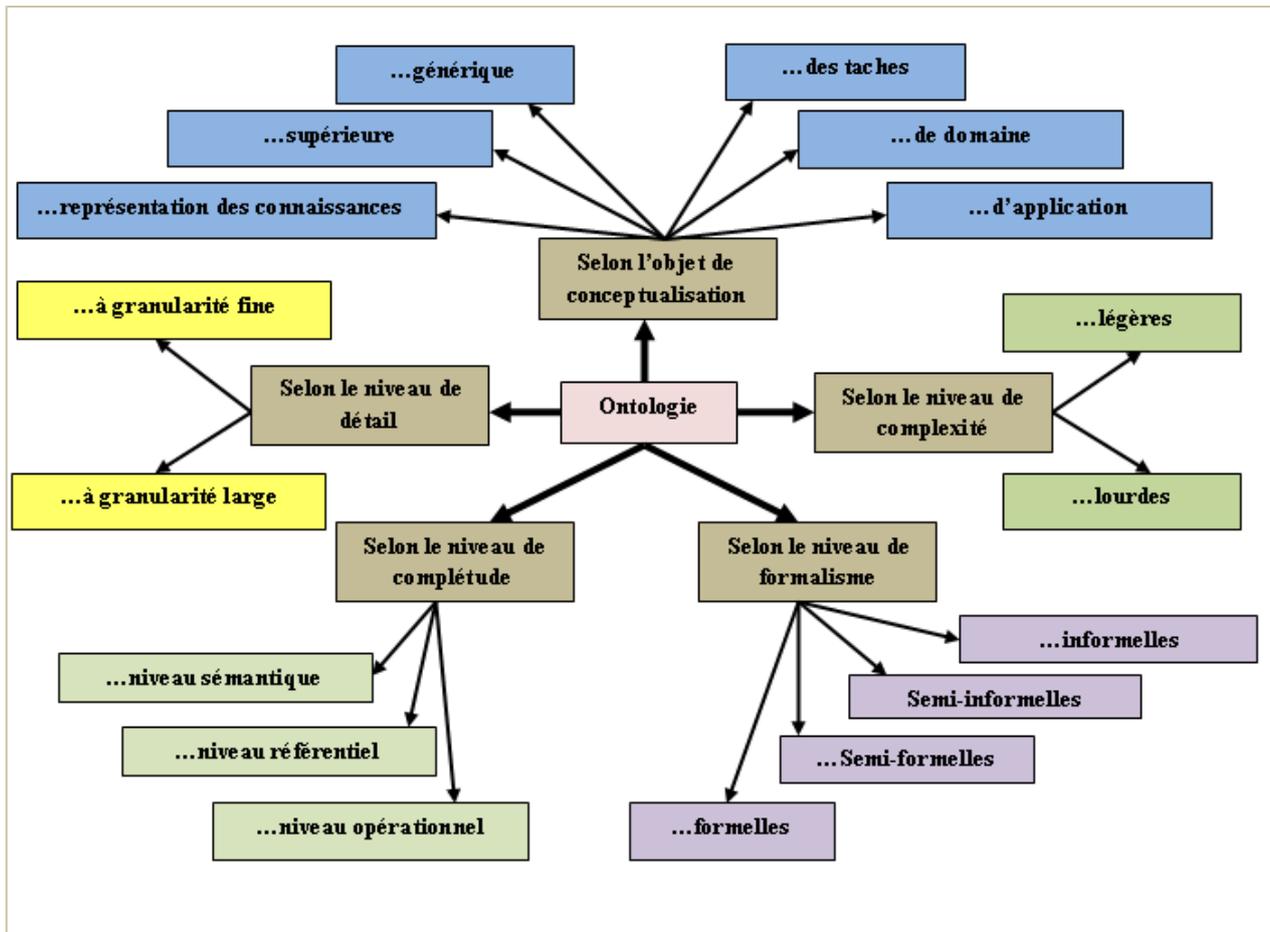


FIGURE II.4 – Classification des ontologies.

II.8.1 Typologie selon le niveau de conceptualisation

- **Ontologie de représentation de connaissances** : elles décrivent les connaissances nécessaires à la modélisation d'un système de représentation des connaissances. [Gru,93].
- **Ontologie haut niveau** : des concepts de très haut niveau sont décrits par ces ontologies, où elles fournissent une structure et un ensemble de concepts généraux sur lesquels les ontologies du domaine pourront être construites [Gua,98].
- **Ontologie générique** : appelées aussi méta-ontologies sont des ontologies décrivant des connaissances de hautes abstractions, qui peuvent concerner les concepts tels que le temps, l'espace, les évènements, etc. qui sont indépendants d'un domaine ou d'un problème particulier. Ce qui les rend réutilisables dans différents domaines [Gua,98].
- **Ontologie de domaine** : la grande majorité des ontologies existantes sont des ontologies du domaine. Ces ontologies contiennent des connaissances spécifiques à un domaine particulier. Néanmoins, elles sont assez génériques pour être utilisées et réutilisées sur différentes applications [Izo et al,00].

- **Ontologie de tâche** : c'est lorsqu'on décrit des connaissances relatives à une tâche générique ou une activité telle que par exemple le diagnostique ou la planification. [Gua, 98].
- **Ontologie d'application** : elles sont plus spécifiques et conçues pour une application précise dans un domaine particulier. Elles ne sont pas réutilisables.

II.8.2 Typologie selon le niveau de formalisation utilisée

Ushold et Grüninger ont proposé une classification comprenant quatre catégories [Usc et Gru,96] afin de rendre l'ontologie opérationnelle :

- **Informelles** : exprimée en langage naturelle. Ce qui permet de la rendre plus compréhensible pour l'utilisateur, cependant, la vérification de l'absence de redondances ou de contradiction devient plus difficile.
- **Semi-informelles** : elle est exprimée dans une forme restreinte et structurée de la langue naturelle, on peut garantir ici d'augmenter la clarté de l'ontologie tout en réduisant l'ambiguïté.
- **Semi-formelles** : l'ontologie est exprimée dans un langage artificiel définit formellement.
- **Formelles** : on exprime l'ontologie ici dans un langage artificiel disposant d'une sémantique formelle, permettant de prouver des propriétés de cette ontologie. On pourra effectuer des vérifications portant sur la complétude, non redondance, consistance, cohérence...etc.

II.8.3 Typologie selon le niveau de détail de l'ontologie

Deux catégories peuvent être définies dans cette [Psy et al,03] :

- **Granularité fine** : ce niveau correspond à des ontologies très détaillées, possédant ainsi un vocabulaire plus riche capable d'assurer une description détaillée des concepts pertinents d'un domaine ou d'une tâche.
- **Granularité large** : ce niveau correspond à des vocabulaires moins détaillés. Les ontologies de haut niveau possèdent généralement une granularité large, compte tenu que les concepts qu'elles traduisent sont normalement raffinés ensuite dans d'autres ontologies de domaine ou d'application.

II.8.4 Typologie selon le niveau de complétude

D'après [Bac,00], On peut définir trois niveaux de complétude :

- **Niveau sémantique** : On définit des principes dits différentiels dont tous les concepts, caractérisés par un terme/libellé doivent les respecter :
 - Communauté avec l'ancêtre ;
 - Différence (spécification) par rapport à l'ancêtre ;
 - Communauté avec les concepts frères (situés au même niveau) ;
 - Différence par rapport aux concepts frères.
- **Niveau-Référentiel** : Les concepts référentiels ou formels, se caractérisent par un terme/libellé dont la sémantique est définie par une extension d'objets. L'engagement ontologique spécifie les objets du domaine qui peuvent être associés au concept, conformément à sa signification formelle.
- **Niveau-Opérationnel** : Les concepts du niveau opérationnel ou computationnel sont caractérisés par les opérations qu'il est possible de leur appliquer pour générer des interfaces ou engagement computationnel.

II.8.5 Typologie selon le niveau de complexité de l'ontologie

Lavoie énonce les classes suivantes [Lav,07] :

- **Ontologies légères (light weight ontologies)** : incluent des concepts comprenant des propriétés et sont organisées en taxonomies avec des relations conceptuelles.
- **Ontologies lourdes (heavy weight ontologies)** : ajoutent aux ontologies légères des axiomes et des restrictions clarifiant le sens. Elles modélisent un domaine de façon plus profonde avec plus de restrictions basées sur la sémantique du domaine.

II.9 Cycle de vie d'une ontologie

Le développement des ontologies s'appuie sur le même principe appliqué au génie logiciel vu qu'elles sont considérées comme des composants logiciels dans les systèmes. Les activités liées au cycle de vie des ontologies regroupent à la fois celles liées aux activités de gestion de projet telle que : [Ban,10]

- Planification,
- Contrôle,
- Assurance qualité,

Et d'autre part, des activités de développement :

- Spécification,
- Conceptualisation,
- Formalisation.

Ajoutées à cela, des activités transversales de support telles que l'évaluation, la documentation et la gestion de la configuration.

Ainsi, son cycle s'articule autour de :

- Étape de détection et spécification des besoins,
- Étape de conception,
- Étape d'utilisation,
- Étape d'évaluation,
- Étape d'évolution et de maintenance.

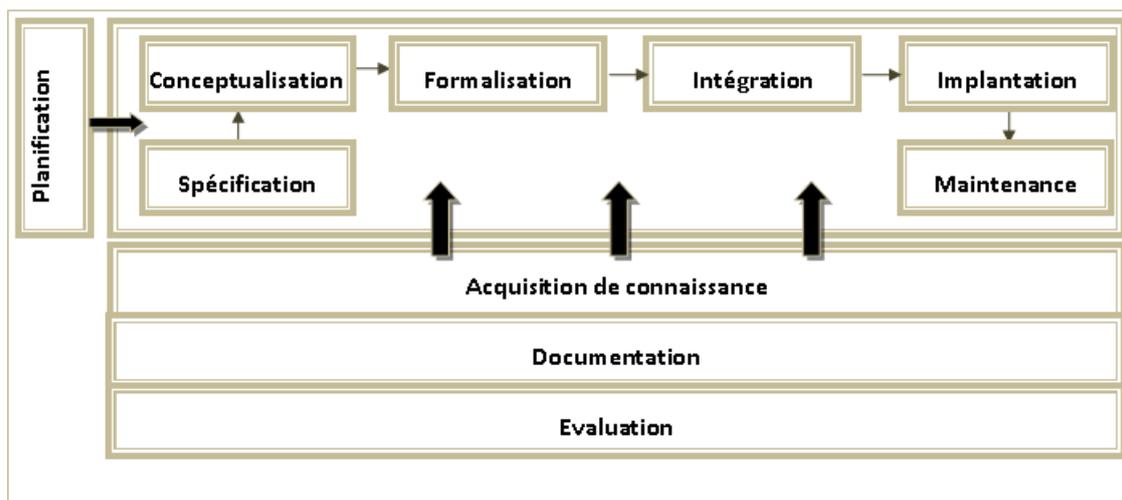


FIGURE II.5 – Cycle de vie des ontologies.

II.10 Processus de construction d'une ontologie

Cette phase de cycle de vie est considérée comme la phase motrice dans le développement. Elle fait collaborer des experts du domaine de connaissance, des ingénieurs de la connaissance, voire les futurs utilisateurs de l'ontologie. Cette collaboration ne peut être fructueuse que si les objectifs du processus ont été clairement définis.

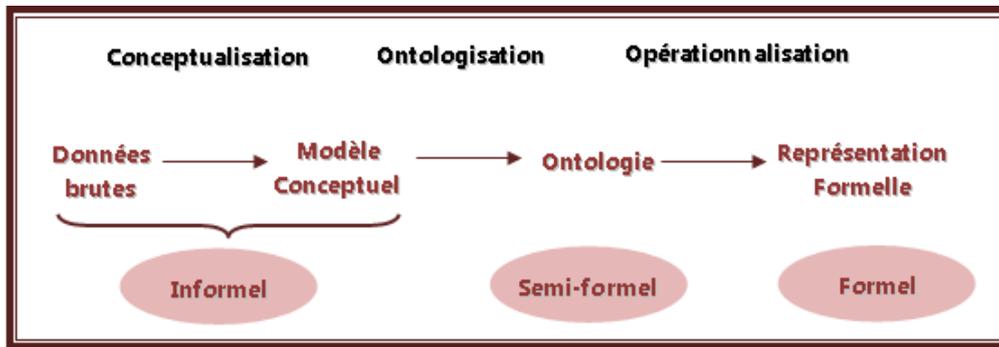


FIGURE II.6 – Processus de construction ontologies.

II.10.1 Spécification (évaluation des besoins)

Dans cette phase de développement, on se pose des questions à lesquelles on doit trouver les éléments de réponse qui nous mènent à spécifier nos besoins, ces questions peuvent être : Quel est le domaine que l'ontologie va couvrir ? À quoi cette ontologie va servir ? À quels types de questions les informations de l'ontologie doivent fournir des réponses ? Qui va utiliser et maintenir l'ontologie ?, etc. Ces réponses peuvent être modifiées au cours du processus.

II.10.2 Conceptualisation

Elle permet d'aboutir à un modèle informel, sémantiquement ambiguë et généralement exprimé en langage naturel. Dans cette phase, on doit dégager les concepts et les relations les reliant à partir des données brutes[Bel,15].

II.10.3 Ontologisation

C'est une formalisation partielle, sans perte d'information, du modèle conceptuel obtenu dans l'étape précédente. Cela permet de faciliter sa représentation ultérieure dans un langage complètement formel et opérationnel. Le modèle obtenu est souvent qualifié de semi-formel[Bel,15].

II.10.4 Opérationnalisation

C'est une spécification informatique des opérations applicables aux concepts dans un langage opérationnel. Cette phase doit intégrer des outils permettant d'opérationnaliser l'ontologie[Bel,19].

II.11 Méthodes et méthodologies de construction d'ontologies

II.11.1 Démarches de construction d'une ontologie

On peut trouver trois démarches :

- **Ascendante** : débute par les concepts les plus spécifiques aux concepts plus génériques,
- **Descendante** : débute par les concepts les plus génériques aux concepts plus spécifiques,
- **Mixte** : combinaison des deux stratégies, on commence par un concept connu et l'étendre vers un concept générique ou spécifique..

II.11.2 Méthodologies de construction

La méthode la plus célèbre de construction d'ontologie et celle d'Uschold et Grüninger :

(a). **Méthodologie d'Uschold et Grüninger [Usc et Gru,96]** : elle repose sur les points suivants :

- Identifier l'objectif souhaité et spécifier le domaine concerné,
- Construire l'ontologie et pour cela définir les concepts, les relations clés et produire des définitions textuelles précises et non ambiguës de ces concepts,
- Evaluer le résultat,
- Documenter le modèle en éditant des recommandations précises pour chaque étape .

(b). **La méthode Methontology [Fer et al,97]** : elle est utilisée pour :

- Construire le glossaire des termes qui seront inclus dans l'ontologie, préciser leurs définitions en langage naturel, identifier leurs synonymes et leurs acronymes,
- Construire des taxonomies de concepts,
- Construire des diagrammes de relations binaires,
- Construire le dictionnaire de concepts qui inclut, pour chaque concept, ses attributs d'instance, ses attributs de classe et ses relations,
- Décrire en détail chaque relation binaire,
- Décrire en détail chaque attribut d'instance,
- Décrire en détail chaque attribut de classe,
- Décrire en détail chaque constante,

- Décrire les axiomes formels,
 - Décrire les règles utilisées pour contraindre le contrôle et pour inférer des valeurs aux attributs.
- (c). **Méthodologie de Guarino et Welty** : cette méthode est une étape qui permet la vérification et la correction d'une structure ontologique construite un peu anarchiquement [Wel et Gua,01].
- (d). **Méthode ARCHONTE (ARCHitecture for ONTological Elaborating)** : proposée par Bruno Bachimont [Bac,00], consistant à :
- Choisir les termes pertinents du domaine et normaliser leur sens en précisant les relations de similarités et de différences que chaque concept entretient avec ses concepts frères et son concept père,
 - Formaliser les connaissances, c'est-à-dire ajouter éventuellement des propriétés et des axiomes à des concepts,
 - Opérationnaliser dans un langage de représentation des connaissances.

II.12 Langages d'ontologie

Pour que l'ontologie soit exploitée et partagée par un grand nombre d'utilisateurs, elle doit être obligatoirement représentée dans un langage qui peut être utilisée par les applications et les plateformes et doit répondre aux exigences des utilisateurs potentiels de cette ontologie.

II.12.1 Langages traditionnels

Au début, les langages de représentation des ontologies étaient fondus sur l'IA (vers les années 90). Ils se basaient principalement sur les formalismes de représentation des connaissances, tels que :

- La logique du premier ordre tel que KIF,
- Les frames combinés avec la logique du premier ordre tel qu'Ontolingua, OCML et Flogic,
- La logique de description telle que Loom [Kha,09].

II.12.2 Langage web standards ou basé sur xml

Plusieurs langages de description des ontologies ont vu le jour. Ils regroupent des langages tels que le langage de représentation des hiérarchies XML (eXtensible Markup Language), le langage RDF

(Resource Description Framework), le langage DAML+OIL (Darpa Modeling Language of Ontology+ Ontology Inference Layer) et le langage OWL (Ontology Web Language). Ces langages offrent différents niveaux d'expressivité [Dan et al ,04], [Ton,07], [Hor et al,01].

(a). **XML**

XML est utilisé pour la description des documents semi structurés. Il ne permet pas d'imposer des contraintes sémantiques à la signification des documents décrits. Cependant, XML Schéma permet de définir des balises ainsi que l'agencement de celles ci autorisées pour définir la validité d'un document XML.

(b). **RDF et RDFs**

- **RDF** : « Resource Description Framework», est un langage basé sur la syntaxe de XML permettant la description des objets appelés aussi ressources et de les reliant sur le web. RDF permet de définir, sous forme de graphes de triplets, des données ou des métadonnées.
- **Le RDFs (RDF schéma)** : c'est l'extension de RDF intégrant des classes, sous-classes, propriétés, sous-propriétés et des règles d'héritage de propriétés.

RDF et RDFs fournissent les langages de base pour le web sémantiques.

(c). **DAML+OIL** : désigne la fusion de deux langages de représentation des connaissances DAML et OIL basés essentiellement sur les logiques de descriptions, ils ont été proposés comme langage de description d'ontologies.

Le but ce langage est d'étendre RDFS en lui ajoutant des primitives plus expressives pour la définition des classes et des propriétés d'une ontologie.

(d). **OWL (Ontology Web Language)** : utilisés pour palier aux limites des RDFs touchant l'hierarchie des sous-classes et de sous propriétés avec restriction du domaine, il existe plusieurs cas particuliers d'utilisation du web sémantiques qui nécessitent une grande expressivité, ce type d'extension comprend :

- La disjonction,
- combinaison booléenne des classes,
- les restrictions de cardinalités et les caractéristiques des propriétés.

Créé en 2001 par le W3C, OWL est un nouveau langage d'ontologies, c'est l'héritier du langage DAML+OIL. Il se base sur RDFs. OWL est doté de trois sous langages offrant des capacités d'expression croissantes.

- **OWL Lite** : sous langage d'OWL DL, il définit des concepts simples d'OWL.
- **OWL Full** : autorise le mélange arbitraire avec le schéma RDF. Il est plus complet qu'OWL Lite et OWL DL et il est utilisé pour la transformation des concepts des diagrammes de classe UML.
- **OWL DL** : est un sous langage de OWL Full et se base sur la logique descriptive. Il définit les concepts plus complexes comme le complément, l'intersection, l'union, etc.

La puissance d'OWL lui permet de :

- Vérifier la cohérence d'une ontologie et d'une connaissance,
- Vérifier la présence de relations non voulues entre classes,
- Classer automatiquement les instances en classes.

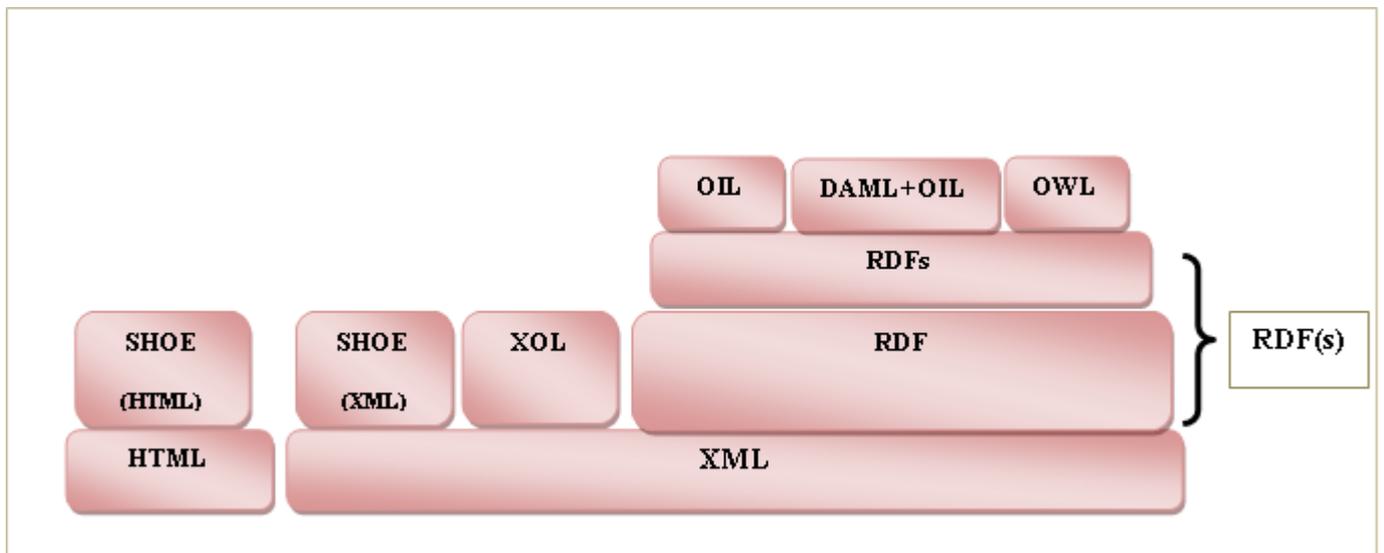


FIGURE II.7 – Langage d'exploitation des ontologies[Gom,04].

II.12.3 Langages d'interrogation des ontologies

II.12.3.1 Protocole de requête SPARQL

Certes le processus de constructions de l'ontologie est très fastidieux et primordial, mais il faudra aussi, trouver des mécanismes pouvant interroger ces ontologies et extraire des connaissances. Des

protocoles de requêtes spécifiques au langage RDF construits sur des graphes de données sont utilisés pour l'interrogation, à savoir le SPARQL.

Le SPARQL (prononcer « sparkle », acronyme de SPARQL Protocol And RDF Query Language), recommandé depuis le 15 Janvier 2008 par le W3C. SPARQL est un langage de requête et un protocole qui permettant la recherche, l'ajout, la modification et la suppression des données RDF/RDFS disponible[Ghe,17].

II.12.4 Approches du raisonnement

Ces techniques s'inspirent des celles utilisées depuis le développement des algorithmes de décision pour les problèmes d'inférence.

Nous distinguons deux grandes classes d'algorithmes, la première est celle des algorithmes structurels, qui comparent la structure syntaxique des concepts. Et la seconde classe qui comprend les algorithmes de tableaux qui ont été proposés la première fois pas SchmidtSchauf et Smolka [Sch et Smo,88], et qui deviennent réellement un outil principal pour les problèmes de satisfaisabilité et de subsomption de concepts dans les logiques de Descriptions.

A). Algorithmes de subsomption structurelle

Cette famille d'algorithmes repose sur la comparaison structurelle entre les expressions de concepts. L'idée centrale est que si deux expressions de concept sont faites de sous-expressions, alors elles peuvent être comparées séparément, en comparant la sous-expression d'un concept avec toutes celles de l'autre.

B). Algorithmes de tableaux

Principalement, ces algorithmes visent l'adaptation de calculs aux analyses de la complexité. Les structures de tableaux obtenues en raisonnant avec un langage de LD donné sont soigneusement analysées, des vérifications redondantes dans les tableaux sont éliminées afin de donner une limite supérieure strictement sur la complexité de la méthode. L'idée principale du calcul de tableau est de vérifier si une formule donnée F est une conséquence logique d'une théorie donnée T. On essaye de construire, avec des règles de propagation convenables, le modèle le plus générique de T où F est faux.

II.13 Outils de développement d'ontologies

Les outils de développement des ontologies doivent permettre leur documentation et la disposition d'un support graphique pour les construire. On dispose de quelques outils aidant l'utilisateur pour les créer une ontologie et les exporter dans un langage informatisé, ainsi de les exploiter via d'autres API.

II.13.1 PROTÉGÉ

Développé par le Stanford Medical Informatics de l'Université de Stanford, il fournit un environnement graphique permettant l'édition, la visualisation et le contrôle (vérification des contraintes) d'ontologies. Son modèle de représentation de connaissances est issu du modèle des frames. Ce dernier contient des classes (pour modéliser les concepts), des slots (pour modéliser les attributs des concepts) et des facettes (pour définir les valeurs des propriétés et des contraintes sur ces valeurs), ainsi que des instances des classes. PROTÉGÉ¹ introduit la notion de méta-classe, dont les instances sont des classes. L'interface très complète ainsi que l'architecture logicielle extensible permettant l'insertion de plusieurs plug-ins offrant de nouvelles fonctionnalités, notamment des plug-ins pour gérer les représentations sous forme graphique [Kha,09].

II.13.2 ODE (ONTOLOGY DESIGN ENVIRONMENT)

Cet outil se charge de la construction des ontologies au niveau connaissances (Méthodologie METHONTOLOGY). Son objectif est d'assister l'ontologue tout au long du processus de développement, à partir de la spécification des exigences jusqu'à la mise en œuvre, avec des éventuelles évaluations dès que possible [Arp et al,03].

II.13.3 OntoEdit

L'institut AIFB de l'université de Karlsruhe a mis au point un environnement d'ingénierie ontologique qui l'a nommé ONTOEDI². Actuellement, il est commercialisé par la société Ontoprise GmbH. C'est un outil fondé sur le processus de développement d'ontologies, il suit les diverses étapes de la méthode de construction ONTO-KNOWLEDGE tout en mettant à la disposition de l'utilisateur plusieurs vues graphiques correspondantes aux différentes phases de conception de l'ontologie [Sur,02].

1. PROTÉGÉ est disponible à l'adresse suivante : <http://protege.stanford.edu/>.

2. ONTOEDI's XML-based Ontology representation Language.

II.13.4 WebOde

Un groupe dénommé Ontological Engineering du département d'intelligence artificielle de la faculté d'informatique de l'université polytechnique de Madrid ont développé une plateforme en ligne qui l'ont appelée WebOde³. C'est un éditeur qui assure le support de Methontology. L'éditeur d'ontologie de WebODE permet d'éditer et de naviguer dans les ontologies WebODE et il se base sur des formulaires HTML et des applets Java [Arp et al,03].

II.13.5 DoE

En 2002, deux chercheurs, R. Troncy et A. Isaac de l'Institut National de l'Audiovisuel ont développé DoE⁴(Differential Ontologies Editor) [Isa,95]. Plusieurs fonctionnalités sont offertes par l'éditeur DOE telles que, les interfaces de création, la modification et suppression de concepts et de relations, la représentation graphique de l'arbre ontologique, et des fonctionnalités de recherche et de navigation dans la structure créée. L'ontologie est documentée par des définitions encyclopédiques avec des synonymes et les principes différentiels en plusieurs langues [Ban et Cha,07].

II.13.6 Framwork jena

Apache Jena (ou Jena en bref) est une plateforme Java gratuite et open source pour la création des applications Web sémantiques. JENA est composé de différentes API interagissant ensemble pour traiter les données dans des documents RDF, RDFS, OWL et SPARQL. La plateforme fournit un moteur d'inférences permettant des raisonnements sur les ontologies [Ben,17].

II.14 Conclusion

Nous avons évoqué dans ce chapitre, l'apport de l'ingénierie des connaissances aux systèmes de recommandation qui ont été abordé dans le chapitre précédent. Cet apport est mis en évidence par la possibilité d'envisager une approche orientée connaissances, telle que, les ontologies, le Web sémantique, Linked Data, les graphes de connaissances, la logique de description et d'autres qui permettent de mieux représenter les utilisateurs ou/et les items et de mieux capturer les interactions entre eux.

Nous avons présenté également les systèmes à base de connaissances à savoir leurs rôles, objectifs...etc., ainsi que la manière d'inférence sur ces connaissances qui repose principalement dans

3. <http://webode.dia.fi.upm.es/WebODEWeb/index.html>

4. DOE est un logiciel libre sous licence GPL disponible à l'adresse : <http://opales.ina.fr/public>.

notre projet sur le raisonnement à base de cas RàPC.

Notre contribution était centrée "ontologies", donc il était bien évident de citer les concepts de base relatifs à celles-ci. Nous avons clarifié le principe d'ontologies en présentant certaines définitions. Les composantes de l'ontologie ainsi que leurs différentes classifications, cycle de vie et leurs méthodologies de construction ont été exposés par la suite.

Et pareillement, les différents langages d'ontologie tel que les langages de représentation, les langages d'interrogation, et de raisonnement ainsi les outils de développement d'ontologie ont été présentés.

Dans le chapitre qui suivra, nous allons présenter un certain nombre des travaux connexes aux systèmes de recommandation.

Chapitre III

État de l'art des systèmes de recommandation

CHAPITRE III

État de l'art des systèmes de recommandation

Sommaire

III.1 Introduction	48
III.2 Les travaux connexes aux systèmes de recommandation	48
III.3 Comparaison	55
III.4 Synthèse	57
III.5 Conclusion	58

III.1 Introduction

Les systèmes de recommandation appliquent plusieurs techniques d'exploration de données telles que le filtrage collaboratif, le filtrage basé sur le contenu, les techniques hybrides ou les méthodes basées sur les connaissances en fonction des caractéristiques du domaine, de la qualité des données disponibles et des objectifs commerciaux. Dans ce chapitre nous allons dénombrer un certain nombre de travaux connexes aux systèmes de recommandation.

III.2 Les travaux connexes aux systèmes de recommandation

« **Information Lens System** » [Mal et al,87] est le premier système de recommandation développé. A l'époque, l'approche la plus commune pour le problème de partage d'informations dans

l'environnement de messagerie électronique était la liste de distributions basée sur les groupes d'intérêt. La technique du filtrage est utilisée dans un sens plus général qui consiste à sélectionner les choses à partir d'un ensemble plus large de possibilités (filtrage positif : sélection).

Tapestry [Gol et al,92] est parmi les premiers systèmes à émettre l'hypothèse qu'une intervention humaine pourrait améliorer les résultats de filtrage d'items, des e-mails dans leur cas. L'objectif est donc d'améliorer les résultats de filtrage d'e-mail en proposant le système de filtrage collaboratif Tapestry. Le principe, novateur à l'époque, est de s'appuyer sur les précédentes requêtes d'utilisateurs du système Tapestry et sur leurs appréciations en fonction des documents qu'ils ont jugés pertinents afin de filtrer les résultats proposés à de futurs utilisateurs qui feront la même requête.

Dans **GroupLens** [Res et al,94], les auteurs proposent une approche assez similaire en se focalisant sur la recherche documentaire par le biais du système GroupLens. Notons cependant que comme lors de l'utilisation de l'outil Tapestry, GroupLens se fonde sur une appréciation humaine active d'article. Le terme d'actif signifie dans ce cas que l'utilisateur doit faire la démarche d'annoter un document afin de construire des classes de documents pertinents. Ces approches ne sont donc pas fondées sur un apprentissage implicite du comportement d'utilisateurs comme dans le cas d'Amazon [Lin et al,03].

Les applications web ont également fait l'objet d'intégration des systèmes de recommandation. Les travaux de [Lie,95] présentent le système **Letizia** qui vise à aider les utilisateurs à naviguer sur le Web. Son authenticité réside su fait qu'il soit l'un des premiers à proposer des recommandations via un système de filtrage basé sur le contenu sans intervention explicite de l'utilisateur. Plus formellement, le système utilise des techniques statistiques pour créer un profil d'utilisateur basé sur des outils d'extraction de descripteurs ou de mots-clés. Les informations ciblées sont les liens vers les pages que les utilisateurs visitent.

Encore, dans le même système de filtrage par contenu, [Paz et al,97] proposent d'identifier les sites pertinents des utilisateurs dans un domaine fixe, en démontrant la qualité du classificateur **bayésien** [Dud et Har,73] par rapport aux autres systèmes de classification. Les données apprises sont affichées en premier dans un espace vectoriel de cooccurrences des mots selon les pages Web dans lesquels elles apparaissent. Le classificateur bayésien est utilisé pour construire des classes d'individus. Les auteurs suggèrent également de saisir des informations sémantiques à l'aide de WordNet [Mil et al,90], ce qui augmente les performances.

Le système **WebMate** [Che et Syc,98] permet à ses utilisateurs de rendre la navigation Web plus efficace, en proposant notamment un agent les guidant dans leurs recherches. La méthode utilisée est vectorielle et propose de construire, dans un premier temps, les profils utilisateurs avec un système Saltonien reposant sur une pondération de type Tf-Idf. Les mots clés représentatifs des choix des utilisateurs sont alors extraits. Des lors, l'originalité de la méthode est de réintroduire dans la requête de l'utilisateur des résultats jugés pertinents par d'autres utilisateurs proches de lui, utilisant le principe du retour de pertinence (Relevance Feed-back). Finalement, lors de la première utilisation du système par un utilisateur, son profil est alors vide et aucune recommandation ne peut lui être faite. Notons que leur système a été évalué avec les moteurs de recherche Altavista et Lycos.

L'auteur Cunningham et ses collègues [Cun et al,00] ont proposé **Websell**, un système d'aide à la navigation sur le Web s'intéressant à la vente en ligne. Le principe est de produire un assistant de vente en ligne pour un utilisateur en ciblant dans un premier temps ces choix et en lui recommandant dans un second temps des produits relatifs à son profil. Le projet Websell utilise notamment un filtrage collaboratif.

[Fis et Ste,91] ont proposé le système **Infoscope** afin de faire la recommandation à des utilisateurs de groupes de discussions en fonction des requêtes qu'ils soumettent. Le principe est de construire une représentation sémantique d'un utilisateur en fonction de ses préférences en s'appuyant sur une structure d'arbre afin de catégoriser les articles des NewsGroups.

[She et Mae,93] présentent le système **Newt**, se basant sur le retour de pertinences d'utilisateurs (relevance feedback) et d'un algorithme de classification afin d'effectuer un filtrage personnalisé d'informations. Le retour de pertinence utilisateur a été choisi afin de permettre une mise à jour dynamique des choix de l'utilisateur. Un algorithme génétique, a été utilisé pour sa capacité de sortir des minimums locaux lors des phases de mutations.

[Moo et Roy,99], partent de la constatation que les systèmes de recommandations collaboratifs peuvent être limités, ils ont proposé le système **Libra** qui est destiné à faire de la recommandation de livres à partir du site Web d'Amazon. Le processus de recommandation de Libra se déroule en plusieurs phases, telles que, la phase d'extraction d'informations, la phase d'évaluation de l'utilisateur, une phase d'apprentissage de profils et la phase de recommandation.

RARE (Recommender system based on Association RulEs) [Ben et al,06] est un système de recommandation de cours. RARE se base sur le principe de faire profiter ses utilisateurs d'une expérience collaborative d'étudiants. Vu que le volume des données mémorisées des étudiants qui ont

été enregistrés et qui ont achevé leurs études est important, des techniques de forage de données sont appliquées sur cet ensemble de données. Elles permettraient de capturer le comportement des étudiants dans leurs choix de cours et, par conséquent, révéler des relations cachées entre les cours suivis.

[Alv et al ,14] ont proposé un nouveau système de recommandation basé sur la qualité. Ce système utilise la qualité des items pour estimer leur pertinence, cette mesure de qualité est prise comme un nouveau facteur à considérer dans le processus de recommandation. Cela aide les utilisateurs à accéder aux ressources de recherche pertinentes. Ce système de recommandation est développé en utilisant une approche linguistique floue et il a été testé de manière satisfaisante dans une bibliothèque numérique universitaire.

Le système **Cora** [Flo et al,20] est un système conversationnel qui recommande des recettes alignées sur les habitudes alimentaires et les préférences actuelles de ses utilisateurs. Les utilisateurs peuvent interagir avec Cora en cliquant sur des boutons ou en écrivant du texte en langage naturel. De son côté, Cora peut engager les utilisateurs via un dialogue social ou aller droit au but. Une expérience a été menée pour évaluer l'impact du style conversationnel de Cora et du mode d'interaction avec les utilisateurs sur la qualité perçue de l'interaction et du système, et sur l'intention de l'utilisateur de cuisiner les recettes recommandées. Les résultats montrent qu'un système de recommandation conversationnel qui engage ses utilisateurs à travers un dialogue social améliore la qualité perçue de l'interaction ainsi que celle du système.

[Jar et al ,16], proposent dans leur étude une architecture d'un système de recommandation basé sur le contenu visant à sélectionner des examinateurs (experts) pour évaluer des propositions de recherche ou des articles. Ils introduisent un cadre algorithmique complet supporté par diverses techniques de recherche d'informations. Ils proposent une méthodologie complète qui explore les concepts de données, d'informations, de connaissances et de relations entre eux pour soutenir la formation d'une recommandation appropriée. Chaque partie essentielle du système est traitée comme un module séparé, tandis que chaque couche prend en charge une certaine fonctionnalité du système. La modularité de l'architecture facilite la maintenabilité.

Entree est un exemple de ce qu'on appelle un système FindMe [Bur et al,97]. Les systèmes **FindMe** sont considérés comme des systèmes de recommandations avec les caractéristiques distinctives suivantes :

1. Ils sont principalement basés sur des connaissances,

2. Possibilité de critiquer une suggestion de la part de l'utilisateur,
3. Ils classent les produits en fonction des objectifs attendus d'utilisateurs.

[**Mor et al,13**] proposent dans leur système de recommandation des activités touristiques, de ne questionner l'utilisateur que sur quelques concepts généraux mais suffisamment significatifs pour représenter les principaux centres d'intérêt des touristes (par exemple Plage, Shopping, Culture, Gastronomie, etc.). L'avantage de demander directement à l'utilisateur d'exprimer ses goûts pour construire son profil est que cela permet d'obtenir des recommandations précises, exemple : s'il aime la plage, on lui recommandera des activités près de la mer.

D'autres SdR touchant le domaine du multimédias ont été ainsi proposés, tels que :

Les travaux des chercheurs dans [**Ter,93**] et [**Har,94**] s'appuyaient sur les applications de filtrages collaboratifs, ils ont proposé le système **MovieLens**, un système de recommandation de films, et le système Jeter recommandant des farces.

Le système **Ringo** de [**Sha et Mae,195a**] s'intéresse quant à lui aux albums musicaux en utilisant le principe du filtrage collaboratif mais utilise également un système d'annotation manuel des items. Les auteurs proposent alors une comparaison de quatre algorithmes de recommandation, le premier étant une recommandation sans filtrage collaboratif et les trois autres proposant différents scores de similarités s'intéressant à la proximité dite "sociale" entre individus comparant alors un utilisateur à un autre. Ainsi, pour produire une recommandation, un utilisateur est comparé à l'ensemble des utilisateurs. Les utilisateurs dont le score obtenu est supérieur à un certain seuil sont retenus.

[**Hau et Fre,01**] ont proposé le système **FlyCasting** recommandant des radios en lignes en se basant sur le système de recommandation collaboratif.

[**Bla et al,08**] présentent un système **CBF** pour recommander des programmes de télévision numérique. La spécificité de leurs travaux est que le profil d'utilisateur est modélisé par un sous-graphe extrait à partir d'une base de connaissance associée à un domaine, i.e. celui de la TV dans leur cas. Un tel profil d'utilisateur contient des instances (des programmes) pertinentes, leurs principaux attributs et les genres sous lesquels ces programmes sont classés dans l'ontologie. L'intégration des différents types de nœuds dans le profil permettra un filtrage plus fin puisque l'on dispose de plus d'information.

L'**hybridation** des approches de recommandation a été grandement exploitée afin de palier aux inconvénients des celles-ci (utilisées seules). Les travaux de [Cla et al,99] reposaient sur ceux de [Vog et al,97] où ils ont montré que la combinaison linéaire de scores retournés par différentes approches de recherche d'information pouvait améliorer significativement les résultats.

Dans l'article [Ric,79], l'auteur propose le système **Grundy** qui visait à l'époque la construction de profils utilisateurs sans parler nécessairement de recommandation mais plutôt de personnalisation. Le principe est de construire des profils "stéréotypé" et d'activer ces derniers en fonction du comportement d'un utilisateur, selon le type de requête qu'il formule. Cette approche reste toujours utilisable afin de construire des profils initiaux d'utilisateurs comme dans [Kob et al,01], où les auteurs proposent une approche collaborative favorisant ainsi les relations entre clients de sites en ligne. Ce travail a été étendu par un large panel d'approches de recommandation telles que, les approches hybrides et celles fondées sur le contenu.

Dans l'article de [Rha et al,20], les auteurs proposent un système de recommandation de **bibliothèque** hybride piloté par les usagers qui utilise des techniques d'apprentissage automatique pour recommander et aider aux opérations d'acquisition et de désherbage en extrayant et en analysant les opinions et les évaluations des utilisateurs. Le système de recommandation est appliqué et validé dans un cas réel de bibliothèque nationale en utilisant la bibliothèque numérique d'Amazon et le catalogue de la bibliothèque comme source de données.

[Sch et al,01] s'intéressaient à l'**apprentissage de profils** en se basant sur un filtrage collaboratif et une analyse du contenu du profil de l'utilisateur. A noter que cette approche peut être apparentée à une méthode hybride, utilisant une méthode de filtrage collaboratif et une méthode basée sur le contenu.

[Bas et al,01] qui proposent une méthode à base de règle regroupant dans un même classificateur des notions basées sur le contenu et collaboratives. A noter que la partie collaborative est dans cette approche passive, les auteurs récupérant des informations sur l'utilisateur via le Web, sans interaction avec eux.

De même, [Pop et al,01] et [Sch et al,02] proposent une approche semblable en utilisant un modèle probabiliste afin de combiner les méthodes dans un modèle unique. Les auteurs proposent d'utiliser ici l'analyse latente probabiliste (PLSA).

Par ailleurs, [Li et Kim,03] s'intéressent également à cette problématique d'hybridation en proposant une approche se fondant sur la construction de classes, en résolvant également le problème du démarrage à froid. Cette approche se décompose en trois points qui sont l'utilisation d'un algorithme de **clustering**, le calcul des distances entre les différents groupes et fournir une prédiction à un utilisateur.

Le système de recommandation **Entree** [Bur et al,97] a été renforcé par un filtrage collaboratif créant ainsi un hybride en cascade basé sur les connaissances/collaboratif.

Plusieurs auteurs ont intégré les approches de l'ingénierie des connaissances aux systèmes de recommandation et ils ont montré que cet apport est fructueux. Nous allons citer dans la section suivante quelques travaux de recherche sur les systèmes de recommandation à base de connaissances et plus précisément les ontologies.

Les travaux les plus anciens s'appuient notamment sur les **ontologies** du domaine en exploitant les relations hiérarchiques et sémantiques entre les concepts, et le calcul de similarité sémantique basée sur la structure ontologique [Mid et al,01];[Mid et al,04];[Can et al,08];[Sie et al,10]; [El-D et al,12]; [Car et al,12]; [Har et al,13]; [Mor et al,13];[Rod et al,15].

Avec la standardisation des protocoles du Web sémantique (triplets RDF, langages OWL et SPARQL, etc.), il est devenu facile de représenter les connaissances et ses liaisons.

Récemment, les auteurs [Di et al,12b];[Di et al,12a]; [Ost et al,13]; [Di et Ost,15]; [Mus et al,16a]; [Pal et al,18b];[Pal et al,18a];[Pal et al,20], ont proposé des approches s'appuyant sur **les données liées (linked data)** appliquées aux systèmes de recommandation.

les auteurs [Mid et al,01]; [Sie et al,10]; [Mor et al,13]; [Car et al,12]; [Rod et al,15], les auteurs ont utilisé des ontologies comme support à la représentation des profils utilisateurs (resp. des profils des items). Ils ont défini deux types d'approches : l'une consiste à considérer les items comme des instances, la seconde associe un ensemble de concepts/instances de la base de connaissances aux éléments dont on veut définir le profil (items et/ou utilisateurs).

Les auteurs [Di et al,12b]; [Ost et al,13]; [Pal et al,20], visaient à générer des **mappings** visent à générer des mappings entre les items et les objets au sein d'une ontologie, comme d'ailleurs la plupart des approches de la littérature.

Les auteurs de [Sie et al,10] définissent le profil utilisateur comme un ensemble de nœuds dont chacun est représenté sous la forme d'une paire, $\langle C_j, IS(C_j) \rangle$, où C_j est un concept défini dans l'ontologie et $IS(C_j)$, i.e., Interest Score, est le taux d'intérêt d'un utilisateur pour le concept C_j . Ils utilisent ensuite ces profils pour estimer la similarité entre utilisateurs dans une approche hybride. Ce formalisme se base sur le travail de [Middleton et al., 2004] qui adopte des triplets $\langle \text{utilisateur}, \text{thème}, \text{taux d'intérêt} \rangle$ pour modéliser l'intérêt des utilisateurs pour des articles scientifiques de différents domaines.

Dans [Car et al,12], les auteurs proposent un autre moyen de représenter le profil utilisateur, nommé **Recon**, ce profil est composé d'un vecteur associé à chaque utilisateur, mais dont chaque dimension traduit un degré d'intérêt pour une instance (un item particulier) et non pas un concept. L'avantage d'encapsuler les instances dans le profil utilisateur est que chaque nouvelle évaluation affine la définition du profil. Ainsi une part du filtrage est déjà réalisée puisqu'il suffira d'ordonner les composantes du vecteur pour déterminer les items (instances) qui ont le plus de poids et en faire la recommandation.

Et beaucoup plus, plusieurs travaux ont été proposés intégrant notamment les ontologies de différentes manières et dans différentes phases du processus de recommandation. Bien évidemment, nous ne pouvons pas détailler tous les travaux connexes à ce domaine, Nous nous contentons de citer quelques références de base telles que [Har et al,13]; [Al et al,11]; [Car et al,12]; [Ben-I,17]; [Har et al,13]; [Pes et al,09].

D'autres combinant les ontologies et éventuellement les graphes de connaissances et/ou linked data, sont donnés par les références suivantes : [Ber,06]; [Aue et al,07]; [Pas,10a]; [Hei et Hay,10]; [Di et al,12b], [Di et al,16].

III.3 Comparaison

Nous avons tenté de procéder à une comparaison des approches des systèmes de recommandation en se basant sur quelques critères. Cette comparaison est présentée dans le tableau suivant.

Approche	Objectifs	Algorithme	Technique	Sémantique
[Gol et al,92] Système Tapestry	Recommandation / Comparaison	filtrage collaboratif / filtrage de l'information	Annotation	Non
[Che et Syc,98] Système WebMate	Recommandation / Sélection	Filtrage collaboratif	Relevance Feed-back	Non
[Cun et al,00]	Choix / Recom- mandation	Filtrage collaboratif	/	Oui
[Ric,79] Système Grundy	Personnalisation	Filtrage collaboratif	/	/
[Sch et al,01]	Analyse	Filtrage collaboratif / filtrage basée sur le contenu	apprentissage	Non
[Fis et Ste,91] Système Infoscope	Recommandation	Filtrage collaboratif	/	Oui
[She et Mae, 93] SystèmeNewt	Choix	filtrage personnalisé d'informations	Relevance feedback	Non
[Lie,95] Système Letizia	Recommandation / Choix	filtrage basée sur le contenu	Extraction de descripteurs ou de mots-clés	Non
[Bas et al,01]	Regroupement	Méthode à base de règle / Hybride	/	Non
[Alv et al ,14] Système linguistique floue	Recommandation	Basé sur la qualité	/	Non

[Rha et al,20] Système bibliothèque	Recommandation	Filtrage Hybride	Apprentissage automatique	Non
[Ben et al,06] Système de cours RARE	Recommandation / Choix	Filtrage collaboratif	/	Non
[Sha et Mae,95a] Système Ringo	Recommandation / Comparaison	filtrage collaboratif	Annotation	Oui
[Jar et al ,16] Système des examineurs (experts)	Recommandation / Recommandation	basé sur le contenu / connaissance	/	/
[Bur et al,97] Système Entree	Recommandation	Filtrage basée sur connaissance	Raisonnement à base de cas	Oui
[Bur et al,97] [Bur,99] [Bur,00] Système EntreeC	Recommandation	Filtrage basée sur connaissance /Collaboratif (hybride)	/	Non
[Bla et al,08] Système de Télévision numérique	Recommandation	Filtrage basée sur connaissance	Raisonnement à base de cas / Ontologie	Oui

TABLE III.1 – Comparaison entre les systemes de recommandation .

III.4 Synthèse

Les systèmes collaboratifs et basés sur le contenu sont faciles à configurer car seules les informations de base sur les noms d'éléments, les descriptions et les représentations graphiques sont nécessaires. Les méthodes de filtrage hybride (HF) peuvent être plus précises que les méthodes conventionnelles ; cependant, la mise en œuvre efficace de telles solutions peut être très difficile pour des problèmes complexes. Les deux approches, si elles ne sont pas formées avec de nombreux exemples (évaluations des éléments ou modèle de préférences des utilisateurs), fournissent de mauvaises re-

commandations.

Cette limitation a principalement motivé une quatrième approche, basée sur les connaissances, qui tente de mieux utiliser les connaissances préexistantes du domaine d'application pour construire un modèle plus précis nécessitant moins d'instances de formation.

Dans notre travail, nous avons opté pour l'intégration des ontologies comme modèle de connaissance en considérant une sémantique des recommandations pour palier aux limites des approches cités dans le début de cet état de l'art tels que le filtrage collaboratif, à base de contenu...etc.

III.5 Conclusion

Dans ce chapitre, nous avons exposé quelques travaux connexes aux systèmes de recommandation utilisant différentes techniques de recommandation. Dans la dernière section, nous avons mis le point sur les SdR à base de connaissances et plus précisément les ontologies.

Dans le chapitre suivant, nous détaillons notre vision orientée connaissances en présentant les aspects conceptuels et de réalisation de notre système de recommandation à base de connaissance reposant sur les ontologies, les outils utilisés,...etc.

Chapitre IV

Systeme à base de connaissances pour la recommandation de résolution de problèmes en maintenance informatique

CHAPITRE IV

Système à base de connaissances pour la recommandation de résolution de problèmes en maintenance informatique

Sommaire

IV.1 Introduction	61
IV.2 Présentation de l'application	61
IV.3 Proposition d'une approche basée Ontologie du SdR	62
IV.3.1 Architecture de l'approche	62
IV.3.2 Déroulement de l'approche	63
IV.3.3 Description du modèle ontologique	65
IV.3.4 Protocole d'évaluation des recommandations	66
IV.4 Aspects de modélisation	68
IV.4.1 Les cas d'utilisation	68
IV.4.1.1 Diagramme de cas d'utilisation de l'utilisateur	69
IV.4.1.2 Diagramme de cas d'utilisation de l'expert	70
IV.4.1.3 Diagramme de cas d'utilisation de l'Administrateur	71
IV.4.2 Les diagrammes de séquences	72
IV.4.2.1 Diagramme de séquence d'Authentification	72
IV.4.2.2 Diagramme de séquence d'identification et d'envoi du problème	73
IV.4.2.3 Diagramme de séquence de réception et résolution du problème	74

IV.4.3	Diagramme de classe	74
IV.5	Mise en œuvre de l'approche	75
IV.5.1	Environnement et outils de programmation	75
IV.5.1.1	IDE Netbeans	76
IV.5.1.2	Java	76
IV.5.1.3	API Jena	76
IV.5.1.4	Protégé	77
IV.5.1.5	WampServer	77
IV.5.1.6	L'API JFreeChart	77
IV.6	Réalisation de l'application(scénarios d'exécution)	77
IV.6.1	Raisonnement à base de cas	77
IV.6.2	Scénarios de la démarche décrite dans la section (IV.3.2)	78
IV.7	Conclusion	92

IV.1 Introduction

Dans ce chapitre, nous présentons l'architecture du système de recommandation à base de connaissances. Nous avons adopté une approche de conception fondée sur la modélisation UML. Nous allons décrire ainsi les principaux paliers de notre démarche pour la mise en œuvre d'une solution logicielle adéquate à ce contexte.

Afin d'atteindre notre objectif, nous avons mis en place une base de connaissances reposant sur un modèle ontologique intégrant les mécanismes de raisonnement à base de cas afin d'extraire les cas les plus similaires à un problème donné. Un protocole d'évaluation des recommandations proposées a été développé en se servant sur des mesures de similarité.

IV.2 Présentation de l'application

Nous avons développé un SdR destiné à résoudre les problèmes du dysfonctionnement du matériel informatique, autrement dit, procéder à sa maintenance.

Par définition, « la maintenance vise à maintenir ou à rétablir un bien dans un état spécifié afin que celui-ci soit en mesure d'assurer un service déterminé ». Elle regroupe ainsi les actions de dépannage et de réparation, de réglage, de contrôle et de vérification des équipements matériels (machines, véhicules, objets manufacturés, etc.) ou même immatériels (logiciels).

Dans le système de recommandation développé, La technique de maintenance permet l'acquisition de connaissances relatives au problème (configuration/description du problème à résoudre) en vue d'obtenir des propositions (recommandations) afin de résoudre le problème surgi.

Plusieurs classes de maintenances ont été définies telles, la maintenance corrective, palliative, réparatrice, préventive, systématique, prévisionnelle.

IV.3 Proposition d'une approche basée Ontologie du SdR

IV.3.1 Architecture de l'approche

Nous avons proposé l'architecture schématisée dans la figure suivante pour notre système de recommandation.

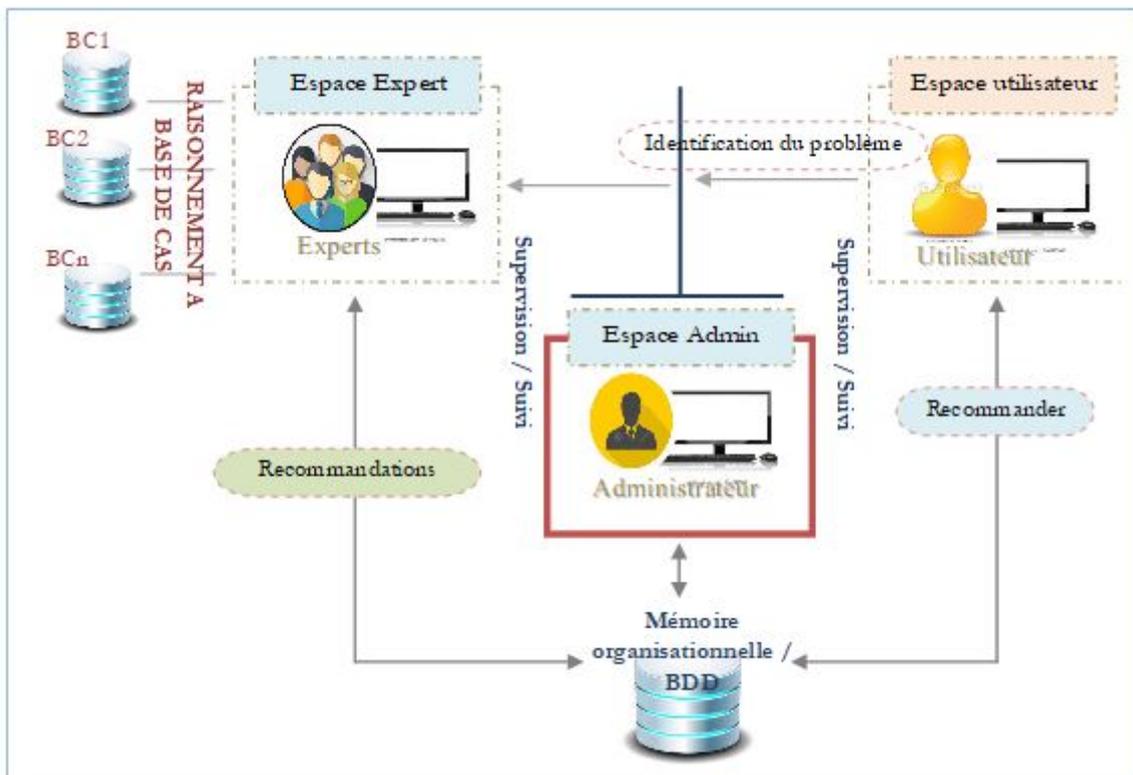


FIGURE IV.1 – Architecture de notre système.

IV.3.2 Déroulement de l'approche

Nous avons choisi un système de recommandation à base de connaissances, cette base est représentée par un modèle ontologique. Ainsi, pour évaluer la qualité des recommandations proposées, nous utilisons un protocole afin d'explicitier le calcul de similarité sémantique entre le problème posé et ceux trouvés, autrement dit, dans le vocabulaire habituel des SdR, entre l'item i demandé par un utilisateur j et l'ensemble I_k des items proposés.

L'utilisateur doit d'abord sélectionner la catégorie du problème à résoudre (selon son profil), puis attendre par la suite les différentes recommandations des experts. Cette session de recommandation se déroule en plusieurs phases qui seront décrites dans la figure suivante.

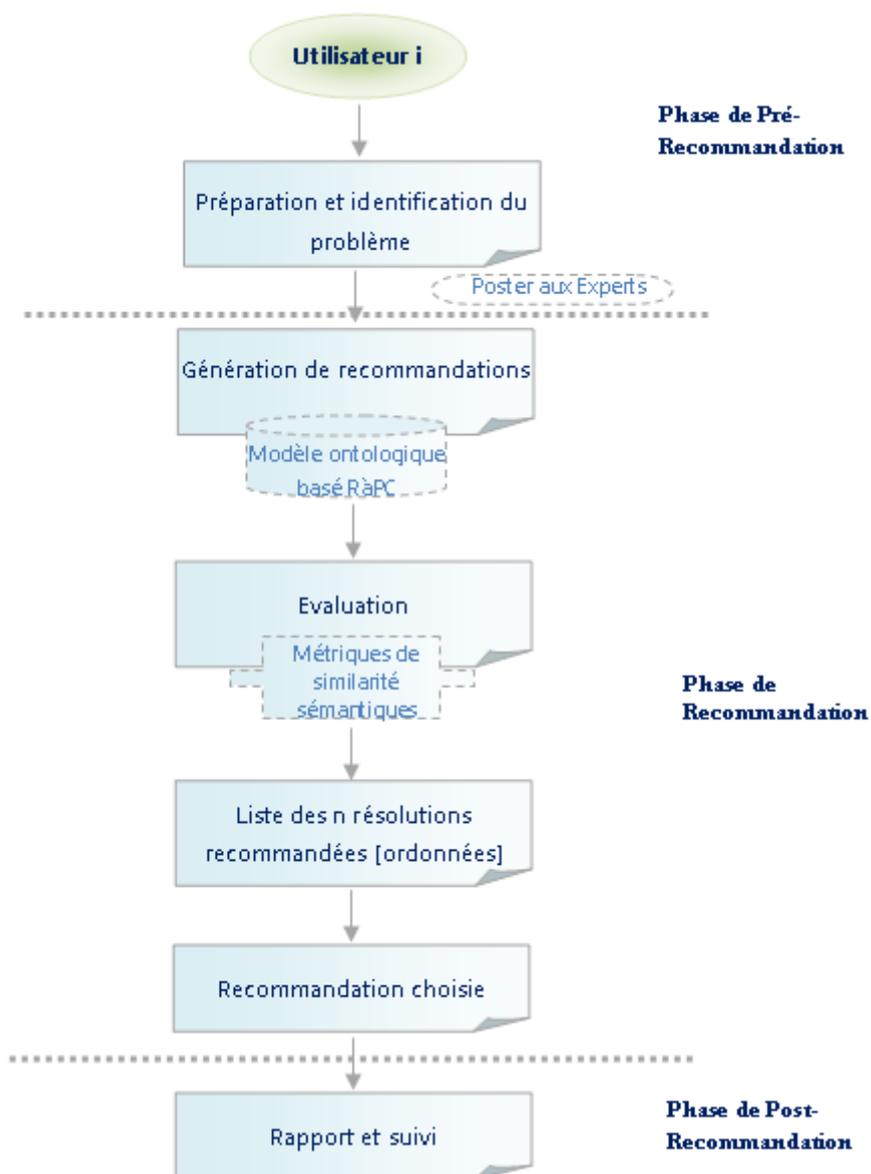


FIGURE IV.2 – Système de recommandation basé d'Ontologie.

Dans la section suivante, nous allons présenter les différentes interactions qui rentrent dans le cadre du processus de recommandation (Figure IV.2).

A). Phase de Pré-recommandation : ouverture d'une session de recommandation.

- Sollicitation des experts :
 - Une fois qu'un utilisateur *i* identifie le problème à résoudre, il consulte par la suite la liste des experts connectés intégrant la session de recommandation et sollicite ceux, destinataires de sa requête, cela devra se faire selon les profils des experts potentiels déjà stockés dans une base de données.
 - Une fois que la liste des experts est désignée, ces derniers seront contactés par courriers.
- Démarrage du processus de recommandation :
 - Tout expert sollicité par un utilisateur *i*, reçoit la demande de recommandation de celui-ci, et, décide par la suite d'accepter de collaborer à la recommandation ou pas.
 - Dans le cas d'acceptation, il se voit notifié par l'administrateur.

B). Phase de recommandation

- Proposition de recommandation :
 - Les experts proposent des résolutions pour le problème posé, l'élaboration de cette solution est fournie par l'expert lui-même, comme il pourra être assisté par un co-expert.
 - La recommandation est proposée en fonction d'un mécanisme d'inférence en recherchant les solutions similaires dans le modèle ontologique, on note que chaque expert est doté de sa propre base de connaissances.
 - La base de cas (connaissances) sera décrite en détail dans la section (IV.3.3).
- Organisation des recommandations proposées :
 - Toute opération exécutée dans une session de recommandation est notifiée au niveau de l'administrateur.
 - L'admin procède à une éventuelle épuration des recommandations proposées.
- Evaluation des recommandations :
 - Une fois que la liste des recommandations proposées par les experts est éventuellement épurée, l'administrateur évalue ces alternatives de solutions. Le protocole

adopté s'appuie sur des mesures de similarité afin que l'utilisateur ait plus de précision sur le taux exact de similarité de la recommandation proposée par rapport au problème posé, ceci lui permet de prendre la bonne décision sur la résolution à choisir. Ce protocole de similarité sera décrit dans la section (IV.3.4).

- Choix de la recommandation
 - Dès que l'utilisateur reçoive les différentes recommandations évaluées, il pourra avoir une vue globale sur celle la mieux adaptée à son problème.

C). Phase de Post-recommandation

- Clôture de la session de recommandation :
 - L'utilisateur pourra quitter la session avec la possibilité d'archiver la recommandation choisie afin de l'utiliser ultérieurement si nécessaire.

IV.3.3 Description du modèle ontologique

Les connaissances représentées par le modèle intègrent implicitement trois types d'ontologies [ontologie de domaine, ontologie de domaine d'application, ainsi que celle de tâche]. Ces types sont schématisés dans la figure suivante :

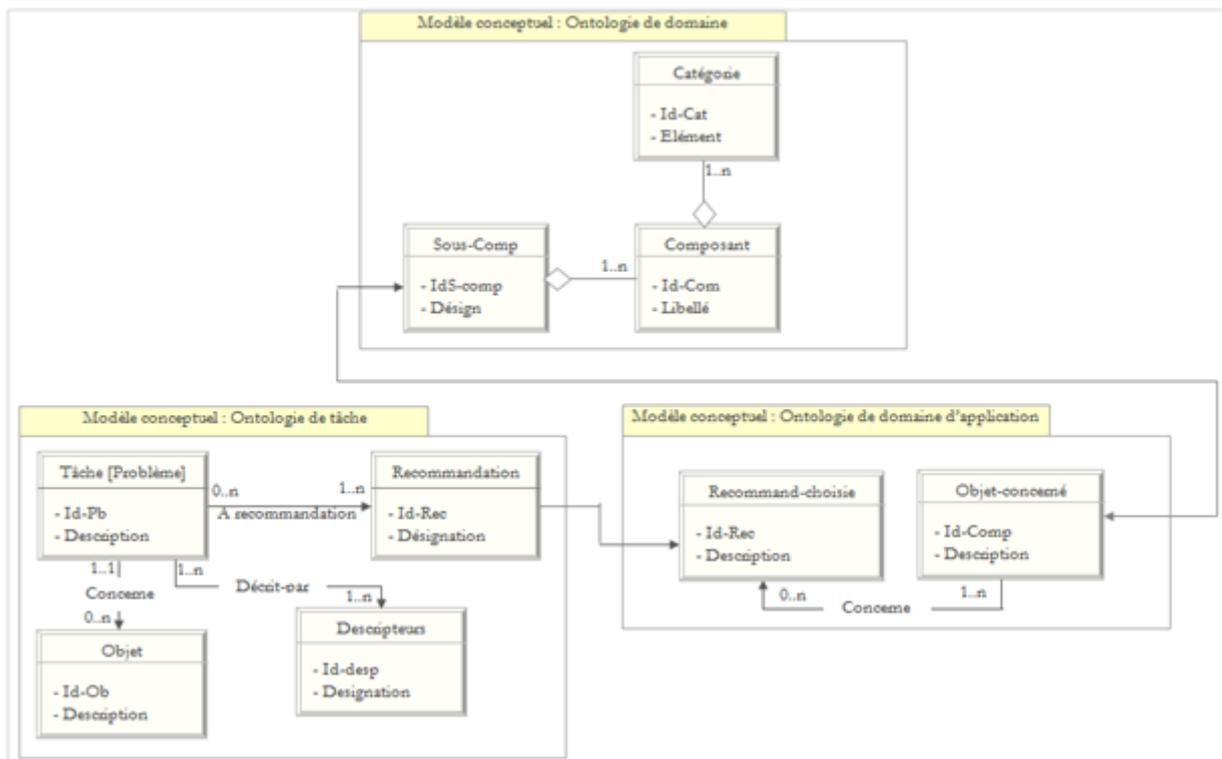


FIGURE IV.3 – Différents sous-modèles ontologiques.

IV.3.4 Protocole d'évaluation des recommandations

Ce protocole se base sur les similarités sémantiques et il devrait augmenter la crédibilité du processus de recommandation vis-à-vis de l'utilisateur, il doit l'amener aussi à :

- Bonne compréhension de la recommandation ;
- Plus de persuasion envers la recommandation ;
- Apprendre de nouvelles informations sur les items recommandés ;
- Avoir plus de confiances aux SdR et ;
- Bonne évaluation de la qualité de l'item recommandé.

Formulation

Comme nous avons cité précédemment, l'item recommandé dans notre approche représente la recherche d'un cas similaire au cas problématique. Cet item est défini par le triplet Cas (Description, Objet, Descripteur).

Où :

- Le champ Description : représente une description du problème (panne, dysfonctionnement...etc.).
- Le champ Objet : désigne l'objet sujet de la panne.
- Le champ Descripteurs : désigne les caractéristiques (signes, indicateurs...) du problème.

Il existe diverse techniques de calcul de similarité, ces mesures peuvent être locales ou globales[Coh et al,03].

1. Similarités locales : elles dépendent du type des caractéristiques et des rangs des valeurs des caractéristiques. Généralement, elles sont basées sur la distance.
2. Similarités globales : elles se calculent en agrégeant les similarités locales au niveau des cas.

Nous avons implémenté et testé quelques mesures, nous allons citer quelques unes :

- *Similarité Cosinus* qui est donnée par la formule suivante :

$$Similarity = \cos(\theta) = \frac{A \cdot B}{\|A\| \|B\|} = \frac{\sum_{i=1}^n A_i B_i}{\sqrt{\sum_{i=1}^n A_i^2} \sqrt{\sum_{i=1}^n B_i^2}} \quad (IV.1)$$

A_i : fréquence de l'élément i dans le vecteur de triplet de la description du 1er cas source (source).

B_i : fréquence de l'élément i dans le vecteur de triplet de la description du 2ème cas source (cible).

– *Similarité Jaro* définie comme suit :

$$d_j = \frac{1}{3} \left(\frac{m}{|s_1|} + \frac{m}{|s_2|} + \frac{m-t}{m} \right) \quad (\text{IV.2})$$

Où :

$|s_i|$: la longueur d'un élément s_i , désigne dans notre cas la longueur d'un élément du triplet d'entrée (respectivement de sortie).

t : nombre de caractères correspondants.

m : nombre de transpositions.

Deux chaînes caractères identiques de s_1 et de s_2 sont considérés comme correspondants si leur éloignement (i.e. la différence entre leurs positions dans leurs chaînes respectives) ne dépasse pas :

$$\left[\frac{\max(|s_1|, |s_2|)}{2} \right] - 1 \quad (\text{IV.3})$$

Le nombre de transpositions est obtenu en comparant le i -ème caractère correspondant de s_1 avec le i -ème caractère correspondant de s_2 . Le nombre de fois où ces caractères sont différents, divisé par deux, donne le nombre de transpositions.

– *Similarité Dice* connue aussi sous le nom indice de **Sørensen** ou coefficient de **Dice**. Plusieurs variantes de cette variante ont été proposées, notamment, pour le domaine de la recherche d'information, elle se calcule comme suit :

$$s = \frac{2n_t}{n_x + n_y} \quad (\text{IV.4})$$

Où :

n_t est le nombre de digrammes (formés de deux caractères consécutifs) communs aux deux triplets (cas source, cas cible),

n_x est le nombre de digrammes dans le triplet d'entrée (cas source).

n_y le nombre de digrammes dans triplet de sortie (cas cible).

- *Similarité euclidienne* elle calcule la similarité entre deux documents d_1 et d_2 comme la distance entre leurs représentations vectorielles ramenées à un seul point :

$$sim_{euclidienne}(d_1, d_2) = \|\vec{d}_1 - \vec{d}_2\| = \sqrt{\sum_{i=1}^n (d_{1i} - d_{2i})^2} \quad (IV.5)$$

Où :

d_1 et d_2 : triplets d'entrée et sortie (respectivement).

n est le nombre total de termes représentés, i.e. la taille des vecteurs.

Les résultats de l'utilisation de ces mesures seront exposés plus bas dans la section de la mise en œuvre.

IV.4 Aspects de modélisation

Depuis quelques années, la notation UML (Unified Modeling language) s'est imposée pour le développement et la modélisation des applications. Cet outil de modélisation présente plusieurs atouts notamment pour la spécification, la construction, la visualisation et la documentation des systèmes informatiques. Nous avons opté pour cette notation afin de modéliser notre application. Nous avons choisi un type de diagramme pour chacun des aspects de modélisation d'UML, à savoir le diagramme de cas d'utilisation, de séquences et de classe.

IV.4.1 Les cas d'utilisation

Selon la notation UML, les cas d'utilisation ou (use case) représentent la première phase d'analyse du système. Les use case servent à récolter, analyser et d'organier les besoins, et de recenser les grandes fonctionnalités du système. Ils décrivent comment les utilisateurs interagissent avec le système. Dans cet aspect fonctionnel, on décrit les liens fonctionnels entre le système étudié et les acteurs. Ils permettent aussi, de concevoir des systèmes adaptés aux besoins des utilisateurs.

Un cas d'utilisation décrit donc, les différentes utilisations et actions produite par un acteur du système. Un acteur désigne une entité externe qui interagit avec le système et qui peut jouer plusieurs rôles vis-à-vis de ce dernier.

Nous allons présenter les cas d'utilisation des principaux acteurs dans notre application qui sont désignés par l'Administrateur, les Experts et les Utilisateurs.

IV.4.1.1 Diagramme de cas d'utilisation de l'utilisateur

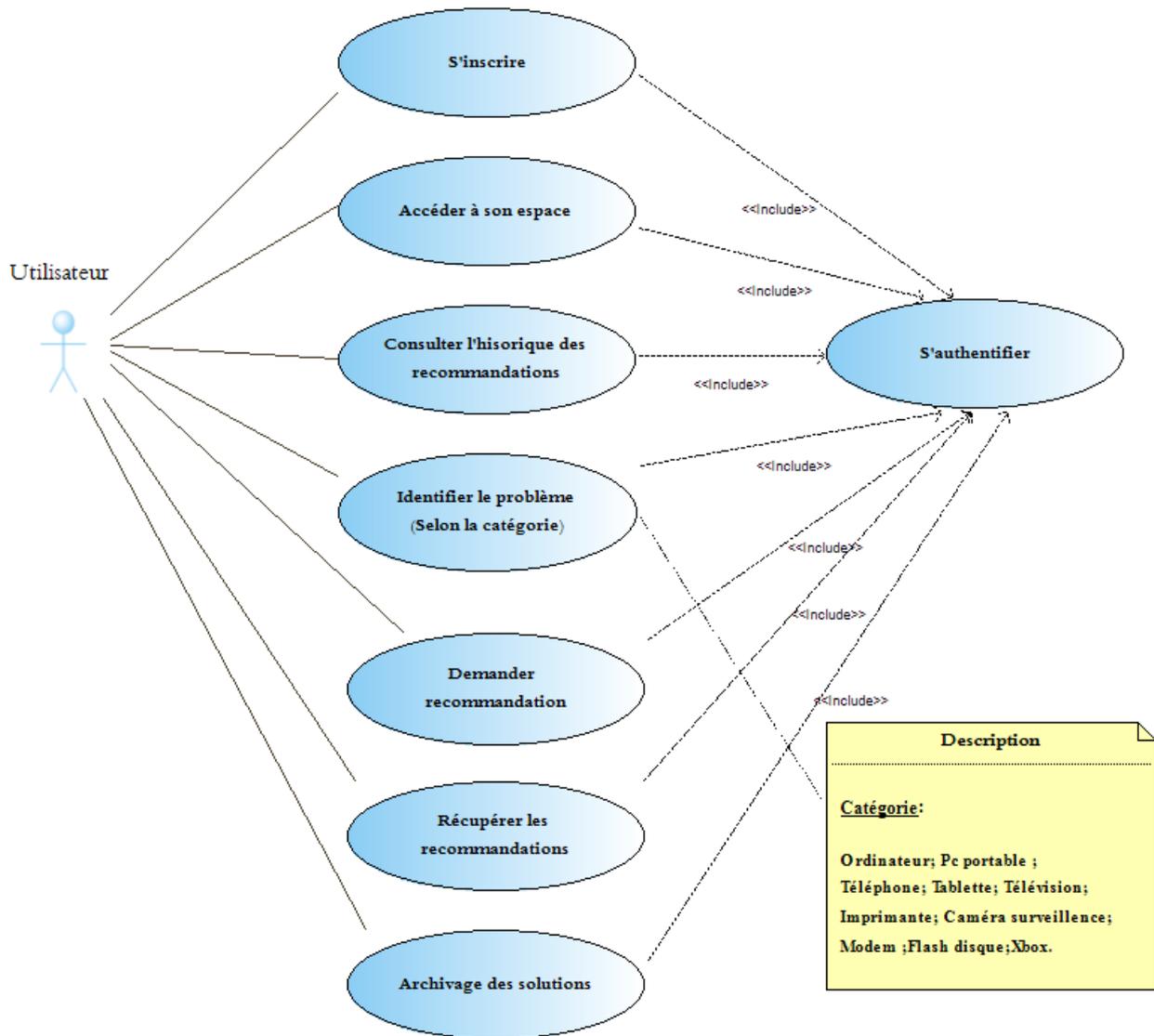


FIGURE IV.4 – Diagramme de cas d'utilisation de l'utilisateur.

IV.4.1.2 Diagramme de cas d'utilisation de l'expert

La figure suivante décrit le diagramme de cas d'utilisation d'un expert :

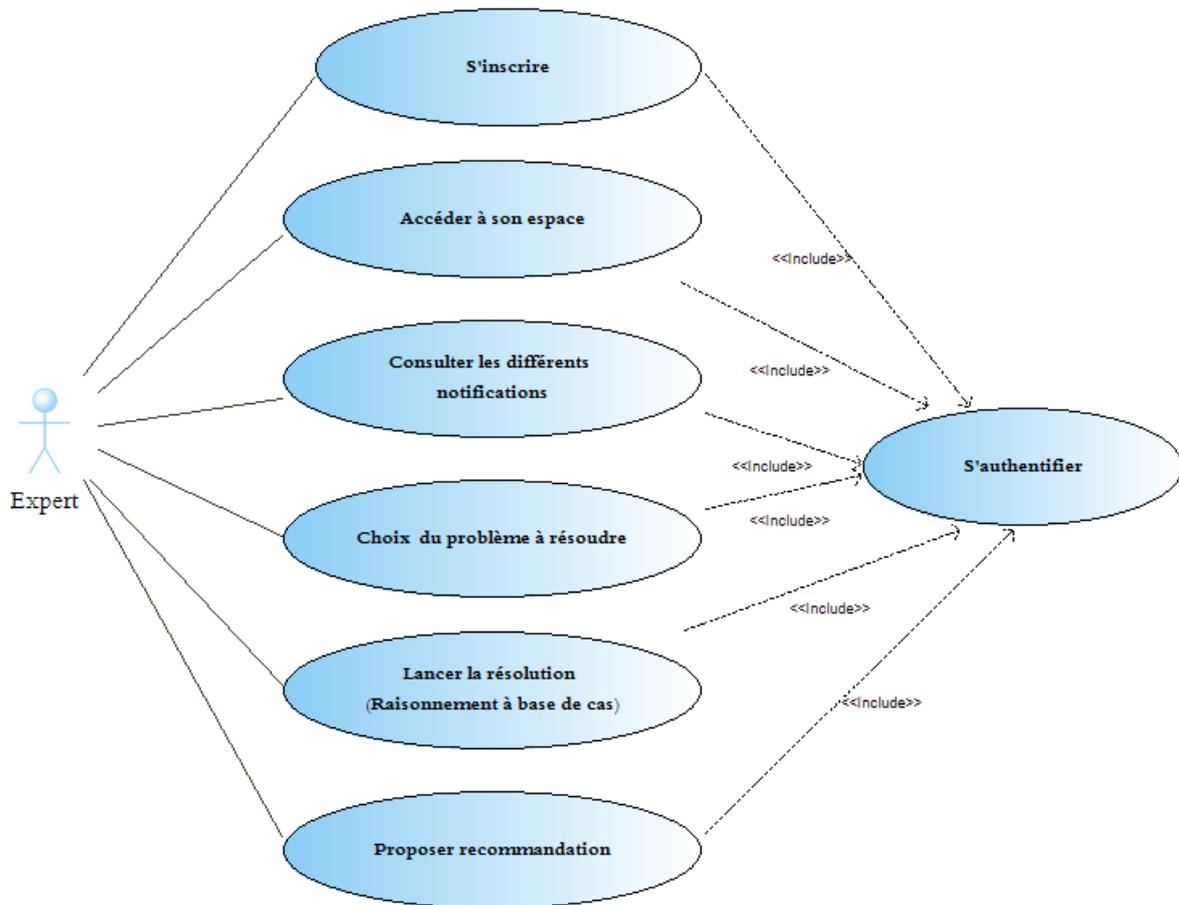


FIGURE IV.5 – Diagramme de cas d'utilisation de l'expert.

IV.4.1.3 Diagramme de cas d'utilisation de l'Administrateur

Et finalement, les principales fonctionnalités de l'Admin sont décrites dans le diagramme de la figure IV.6.

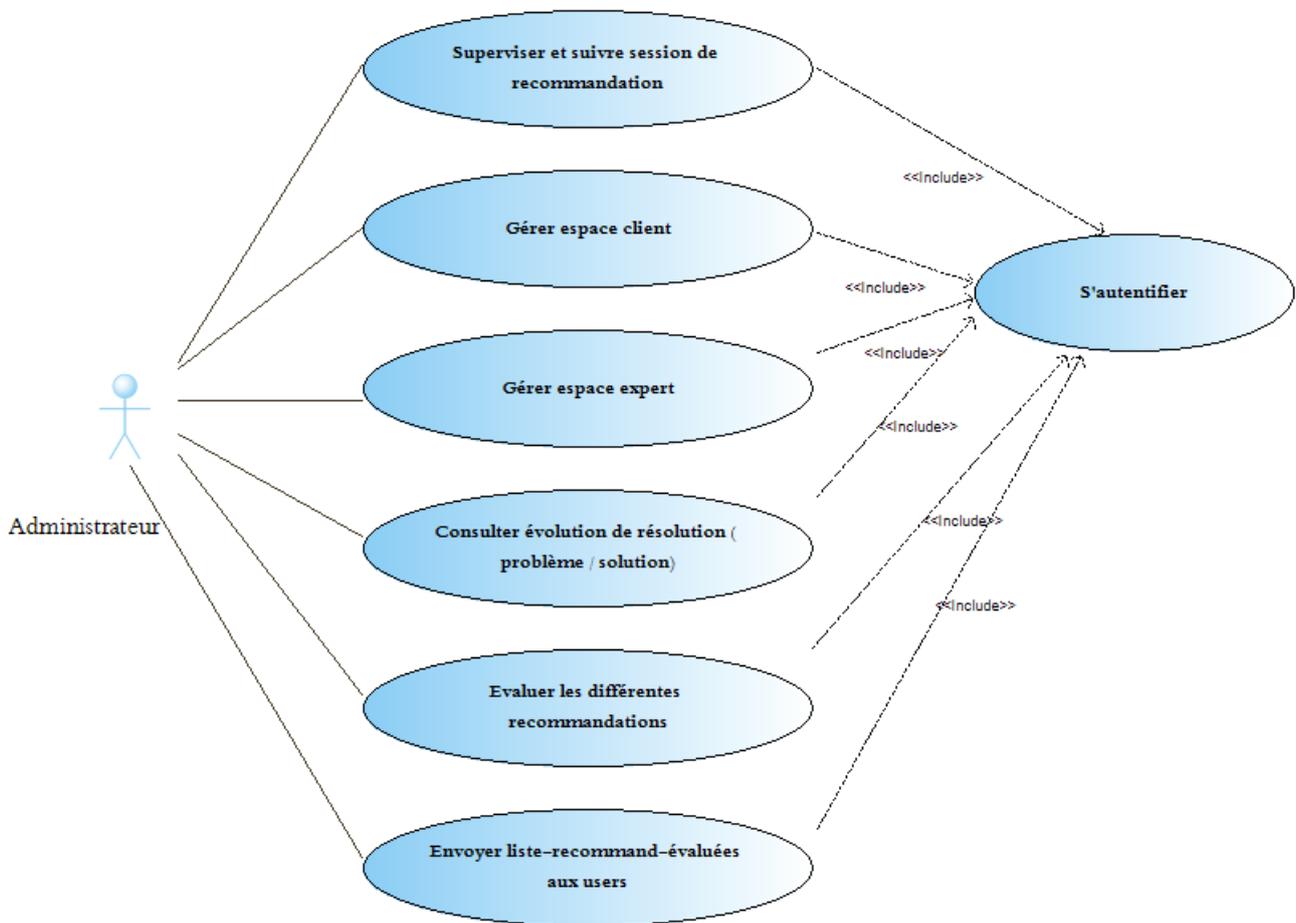


FIGURE IV.6 – Diagramme de cas d'utilisation de l'Administrateur.

IV.4.2 Les diagrammes de séquences

Nous allons décrire les interactions entre les différents acteurs et le système par des chronogrammes qui seront illustrés par des diagrammes de séquence. Ils sont considérés comme diagrammes dynamique selon la notation d'UML, car ils se contentent sur des lignes de vie des processus et objets qui vivent simultanément, et les messages échangés entre eux.

Nous avons décrit ces interactions textuellement et graphiquement.

IV.4.2.1 Diagramme de séquence d'Authentification

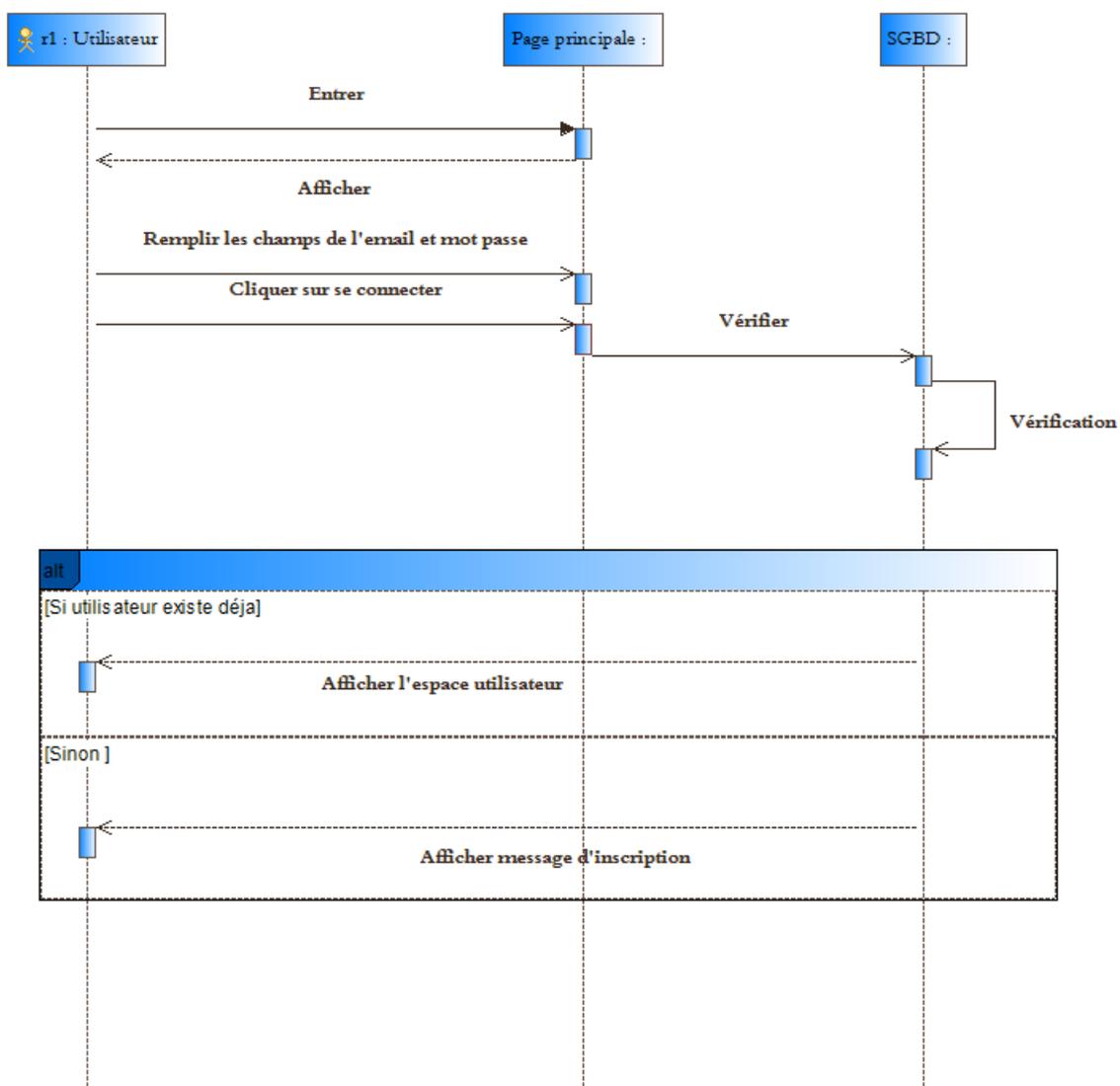


FIGURE IV.7 – Diagramme de séquence d'authentification.

IV.4.2.2 Diagramme de séquence d'identification et d'envoi du problème

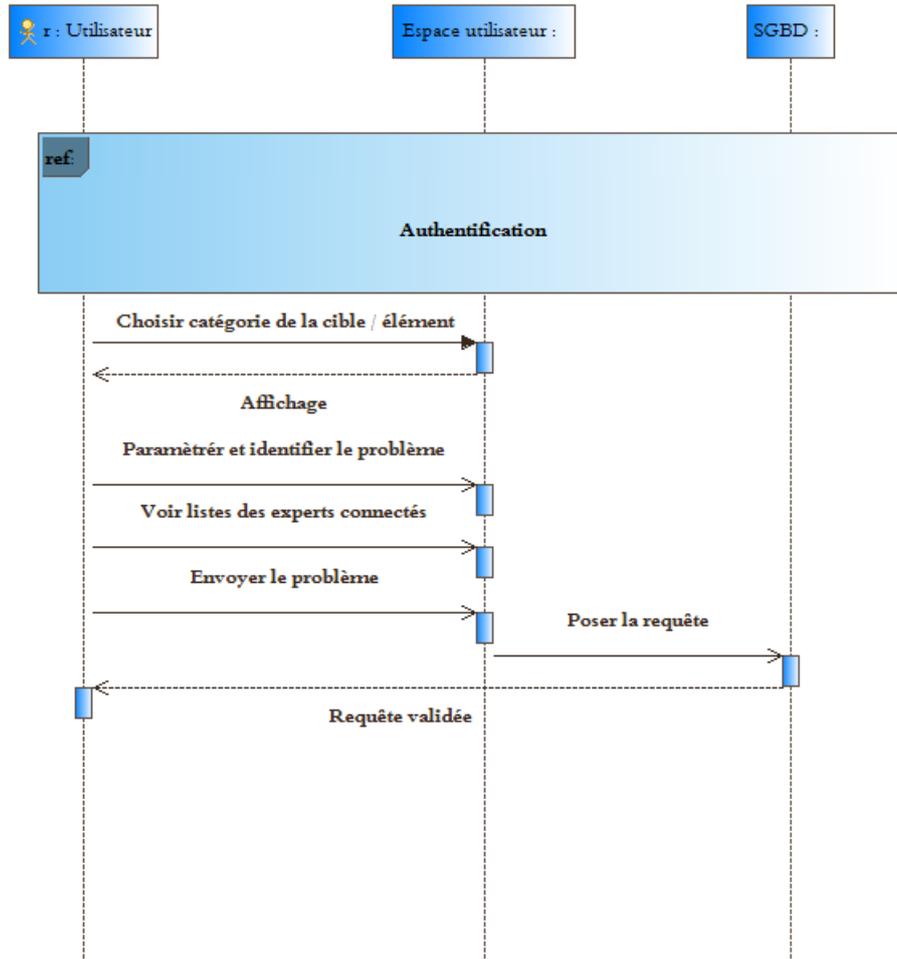


FIGURE IV.8 – Diagramme de séquence d'identification et d'envoi du problème.

IV.4.2.3 Diagramme de séquence de réception et résolution du problème

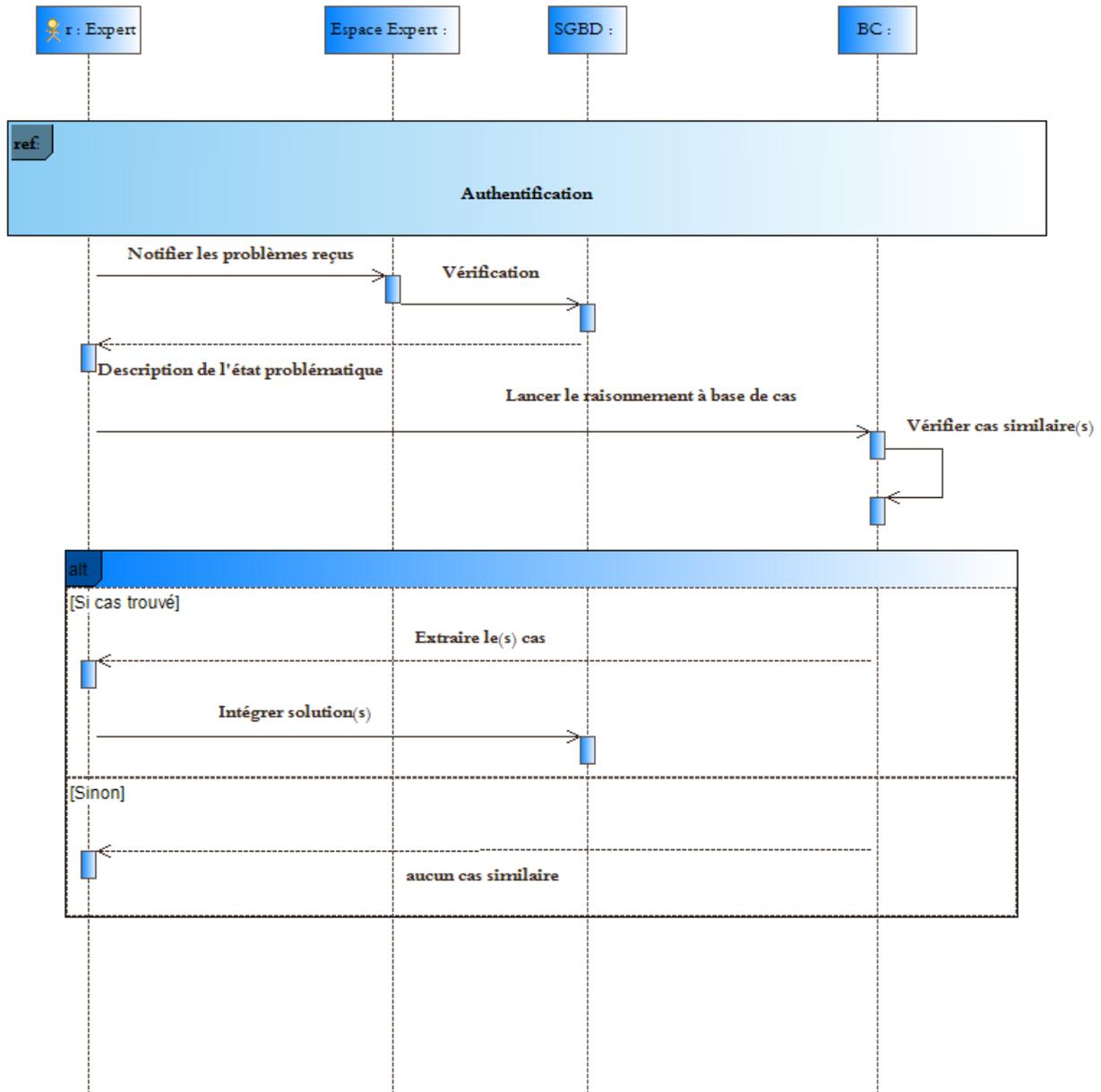


FIGURE IV.9 – Diagramme de réception et résolution du problème.

IV.4.3 Diagramme de classe

Ces diagrammes représentent le pivot de modélisation et ils sont considérés comme les diagrammes UML les plus utiles, car ils décrivent clairement la structure d'un système particulier en modélisant ses classes, ses attributs, ses opérations et les relations entre ses objets.

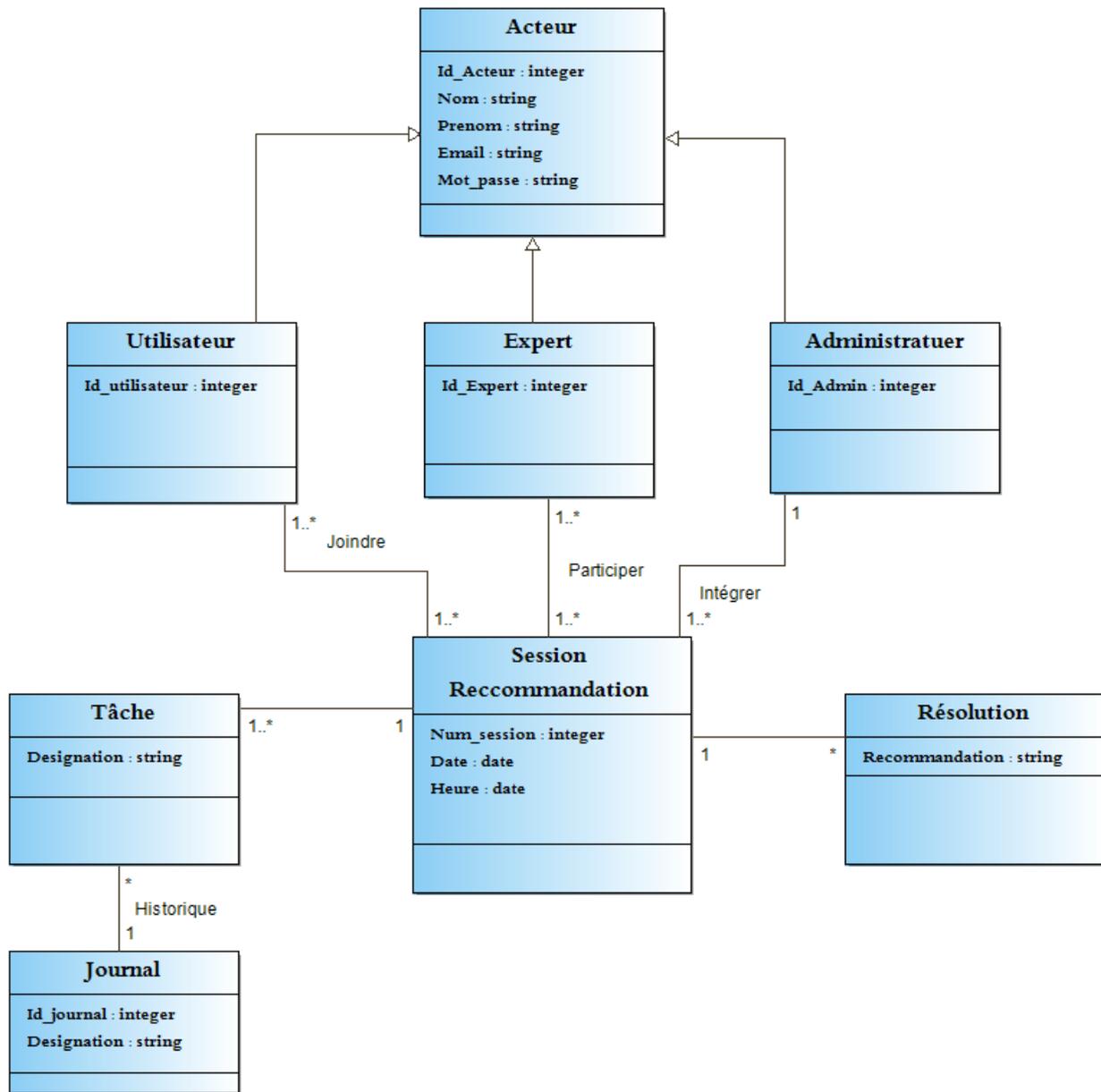


FIGURE IV.10 – Diagramme de classe.

IV.5 Mise en œuvre de l’approche

IV.5.1 Environnement et outils de programmation

Pour réaliser notre application, nous nous sommes servis d’une panoplie d’outils tels que, langages de programmation, plateformes à intégrer, éditeurs d’ontologie...etc. Nous allons décrire ces outils dans la section suivante.

IV.5.1.1 IDE Netbeans

NetBeans⁵ est un environnement de développement intégré (EDI), placé en open source en juin 2000 sous licence CDDL (Common Development and Distribution License). En plus de Java, NetBeans permet également de supporter différents autres langages, comme C, C++, JavaScript, XML, et HTML. Il comprend toutes les caractéristiques d'un IDE moderne (éditeur en couleur, projets multi-langage, éditeur graphique d'interfaces et de pages Web). Conçu en Java, NetBeans est disponible sous Windows, Linux, ou sous une version indépendante des systèmes d'exploitation (requérant une machine virtuelle Java). NetBeans constitue par ailleurs une plate-forme qui permet le développement d'applications spécifiques (bibliothèque Swing (Java)). L'IDE NetBeans s'appuie sur cette plate-forme, il s'enrichit à l'aide de plugins.

IV.5.1.2 Java

Java⁶ est un langage de programmation informatique orienté objet créé par James Gosling et Patrick Naughton, employés de Sun Microsystems, avec le soutien de Bill Joy. Il fut présenté officiellement en 1995. Selon les développeurs de Sun, Java (qui signifie café en argot américain) est un langage : simple, orienté-objet, distribué, interprété, robuste sécurisé, neutre vis à vis de l'architecture, portable, à haute performance, multi-threaded et dynamique. Le langage Java était à la base un langage pour Internet, pour pouvoir rendre plus dynamiques les pages (tout comme le JavaScript aujourd'hui). Ce langage s'est beaucoup évolué et est devenu un langage très puissant. Java met à la disposition du développeur une API très riche lui permettant de faire de très nombreuses choses. Il existe plusieurs IDE (Integrated Development Environment) pour le langage Java tels que, Eclipse, JBuilder et NetBeans.

IV.5.1.3 API Jena

Jena⁷ est un environnement de travail ou une API (interface de programmation d'application) pour les applications Java pour le web sémantiques. Ainsi, Les graphiques sont représentés comme un «modèle» abstrait. Un modèle peut interagir avec les données des fichiers, bases de données, des URL ou une combinaison de ceux-ci. Un modèle peut également être interrogé par SPARQL. JENA permet de manipuler des documents RDF, RDFS, OWL et SPARQL et elle fournit des moteurs permettant le raisonnement sur les ontologies.

5. www.netbeans.org

6. <https://www.futura-sciences.com/tech/definitions/informatique-java>

7. <https://jena.apache.org/>

IV.5.1.4 Protégé

Protégé est une interface modulaire, développée par la Stanford Medical Informatics Division de l'Université de Stanford, qui permet l'édition, la visualisation, le contrôle, l'extraction à partir de sources textuelles et la fusion semi-automatique d'ontologies. PROTEGE OWL permet la définition de métaclasse, dont les instances sont des classes, qui permettent de créer ses propres modèles de connaissances avant de construire des ontologies. De nombreux plugins sont disponibles ou peuvent être ajoutés par l'utilisateur.

IV.5.1.5 WampServer

WampServer⁸ est une plateforme de développement Web de type WAMP, servant à faire fonctionner localement (sans se connecter à un serveur externe) des scripts PHP. WampServer n'est pas en soi un logiciel, mais un environnement comprenant deux serveurs (Apache et MySQL), un interpréteur de script (PHP), ainsi qu'une administration pour les deux bases SQL PhpMyAdmin et SQLiteManager. Il dispose d'une interface d'administration servant à gérer et d'administrer ses serveurs au travers d'un tray icon (icône près de l'horloge de Windows).

IV.5.1.6 L'API JFreeChart

JFreeChart⁹ est une API Java permettant de créer des graphiques et des diagrammes de très bonne qualité. Cette API est open source et sous licence LGPL.

IV.6 Réalisation de l'application(scénarios d'exécution)

Le principe de fonctionnement global de notre application sera décrit à travers les scénarios suivants :

IV.6.1 Raisonnement à base de cas

Pour tester notre prototype, nous étions amenés à créer plusieurs ontologies vu que chaque expert utilise sa propre base de connaissance afin de trouver le(s) cas similaires item j à un nouveau problème. Une vue partielle d'une ontologie de tâche est représentée dans la figure IV.11

8. <http://fr.wikipedia.org/wiki/WampServer>

9. <http://www.jfree.org/jfreechart/>

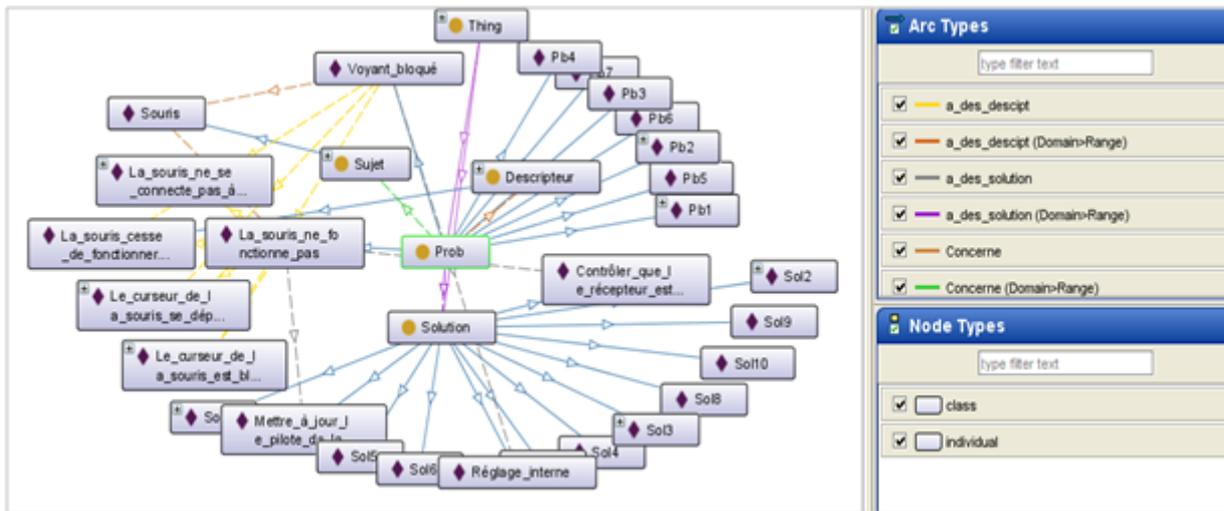


FIGURE IV.11 – Vue partielle de l'ontologie de tâche.

IV.6.2 Scénarios de la démarche décrite dans la section (IV.3.2)

Scénario n°01 : Phase de Pré-recommandation

Pour ouvrir une session de recommandation, les acteurs principaux de l'application doivent accéder à leurs espaces. La possibilité de création de nouveaux comptes utilisateurs et/ou experts est faisable. Les figures suivantes montrent ces interfaces d'accès.



FIGURE IV.12 – Interface principale de l'Admin.

Les interfaces d'authentification des utilisateurs et les experts ont le même principe.

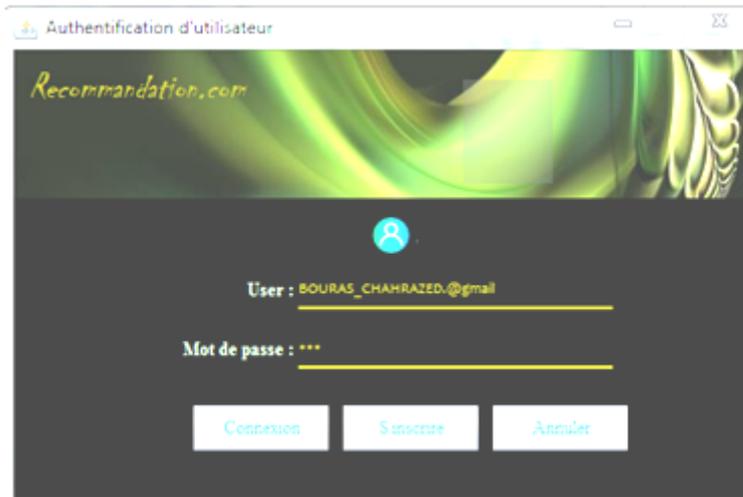


FIGURE IV.13 – Interface d'authentification de l'utilisateur.

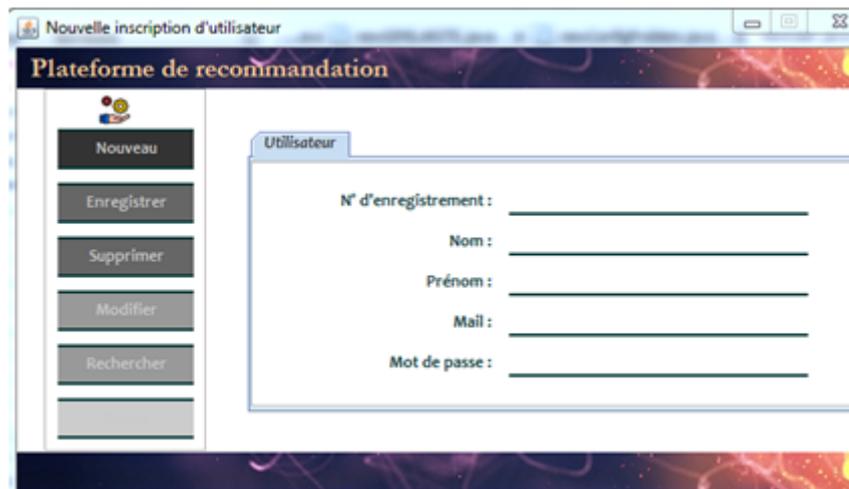


FIGURE IV.14 – Inscription d'un nouvel utilisateur.

Dès que l'utilisateur accède à son espace, il doit sélectionner le type [catégorie] de l'appareil sujet de problème. Par la suite, il procède au paramétrage et identification du problème à résoudre.

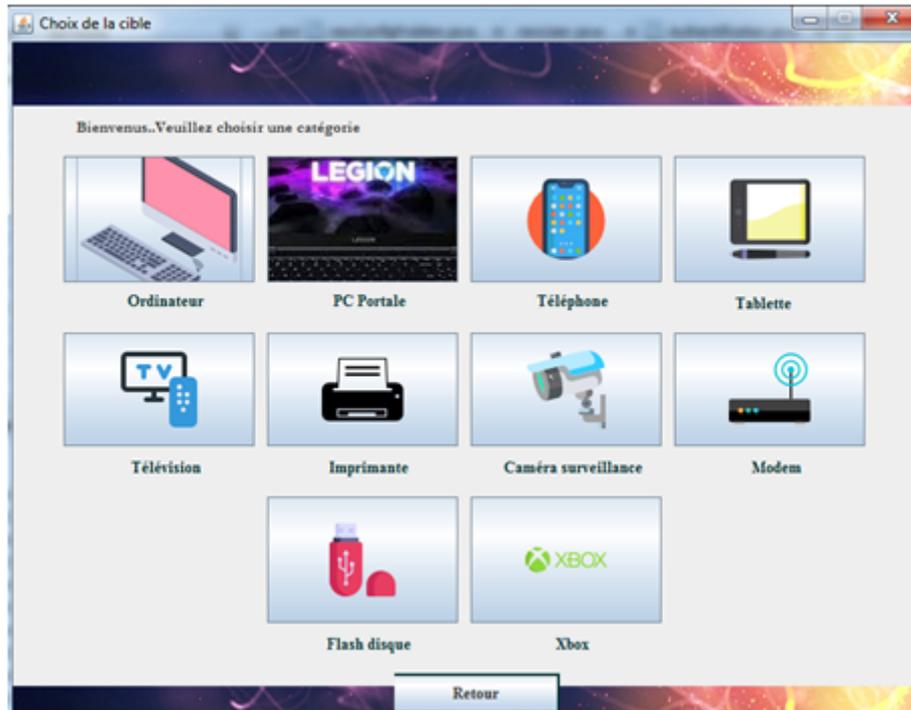


FIGURE IV.15 – Choix de la catégorie de l'appareil.

Nous supposons que l'utilisateur a sélectionné la catégorie ordinateur et comme problème à résoudre : la souris qui ne fonctionne pas. Il va faire dans un premier lieu une recherche dans l'historique des recommandations afin de trouver une solution, dans le cas contraire, il solliciter des experts afin de lui proposer des recommandations.

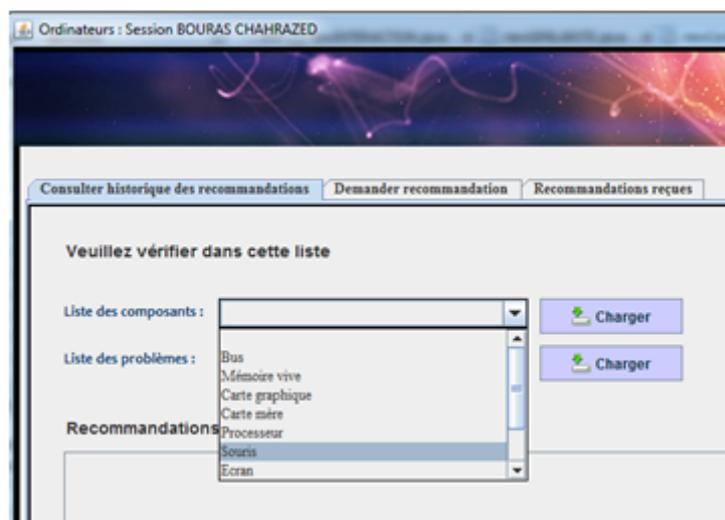


FIGURE IV.16 – Consultation de l'historique des recommandations.

Les éventuels problèmes ainsi que Les anciennes recommandations proposées concernant la souris par exemple seront visibles en cliquant sur les deux boutons « Charger ».

Identification du problème et sollicitation des experts :

FIGURE IV.17 – Identification du problème.

La configuration (identification) du problème se fait en renseignant les champs du triplet (Objet, Desc- Problème, Descripteurs- Problème). Le système peut suggérer des champs prédéfinis, comme il existe aussi la possibilité d'ajouter d'autres.

L'utilisateur consulte par la suite la liste des experts connectés intégrant la session de recommandation et sollicite uniquement les destinataires voulus (selon leurs profils).

Liste des experts connectés			
Nom	Prénom	Mail	Destinataires d'envoi
BEDJAOUTI	MOHAMED	BEDJAOUTI_MOHAMED_@gmail	1
BELBACHIR	MUSTAPHA	BELBACHIR_MUSTAPHA_@gmail	0
BENKHALFALAH	NABIL	BENKHALFALAH_NABIL_@gmail	1
BAHOUSI	ZAKARIA	BAHOUSI_ZAKARIA_@gmail	1
BOUZIANE	RIAD	BOUZIANE_RIAD_@gmail	0

FIGURE IV.18 – Requêtes envoyée aux experts.

Les experts désignés par l'envoi sont choisis de la liste et marqués par « 1 ». Ils décideront par la suite s'ils collaborent à la recommandation ou non.

L'identification détaillée du problème ainsi que les destinataires sont enregistrés dans la base de données dans une table spéciale de messagerie.

Scénario n°02 : Phase de recommandation

Tout expert ayant accepté la collaboration à la résolution du problème posé par un utilisateur i , lance son propre mécanisme de raisonnement à base de cas afin de trouver les éventuelles solutions.

Un expert peut recevoir plusieurs demandes venant des utilisateurs différents ou voire du même utilisateur.

Nous récapitulons la situation problématique comme suit :

Utilisateur / Émetteur	Problème	Objet	Descripteurs	Experts / Récepteurs
BOURAS Chahrazed	La souris a arrêté de fonctionner	Souris.	- La souris ne se connecte pas à l'ordinateur. - Le curseur de la souris est bloqué.	- BEDJAOUI Mohammed - BAHOUSI Zakaria

L'expert, en cliquant sur le bouton « Notification » pourra visualiser tous les messages qui lui sont destinés et sélectionnera après le problème à résoudre.



FIGURE IV.19 – L'interface de notification de l'espace Expert.

Le lancement du moteur de raisonnement se fait en chargeant la BC et en lançant le processus.

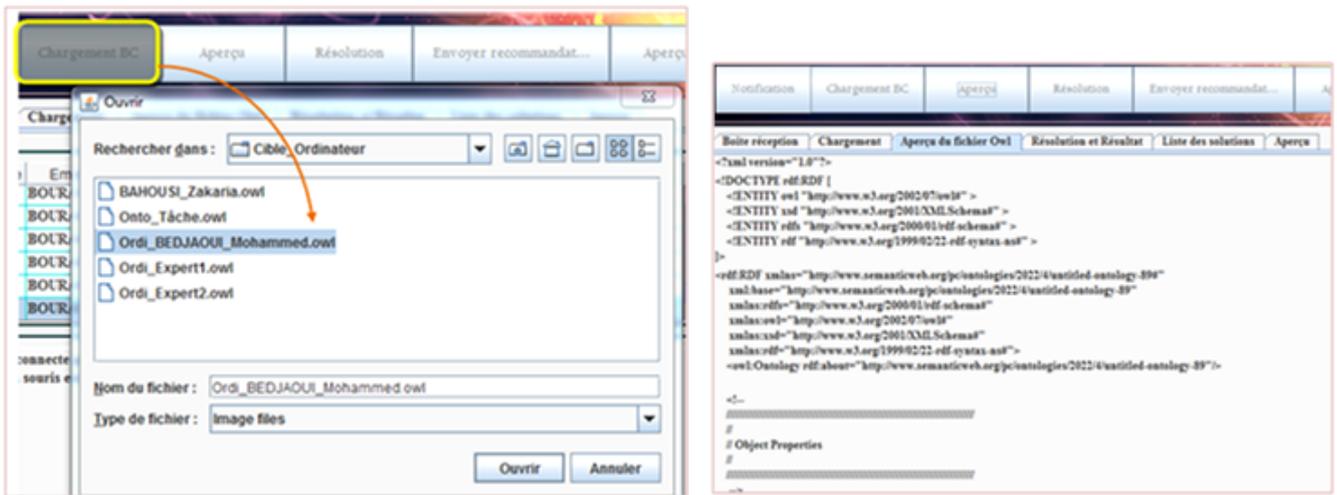


FIGURE IV.20 – Chargement de la BC de l'expert et aperçu de son fichier owl.

Les solutions des cas sources similaires sont ensuite extraites et affichées, comme le montre la figure (IV.20).

La figure suivante montre l'extraction des solutions des cas sources similaires.

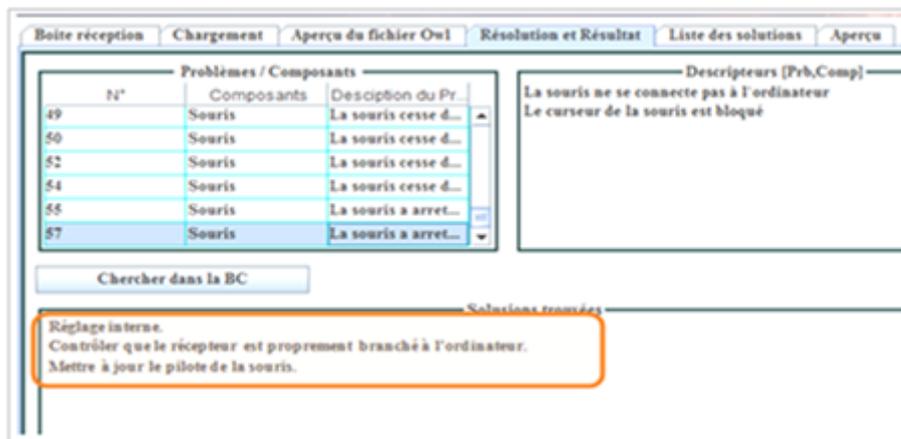


FIGURE IV.21 – Extraction de solutions de cas sources similaires.

Le détail des cas sources extraits est montré par la figure (IV.22).

Boîte réception	Chargement	Aperçu du fichier Owl	Résolution et Résultat	Liste des solutions	Aperçu
Probleme	Compo...	Symptome		Solution	
Voyant bloqué	Souris	La souris ne se connecte pas à l'ordinateur		Reglage interne	
Voyant bloqué	Souris	La souris cesse de fonctionner après une certaine période d'utilisation		Reglage interne	
Voyant bloqué	Souris	Le curseur de la souris se déplacer de manière aléatoire		Reglage interne	
La souris ne fonctionne pas	Souris	Le curseur de la souris est bloqué		Contrôler que le récepteur est correctement branché à l'ordinateur.	
La souris ne fonctionne pas	Souris	Le curseur de la souris est bloqué		Mettre à jour le pilote de la souris.	
La souris ne fonctionne pas	Souris	La souris ne se connecte pas à l'ordinateur		Contrôler que le récepteur est correctement branché à l'ordinateur.	
La souris ne fonctionne pas	Souris	La souris ne se connecte pas à l'ordinateur		Mettre à jour le pilote de la souris.	
La souris ne fonctionne pas	Souris	La souris cesse de fonctionner après une certaine période d'utilisation		Contrôler que le récepteur est correctement branché à l'ordinateur.	
La souris ne fonctionne pas	Souris	La souris cesse de fonctionner après une certaine période d'utilisation		Mettre à jour le pilote de la souris.	
La souris ne fonctionne pas	Souris	Le curseur de la souris se déplacer de manière aléatoire		Contrôler que le récepteur est correctement branché à l'ordinateur.	
La souris ne fonctionne pas	Souris	Le curseur de la souris se déplacer de manière aléatoire		Mettre à jour le pilote de la souris.	

FIGURE IV.22 – Cas sources récupérés du 1^{er} Expert.

De même pour le deuxième expert « BAHOUSSI Zakaria » qui va suivre les mêmes étapes pour extraire les éventuels cas de solutions. La figure suivante montre le détail des cas sources extraits de la BC.

Boîte réception	Chargement	Aperçu du fichier Owl	Résolution et Résultat	Liste des solutions	Ap
Probleme	Co...	Symptome		Solution	
La souris ne fonctionne pas	Souris	Le curseur de la souris est bloqué		Brancher la souris à un autre ordinateur pour voir si elle fonctionne.	
La souris ne fonctionne pas	Souris	Le curseur de la souris est bloqué		Supprimer le virus souris qui bouge toute seule.	
La souris ne fonctionne pas	Souris	Le curseur de la souris est bloqué		Débrancher l'adaptateur USB et brancher-le sur un autre port.	
La souris ne fonctionne pas	Souris	Le curseur de la souris se déplace de manière aléatoire		Brancher la souris à un autre ordinateur pour voir si elle fonctionne.	
La souris ne fonctionne pas	Souris	Le curseur de la souris se déplace de manière aléatoire		Supprimer le virus souris qui bouge toute seule.	
La souris ne fonctionne pas	Souris	Le curseur de la souris se déplace de manière aléatoire		Débrancher l'adaptateur USB et brancher-le sur un autre port.	
La souris ne fonctionne pas	Souris	Aucun voyant ne s'allume sur la souris quand elle est branchée.		Brancher la souris à un autre ordinateur pour voir si elle fonctionne.	
La souris ne fonctionne pas	Souris	Aucun voyant ne s'allume sur la souris quand elle est branchée.		Supprimer le virus souris qui bouge toute seule.	
La souris ne fonctionne pas	Souris	Aucun voyant ne s'allume sur la souris quand elle est branchée.		Débrancher l'adaptateur USB et brancher-le sur un autre port.	
La souris ne fonctionne pas	Souris	La souris ne se connecte pas à l'ordinateur		Brancher la souris à un autre ordinateur pour voir si elle fonctionne.	
La souris ne fonctionne pas	Souris	La souris ne se connecte pas à l'ordinateur		Supprimer le virus souris qui bouge toute seule.	
La souris ne fonctionne pas	Souris	La souris ne se connecte pas à l'ordinateur		Débrancher l'adaptateur USB et brancher-le sur un autre port.	

FIGURE IV.23 – Cas sources récupérés du 2^{me} Expert.

Le tableau suivant illustre les cas sources récupérés des deux experts [BEDJAOUI Mohammed et BAHHOUSI Zakaria] pour un seul triplet d'entrée émis par l'utilisateur [BOURAS Chahrazed].

Problème d'entrée

Utilisateur / Émetteur	Problème	Objet	Descripteurs	Experts / Récepteurs
BOURAS Chahrazed	La souris a arrêté de fonctionner	Souris.	- Le curseur de la souris est bloqué. - Le curseur de la souris se déplace de manière aléatoire.	- BEDJAOUI Mohammed - BAHOUSI Zakaria

Cas sources récupérés

Expert	Problème	Objet	Descripteurs	Solution
BEDJAOUI Mohammed	Cas 1 : Voyant bloqué	Souris	- La souris ne se connecte pas à l'ordinateur. - Le curseur de cesse de fonctionner après une certaine période d'utilisation. - Le curseur de la souris se déplace de manière aléatoire.	Réglage interne.

	Cas 2 : La souris ne fonctionne pas	Souris	<ul style="list-style-type: none"> - Le curseur de la souris est bloqué. - Le curseur ne se connecte pas à l'ordinateur. - La souris cesse de fonctionner après une certaine période d'utilisation. - Le curseur de la souris se déplace de manière aléatoire 	<ul style="list-style-type: none"> - Contrôler que le récepteur est proprement branché à l'ordinateur. - Mettre à jour le pilote de la souris.
BAHHOUSI Zakaria	Cas 1 : La souris ne fonctionne pas	Souris	<ul style="list-style-type: none"> - Le curseur de la souris est bloqué. - Le curseur de la souris se déplace de manière aléatoire. - La souris ne se connecte pas à l'ordinateur. - Aucun voyant ne s'allume sur la souris quand elle est connectée et le curseur n'apparaît pas à l'écran. 	<ul style="list-style-type: none"> - Brancher la souris sur un autre ordinateur pour voir si elle fonctionne. - Supprimer le virus souris qui bouge toute seule. - Débrancher l'adaptateur USB et brancher le sur un autre port.

TABLE IV.1 – Cas sources récupérés.

Ces solutions seront ensuite enregistrées dans la BDD. Le système (représenté par l'admin) évalue les recommandations proposées par chaque expert afin que l'utilisateur soit guidé lors du choix de la recommandation à prendre en considération.

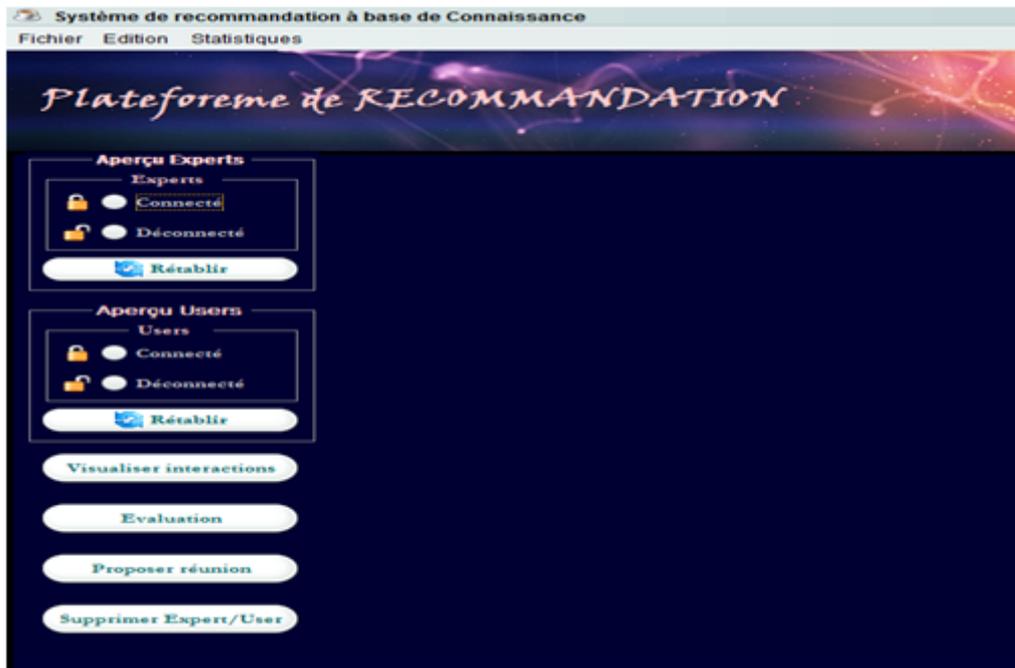


FIGURE IV.24 – Espace principale de l'admin.

Comme nous avons cité précédemment, l'admin est aux frontières du processus de recommandation, il n'intervient pas directement, seulement, il contrôle et supervise l'évolution des opérations, visualise les acteurs qui composent les groupes lors d'une session.

L'admin a d'autres fonctionnalités, nous allons présenter celle concernant l'évaluation des solutions en utilisant des mesures de similarité afin de finaliser le cycle de vie d'une session de recommandation.



FIGURE IV.25 – Évaluation des recommandations.

Scénario n°03 : Protocole d’expérimentation

Notre objectif ici est de donner plus de crédibilité pour appuyer le raisonnement à base de cas. De ce fait, nous avons testé certaines mesures de similarité comme indiqué dans la figure précédente. La démarche de cette expérimentation est décrite comme suit :

Étape 1

Expérimentation sur les données ontologiques à travers les cas sources extraits de la BC. Ce jeu est illustré par le tableau IV.1 .

Étape 2

Métriques à évaluer : nous avons testé les métriques suivantes (Cosinus, Jaro, Dice, Jaccard, Jaccard Généralisé, Euclidienne).

La pertinence des résultats des mesures de similarité est illustrée dans le tableau suivant :

Expert	Cas source extrait	SimCos	SimDice	SimJaro	Sim-Jaccard	SimGen-Jaccard	Euclidienne
BEDJAOUI Mohamed	Réglage interne	0.38111	0.38112	0.5779	0.113751	0.115052	0.4710819
	Contrôler que le récepteur est proprement branché à l'ordinateur. Mettre à jour le pilote de la souris.	0.7834776	0.366008585	0.62202	0.253715	0.25705	0.68293
BAHHOUSI Zakaria	Brancher la souris sur un autre ordinateur pour voir si elle fonctionne. Supprimer le virus souris qui bouge toute seule.	0.7834776	0.37692338	0.633640	0.26742	0.271805	0.68078

	Débrancher l'adapta- teur USB et brancher le sur un autre port.						
--	-----------------------------------------------------------------------------------	--	--	--	--	--	--

TABLE IV.2 – Similarité [cas source,cas cible].

Discussion des résultats

Nous allons tracer des graphes de comparaison afin d’expliciter les résultats obtenus.

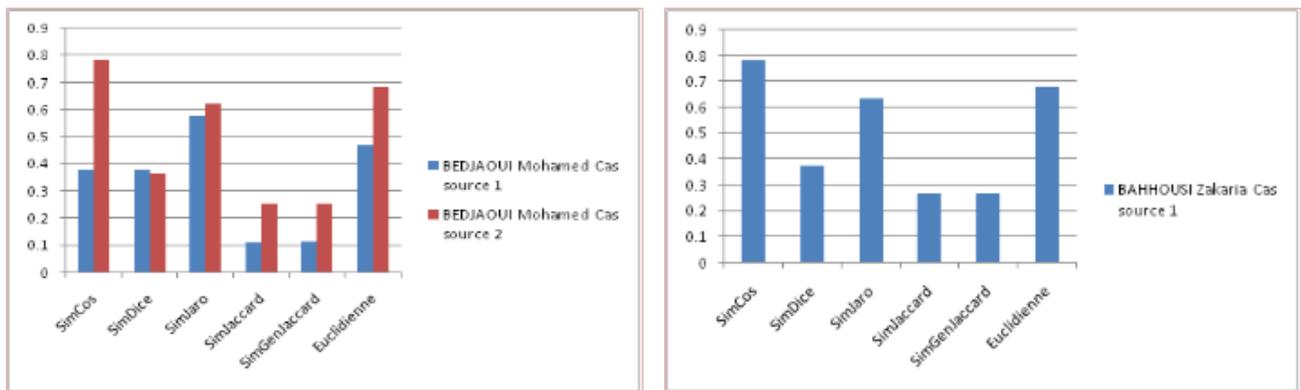


FIGURE IV.26 – Représentation des cas sources trouvés pour chaque expert.

Interprétation

Si nous se référons au raisonnement ontologique d’un point de vue fiabilité de l’extraction des cas source indépendamment de toute comparaison entre les métriques de similarité, nous pouvons conclure que toute les mesures sont fiables à un degré élevé car si nous prenons par eg., le cas du premier expert, les deux cas extraits présentent de taux de similarités différents où le deuxième est plus similaire au cas cible, par rapport au premier et cela est déjà illustré dans le tableau qui précise le jeu de données utilisé. Cela permet de classer les recommandations proposées de la plus pertinente à la moins pertinente.

D’un point de vue quantitatif, nous avons essayé de comparer la pertinence de ces mesures en se fiant à une des mesure de pertinence qui est la précision P, qui se calcule selon la formule suivante :

$P = (a/a+b)$ qui permet de calculer le ratio des résultats pertinents retrouvés (a) par rapport aux résultats non pertinents (b). Cette mesure précise la capacité qu'un système donné refuse des solutions non pertinentes.

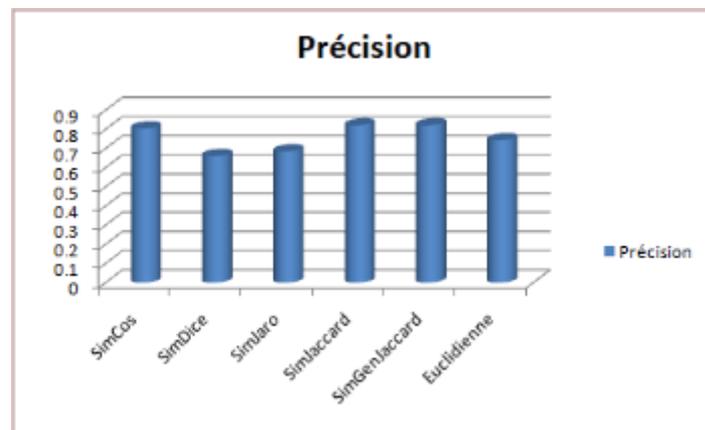


FIGURE IV.27 – Précision moyenne des métriques.

La figure (IV.27) présente les métriques en fonction de la précision moyenne, nous remarquons que les métriques suivantes Jaccard, GénéralJaccard, Cosinus et l'Euclidienne sont considérées les plus pertinentes, et viennent par la suite Dice et Jaro, si nous examinons les résultats données par les métriques individuellement et par le raisonnement ontologique, nous remarquons que cette mesure s'alterne entre les deux extrémités [cas le(s) plus similaire(s)], [cas le(s) moins similaire(s)] et cela est expliqué par le fait que les deux premières métriques ne sont pas très bien adaptées dans le cas des chaînes de caractères longues (s'adaptent principalement pour les chaînes courtes (token)), et donc ce taux de précision élevé interprète le refus des cas pour lesquelles la comparaison ne marchait pas bien. Cependant, Cosinus et l'Euclidienne indiquent approximativement un taux réel de précision. Par conséquent, nous ne pouvons pas considérer ce facteur de précision (P) comme indicateur de pertinence, vu que l'interprétation des résultats dépend de la connaissance et l'analyse des résultats précédemment donnés.

D'un point de qualitatif, le graphe donné par la figure (IV.28) présente les cas qui ne marchaient pas pour les meilleures métriques si on se base sur le facteur de précision uniquement (comme c'est déjà expliqué dans le paragraphe précédent). Mais, nous remarquons que l'écart (qui tend vers le zéro) entre les valeurs calculées pour les métriques utilisées reste quasiment le même, ce qui explique la fiabilité de la métrique de similarité en elle-même pour déterminer le taux de similitude entre les cas malgré la non adaptation complètes de quelques une pour des problèmes présentant des contraintes de taille de données.

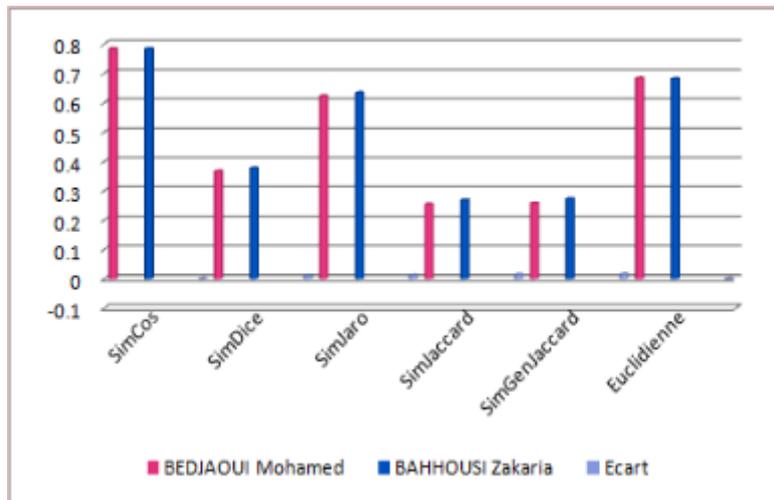


FIGURE IV.28 – Écart entre les mesures de similarité.

Pour résumer, l'utilisation de ces métriques de similarité a montré son efficacité pour l'extraction des cas similaires en explicitant les calculs de ces différentes mesures et qui ont été analysées afin d'appuyer et valider par la suite le raisonnement à base de cas.

Scénario n°03 : Validation de la recommandation choisie et clôture de la session Dans cette phase l'utilisateur choisie la solution à appliquer et il peut éventuellement l'archiver afin de s'en servir ultérieurement. Et La session sera clôturée.

IV.7 Conclusion

Nous avons présenté dans ce chapitre notre approche de recommandation à base de connaissances et nous avons montré, grâce aux mécanismes d'inférence, l'exploitation de notre base de connaissances qui est structurée sous forme d'une ontologie, en faisant un raisonnement à base de cas dans une hiérarchie de concepts et de connaissances.

Nous avons rapporté les différentes étapes de réalisation de notre projet commençant par une démarche de modélisation, qui a été suivie par une implémentation de la solution logicielle, cette dernière a été illustrée par des captures d'écran de l'application et quelques scénarios d'exécution.

Conclusion générale et perspectives

Conclusion générale et perspectives

Conclusion

La recherche de l'information est devenue au cœur de nombreux débats actuels, l'un de ses enjeux majeurs est de transformer et organiser cette masse d'information en connaissances permettant d'améliorer la qualité des recommandations recherchées. Dans ce travail, nous nous sommes intéressés aux systèmes de recommandation (SdR), qui occupent une place très importante qui ne cesse d'accroître et constituent actuellement un élément clé de diverses applications de domaines différents. Ainsi, ils jouent un rôle primordial pour assister et guider les choix des utilisateurs en leur proposant des recommandations en adéquation avec leurs besoins et exigences recherchés dans le but de les aider à repérer facilement les informations utiles.

En nous appuyant sur les techniques de l'ingénierie des connaissances, nous avons mené un travail visant à : i) améliorer la qualité des recommandations proposées à travers l'extraction de connaissances faisant de l'inférence sur une base de cas, ii) mieux comprendre le classement des recommandations proposées à travers l'explication des détails du raisonnement à base cas, iii) rendre l'approche de recommandation plus transparente et explicite, iv) construire un modèle ontologique qui sert d'un côté à la structuration de notre base de connaissance, et d'un autre côté, de support d'inférence, v) palier aux inconvénients de certaines approches de recommandation, vi) proposer un protocole d'évaluation des recommandations trouvées en testant des métriques de similarité, vii) validation des solutions extraites à partir du raisonnement comparativement aux résultats des métriques d'évaluation. Cette section récapitule l'explication de notre approche ainsi que quelques fins qui en découlent.

Dans notre travail, nous avons réalisé une application de recommandation à base de connaissances pour la résolution des problèmes de panne / dysfonctionnement du matériel informatique et les accessoires qui y sont liées, nous devons recommander à un utilisateur donné des solutions qui peuvent satisfaire ses attentes en matière de dépannage et d'acquisition d'un savoir faire (connaissance) à utiliser dans le futur. Le processus de recommandation se déroule en trois grandes phases, une phase de pré-recommandation qui consiste à identifier le problème à résoudre et solliciter les experts susceptibles de proposer des solutions, la seconde phase, celle de la recommandation proprement dite, où des solutions similaires devront être recherchées dans la base de connaissances et communiquées ensuite aux demandeurs, finalement, la phase de clôture de la session de recommandation avec l'archivage / historisation des solutions retenues.

Depuis longtemps, les systèmes de recommandation étaient évalués d'une manière systématique sur leur manière à proposer des solutions pertinentes aux utilisateurs, ce critère est assez important, néanmoins, la mesure réelle de la concordance de la solution proposée avec le problème posé devient un facteur très important pour la satisfaction de l'utilisateur, de ce fait, nous avons contribué à créer un protocole d'évaluation des recommandations par des métriques de similarités qui ont été testées, ce qui nous a permis de valider notre mécanisme de raisonnement à base de cas sur une structure ontologique.

Perspectives de travaux futurs

Nous présentons dans cette section quelques terrains de recherche relatifs à l'approche que nous avons traitée dans ce projet.

Dans le cadre de la diversité des SdR, nous proposons des systèmes de recommandations appartenant à des familles différentes telles que, le filtrage collaboratif, le filtrage basé contenu, filtrage basé items,...etc.

Proposer des mécanismes d'explication détaillée des recommandations proposées à l'utilisateur dans le but de le convaincre.

Intégrer les théories de confiance entre le SdR et les utilisateurs.

Dans le but d'améliorer la dimension collective / collaborative, nous proposons d'étendre la résolution individuelle d'un expert (indépendamment des autres experts) du problème, à une résolution collaborative au vrai sens du mot.

Enfin, nous souhaitons que d'autres travaux soient inspirés par notre approche réalisée proposant ainsi de nouvelles ouvertures dans le domaine de la recommandation.

Bibliographie

- [**Ado et Tuz,05**] Adomavicius, G. and Tuzhilin.,(2005). A. 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, 17(6) :734–749.
- [**Alv et al ,14**],.(10 mars 2014) « Un système de recommandation basé sur la qualité pour diffuser des informations dans une bibliothèque numérique universitaire »Sciences de l’information volume 261, pages 52-69.
- [**Al et al,11**] AL-HASSAN M., LU H. and LU J.,(2011). Personalized e-government services : Tourism recommendersystem framework. In Lecture Notes in Business Information Processing, volume 75 LNBIP, p.173–187 : Springer, Berlin, Heidelberg.
- [**Ami,09**] N.Amine.,(Juillet,2009). Papyrus : Un système de gestion et de recommandation d’articles de recherche, Master’s thesis, Université de Montréal Faculté des arts des sciences.
- [**Arp et al,03**] Arpirez, J. C., Corcho, O., Fernández-López, M.,et Gómez-Pérez, A., (2003). WebODE in a nutshell. AI Magazine. To be published in 2003.
- [**Aue et al,07**] Auer, S., Bizer, C., Kobilarov, G., Lehmann, J., Cyganiak, R., and Ives, Z., (2007). Dbpedia : A nucleus for a web of open data. In The semantic web, volume 4825, pages 722–735. Springer.
- [**Bac,00**] Bachimont, B.,(2000).Engagement sémantique et engagement ontologique conception et réalisation d’ontologie en ingénierie des connaissances.
- [**Bal et Sho,97**] Balabanović, M. and Shoham Y.,(1997). Fab : content-based, collaborative recommendation. Communications of the ACM, 40(3) :66–72.
- [**Ban,10**] BANDOUE FARIDA.,(2010). Diagnostic de pannes d’un système de pompage photovoltaïque.
- [**Ban et Cha,07**] BANEYX Audrey, CHARLET Jean.,(2007). Évaluation, évolution et Maintenance d’une ontologie en médecine : état des lieux et expérimentation.
- [**Bas et al,01**] Basu, C., Hirsh, H., Cohen, W. and Manning, N. C.,(2001). Technical paper

recommendation : A study in combining multiple information sources.

[Bel,15] ELABBES FATIMA.,(2015). Conception et implémentation d'une ontologie médicale
Cas : insuffisance cardiaque.

[Bel,19] Belhamiani Fatima Zohra.,(2019). « Outil intelligent d'aide à Maintenance Industrielle : raisonnement à partir de cas et ontologie », Mémoire de Master en Informatique, Université Belhadj Bouchaib, Ain Temouchent.

[Bel et Cro,92] Belkin, N.J. and Croft, B.W.:(1992) Information Filtering and Information Retrieval Two Sides of the Same Coin Communications of the ACM, 35, 29-38.

[Ben et al,06] Bendakir, N., Airneur, E.,(2006). Using Association Rules for Course Recommendation. Workshop on Educational Data Mining. Conference AAAI 2006 (Twenty-first National Conference on Artificial Intelligence, Boston, Massachusetts USA, 31- 40.

[Ben,17] BENABDALLAH Ali.,(2017). Construction semi-automatique des ontologies à partir des documents textuels arabes.

[Ben-I,17] BENOURET I.,(2017). A contextual and composite recommender system for the personalization of cultural sites visit. Thèse de doctorat en informatique, Université de Technologie de Compiègne.

[Ber,02] Berti-Equille L.,(2002). Annotation et recommandation collaboratives des documents selon leur qualité, RSTI série ISI-NIS Recherche de filtrage d'information, 7(12), pp. 125-155.

[Ber,06] Berners-Lee, T.,(2006). Linked data. International Journal on Semantic Web and Information Systems, 4(2).

[Bla et al,08] BLANCO-FERNANDEZ Y., PAZOS-ARIAS J., GIL-SOLLA A., RAMOS-CABRER M. and LOPEZNORES M.,(2008). Providing entertainment by content-based.

[Bob et al,10] J. Bobadilla,F. Serradila, J. Bernal.,(août 2010). « Une nouvelle métrique de filtrage collaboratif qui améliore le comportement des systèmes de recommandation ». Systèmes basés sur la connaissance Volume 23, numéro 6 , pages 520-528.

[Bob,13] Jesús Bobadilla.,(2013), Fernando Ortega, Antonio Hernando, and Abraham Gutiérrez. Recommender systems survey. Knowledge-based systems, 46 :109–132.

[Bor,97] (1997).Construction of Engineering Ontologies. Centre for Telematica and Information Technology, University of Twente, Enschede, The Netherlands.

[Bou ,05] Bouzeghoub, M., et Kostadinov, D.,(2005). Personnalisation de l'information : Aperçu de l'état de l'art et définition d'un modèle flexible de définition de profils. Actes de la 2nde Conférence en Recherche d'Information et Applications CORIA, 2005, p. 201-218.

[Bre et al,98] Breese J. S., Heckerman D., and Kadie C., (1998). Empirical analysis of predictive algorithms for collaborative filtering, In Proc. Of the 14th Conf. on UAI, pp. 43–52.

- [Bur et al,97] Burke, R., Hammond, K., and Young, B.,(1997). The FindMe Approach to Assisted Browsing, *IEEE Expert*, 12(4), pages 32-40.
- [Bur,99] BURKE R.,(1999). The Wasabi Personal Shopper : A Case-Based Recommender System. Submitted to the 11 Annual Conference on Innovative Applications of Artificial Intelligence.
- [Bur,00] BURKE R.,(2000). Knowledge-based recommender systems. *Encyclopedia of library and information systems*, 69.
- [Bur,02] Robin Burke.,(2002). Hybrid recommender systems : Survey and experiments. *User modeling and user-adapted interaction*, 12(4) :331–370.
- [Can et al,08] Cantador et al., 2008 CANTADOR I., BELLOGÍN A. and CASTELLS P.,(2008). News@hand : A semantic web approach to recommending news. In *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, volume 5149.
- [Car et al,12] CARRER-NETO W., HERNÁNDEZ-ALCARAZ M. L., VALENCIA-GARCÍA R. and GARCÍA-SÁNCHEZ F. (2012). Social knowledge-based recommender system. Application to the movies domain. *Expert Systems with Applications*, 39(12), 10990–11000.
- [Che et Syc,98] Chen, L. and Sycara, K.,(1998). WebMate : a personal agent for browsing and searching. In *Proceedings of the second international conference on Autonomous agents AGENTS '98* pp. 132-139, ACM, New York, NY, USA.
- [Cho et al,02] Cho Y. H., Kim S.H., Kim J.K.,(2002). A personalized recommender system based on web usage mining and decision tree induction, *ESA*, 23(3), pp. 329-342.
- [Cla et al,99] Claypool, M., Gokhale, A., Miranda, T., Murnikov, P., Netes, D. and Sartin, M.,(1999). Combining Content-Based and Collaborative Filters in an Online Newspaper.
- [Coh et al , 03] Cohen W. W., Ravikumar P., Fienberg S.,(2003) « A Comparison of String Distance Metrics for NameMatching Tasks », *IJCAI-03 Workshop on Information Integration*, p. 73-78.
- [Con et Her,99] Conner, M. and Herlocker, J.,(1999). Clustering items for collaborative filtering. In the *ACM SIGIR Workshop on Recommender Systems*.
- [Cor,06] Cortes Robles G.,(2006).“Management de l’innovation technologique et des connaissances : Synergie entre la théorie TRIZ et le Raisonnement à Partir de Cas. Application en génie des procédés et systèmes industriels“, Thèse doctorale. Institut National Polytechnique de Toulouse.
- [Cos et al,02] Cosley, D., Lawrence, S., and Pennock, D.,(2002). Referee : An open framework for practical testing of recommender systems using research index. In the 28th international

conference on Very Large Data Bases, page 46.

[Cro,93] W.B. Croft.,(Avril 1993) Knowledge-based and Statistical approaches to Text Retrieval. IEEE EXPERT, vol. 8, n° 2.

[Cun et al,00] Cunningham, P., Bergmann, R., Schmitt, S., Traphoner, R., Breen,S. and Smyth, B.,(2000). WEBSSELL : Intelligent Sales Assistants for the World Wide Web.In Procs. of the Workshop Programme at the Fourth International Conference on Case-Based Reasoning.

[Dan et al ,04] Dan Brickley and R. V. Guha, Editors.,(10 Février 2004). RDF Vocabulary Description Language 1.0 : RDF Schema, W3C Recommendation.[http ://www.w3.org/TR/rdf-schema/](http://www.w3.org/TR/rdf-schema/).

[Des et Kar,11] Desrosiers, C. and Karypis, G.,(2011). A comprehensive survey of neighborhood-based recommendation methods. In Recommender systems handbook, pages 107–144. Springer.

[Di et al,12a] Di Noia, T., Mirizzi, R., Ostuni, V. C., and Romito, D.,(2012a). Exploiting the web of data in model-based recommender systems. In Proceedings of the sixth ACM conference on Recommender systems, pages 253–256.

[Di et al,12b] Di Noia, T., Mirizzi, R., Ostuni, V. C., Romito, D., and Zanker, M., (2012b). Linked open data to support content-based recommender systems. In Proceedings of the 8th International Conference on Semantic Systems, I-SEMANTICS '12, page 1–8.

[Di et Ost,15] Di Noia, T. and Ostuni, V. C.,(2015). Recommender systems and linked open data. In Reasoning Web International Summer School, pages 88–113. Springer.

[Di et al,16] Di Noia, T., Ostuni, V. C., Tomeo, P., and Sciascio, E. D., (2016). Sprank : Semantic pathbased ranking for top-n recommendations using linked open data. ACM Transactions on Intelligent Systems and Technology (TIST), 8(1) :1–34.

[Dud et Har,73] Duda, R. O. and Hart, P. E.,(1973). Pattern Classification and Scene Analysis. John Wiley et Sons Inc.

[El-D et al,12] EL-DOSUKY M. A., RASHAD M. Z., HAMZA T. T. and EL-BASSIOUNY A. H. (2012). Food recommendation using ontology and heuristics. In International conference on advanced machine learning technologies and applications, p. 423–429 : Springer.

[Fer et al,97] FERNANDEZ M., A. GOMEZ-PEREZ et al.,(1997). METHONTOLOGY : from ontological art owards ontological engineering.

[Fis et Ste,91] Fischer, G. and Stevens, C.,(1991). Information Access in Complex, Poorly Structured Information Spaces. pp. 63-70, ACM Press.

[Flo, 06] : Florence Le Ber.,(2006). « Les systèmes à base de connaissances », Article publié dans l'Encyclopédie de l'informatique et des systèmes d'information, J. Akoka and I. Comyn-

Wattiau,(Ed.) 1197–1208.

[**Flo et al,20**] ,(Jun 2020). « Un système de recommandation social pour des recommandations personnalisées de recettes »Workshop sur les Affects, Compagnons artificiels et Interactions, CNRS, Université Toulouse Jean Jaurès, Université de Bordeaux, Saint Pierre d’Oléron, France.

[**Ghe, 17**] GHERZOULI Imane.,(2017). Système d’aide au diagnostic médical à base d’ontologie.

[**Gol et al,92**] Goldberg, D., Nichols, D., Oki, B. M., and Terry, D.,(1992). Using collaborative filtering to weave an information tapestry. *Communications of the ACM*, 35(12) :61–70.

[**Gom,99**] Gómez-Pérez A.,(1999) “Ontological Engineering : A state of the art”. *Expert Update*.

[**Gom et al,04**] Gómez-Pérez A., Fernández-López M., Corcho O.,(2004). "Ontological Engineering" Springer Verlag Advanced Information and Knowledge Processing, 1st ed. 2004. 2nd printing, 2004, XII, 403 p. 159 illus., Hardcover ISBN : 1-85233-551-3.

[**Gru,93**] Gruber, T. R.,(1993). “A translation approach to portable ontology specifications. *Knowledge Acquisition*”, 5(2) :199 – 220.

[**Gua,97**] Guarino, N.,(1997). Formal Ontology in Information Systems. In N. Guarino (Ed.), 1st International Conference on Formal Ontology in Information Systems (FOIS’98) (pp. 3-15). IOS Press.

[**Gua,98**] Guarino N.,(1998) “Formal Ontology and Information Systems”, In N. Guarino (Eds), *Formal Ontology in Information Systems*, IOS Press, Amsterdam.

[**Har,94**] Harman, D.,(1994). Overview of the Third Text REtrieval Conference (TREC-3). In TREC’94.

[**Hau et Fre,01**] Hauver, D. B. and French, J. C.,(2001). Flycasting : Using Collaborative Filtering to Generate a Playlist for Online Radio. In *Proc. Int. Conf. Web Delivering of Music* pp. 123-130.

[**Hei et Hay,10**] HEITMANN B. and HAYES C.,(2010). Using Linked Data to Build Open, Collaborative Recommender Systems. In *AAAI spring symposium : linked data meets artificial intelligence*, volume 2010.

[**Hil et al,95**] Hill, W., Stead, L., Rosenstein, M., and Furnas, G.,(1995). Recommending and evaluating choices in a virtual community of use. In *Proceedings of the SIGCHI conference on Human factors in computing systems*, pages 194–201. ACM Press/Addison-Wesley Publishing Co.

[**Hor et al,01**] I. Horrocks, F. van Harmelen, P.F. Patel-Schneider.,(2001). Reference description of the DAML+OIL (March2001) ontology markup language,

<http://www.daml.org/2001/03/reference.html>.

[Hu et al., 08] Hu, Y., Koren, Y., Volinsky, C.(2008). Filtrage collaboratif pour les ensembles de données de rétroaction implicites. Dans : 2008 Huitième Conférence internationale de l'IEEE sur l'exploration de données, pp. 263–272. Pise, Italie.

[Isa,95] : Isaac.,(1995). Conception et utilisation d'ontologies pour l'indexation de documents audiovisuels.

[Izo et al,00]Izoguchi R., Kozaki K., Sano T. et Kitamura Y.,(2000) . Construction and Deployment of a plant Ontology. The 12th International Conference, EKAW2000,(Lecture Notes in Artificial Intelligence 1937).

[Jar et al ,16] ,(15 aout 2016). « Un système de recommandation d'examineurs et d'experts dans l'examen des problèmes » knowledge-BasedSystems(Systèmes à base de connaissances) volume106,pages 164-178.

[Kha,09] Souheila KHALFI.,(2009). Construction d'une ontologie pour la prise en charge des patients à domicile.

[Kob et al,01] Kobsa, A., Koenemann, J. and Pohl, W.,(2001). Personalised hypermedia presentation techniques for improving online customer relationships. Knowl. Eng. Rev. 16,111-155.

[Lao et Kor,16] Laouar Assia , Korichi Khadidja.,(2016).« Un système de recommandation pour l'assistance à la navigation sur Internet ». Mémoire de Master, Université Kasdi Merbah, Ourgla, Département d'Informatique et des Technologies de l'Information.

[Lav,07] Lavoie B.,(8 février 2007) .« Synthèse de lectures, Notion d'ontologie et construction d'ontologie à partir de corpus de textes », Université du Québec à Montréal.

[Li et Kim, 03] Li, Q. and Kim, B. M.,(2003). An approach for combining content-based and collaborative-filters. In Proceedings of the sixth international workshop on Information retrieval with Asian languages - Volume 11 AsianIR 03 pp. 17-24, Association for Computational Linguistics, Stroudsburg, PA, USA.

[Lie,95] Lieberman, H. (1995). Letizia : An Agent That Assists Web Browsing. In INTERNATIONAL JOINT CONFERENCE ON ARTIFICIAL INTELLIGENCE pp. 924-929.

[Lin et al,03] Linden G., Smith B., and York J.,(2003). Amazon.com recommendations : Item-to-item collaborative filtering. In Proceedings of IEEE Internet Computing, pp. 76–80.

[Maa,16] Maatallah Majda.,(2016). « Une Technique Hybride pour les Systèmes de Recommandation », thèse de doctorat, Faculté des sciences de l'ingénieur, département d'informatique, Université Badji Mokhtar.

[Mal et al,87] Malone, T., Brobst, S., Cohen, S., Grant, K., and Turbak, F.,(1987). Intelligent

- information des systemes de partage. In *Communications of the ACM*, volume 30, pages 390–402.
- [McN et al,02]** McNee, S., Albert, I., Cosley, D., Gopalkrishnan, P., Lam, S., Rashid, A., Konstan, J., and Riedl, J.,(2002). On the recommending of citations for research papers. In the 2002 ACM conference on Computer supported cooperative work, page 125. ACM.
- [Mid et al,04]** MIDDLETON S. E., SHADBOLT N. R. and DE ROURE D. C.,(2004). Ontological user profiling in recommender systems. *ACM Trans. Inf. Syst.*, 22(1), 54–88.
- [Mid et al,01]** MIDDLETON S. E., DE ROURE D. C. and SHADBOLT N. R.,(2001). Capturing knowledge of user preferences : ontologies in recommender systems. In *Proceedings of the 1st international conference on Knowledge capture*, p. 100–107 : ACM.
- [Mil et al,90]** Miller, G. A., Beckwith, R., Fellbaum, C., Gross, D. and Miller, K.,(1990). WordNet : An on-line lexical database. *International Journal of Lexicography* 3, 235-244.
- [Moo et Roy,99]** Mooney, R. J. and Roy, L.,(1999). Content-Based Book Recommending Using Learning for Text Categorization. In *Proceedings of the Fifth ACM Conference on Digital Libraries* pp. 195 (204, ACM Press).
- [Mor et al,13]** MORENO A., VALLS A., ISERN D., MARIN L. and BORRÀS J.,(2013). SigTur/E-Destination : Ontology-based personalized recommendation of Tourism and Leisure Activities. *Engineering Applications of Artificial Intelligence*, 26(1), 633–651.
- [Mus et al,16a]** Musto, C., Semeraro, G., de Gemmis, M., Lops, P.,(2016). Learning Word Embeddings from Wikipedia for Content-Based Recommender Systems. In : , et al. *Advances in Information Retrieval. ECIR 2016. Lecture Notes in Computer Science()*, vol 9626. Springer, Cham. <https://doi.org/10.1007/978-3-319-30671-1-60>.
- [Nec et al ,91]**(1991).Enabling technology for knowledge sharing. *AI Magazine* , 12 (3), 36–56.
- [Ost et al,13]** Ostuni, V. C., Di Noia, T., Di Sciascio, E., and Mirizzi, R.,(2013). Top-n recommendations from implicit feedback leveraging linked open data. In *Proceedings of the 7th ACM Conference on Recommender Systems, RecSys '13*, pages 85–92.
- [Pal et al,18a]** Palumbo, E., Rizzo, G., Troncy, R., Baralis, E., Osella, M., and Ferro, E.,(2018a). An empirical comparison of knowledge graph embeddings for item recommendation. In *DL4KGS 2018, 1st Workshop on Deep Learning for Knowledge Graphs and Semantic Technologies 2018, co-located with the 15th Extended Semantic Web Conference (ESWC 2018)*, pages 14–20.
- [Pal et al,18b]** Palumbo, E., Rizzo, G., Troncy, R., Baralis, E., Osella, M., and Ferro, E.,(2018b). Translational models for item recommendation. In *European Semantic Web Conference (ESWC '18)*, pages 478–490.

- [**Pal et al,20**] Palumbo, E., Monti, D., Rizzo, G., Troncy, R., and Baralis, E. (2020). entity2rec : Property-specific knowledge graph embeddings for item recommendation. *Expert Systems with Applications*, 151 :113235.
- [**Pas,10a**] Passant, A.,(2010a). dbrec—music recommendations using dbpedia. In *International Semantic Web Conference (ISWC '10)*, pages 209–224.
- [**Paz et al,97**] Pazzani, M., Billsus, D., Michalski, S. and Wnek, J.,(1997). Learning and Revising User Profiles : The Identification of Interesting Web Sites. In *Machine Learning* pp. 313-331.
- [**Paz,99**] Pazzani, M. J.,(1999). A Framework for Collaborative, Content-based and Demographic Filtering. *Artificial Intelligence Review*, vol. 13, no. 5-6, 393-408.
- [**Paz et Bil, 07**] Michael J. Pazzani and Daniel Billsus.,(January 2007). Content-based recommendation systems. In Peter Brusilovsky, Alfred Kobsa, and Wolfgang Nejdl, editors, *The Adaptive Web*, number 4321 in *Lecture Notes in Computer Science*, pages 325–341. Springer Berlin Heidelberg.
- [**Per et al,04**] Perugini, S., Gonçalves, M, A., Fox, E. A.,(2004). Recommender Systems Research : A Connection-Centric Survey. *Journal of Intelligent Information Systems*, vol. 23, no. 2, 107-143.
- [**Pes et al,09**] PESQUITA C., FARIA D., FALCÃO A. O., LORD P. and COUTO F. M.,(2009). Semantic similarity in biomedical ontologies. *PLOS Computational Biology*, 5(7), 1–12.
- [**Pop et al,01**] Popescul, A., Ungar, L. H., Pennock, D. M. and Lawrence, S., (2001). Probabilistic Models for Unified Collaborative and Content-Based Recommendation in Sparse-Data Environments. In *Proceedings of the 17th Conference in Uncertainty in Artificial Intelligence UAI '01* pp. 437-444, Morgan Kaufmann Publishers Inc., San Francisco, CA, USA.
- [**Psy et al,03**] syché V, Mendes O. et Bourdeau J.,(2003). Apport de l'ingénierie ontologique aux environnements de formation à distance.
- [**Rac,04**] Rakoto H.,(2004). "Intégration du Retour d'Expérience dans les processus industriels. Application à Alstom Transport", Thèse de doctorat. Institut National Polytechnique de Toulouse.
- [**Res et al,94**] Resnick, P., Iacovou, N., Suchak, M., Bergstrom, P., and Riedl, J.,(1994). GroupLens : an open architecture for collaborative filtering of netnews. In *Proceedings of the 1994 ACM conference on Computer supported cooperative work*, pages 175–186. ACM.
- [**Res et Var,97**] Resnick, P. and Varian, H. R.,(1997). Recommender systems. *Commun. ACM* 40, 56-58.
- [**Rha et al,20**].(2020). « Un système de recommandation hybride pour l'acquisition et désherbage des bibliothèques par les usagers », *Journal of King Saud University - Informatique*

et sciences de l'information.

[Ric,79] Rich, E.,(1979). User modeling via stereotypes*. *Cognitive science*, 3(4) :329–354

[Rij,79] Rijsbergen C.J.V.,(1979). *Information Retrieval*. Second edition, Butterworks.

[Rod et al,15] RODRÍGUEZ-GARCÍA M. Á., COLOMBO-MENDOZA L. O., VALENCIA-GARCÍA R., LOPEZLORCA A. A. and BEYDOUN G.,(2015). Ontology-based music recommender system. In S. OMATU, Q. M. MALLUHI, S. R. GONZALEZ, G. BOCEWICZ, E. BUCCIARELLI, G. GIULIONI and F. IQBA, Eds., *Distributed Computing and Artificial Intelligence*, 12th International Conference, p. 39–46, Cham : Springer International Publishing.

[Ros,06] Rasovska I.,(2006). “Contribution à une méthodologie de capitalisation des connaissances basée sur le raisonnement à partir de cas. Application au diagnostic dans une plateforme d'e-maintenance“. Thèse doctorale. L'UFR des Sciences et Techniques de l'Université de Franche-Comté.

[Sal ,13] Salehi, M. ;(2013). Application of implicit and explicit attribute based collaborative filtering and BIDE for learning resource recommendation. *Data and Knowledge Engineering*, 87, 130-145.

[SAL et al,13] Salehi, M. and Kamalabadi I. N., Ghaznavi M. B.,(2013). Attribute-based Collaborative filtering using genetic algorithm and weighted C-means algorithm. *IJBIS*, 13(3), pp. 265 – 283.

[Sal et al,13] G. Salton et C. S. Yang.,(2013) On the specification of term values in automatic indexing. *Journal of Documentation*., vol. 29, no. 4, pages 351– 372.

[Sal et Kam,13] Salehi, M. and Kamalabadi I. N.,(2013). A hybrid recommendation approach based on attributes of products using genetic algorithm and naive Bayes classifier. *IJBIS*, 13(4), pp. 381 – 399.

[Sar et al,01] B. Sarwar, G. Karypis, J. Konstan, and J. Riedl.,(2001). Item-based collaborative filtering recommendation algorithms. In *Proceedings of the 10th international conference on World Wide Web*, page 285–295.

[Sch et Smo,88] : Schmidt-Schauß and Smolka, Gert Smolka.,(1988). *Attributive concept descriptions with unions and complements*.

[Sch,82] : Schank R.C.,(1982). « *Dynamic Memory : A Theory of Reminding and Learning in Computers and People* ». Cambridge University Press, New York, NY.

[Sch et al,02] Schein, A. I., Popescul, A., H., L., Popescul, R., Ungar, L. H. and Pennock, D. M.,(2002). *Methods and Metrics for Cold-Start Recommendations*. In *Proceedings of the 25th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval* pp. 253-260, ACM Press.

- [Sch et al,01] Schwab, I., Kobsa, A. and Koychev, I.;(2001). Learning User Interests through Positive Examples Using Content Analysis and Collaborative Filtering. In 30 2001.Internal Memo, GMD.
- [Sha et Mae,95a] Shardanand, U. and Maes, P.,(1995a). Social information filtering : algorithms for automating ?word of mouth? In Proceedings of the SIGCHI conference on Human factors in computing systems, pages 210– 217. ACM Press/Addison-Wesley Publishing Co.
- [Sha et al,97] Shapira B., Shoal P., Hannani U.,(1997). Stereotypes in Information Filtering Systems. Information Proceedings 1 Management, 33(3), pp. 273-287.
- [Sha et Sum,13] Sharma, S. K., and Suman, U. (2013). A framework of hybrid recommender system for web personalisation. Journal of Business Information Systems, 13(3), pp. 284-316.
- [She et Mae,93] Sheth, B. and Maes, P.,(1993). Evolving agents for personalized information filtering. pp. 345-352,
- [Sie et al,10] SIEG A., MOBASHER B. and BURKE R.,(2010). Improving the effectiveness of collaborative recommendation with ontology-based user profiles. In proceedings of the 1st International Workshop on Information Heterogeneity and Fusion in Recommender Systems, p. 39–46 : ACM.
- [Stu et al,98] (1998).Knowledge Engineering : Principles and Methods. IEEE Transactions on Data and Knowledge Engineering , 25 (1-2), 161-197.
- [Sur,02] Y. Sure, J. Angele, S. Staab.,(2002). OntoEdit : Guiding Ontology Development by Methodology and Inferencing, In Proceedings of the Confederated International Conferences CoopIS, DOA and ODBASE 2002, volume 2519, pp 1205–1222, Springer-Verlag LNCS.
- [Su et al,09] Su, X., et Khoshgoftaar, T. M., (2009). A survey of collaborative filtering techniques. Advances in artificial intelligence, vol. 2009, p. 19.
- [Ter,93] Terry, D. B.,(1993). A tour through Tapestry. In Proceedings of the conference on Organizational computing systems COCS '93 pp. 21-30, ACM, New York, NY, USA.
- [Ton,07] (2007). De l'apport des ontologies pour la conception de systèmes multi-agents ouverts. Mémoire de Master Recherche Informatique mention Intelligence Artificielle et Extraction de Connaissances, Université de Lyon.
- [Usc et Gru,96] schold M et Grüninger M.,(1996). Ontologies : Principles, Methods and Applications.
- [Upe,94] Upendre,S.,(1994). Social information filtering for music recommendation, Technical Report TR-94-04, MIT EECS, Learning and Common Sense Group, MIT Media Laboratory
- [Vog et al,97] Vogt, C. C., Cottrell, G. W., Belew, R. K. and Bartell, B. T.,(1997).Using Relevance to Train a Linear Mixture of Experts. In [Harman pp. 503-515].

[**Wel et Gua,01**] Welty C.,Guarino N.,(2001). Supporting ontological analysis of taxonomic relationships.

[**Wen et al,08**] Weng L. T., Xu Y., Li Y., Nayak R.,(2008). Exploiting Item Taxonomy for Solving Cold-Start Problem in Recommendation Making. In the 20th IEEE Inter. Conf., Vol. 2, pp. 113-120.