REPUBLIQUE ALGERIENNE DEMOCRATIQUE ET POPULAIRE

MINISTERE DE L'ENSEIGNEMENT SUPERIEUR ET DE LA RECHERCHE SCIENTIFIQUE

UNIVERSITE YAHIA FARES DE MEDEA
FACULTE DES SCIENCES
Département Mathématique et Informatique



Mémoire de fin d'études pour l'obtention du Diplôme Master

Spécialité: Informatique

Option : Ingénierie des Systèmes
et Technologie du Web

Thème:
Système de recommandation hybride

Présenté par

Mlle. BERKANI Rania Sanaa.

Mlle. BOUGHERARA Hafsa.

Proposé et Dirigé par

Mme SADOUKI Fatiha.

Année universitaire : **2018 - 2019**

DEDICACE

Je dédie ce travail :

A la lumière de mes jours, la source de mes efforts, la flamme demon cœur, ma vie et mon bonheur ; maman que j'adore.

A mon père, a mon frère **Amine**, à ma sœur **Lyna**, à la hauteur de l'amour dont ils ne cessent de me combler. Que dieu leur garde pour moi.

A celui qui m'a aidé et soutenu **Mohammed Yassine** tout au long de ce projet.

Aux personnes qui m'ont toujours aidé et encouragé, qui étaient toujours à mes côtés, et qui m'ont accompagnaient durant mon chemin d'études: **Midou, Fidaa, Nazila, Mehdi, Leila.**

A toute ma famille, et mes amis.

A mon binôme **Hafsa.**

Et à tous ceux qui ont contribué de près ou de loin pour que ce projet soit possible, je vous dis merci.

Rania

DEDICACE

Je dédie ce travail :

A mon père
Pour son grand amour, sa patience, son encouragement, son sens du devoir et ses sacrifices pour que je réussisse dans mes études.

A ma mère
Pour son affection, sa patience, son encouragement pendant les épreuves difficiles ainsi que ses prières qui m'apportent bonheur et réussite.

A mes chères sœurs
Pour son amour, pour son soutien, et pour ses encouragements

A mon frère
Qui est toujours à mes côtés, prêt á m'aider.

A ma chère copine **Meriem**
Pour son soutien, et ses encouragements

A mon binôme **Rania**
A mes chères amies

A tous ceux qui m'aiment et à tous ceux que j'aime.
A tous ceux que je connais de près ou de loin.

Hafsa

Remerciement

En tout premier lieu, nous remercions le bon Dieu, Tout Puissant, de nous avoir donné la force pour survivre, ainsi que l'audace pour dépasser toutes les difficultés.

Nous exprimons toute ma gratitude à notre directrice madame Sadouki Fatiha, professeur à l'université DR. Yahia Fares Médéa, pour sa disponibilité.

Nous exprimons nos profonde reconnaissances et nos chaleureux remerciements à nos très chers parents, qui ont toujours été là pour nous, « Vous avez tout sacrifié pour vos enfants n'épargnant ni santé ni efforts. Vous nous avez donné un magnifique modèle de labeur et de persévérance. Nous sommes redevable d'une éducation dont nous sommes fiers ».

Nous désirons aussi remercier nos frères et nos sœurs pour leur soutien inconditionnel et leur encouragement.
Nous tenons à remercier toutes nos amies et nos collègues qui nous ont apporté leur support moral et intellectuel tout au long de notre démarche. Un grand merci pour leur sincère amitié et confiance.

À tous ces intervenants, nous présentons nos remerciements, notre respect et notre gratitude.

ملخص:

توفر أنظمة التوصية حلاً لمشكلة التحميل الزائد للمعلومات، وهي قادرة على تقدير اهتمام المستخدم بمورد معين استنادًا إلى معلومات معينة حول مستخدمين آخرين متشابهين وخصائص الموارد. الغرض من هذه العمل هو اقتراح نهج توصية مختلطة جديد يجمع بين عدة خوارزميات توصية للحصول على توصية أكثر دقة. بالإضافة إلى ذلك ، يعتمد النهج المقترح على بنية الأنطولوجيا المقدمة في بداية النظام ، مما يجعله قابلاً لإعادة الاستخدام وقابل للتكيف بسهولة وقابل للتطبيق على جميع المجالات (الموسيقى ، والمنشورات العلمية ، والأفلام ، إلخ). تم تجربة نظام التوصية المحدد في وضع عدم الاتصال باستخدام التحقق من الصحة على مجموعة بيانات. movieLens

الكلمات المفتاحية: نظام التوصية، التصفية التعاونية، ملف تعريف المستخدم، التشابه الدلالي، علم الوجود.

Résumé :

Les systèmes de recommandation apportent une solution au problème de surcharge d'information et sont capables d'estimer l'intérêt d'un utilisateur pour une ressource donnée à partir de certaines informations relatives à d'autres utilisateurs similaires et aux propriétés des ressources. L'objet de ce mémoire consiste à proposer une nouvelle approche de recommandation hybride combinant plusieurs algorithmes de recommandation afin d'obtenir une recommandation plus précise. De plus, l'approche proposée repose sur la structure de l'ontologie donnée en entrée du système, ce qui la rend réutilisable, facilement adaptable et applicable à tous les domaines (musique, publications scientifiques, films, etc.). Le système de recommandation défini a été expérimenté hors-ligne sur le dataset de MovieLens.

Mots Clés : Système de Recommandation, Filtrage Collaboratif, profil utilisateur, similarité sémantique, ontologie.

Abstract:

The recommendation systems provide a solution to the problem of information overload and are able to estimate the interest of a user for a given resource based on certain information relating to other similar users and the properties of the resources.In this thesis, wedefine a hybrid recommender system which combinesseveral recommendation algorithms in order to obtain more accurate recommendations. Moreover,the defined approach relies on the structure of the input ontology, which makes the frameworkreusable, adaptable and domain-independent (music,research papers, films, etc.). Our recommender system has been experimented offline on the MovieLens dataset.

Keyword: Recommendation system, Collaborative Filtering, user profile, semantic similarity, ontology.

Tables des matières

Liste des Figures

Liste des Tableaux

Introduction générale 1

Partie I : Etat de l'art

Chapitre I : Système de recommandation

I.1 Introduction 4

I.2 Filtrage d'information vs Recherche d'information 5

I.3 Système de recommandation 6

I.3.1Objectif des systèmes de recommandation 7

I.3.2 Historique 7

I.3.3 Concept de base 9

I.3.3.1 Les entités Utilisateur et Item 9

I.3.3.2 Evaluation (Note ou Vote) 9

I.3.3.3 Matrice d'évaluation utilisateur-item 11

I.3.3.4 Notion du profil 11

I.3.3.5 Prédiction 12

I.3.3.6 La recommandation 13

I.4 Les techniques de recommandation 13

I.4.1 Approche basé sur le contenue 13

I.4.1.1 La notion de profil dans les systèmes d'informations 14

I.4.1.2 Le model vectoriel 14

I.4.1.3 Les limitations des approches basées sur le contenu 15

I.4.2 Approche basé sur le filtrage collaboratif 16
I.4.2.1 Filtrage collaboratif basé sur la mémoire 18
I.4.2.2 Filtrage collaboratif basé sur un modèle 19
I.4.3 Les approches hybrides 20
I.4.4 Les approches démographiques 25
I.4.5 Les approches sociales 26
I.5 Avantage et inconvénient de système de recommandation 26
I.6 L'évaluation de la recommandation 28
I.6.1 Techniques d'évaluation 28
I.6.2 Précision de la prédiction de notes 28
I.7 Les problèmes des systèmes de recommandations 31
I.7.1 Le problème du démarrage à froid (cold start problem) 31
I.7.2 Le problème de la Stabilité Vs. Dynamicité (Stability Vs. Plasticity problem) 32
I.7.3 Les Systèmes Statiques 33
I.7.4 Le problème de la parcimonie de données (Sparcity problem) 34
I.7.5 Le problème de l'évolutivité de données 34
I.7.6 Le problème de la sur-spécialisation 34
I.8 Conclusion 35

Chapitre II :Les ontologies et les relations sémantiques

II.1 Introduction 36
II.2 La notion d'ontologie 36
II.3 Définition d'Ontologie 36
II.3.1 Les composantes de l'ontologie 37
II.3.1.1 concepts 37
II.3.1.2 Relations 38
II.3.1.3 Fonctions 38
II.3.1.1 Axiomes 39
II.4. Différents types d'ontologies 40
II.4.1 ontologies de haut niveau 40

II.4.2 ontologies de domaine 40
II.4.3 ontologie de tâche 40
II.4.4 ontologie d'application 40
II.5 WordNet 40
II.6 La mesure de similarité sémantique 41
II.7 les approches de mesure de similarité 42
II.7.1 Approches basées sur l'espace vectoriel 43
II.7.2 Approches basées sur les arcs 45
II.7.3 Approches hybrides 48
II.7.4 Quelques approches de la littérature 49
II.8 Recommandation basée sur la sémantique 50
II.9 conclusion 51

Partie II : Modélisation &Expérimentation

Chapitre III : Modélisation

III.1 Introduction 52
III.2 L'architecture du système 52
III.3 Module collaboratif 53
III.3.1 Principe 53
III.3.2 Coefficient de corrélation 54
III.3.3 Le coefficient de corrélation de Pearson 54
III.4 Module sémantique 55
III.4.1 Similarité sémantique à base d'ontologie 56
III.4.2 La mesure de similarité WUP 56
III.5 L'hybridation 57
III.5.1 Hybridation pondérée 57

III.6 Conclusion 58

Chapitre IV : Expérimentation et évaluation

IV.1 Introduction 59

IV.2 Base de données 59

IV.3 L'environnement de développement 59

IV.4 Module collaboratif 60

IV.4.1 resultat 62

IV.5 Module sémantique 64

IV.5.1 Résultat 68

IV.6 L'hybridation 68

IV.6.1 Résultat 69

IV.7 Discussion des résultats 70

IV.8 Conclusion 72

Conclusion Générale 75

Références bibliographiques

Annexe

Liste des figures

Partie I : etat d l'art

Figure I.1 : Représentation de SR. 6

Figure I.2 : Représentation vectoriel d'un document. 15

Figure I.3 : Approche basé sur le contenu. 16

Figure I.4 : Approches collaboratives. 17

Figure I.5 : Recommandation basé sur le filtrage collaboratif. 18

Figure I.6 : Approche hybride. 20

Figure I.7 : Hybridation pondéré. 21

Figure I.8 : Hybridation à bascule. 22

Figure I.9 : Hybridation mixée. 22

Figure I.10 : Hybridation par combinaison de caractéristiques. 23

Figure I.11 : Hybridation en cascade. 24

Figure I.12 : Hybridation par ajout de caractéristiques. 24

Figure I.13 : Hybridation méta-niveau. 25

Figure II.1 : Types d'ontologie. 39

Figure II.2 : exemple d'une hiérarchie des concepts d'une ontologie. 40

Figure II.3 : Deux instances a et b non similaires avec une mesure de similarité classique. 41

Figure II.4 : Deux instances a et b similaires avec notre mesure de similarité sémantique. 42

Figure II.5 : Taxonomie des approches de mesure de similarité. 43

Figure II.6 : Exemple d'un extrait d'une ontologie. 46

Partie II : Modelisation &Experimentation

Figure III.1 **:** l'architecture du système de recommandation. 52

Figure III.2 :l'architecture du module collaboratif. 53

Figure III.3 : Principe général du filtrage collaboratif. 54

Figure III.4 : la matrice obtenue dans le module collaboratif 55

Figure III.5 : l'architecture du module sémantique. 55

Figure III.6 : la matrice obtenue dans le module sémantique. 57

Figure IV.1 : histogramme de la distribution des notes. 62

Figure IV.2 : la relation entre le classement d'un film et le nombre de classements. 63

Figure IV.3 : résultat des films à recommander du premier systeme. 64

Figure IV.4 : la matrice obtenue (user_id/genre). 66

Figure IV.5 : exemple de calculer les utilisateur similaires. 67

Figure IV.6 : resultat des films recommandés du deuxiéme système 68

Figure IV.7 : resultat de l'hybridation1 70

Figure IV.8 : resultat de l'hybridation2 71

Figure IV.9 : graphe de nombre de films recommandés par les deux systèmes jouant sur α et β.71

Figure IV.10 : résultat final de notre système. 72

Liste des Tableaux

Partie I : etat de l'art

Tableau I.1 : Les avantages et les inconvénients des techniques de recommandation. 27

Tableau II.2 : Classification des résultats possibles pour la recommandation d'items. 31

Tableau II.3 : Bilan des approches et des problèmes inhérents aux SRs. 35

Partie II : modelisation & experimentation

Tableau III.1 : Classification des caractéristiques de quelques travaux de similarités. 50

TableauIV.1 **:** Les bibliothèque utilisé dans notre projet. 60

Tableau IV.2 **:** Résultat de la mesure de similarité. 65

Liste des acronymes

SR : Systèmes de Recommandation

RI : Recherche d'Information

FC : Filtrage Collaboratif

FBC : Filtrage Basé Contenu

KPPV: K-Plus ProchesVoisins

MAE: Mean Absolute Error

RMSE: Root Mean Squared Error

CI: contenu informationnel

I2I: item to item

U2U: user to user

UMLS: Unified Medical Language System

ID: identificateur

FD: Filtrage Démographique

Introduction générale

Introduction Générale

Les systèmes d'informations actuels sont caractérisés par leur volume croissant, leur hétérogénéité, et par le fait qu'ils ne sont pas suffisamment adaptes aux besoins des utilisateurs. Au vu de l'état actuel de ces systèmes en termes d'hétérogénéité de domaines, de sources, de représentation et de structuration des informations, l'accès à une information pertinente et adaptée aux utilisateurs est un vrai challenge. Les besoins de l'utilisateur sont difficiles à traiter, d'une part, parce qu'ils ne sont pas formulés explicitement et, d'autre part, parce qu'ils sont évolutifs. La popularisation d'internet ainsi que l'explosion des services de recommandation de nos jours ont propulsé la recherche d'information (RI) au premier plan.

Les systèmes de recommandation sont en pleine expansion depuis le début des années 1990. Ils offrent aux utilisateurs la possibilité de mieux choisir les articles parmi les différents contenus disponibles dans les systèmes d'information. De part leur utilité, les systèmes de recommandation ont gagné plusieurs domaines : des sites de e-commerce aux publications scientifiques en passant par la presse et les plateformes de locations de films. Leurs enjeux sont donc devenus considérables, non seulement du point de vue économique pour les systèmes commerciaux, mais également, de façon plus générale, du point de vue de la satisfaction de l'utilisateur. En effet, l'utilisateur représente le cœur de tels systèmes.

La compréhension et l'anticipation de ses attentes et besoins sont essentiels à l'obtention de son contentement.

Pour générer des recommandations, un certain nombre d'approches ont été identifiées**,** où les plus utilisées sont le Filtrage Collaboratif et la Filtrage basé Contenu. Les deux techniques ont leurs forces et faiblesses**,** ou sur une combinaison de ces deux techniques ou d'autres «**approches hybrides qui combinent plusieurs approches de recommandation afin de gagner en performance et de pallier les limitations des approches simples »** [1].]. Le FC recommande des produits en se basant sur la similarité des préférences d'un groupe de clients connus comme étant des voisins. Cela suppose que les utilisateurs ayant des intérêts communs, dans le passé, continueront, probablement, à partager les mêmes goûts dans le futur.

Objectif

Dans ce mémoire nous proposons un système de recommandation hybride combine entre l'approche basé sur FC et l'approche basé sur une mesure de similarité sémantique en présence d'une anthologie. L'approche adoptée est en général dans le sens où nous pouvant l'appliquer pour recommandation de contenus en général (livre, Url, article, produit, film, chanson, etc.). Nous illustrons son principe pour affiner la recommandation à un utilisateur. Deux objectifs principaux ont donc été définis pour l'élaboration de notre système de recommandation : la capacité du système de recommandation à s'adapter aux utilisateurs en fonction de leur comportement. Et la combinaison de plusieurs approches de recommandation afin d'améliorer les performances du système.

STRUCTURE DU MEMOIRE

Le mémoire est organisé en deux parties comme suit :

- **Partie I - Etat de l'art**

Est divisé en deux parties :

- Le premier chapitre présente une vue générale sur les systèmes de recommandation ainsi que leurs avantages et inconvénients. Nous définissons d'abord ce qu'est un système de recommandation, ses origines, les concepts de base et notions liées, sa formalisation, Ensuite nous détaillons ses différents techniques et approches de recommandation existantes, ainsi que leurs limitations, en terminant par les techniques d'évaluation et de validation des performances du système vis-à-vis des objectifs définis.
- Le deuxième chapitre est consacré à définir les différentes mesures de similarité et présenter une classification de ces mesures et éclaircir la notion de l'ontologie.

- **Partie II - Modélisation et Expérimentation**

Cette partie est consacrée à la définition du système que nous proposons. Celui-ci répond aux objectifs fixés dans ce mémoire.

- Dans le premier chapitre nous décrivons dans un premier temps l'architecture globale de notre modèle et le calcul de la recommandation. Le module collaboratif est d'abord détaillé. Puis le module sémantique avec la mesure sémantique intervenant dans le processus de recommandation. Les modules de recommandation, ainsi que leur hybridation, sont ensuite abordés en détail.

- Dans Le deuxième chapitre, Nous avons expérimenté le système de recommandation défini dans le domaine des films.
- Nousdétaillonsdanscettepartielesdifférentesétapesdel'expérimentationet analysons les résultats obtenus.

Nous clôturons ce mémoire par la conclusion générale.

PARTIE I :
Etat de l’art

Chapitre I

Système de recommandation

I.1 Introduction

Dans les dernières décennies, la quantité croissante de produits et services offerts et la possibilité de collecter des quantités importantes de données a nécessité de nouvelles techniques pour en faire le tri. Plutôt que de devoir filtrer manuellement ces quantités importantes d'offres changeant au fil du temps.

Avec la très grande masse d'information disponible sur l'Internet, il reste très difficile pour un utilisateur de trouver l'information qui l'intéresse. Au lieu de laisser ce dernier dépenser son temps à chercher l'information pertinente, le développement du Web a créé un besoin de nouvelles techniques pour aider les utilisateurs à trouver ce qu'ils recherchent mais aussi pour faire savoir qu'une information existe, des systèmes de recommandations ont été créés pour en présenter un petit nombre d'articles ou items à l'utilisateur, personnalisé pour que chacune de ces recommandations soit appropriée et intéressante pour celui-ci. Ces systèmes ont permis le succès de plusieurs entreprises dont par exemple Amazon[1] et Netflix[2].

Nous allons dans ce chapitre dans un premier temps expliquer la différence entre le filtrage d'information et la recherche. Ensuite, d'information détailler le système de recommandation, son objectif et son historique, ainsi que les approches des systèmes de recommandations en se basant sur l'approche hybride, et finalement par expliquer comment évaluer les systèmes de recommandation, et citer les avantages et les problèmes que ses systèmes souffrent.

[1].http ://www.amazon.com/
[2]www.netflix.com

I.2 Filtrage d'information vs Recherche d'information

La différence entre le filtrage d'information et la recherche d'information n'est pas toujours claire, mais une première différence peut être tirée de leur définition. 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 [2], 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 [3]. Un système de recherche d'information a pour but de guider les utilisateurs à retrouver les items qui leur permettent de répondre à leurs besoins, alors qu'un système de filtrage leur recommande directement les items intéressants qui conviennent à leurs préférences et besoins. Dans un système de recherche d'information, l'utilisateur saisie une requête et le système la compare avec les sources d'information disponibles, pour trouver un ensemble de documents répondant à sa requête. Les sources d'information utilisées par le système de recherche d'information sont un ensemble de données collectées dans des bases de données et souvent indexés par un ensemble de mots clés.

Dans un système de filtrage d'information, le système compare l'ensemble des informations disponibles aux profils des individus ou groupe d'individus, afin de leur suggérer les items et\ ou informations correspondants à leurs besoins. Dans www.amazon.com les systèmes de recommandation, les items et\ ou informations proviennent directement de la part des producteurs ou bien collectés dans des bases de données. Par conséquent, un certain nombre de points de distinction entre la recherche d'information et le filtrage d'information peuvent être cités :

- Dans un système de recherche d'information, les données sont sélectionnées à partir d'une base de données statique, alors que le filtrage d'information établit une sélection ou bien une élimination d'information à partir d'une source de données dynamique.
- Un système de recherche d'information ne peut traiter qu'une seule requête à la fois, ce qui le rend capable d'être utilisé que par une seule personne à un moment donné, tandis qu'un système de filtrage peut être utilisé par une ou plusieurs personnes en même temps.
- Un système de recherche d'information collecte et organise les documents selon la recherche des individus, tandis qu'un système de filtrage d'information distribue automatiquement des documents\ et ou des items aux personnes qui en ont besoin.

- Un système de recherche d'information utilise le principe de la découverte de l'information dans la base de données alors que le filtrage d'information utilise un processus d'élimination d'information du flux de données entrant.
- Un système de recherche d'information utilise les requêtes fournies par l'utilisateur qui représentent des intérêts à court terme. Par contre, le filtrage d'information utilise les profils des utilisateurs qui représentent généralement des intérêts à long terme, et qui peuvent être considérés comme des descriptions plus exactes des intérêts des utilisateurs que les requêtes.
- Un système de recherche d'information autorise l'utilisateur à interagir avec un document pendant une seule session de recherche, alors que le filtrage d'information permet l'interaction dans différentes sessions et prend en considération les changements de profils à travers les différentes sessions.

I.3 Système de recommandation

Les systèmes de recommandation sont des techniques spécifiques de filtrer les informations visant à présenter et proposer les éléments d'information qui sont susceptibles d'intéresse à un utilisateur actif, pour résoudre le problème de surcharge et de richesse d'information disponible.

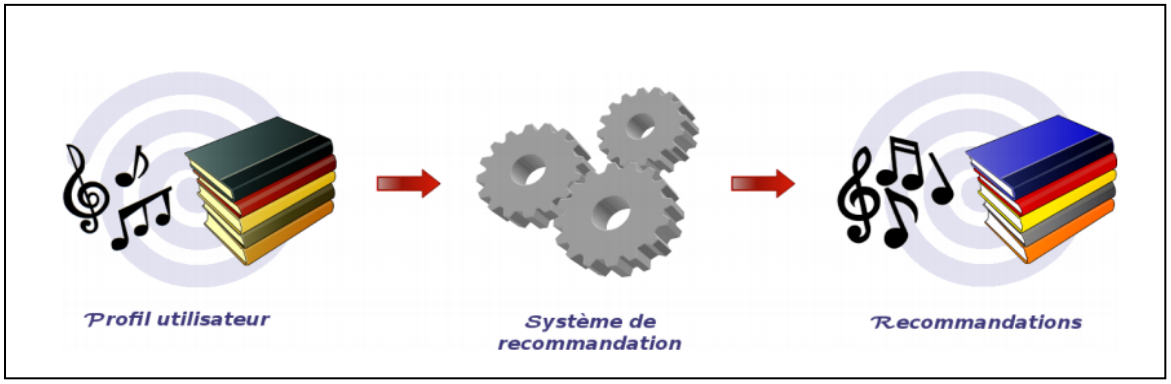


Figure I.1. *Représentation de SR*

Dans tous les systèmes de recommandations, il y a deux entités de base apparaissent sont l'utilisateur, C'est la personne qui utilise le système de recommandation qui donne son opinion et l'item qui est le terme général utilisé pour désigner ce que le système recommande au l'utilisateur.

Les systèmes de recommandation offrent aux utilisateurs la possibilité de mieux choisir parmi les différents contenus disponibles dans les systèmes d'information. Ils sont donc un outil de personnalisation de l'accès à l'information. Ils ont été étudiés dans des domaines divers et variées comme le web, le e-commerce et bien autres.

I.3.1 *Objectif des systèmes de recommandation*

Un système de recommandation a pour objectif de fournir à un utilisateur des ressources pertinentes en fonction de ses préférences. Ce dernier voit ainsi réduit son temps de recherche mais reçoit également des suggestions de la part du système auxquelles il n'aurait pas spontanément prêté attention. L'essor du Web et sa popularité ont notamment contribue à la mise en place de tels systèmes comme dans le domaine du e-commerce. Les systèmes de recommandation peuvent être vus initialement comme une réponse donnée aux utilisateurs ayant des difficultés à prendre une décision dans le cadre d'utilisation d'un système de recherche d'information "classique"[4].

La recherche d'information [5] sur fonde sur un principe d'indexation des données afin de répondre aux requêtes d'utilisateurs. Plus spécifiquement, la recherche documentaire, sous discipline de la recherche d'information, consiste à interroger une base de connaissance par le bais de requêtes écrites en langues naturelles ou bien sous forme de mots clefs (nommées requêtes ad hoc). Un exemple bien connu d'outils utilisant les fondements de cette discipline sont les moteurs de recherche Web comme Google ou Yahoo!

Avant de présenter plus en détail ces types de recommandation, revenons sur l'historique de la mise en place de systèmes de recommandation.

I.3.2 *Historique*

Les systèmes de recommandation ont été utilisés afin de faire face au problème 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.

≪Information Lens System≫ [6] peut être considère comme le premier système de recommandation. 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 première définition pour le filtrage a été donnée aussi par Malone :" *Même si le terme a une connotation littérale de laisser les choses dehors (Filtrage négatif : enlèvement), nous l'utilisons ici dans un sens plus général qui consiste à sélectionner les choses à partir d'un ensemble plus large de possibilités (filtrage positif : sélection)* ".

La littérature académique a introduit le terme de filtrage collaboratif par le système ≪Tapestry≫ [7], Il a été développe en 1992 par le centre de recherche de "Xerox" aux Etats Unis, il s'agit d'un système de recommandation intègre aucune application de mail électronique qui a permis aux utilisateurs de créer des requêtes permanentes, basées sur les annotations (les tags) des utilisateurs. Quelques années plus tard, un certain nombre de systèmes académiques de recommandation ont vu le jour en 1994 et en 1995, tels que le système de recommandation d'articles d'actualités et de films développer par "GroupLens[1]" [8] et le système de recommandation de musique "Ringo" propose par [9]. Ces deux systèmes sont également basés sur le filtrage collaboratif, des livres [8], des vidéos, des films, des pages Web, des articles de nouvelles Usenet et des liens Internet.

Par la suite, avec l'essor de l'Internet et des applications Web, il y a eu un engouement pour les systèmes de recommandation qui se sont développés dans différents domaines d'applications:

- Les systèmes de recommandation de films, tels que : Movielens[2] [10] et Eachmovie [11].
- Les systèmes de recommandation de livres (Bookcrossing).
- Les systèmes de recommandation de musique (LastFM6).
- Les systèmes de recommandation d'articles d'actualités.
- Les systèmes de recommandation de blagues (Jester7 [7]).
- Les systèmes de recommandations introduits sur des sites e-commerce (Amazon).
- Les systèmes de recommandation de restaurants [1].
- Les systèmes de recommandation intègres aux Extranets documentaires (l'Extranet documentaire du Crédit Agricole).

[1].http://www.grouplens.org/
[2]wwww.movielens.org

• Les systèmes de recommandations intègres aux moteurs de recherche(Le moteur de recherche d'AOL).

Pour tous les systèmes de recommandation développés jusqu'au nos jours, la collecte de données relatives aux utilisateurs et/ou aux items, représente une phase clé dans le processus de personnalisation. La section qui suit décrit en détails la typologie de données exploitables par les systèmes de recommandation ainsi que les enjeux liés à leur collecte [12].

I.3.3 *Concept de base*

Quelques concepts relatifs aux systèmes de recommandation : [13]

I.3.3.1 *Les entités Utilisateur et Item*

Dans tout système de recommandation, il existe deux entités importantes qui sont les utilisateurs et les items.

- ✓ *Utilisateur* : est une personne qui accède au système est fait l'enregistrement, en saisissant ses informations démographiques, ses centres d'intérêts et d'autres informations personnelles. L'ensemble des utilisateurs dans le système est représenté par U, où un utilisateur donné $u \in U$.
- ✓ *Item*: dans les systèmes de recommandation, un item est l'entité qui représente tout élément constituant une liste de recommandation et qui correspond aux besoins de l'utilisateur, incluant tout produit susceptible d'être vendu (livre, produits,…ect dans les sites du e-commerce tel que Amazon.com), vu (les films dans les sites de TV en ligne tel que Netflix), écouté (la musique) ou lu (tel que les informations dans les journaux en ligne, les revues dans les bibliothèques numériques), ainsi que les destinations de vacance, des restaurants, etc. Notons qu'un item peut être aussi un individu ou un ensemble d'individus suggérés à l'utilisateur dans les réseaux sociaux. L'ensemble des items disponibles dans le système est représenté par I, où $i \in I$.

I.3.3.2. Evaluation (Note ou Vote)

Une évaluation est une valeur numérique dans une échelle quelconque (la plus utilisée est [1-5]) ou binaire (aimer\ Ne pas aimer, bon\ mauvais, etc.) qui représente la préférence ou non d'un item donné par un utilisateur. L'évaluation donné par un utilisateur *u* à un item *i* est représenté par un triplet $<u, i, r>$. Où, une note de 5, par exemple, exprime une grande préférence et une note de 1 indique une faible préférence i.e. l'utilisateur n'a pas aimé l'item.

La qualité des systèmes de recommandations est en général évaluée selon quatre critères :

- **Pertinence :** le facteur de succès le plus important d'un système de recommandations est la pertinence, c'est-à-dire le filtrage de tous les items possibles pour recommander ceux qu'un usager trouvera les plus intéressants. Toute la difficulté du problème provient de cette découverte de ce qu'un utilisateur considèrera intéressant.
- **Nouveauté :** un utilisateur à qui on ne recommandera que des items qu'il a déjà achetés ou consommés se désintéressera bien vite du système de recommandations. Pour éviter cela, le système devra être en mesure de recommander de nouveaux items. Toutefois, pour certains contextes, recommander des items déjà achetés peut-être positif ; on peut penser à des lames de rasoir qui doivent être remplacées fréquemment.
- **Sérendipité :** la sérendipité d'une recommandation s'apparente à l'expérience d'un heureux hasard, c'est-à-dire que l'utilisateur n'aurait jamais pensé à chercher cet item, mais qu'il trouve celui-ci intéressant. La sérendipité diffère de la nouveauté par le fait que, contrairement à cette dernière, elle procure un effet de surprise. Par exemple, si une utilisatrice aimant beaucoup les films d'amour se fait recommander le nouveau film sortant à la Sainte Valentin, il y aura présence de nouveauté, mais pas de sérendipité. Mais si elle voit sur sa liste, *Mr. Nobody*, un film de science-fiction tournant autour d'une relation amoureuse, celui-ci sera perçu comme un heureux hasard si elle n'a pas l'habitude d'écouter de la science-fiction.
- **Diversité :** la diversité est le fait que toutes les recommandations faites ne sont pas similaires, mais varient légèrement sur différents aspects que l'utilisateur trouve intéressants. Par exemple, un utilisateur utilisant un système de recommandations pour trouver des activités pourrait apprécier de se faire suggérer à la fois des groupes de lectures et des excursions en montagne.

La façon dont un système de recommandations offre ses suggestions se décline de deux façons: évaluer le score qu'un utilisateur mettrait à chaque item ou ordonner la liste des items dans l'ordre de leur pertinence.

La première approche peut être vue comme un problème d'imputation de valeurs manquantes dans une matrice de scores d'utilisateurs. Pour u utilisateurs et n items, cette matrice $\boldsymbol{R}$ sera de taille $u \times i$ avec chaque élément Rij correspondant au vote de l'utilisateur i pour l'item j.

Le but du problème devient donc d'utiliser les valeurs déjà présentes et possiblement d'autres données externes pour prédire les valeurs manquantes de cette matrice.

La deuxième consiste à créer une liste de recommandations avec un ordre total ou partiel, c'est-à-dire ordonner tous les items possibles ou simplement retourner les *k* meilleurs dans un ordre arbitraire. Bien que, souvent, cette liste est créée à partir de la première approche, elle peut aussi être faite à partir de mesures de similarité qui ne dépendent pas d'un score quelconque. On peut penser par exemple à la recommandation d'un film basée sur la similarité de sa description avec celle de films précédemment vus par l'utilisateur.

I.3.3.3 Matrice d'évaluation utilisateur-item

L'ensemble de tous les triplets du système <*u, i, r*> sont enregistrés dans une base de données creuse appelée *Matrice d'Evaluation* (*Rating Matrix*) ou encore *Matrice utilisateur-item* (*user-item Matrix*) et elle est notée par *R*, où chaque ligne correspond aux évaluations fournies par un seul utilisateur et une colonne correspond aux évaluations qu'a eu un seul item par l'ensemble des utilisateurs.

La matrice d'évaluation utilisateur-item est l'entrée pour les systèmes de recommandation et la base des techniques du FC, qui utilisent les préférences (votes) pour la génération des recommandations. Afin d'améliorer la performance des systèmes de recommandation, plusieurs recherches ont apparues offrant de nouveaux scénarios pour introduire de nouvelles informations à la matrice d'évaluation [14].

I.3.3.4 Notion du profil

De façon générale, le profil d'un objet est un ensemble de caractéristiques permettant de l'identifier ou de le représenter. Deux types de profils peuvent être exploités dans les systèmes de recommandation, correspondant aux deux entités utilisées dans ces systèmes : l'utilisateur et l'item.

1) Le profil utilisateur: il s'agit d'une description des caractéristiques de l'utilisateur qui peuvent être ses centres d'intérêt, ses données démographiques, ou bien ses préférences exprimées sous forme d'évaluations, etc. Plusieurs approches d'acquisition des informations sur l'utilisateur afin de construire son profil existent, et peuvent être regroupées en approches manuelles et approches automatiques ou semi-automatiques [1].Les approches manuelles se fondent sur l'intervention de l'utilisateur, alors que les approches automatiques déduisent automatiquement le profil

de l'utilisateur. Parmi les approches automatiques ou semi-automatiques, on distinguer le 'profiling' [15] et les approches par stéréotypes [16].

Le 'profiling'consiste à examiner, analyser et enregistrer les actions et successions d'actions d'un utilisateur lors des différentes sessions de recherche et d'interaction avec le système pour déterminer son profil. En revanche, l'approche par stéréotype est fondée sur l'identification des groupes d'utilisateurs et la détermination des critères de chaque groupe. Un stéréotype d'un utilisateur dans un système de recommandation consiste en un vecteur d'items et leurs appréciations qui augmentent d'une façon continue tant que l'utilisateur interagit avec le système tout au long du temps [1].

2) Le profil item: il correspond à la description de l'item avec un ensemble de caractéristiques, appelées aussi attributs ou propriétés, par exemple dans un système de recommandation de films, les items (films) sont représentés par leurs Ids, titre, genre, réalisateur, année de production, les acteurs principaux. Tandis que dans un système de recommandation de document, les caractéristiques sont des mots clés qui décrivent le contenu sémantique du document. Ces mots clés et leurs poids sont obtenus en général par une opération d'indexation [17].

I.3.3.5 Prédiction

La prédiction est le calcul de la note probable que l'utilisateur va attribuer à un item qu'il n'a pas encore vu ou évalué.

En général, les matrices d'évaluation ont seulement quelques cellules contenant des valeurs tandis que les autres ont des valeurs inconnues et dans la majorité des cas elles ont à l'intérieur un"0", ce qui donne des matrices creuses. Donc, la densité de ces matrices ne sera pas suffisante pour générer des recommandations précises. Par conséquent, les méthodes de prédiction des évaluations manquantes sont utilisées pour augmenter la densité de la matrice utilisateur-item en vue de faire des recommandations plus puissantes et plus pertinentes.

Le calcul de la prédiction se base sur l'utilisation des notes données par les voisins de l'utilisateur (prédiction basée utilisateur) ou attribuées aux items voisins de l'item test évalué par l'utilisateur actif (prédiction basée item), ou bien donné par un modèle (prédiction basée modèle). Ensuite, les items ayant les plus grandes valeurs de prédictions seront recommandés à l'utilisateur.

I.3.3.6 La recommandation

La recommandation est l'action de calculer une liste d'items (Top-N items) que l'utilisateur aimera le plus. Le calcul des listes de recommandation se fait en attribuant des scores pour les items selon leurs popularités ou leurs préférences [18], par exemple. Contrairement à la prédiction, le calcul des recommandations ne se base pas strictement sur les évaluations.

I.4 Les techniques de recommandation

Les approches proposées des recommandations à un utilisateur. Les trois approches les plus courantes sont celles basées sur le contenu comme dans Pandora, les approches collaboratives celle d'Amazon.com par exemple et les approches hybrides (qui sont une combinaison des deux précédentes) par exemple celle de Netflix.

I.3.1 *Approche basé sur le contenue*

Le filtrage basé sur le contenu peut être vu comme un système de recherche d'information, dont la fonction de correspondance entre une requête et un corpus de documents joue le rôle d'un filtre permanent entre un profil (sorte de requête à long terme et évolutif) et le flot de documents entrant (sorte de corpus évolutif) [1].

Un système de filtrage basé contenu suit deux fonctionnalités principales :

- La sélection des items et\ ou documents pertinents vis-à-vis du profil de l'utilisateur.
- La mise à jour du profil de l'utilisateur en fonction du retour de pertinence fourni par l'utilisateur sur les items et\ ou documents qu'il a reçus. La mise à jour se fait par l'intégration des caractéristiques des items et\ ou des thèmes abordés dans les documents jugés pertinents.

Le filtrage basé sur le contenu compare premièrement les items non évalués par l'utilisateur avec son profil, représenté par l'ensemble des items qu'il a évalués, en calculant la similarité entre eux, i.e. c'est une corrélation item à item [19].Ensuite, il recommande les items les plus proches des préférences de l'utilisateur [20]. Donc, le processus de filtrage basé sur le contenu nécessite deux constituants essentiels qui sont : les profils des items et les profils des utilisateurs, car les recommandations se génèrent en se basant sur la corrélation entre les profils de ces deux entités. Les plus utilisées dans la littérature pour la construction des profils utilisateurs et items dans les systèmes à base de FBC.

I.4.1.1 La notion de profil dans les systèmes d'informations

Les approches présentées dans [21] ne sont pas considérées comme des systèmes de recommandation car se contentant d'effectuer un classement d'items possibles. Notons en revanche qu'en introduisant un simple seuil au-delà duquel les items sont considérés comme cohérent vis-à-vis du profil de l'utilisateur (dans l'approche de [21] étant définis par les requêtes qu'il a formulées), il est qualifié de recommandation par filtrage basée sur le contenu. En effet, ce type de filtrage est souvent considère comme une évolution du filtrage d'information issu du domaine de la recherche d'informations.

A l'origine, les travaux sur le filtrage par le contenu à l'instar de ceux sur le filtrage collaboratif (qui sera présenté à la suite), proviennent d'un besoin de sélectionner de l'information pertinente dans les groupes de discutions (Newsgroups) pour des utilisateurs. Les méthodes traditionnelles ayant montré leurs limites comme dans les travaux de [22] qui introduisent un certain nombre d'heuristiques afin de réduire le nombre de documents proposées aux utilisateurs en affinant notamment la sélection de descripteurs. Sont alors apparus les premières approches de construction de profils d'utilisateurs basés sur le contenu.

I.4.1.2 Le model vectoriel

Un certain nombre d'approche étant basé sur des modèles vectoriels. L'idée d'une représentation de documents dans un espace vectoriel introduit la notion de matrice numérique. Cette dernière va être constituée de m x n cellules. En d'autres termes, elle possédera n colonnes de dimension m ou bien m rangs de dimension n. Outre la facilité de l'approche afin de représenter des données, cette matrice va également permettre d'effectuer un certain nombre de calculs vectoriels.

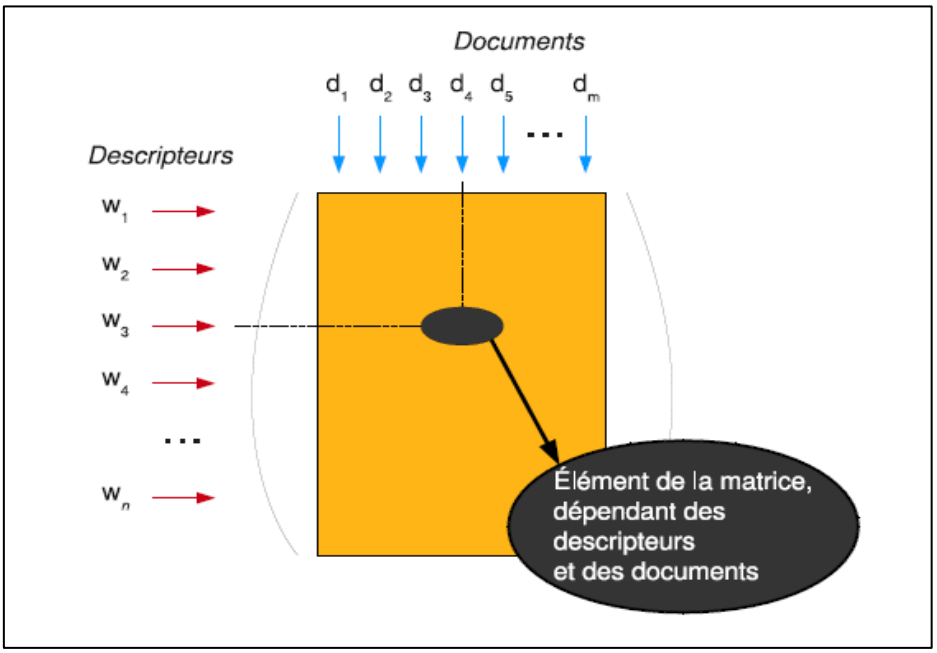


Figure I.2.*Représentation vectoriel d'un document*

Ainsi, les approches de ce type considèrent les mots comme de simples séquences non ordonnées. Le principe général est de représenter un corpus, contenant un ensemble de documents formés de descripteurs (souvent le mot) par une matrice de type documents/descripteurs telle que représentée dans la figure 1. Le principe est de définir chaque vecteur de descripteurs en fonction des documents dans lesquels ils apparaissent.

I.4.1.3 Les limitations des approches basées sur le contenu

Les systèmes de recommandation basés sur le contenu souffrent de limitations, notamment le démarrage à froid et la sur-spécialisation [23].

On parle de démarrage à froid (cold-start) lorsque le système de recommandation peine à recommander des items à un utilisateur ayant consulté et noté peu d'items (nouvel utilisateur par exemple). Le plus souvent, ce problème est contourné en s'aidant des informations démographiques des utilisateurs. En effet, des utilisateurs appartenant à la même tranche d'âge et ayant rempli un profil plus ou moins similaire peuvent avoir les mêmes intérêts. Par conséquent, il est pertinent d'utiliser cette information. D'autres systèmes préfèrent attendre que les utilisateurs aient noté assez d'items avant de leur proposer des recommandations [24].

Des systèmes de recommandation conversationnels reposant sur des approches basées sur la connaissance ou l'utilité sont également utilisés dans ces cas de figure I.3 [25].

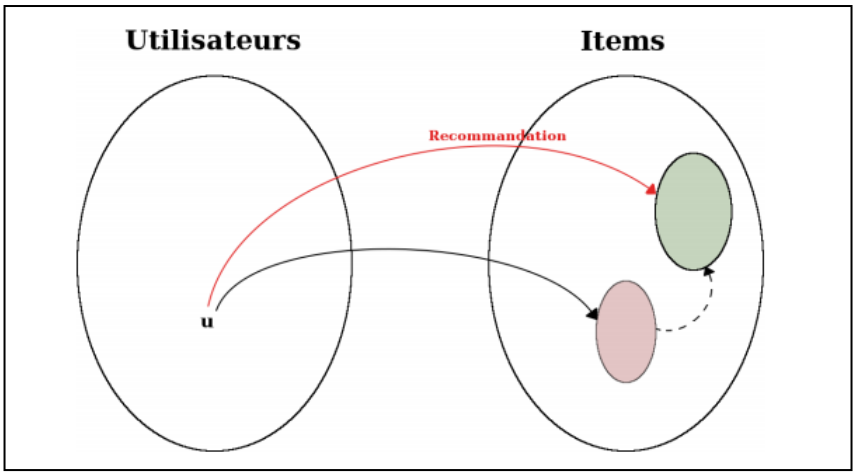


Figure I.3. *Approche basé sur le contenu*

Ils consistent à poser des questions à l'utilisateur sur ce qu'il cherche, et adapter les questions suivantes en fonction des réponses fournies et du moteur de connaissances mis en place.

La sur-spécialisation signifie que la recommandation tend à toujours s'orienter vers des items très similaires du point de vue de leurs caractéristiques. Un système de recommandation de livres pourrait par exemple ne proposer que des livres du même auteur. Afin de pallier ce problème, une touche de hasard est souvent introduite dans les algorithmes en utilisant par exemple des algorithmes génétiques [26]. Autrement, les objets très différents des habitudes (ou simplement de l'historique) de l'utilisateur ne pourront jamais lui être recommandés.

I.4.2 *Approche basé sur le filtrage collaboratif*

S’appuie sur les appréciations données par un ensemble d’utilisateurs sur un ensemble d’articles. Ces appréciations, traduites en valeurs numériques, peuvent être des notes, des comptes d’achats effectues, des nombres de visites, etc. On distingue deux grandes approches de filtrage collaboratif. L’approche se référant aux utilisateurs [8] consiste à comparer les utilisateurs entre eux et à retrouver ceux ayant des gouts en commun, les notes d’un utilisateur étant ensuite prédites selon son voisinage. L’approche se référant aux articles [27] consiste à rapprocher les articles apprécies par les mêmes personnes et àprédire les notes des utilisateurs en fonction des articles les plus proches de ceux qu’ils ont déjà notes.

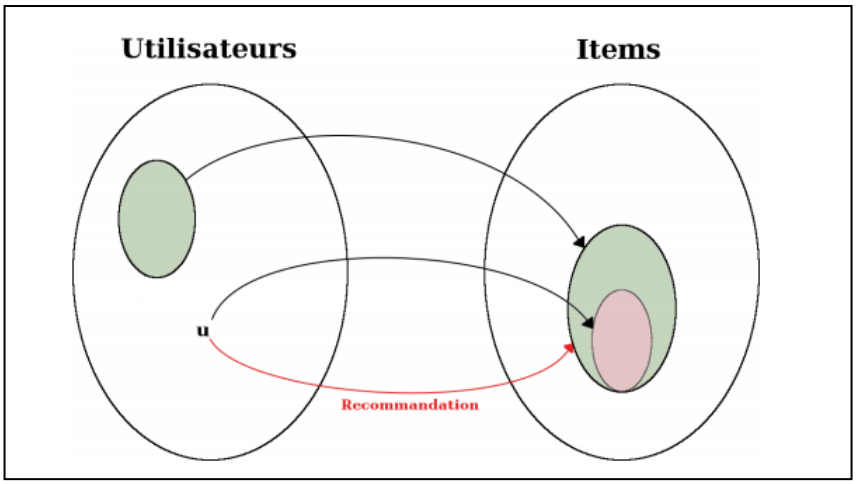


Figure I.4. *Approches collaboratives*

Dans un système du filtrage collaboratif, il faut que les utilisateurs fournissent des évaluations des items qu'ils ont déjà utilisés, sous forme des notes, pour constituer leurs profils. Il n'y a aucune analyse du sujet ou du contenu des objets à recommander. Ce type de système est très efficace au cas où le contenu des objets sont complexes, il est compliqué ou impossible de l'analyser, l'utilisateur peut apercevoir divers domaines intéressants, car le principe du filtrage collaboratif ne se fonde absolument pas sur la dimension thématique des profils, et n'est pas soumis à l'effet ≪ entonnoir ≫.

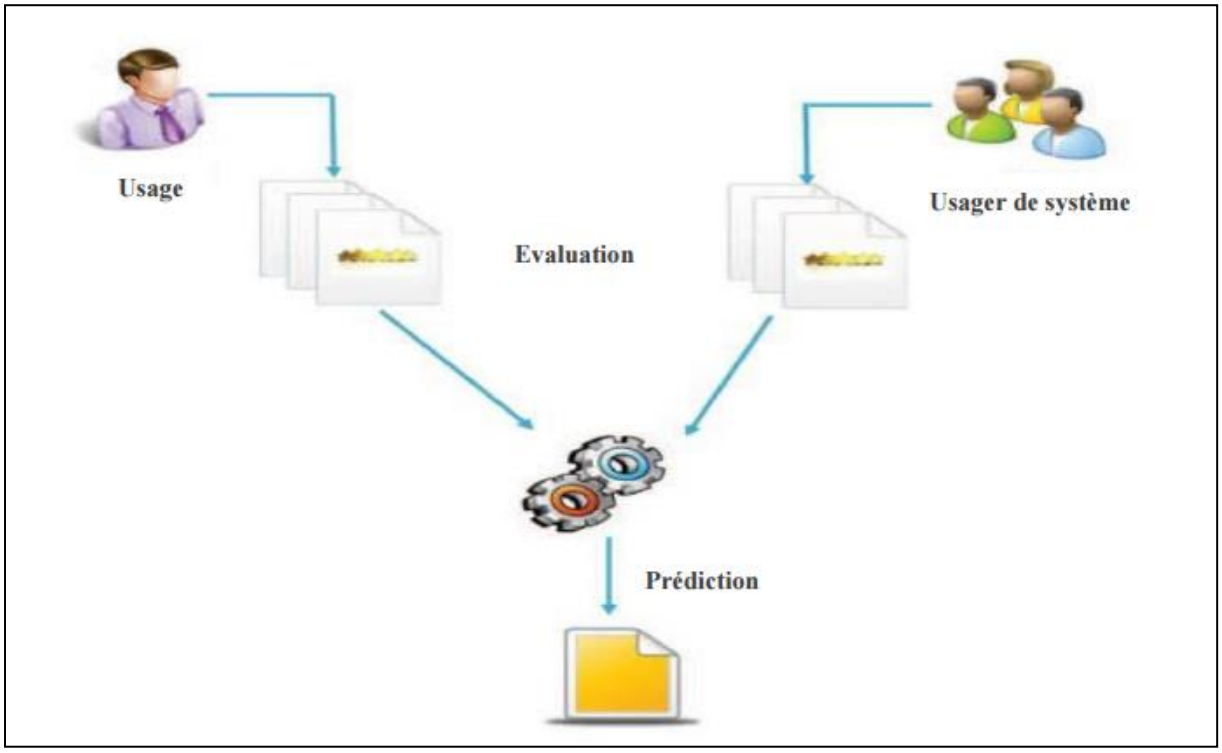


Figure I. 1.*Recommandation basé sur le filtrage collaboratif*

Le filtrage collaboratif est très utilisé par les systèmes vu ses avantages, parmi ces systèmes on peut citer : Amazon, Netflix, MovieLens, Jester, Citeseer, Tapestry, Phoaks, etc.

L'exploitation des données disponibles dans un système de filtrage peut sefaire de plusieurs manières. Ces méthodes sont classées en deux familles principales : les algorithmes bases mémoire est les algorithmes bases modelé.

I.4.2.1 Filtrage collaboratif basé sur la mémoire

Le filtrage collaboratif base sur la mémoire utilise une matrice des votes contenant des préférences des utilisateurs pour prédire des sujets additionnels ou des produits auxquels un nouvel utilisateur peut-être s'intéresse. L'objectif d'un filtrage collaboratif base sur la mémoire est de prédire l'utilité des ressources (items) pour un utilisateur particulier (l'utilisateur actif) base sur la base des votes d'utilisateur. Ainsi, dans le filtrage base sur la mémoire, les notes des utilisateurs stockées par le système sont directement utilisées pour prédire les notes pour de nouveaux items. Cela peut être fait de deux manières connues sous le terme de recommandations basées sur les utilisateurs ou recommandations basées sur les items.

1) Filtrage collaboratif basé sur la mémoire (utilisateurs)

Les systèmes basent sur le voisinage utilisateur, évaluent l'intérêt d'un utilisateur pour un item en utilisant les notes de cet item. Ces notes sont données par d'autres utilisateurs, appelés voisins, qui ont des habitudes de notation similaires. Les voisins d'un utilisateur sont typiquement les utilisateurs dont les notes sur les items sont les plus proches de celles de sur ces items. Les plus proches voisins sont les utilisateurs les plus similaires dans leur notation. En se basant sur le profil d'un utilisateur u_i, le système recherche les utilisateurs u_j (j diffère de i) qui lui sont les plus similaires.

2) Filtrage collaboratif basé sur la mémoire (items)

Alors que les méthodes basées sur le voisinage utilisateur s'appuient sur l'avis d'utilisateurs partageant les mêmes idées pour prédire une note, les approches basées sur les items prédisent la note d'un utilisateur u pour un item *i* en se basant sur les notes de *u* pour des items similaires à *i*. Dans de telles approches, deux items sont similaires si plusieurs utilisateurs du système les ont notés d'une manière similaire.

I.4.2.1 Filtrage collaboratif basé sur un modèle

Le deuxième type d'algorithmes, est comme le nom l'indique bases sur des modèles, supposes réduire la complexité. Ces modèles peuvent être probabilistes et utiliser l'espérance de l'évaluation pour calculer la prédiction. Comme ils peuvent être bases sur des classificateurs permettant de créer des classes pour réduire la complexité.

1. Modèle de Clustering

Les méthodes de Clustering permettent de limiter le nombre d'individus considères dans le calcul de la prédiction. Le temps de traitement sera donc plus court et les résultats seront potentiellement plus pertinents puisque les observations porteront sur un groupe le plus proche de l'utilisateur actif. Autrement dit, au lieu de consulter l'ensemble de la population, estimont la préférence d'un groupe de personnes ayant les mêmes gouts que l'utilisateur.

2.K-Means

La méthode des plus proches voisins K-Means consiste dans un premier temps à choisir aléatoirement k centres dans l'espace de représentation utilisateurs/ressources. Ensuite, chaque utilisateur est mis dans le cluster du centre le plus proche. Quand les groupes de personnes sont formes, la position des centres pour chaque cluster est calculé et réitérons l'opération depuis le début jusqu'à obtenir un état stable ou les centres ne bougent plus. L'algorithme est certes simple

à mettre en œuvre mais présente certains inconvénients, lie à la criticité du choix des clusters initiaux, pouvant influencer sur la qualité de la classification.

3. RecTree

RecTree est un algorithme de filtrage collaboratif appelé l'arbre de recommandation (Recommandation Tree). L'algorithme RecTree fractionne les données dans des cliques d'utilisateurs approximativement semblables. L'objectif est de maximiser les similarités entre les membres d'une même clique et à minimiser celles entre les membres de deux cliques différentes.

I.4.3 *Les approches hybrides*

Les approches hybrides sont des approches qui combinent deux ou plusieurs approches de recommandation. Ces combinaisons permettent de bénéficier des avantages des approches utilisées, de pallier leurs inconvénients et de proposer des recommandations plus pertinentes. Par exemple, le démarrage à froid au niveau des items des approches collaboratives, peut bénéficier des avantages des approches basées sur le contenu. En effet, dans une approche basée sur le contenu, seule la description de l'item est utilisée dans la recommandation, il n'y a donc pas besoin que l'item soit noté par un certain nombre d'utilisateurs avant de pouvoir être recommandé, comme c'est le cas dans les approches collaboratives [28].

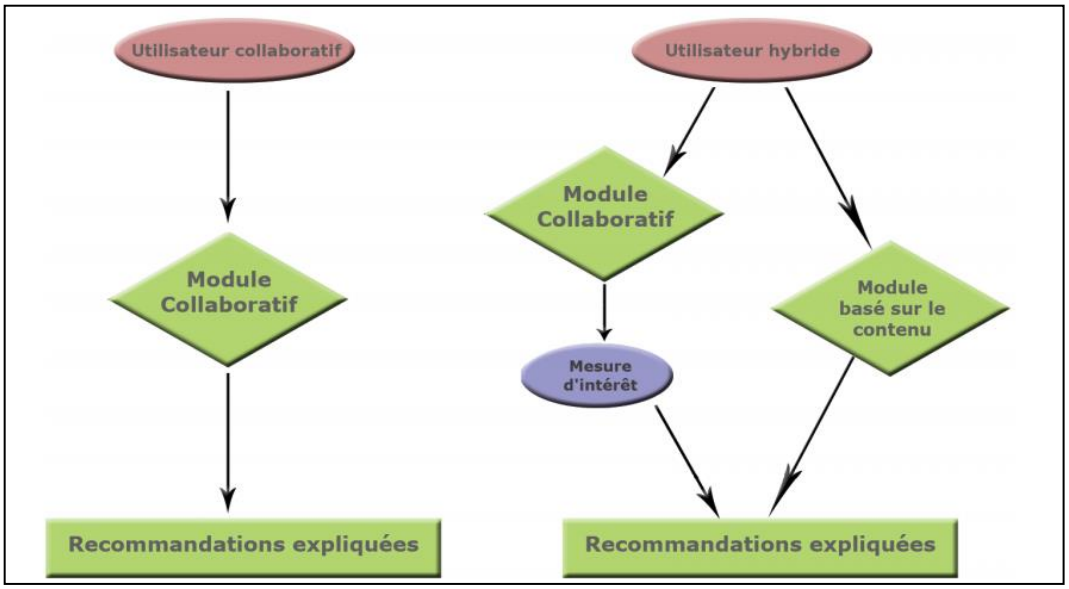


Figure I.6. *Approche hybride*

[29] a recensé les différentes manières d'hybrider des approches de recommandation, décrites ci-dessous :

I.4.3.1 Hybridation pondérée

L'hybridation pondérée [30] consiste à calculer le score d'un item candidat à la recommandation (un item potentiellement pertinent pour l'utilisateur et donc recommandable) grâce à une fonction définie effectuant la somme pondérée du score de l'item dans chaque approche de recommandation présente dans le système. Une fois les scores de tous les items candidats à la recommandation calculés, ceux-ci sont classés par ordre décroissant avant d'être présentés à l'utilisateur (voir figure I.7). Ce type d'hybridation nécessite que les systèmes de recommandations utilisés effectuent une tâche de prédiction de note.

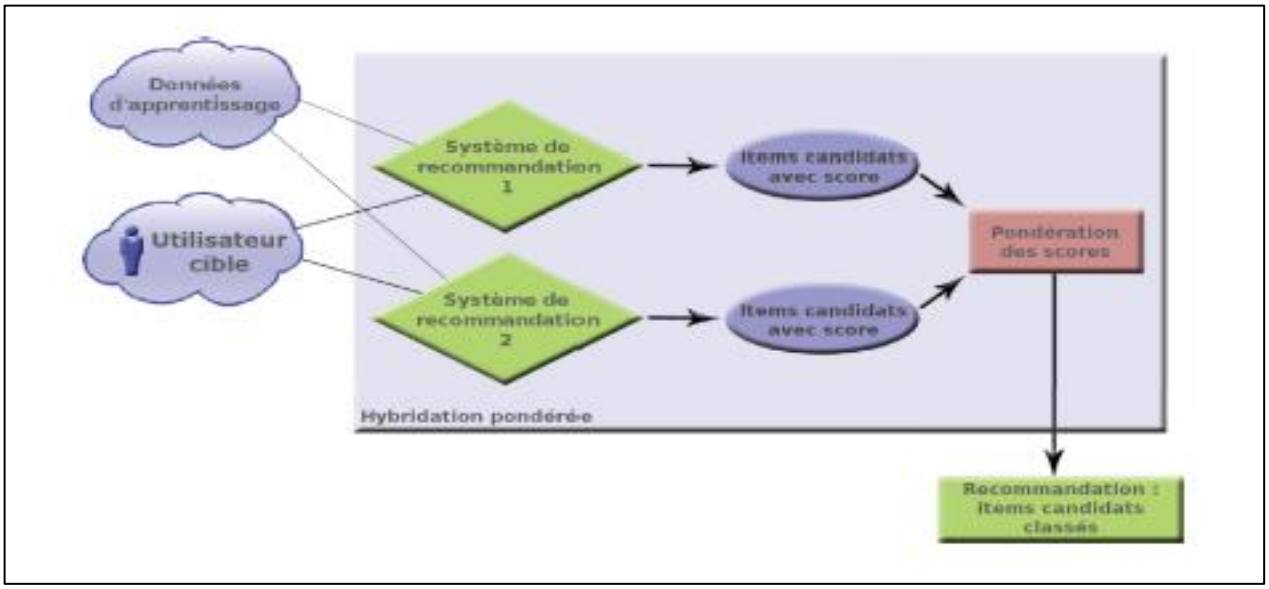


Figure I.7. *Hybridation pondéré*

I.4.3.2 Hybridation à bascule

Dans ce type d'hybridation, le système sélectionne une approche de recommandation plutôt qu'une autre en fonction d'un certain critère. Par exemple, si la confiance du système pour les résultats obtenus est insuffisante, une autre approche sera sélectionnée [31]. Cette hybridation est décrite dans la figure I.8.

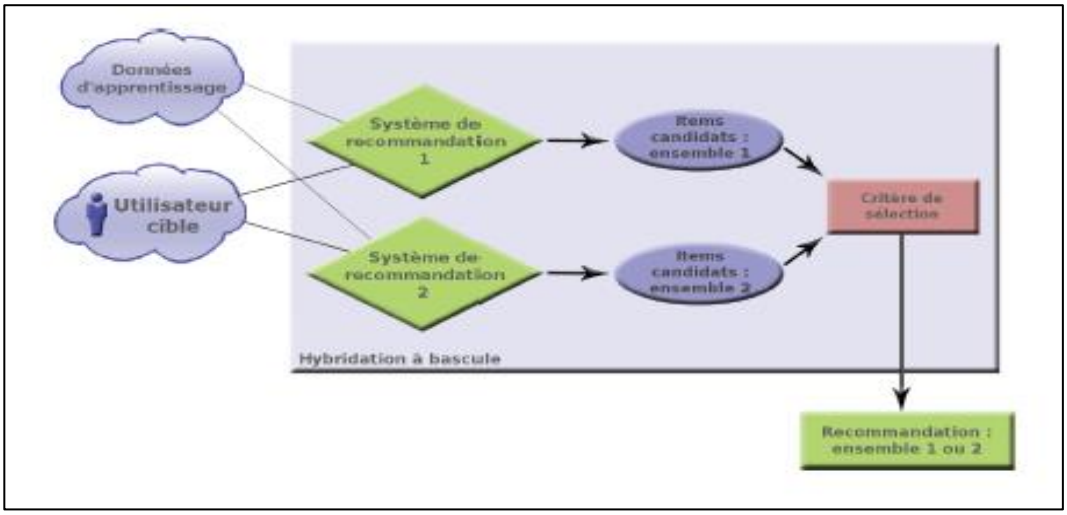


Figure I.8 : *Hybridation à bascule*

I.4.3.3 Hybridation mixée

L'hybridation mixée [32] consiste à présenter à l'utilisateur des recommandations issues de plusieurs approches de recommandation tel que décrit dans la figure I.9.

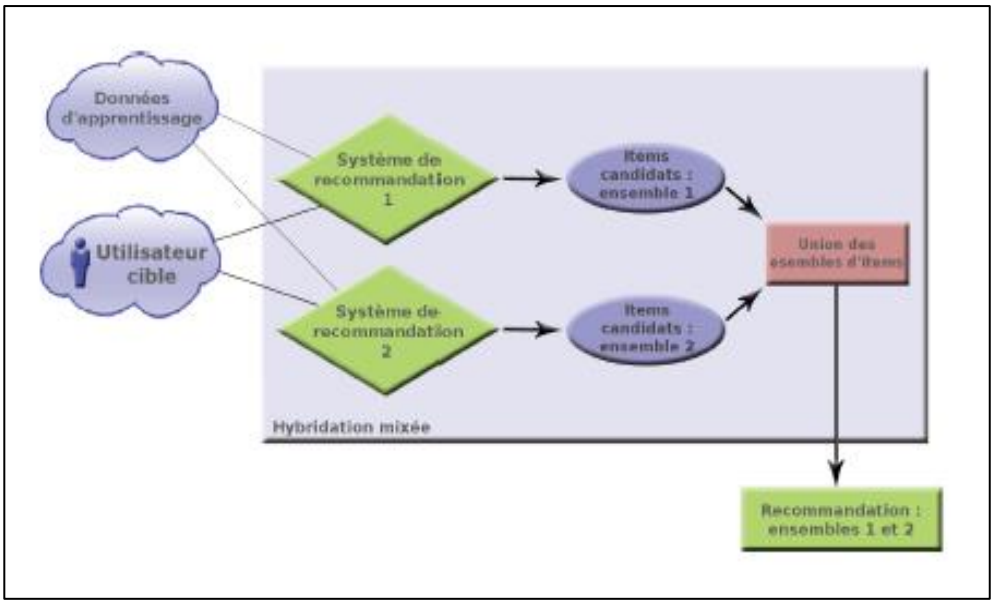


Figure I. 2.*Hybridation mixée*

I.4.3.4 Hybridation par combinaison de caractéristiques

Dans cette hybridation, des caractéristiques d'une approche de recommandation sont injectées dans une autre approche de recommandation [33]. Par exemple, lors de l'utilisation d'une approche collaborative U2U (approche basée sur les utilisateurs), les items peuvent être remplacés par une caractéristique des approches basées sur le contenu, qui est la description des attributs des items. De ce fait, au lieu de traiter l'information « l'utilisateur apprécie l'item i », on traite l'information « l'utilisateur à apprécie l'attribut a ». Le calcul des similarités entre les utilisateurs est ensuite effectué en se basant sur les attributs au lieu de se baser sur les items. Un schéma de cette hybridation est proposé dans la figure I.10.

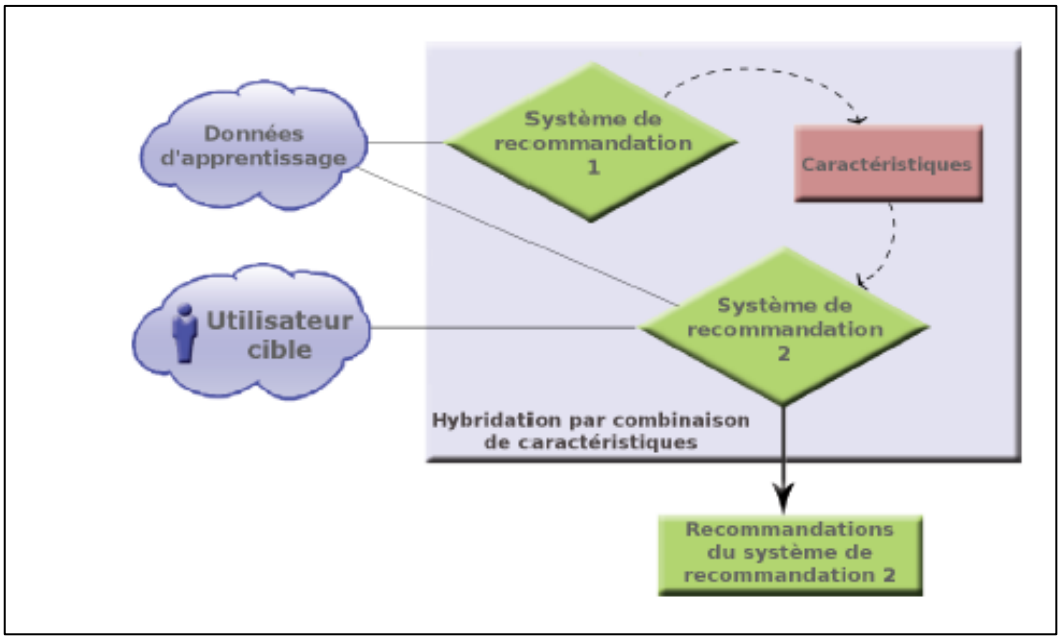


Figure I.10 : *Hybridation par combinaison de caractéristiques*

I.4.3.5 Hybridation en cascade

L'hybridation en cascade implique l'application successive des approches de recommandation du système [1]. La première approche a pour but de générer un ensemble d'items candidats à la recommandation. Chaque approche de recommandation du système est ensuite appliquée à l'ensemble d'items candidats à la recommandation. Les items de cet ensemble sont donc petit à petit filtrés et l'ensemble final est recommandé à l'utilisateur (voir figure I.11).

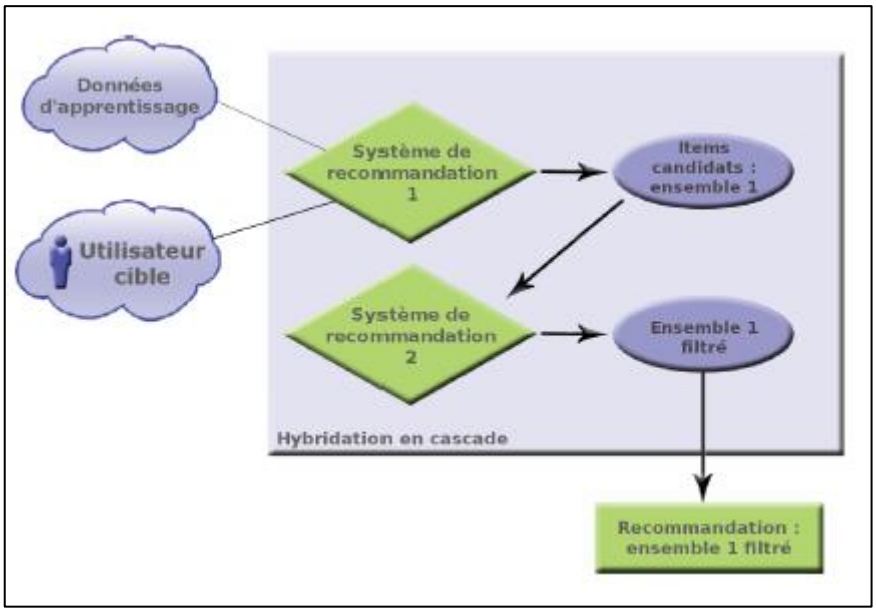


Figure I.11.*Hybridation en cascade*

I.4.3.6 Hybridation par ajout de caractéristiques

Cette hybridation nécessite également une application successive des approches de recommandation disponibles. Cependant, chaque approche prend en paramètre les résultats de l'approche précédente et l'utilise comme information supplémentaire durant son exécution [34].Par exemple, une première approche de recommandation est utilisée afin de générer un ensemble d'items. Ces items sont intégrés au profil de l'utilisateur afin de l'enrichir. La deuxième approche de recommandation est ensuite appliquée sur ce profil enrichi afin de générer des items à proposer à l'utilisateur. La figure I.12 schématise le fonctionnement de cette hybridation.

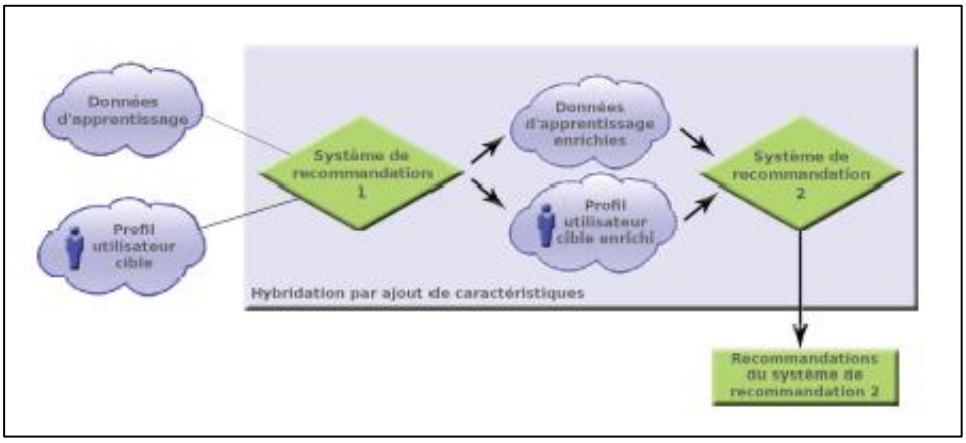


Figure I.12 : *Hybridation par ajout de caractéristiques*

I.4.3.7 Hybridation méta-niveau

Dans ce type d'hybridation, la première approche de recommandation génère un modèle qui est ensuite utilisé par la deuxième approche de recommandation [35]. La deuxième approche de recommandation remplace complètement son entrée (la source des données) par le modèle généré par la première approche. Cette hybridation n'est pas applicable à toutes les approches de recommandation car elle nécessite l'utilisation d'approches basées sur des modèles. La figure I.13 décrit le principe de cette hybridation.

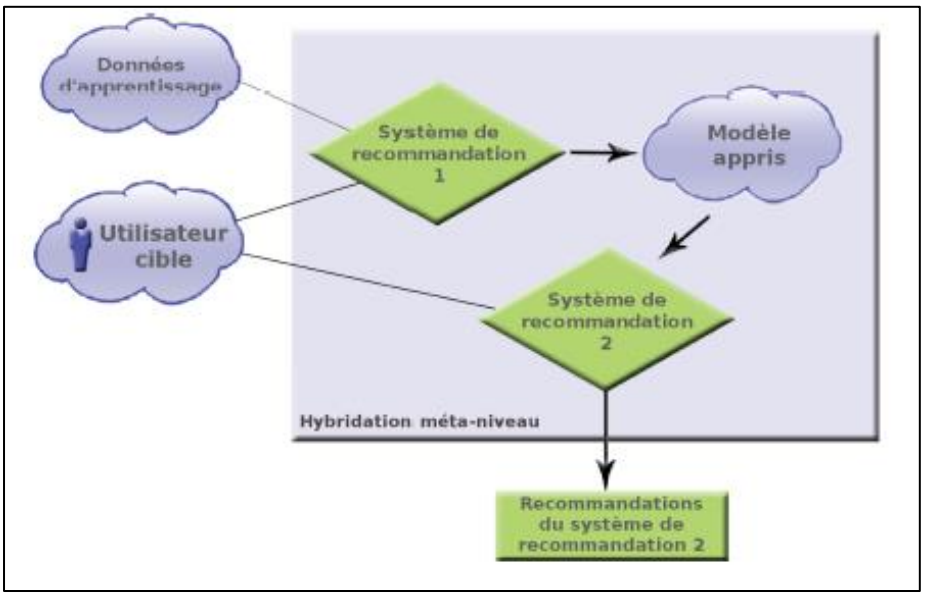


Figure I. 3.*Hybridation méta-niveau*

I.4.4 Les approches démographiques

Dans une approche démographique [35], le système de recommandation demande à l'utilisateur de renseigner un certain nombre de paramètres (âge, sexe, ville, emploi, intérêts, etc.). Ces paramètres permettent de définir des classes démographiques dans lesquelles les utilisateurs sont catégorises. Par la suite, les recommandations de chaque utilisateur proviennent de la classe qui lui a été e attribuée [28].

I.4.5 Les approches sociales

Les réseaux sociaux fournissent à la tâche de recommandation un nouveau paramètre qui est le cercle social de l'utilisateur. De ce fait, un système de recommandation social peut proposer à l'utilisateur des items qui sont populaires dans son cercle social. L'approche peut donc être considérée comme un filtrage collaboratif U2U où le calcul du voisinage del'utilisateur est remplacé par l'utilisation du cercle social de l'utilisateur. Le but étant de calculer le degré de confiance de l'utilisateur envers ses pairs, afin d'identifier ceux qui sont plus proches du profil l'utilisateur [36]. Les items les plus populaires au sein du groupe identifie sont ensuite recommandés à l'utilisateur cible [28].

I.5 Avantage et inconvénient de système de recommandation

Le tableau 1 résume les forces et faiblesses des méthodes traditionnelles utilisent par les systèmes de recommandation, en l'occurrence le Filtrage Collaboratif (FC), le Filtrage Démographique(FD), et le Filtrage à Base de Contenu (FBC).

- **Adaptabilité** : Au fur et à mesure que la base de données des évaluations augmente, la recommandation devient plus précise.
- **Nouvel utilisateur** : un nouvel utilisateur qui n'a pas encore accumule suffisamment d'évaluations ne peut pas avoir de recommandations pertinentes.
- **Nouvel item** : un item doit avoir suffisamment d'évaluations pour qu'il soit pris en considération dans le processus de recommandation.
- **Démarrage à froid** : le démarrage à froid est un problème pour les nouveaux utilisateurs qui commencent à jouer avec le système, parce que le système ne dispose pas d'assez d'informations à leur sujet. Si le profil d'utilisateur est vide, il doit consacrer une somme d'efforts à l'aide du système avant d'obtenir une récompense (les recommandations utiles). D'autre part, quand un nouvel item est ajouté à la collection, le système doit avoir suffisamment d'informations pour être en mesure de recommander cet item aux utilisateurs.

Technique	**Avantage**	**Inconvénient**
Filtrage démographique	N'exige aucun historique d'estimations.	• Problème de confidentialité. • Utilisateur avec un gout unique. • Nouvel Item.
Filtrage a base du contenu	•Pas besoin d'une large communauté d'utilisateurs pour pouvoir effectuer des recommandations. •Une liste de recommandations peut être générée même s'il n'y a qu'un seul utilisateur. •La qualité croit avec le temps. •Pas besoin d'information sur les autres utilisateurs. •Prendre en considération les gouts uniques des utilisateurs.	•L'analyse du contenu est nécessaire pour faire une recommandation. •Problème de recommandation des images et de vidéos en absence de Métadonnées. •Nécessite du profil d'utilisateur.
Filtrage collaboratif	•Ne demande aucune connaissance sur le contenu de l'item ni sa sémantique. •La qualité de la recommandation peut être évaluée. •Plus les nombre d'utilisateurs est grand plus la recommandation est meilleure.	•Démarrage a froid. •Nouvel Item. •Nouvel utilisateur. •Problème de confidentialité. •La complexité : dans les systèmes avec un grand nombre d'items et d'utilisateurs, le calcul croit linéairement.

Tableau I.1. *Les avantages et les inconvénients des techniques de recommandation*

I.6 L'évaluation de la recommandation

Evaluer un système de recommandation permet de mesurer ses performances vis-à-vis de ses objectifs. De ce fait, les dimensions à évaluer différent selon les objectifs fixés (prédiction des notes des utilisateurs pour les items, augmentation des ventes, etc.).

La question de l'évaluation des systèmes de recommandation a été longuement étudiée dans le travail de [37]. Ce dernier répertorie les différents procédés permettant d'évaluer un système, ainsi que les dimensions pouvant être considérées durant l'évaluation.

I.6.1 Techniques d'évaluation

Trois techniques d'évaluation sont proposées par [37]: le test en ligne, les études utilisateurs et les simulations. Chacune présente un certain nombre d'avantages et d'inconvénients :

– Test en ligne

Le test en ligne permet d'évaluer l'application sur de vrais utilisateurs. Dans ce cas de figure, les utilisateurs sont orientés vers plusieurs alternatives du système de recommandation afin de pouvoir comparer leurs performances [38]. L'intérêt présenté par ce genre d'évaluation est l'obtention de résultats réels, dans la mesure où les utilisateurs sont les utilisateurs finaux qui ne sont pas mis au courant de l'exécution du test. De ce fait, cette évaluation mesure la vraie performance du système et permet de choisir l'alternative ayant mieux rempli le cahier des charges défini de l'application. Cependant, ce type d'évaluation présente également un certain nombre d'inconvénients. En effet, comparer plusieurs alternatives du système sur une population d'utilisateurs prend du temps. De plus, un utilisateur ayant eu une alternative non satisfaisante du système peut être dissuadée de l'utiliser par la suite. De ce fait, ce genre d'évaluation doit être utilisé après avoir effectué assez de tests hors-ligne permettant au préalable de filtrer les alternatives infructueuses [28].

– Etudes utilisateurs

Les études utilisateurs consistent à recruter un certain nombre de sujets dont la fonction sera d'effectuer quelques tâches les faisant interagir avec le système. Chaque sujet devra tester et évaluer une ou plusieurs alternatives du système. Ce genre d'évaluations offre la possibilité d'obtenir des données d'évaluations supplémentaires en récoltant les impressions des sujets et en observant leur comportement. Malheureusement, les études utilisateurs peuvent être

coûteuses dans la mesure où les sujets sont le plus souvent rémunérés. De plus, recruter des sujets reflétant la population réelle d'utilisateurs s'avère laborieux et ne permet pas d'évaluer le comportement réel des utilisateurs sur le système dans la mesure où l'effort et l'attention apportés durant l'expérimentation peuvent différer du contexte réel d'utilisation du système [28].

– Simulations (évaluations hors-ligne)

Simuler le comportement des utilisateurs présente une alternative rapide et peu coûteuse aux deux types d'évaluations cités précédemment. Une simulation requiert la disposition d'un jeu de données pouvant servir à reproduire le comportement des utilisateurs. Ce jeu de données contiendra par exemple les notes attribuées par des utilisateurs pour certains items. L'hypothèse de départ implique que le comportement de l'utilisateur lors de la collecte initiale des données sera similaire à son comportement lors de l'exécution de l'évaluation du système de recommandation.

Afin de procéder à une simulation, le jeu de données dont on dispose est échantillonné en deux ensembles : un ensemble d'apprentissage et un ensemble de test. L'échantillon d'apprentissage sert à établir le modèle que l'échantillon de test sert à valider. La validation consiste à appliquer le système de recommandation sur l'échantillon d'apprentissage et à comparer les résultats obtenus avec l'échantillon de test.

L'échantillonnage du jeu de données peut se faire de façon chronologique de sorte que le premier échantillon, dans l'ordre chronologique, serve d'ensemble d'apprentissage et le second soit utilisé comme ensemble de test. Ce partitionnement repose sur le postulat que les systèmes de recommandation influencent le comportement de l'utilisateur et que, par conséquent, certains items sont notés ou consommés grâce à l'influence du système de recommandation. De ce fait, l'évolution chronologique des centres d'intérêt de l'utilisateur est peut-être étroitement liée au système de recommandation et la simulation doit donc respecter ladite chronologie.

Cependant, la validation croisée est plus couramment utilisée. Par exemple, la validation croisée à N plis (N-fold cross-validation) consiste à échantillonner le jeu de données en N parties. Chaque partie sert une fois d'échantillon de test tandis que l'apprentissage s'effectue sur les parties restantes. Il s'agit donc d'une opération qui est répétée N fois, une fois par échantillon. La performance est ensuite calculée en faisant la moyenne des performances sur chaque échantillon de test [28].

I.6.2 Précision de la prédiction de notes

La précision de la prédiction de notes est une des dimensions les plus souvent mesurées dans les systèmes de recommandation. Elle consiste à déterminer si les notes prédites pour les utilisateurs pour des items donnés sont correctes (voir définition 1 dans le chapitre 1). L'hypothèse qui motive la mesure de cette dimension est qu'un utilisateur préférera un système de recommandation aux prédictions les plus précises. Cette dimension est évaluable lors des simulations hors ligne.

Les mesures les plus populaires pour l'estimation de la précision des prédictions de notes sont l'erreur quadratique absolue (MAE) et l'erreur quadratique moyenne (RMSE). La MAE calcule l'écart entre les notes prédites et les vraies notes. Quant à la RMSE, elle repose sur le même principe que la MAE, tout en mettant l'accent sur les écarts importants.

Soit T un ensemble test d'items, p(u;i) une prédiction de note de l'utilisateur u pour l'item i et n(u;i) la note réelle attribuée par l'utilisateur u pour l'item i [28].

– Précision de la recommandation

La précision de la recommandation est également une dimension souvent évaluée. Contrairement à la prédiction de notes, la précision de la recommandation se rapporte à la tâche de recommandation. Il s'agit donc ici de déterminer la performance du système dans la tâche de recommandation consistant à trouver et recommander à l'utilisateur tous les items qui l'intéresseraient. La table I.1 représente les résultats possibles après l'application du processus de recommandation.

La performance du système est évaluée à l'aide de deux mesures : la précision et le rappel.

La précision est une mesure d'exactitude. Elle détermine la proportion des items pertinents recommandés parmi tous les items recommandés : autrement dit, la proportion des recommandations qui se sont avérées pertinentes.

		Prédiction	
		Recommandé	Non recommandé
Réalité	Apprécie	Vrai Positif(VP)	Vrai Négatif(VN)
	Non apprécie	Faux Positif(FP)	Faux Négatif (FN)

Tableau I.2. *Classification des résultats possibles pour la recommandation d'items*

I.7 Les problèmes des systèmes de recommandations

Comme tout domaine d'application, les systèmes de recommandation souffrent de plusieurs problèmes relatifs à la nature de ces systèmes et aux données nécessaires pour leur fonctionnement.

Les problèmes inhérents aux systèmes de recommandation, quelques travaux réalisés en vue de résoudre ou d'alléger les effets de ces problèmes :

I.7.1 Le problème du démarrage à froid (cold startproblem)

Le démarrage à froid est le problème qui se produit dans les premières périodes de l'utilisation d'un système de recommandation, en raison d'un manque dans les données disponibles nécessaires pour la génération d'un processus de recommandation personnalisée de haute qualité. Selon le type de données qui manque, on peut distinguer trois types de problème du démarrage à froid [1] :

- **Le démarrage à froid pour un nouveau système** [39] : Comme son nom l'indique ce type concerne les nouveaux systèmes qui souffrent d'une absence en informations (utilisateurs, items ainsi que les relations entre eux), ce qui conduit certainement à de très mauvaises performances.

  Ce type de problème a été nommé dans [1] par le problème du démarrage (*The Start Up Problem*), et qui a été défini par la nécessité d'accumuler suffisamment de données pour être utilisées comme stéréotypes au démarrage. La solution traditionnelle de ce problème est de spécifier une phase, après la phase du développement du système, à la collecte de données qui seront utilisées comme données d'apprentissage,

et d'utiliser les outils de classification automatique afin de trouver les corrélations entre les entités du système.

- Le démarrage à froid pour un nouvel item : Dans les systèmes à base du FC, les items sont représentés par les évaluations qu'ils ont eues à partir des utilisateurs du système. Donc, un nouvel item qui n'est pas connu pour les utilisateurs, et qui ne reçoit pas suffisamment d'évaluations, il ne sera pas recommandé malgré qu'il convient très bien aux préférences de certains utilisateurs. Une solution efficace pour ce problème est de combiner le FBC avec le FC, afin de calculer les corrélations entre items [40]en se basant sur leur contenu. De plus, les agents intelligents peuvent aussi être utilisés [41] pour évaluer automatiquement les items, malgré que cela peut fausser les recommandations parce qu'elles n'utilisent pas des évaluations réelles.
- Le démarrage à froid pour un nouvel utilisateur: Le profil d'un utilisateur est constitué des évaluations qu'il a faites dans le système. Donc, si l'utilisateur est nouveau pour le système, cela veut dire qu'il n'a pas encore suffisamment d'évaluations pour construire son profil dans le système. Par conséquent, le système ne pourra pas fournir des recommandations pertinentes à cet utilisateur [42], parce qu'il a peu de connaissances sur ses préférences, et par la suite il n'arrivera pas à calculer sa similarité avec les autres utilisateurs afin de lui affecté à un groupe dans le cas du FC. Dans le cas du FBC, le système ne peut pas extraire les caractéristiques du contenu qui intéresse le nouvel utilisateur, ce qui conduit à des recommandations de mauvaise qualité.

I.7.2 Le problème de la Stabilité Vs. Dynamicité (Stability Vs. Plasticityproblem)

Le problème de la stabilité des systèmes de recommandation revient à l'incapacité du système à suivre l'évolution du profil des utilisateurs, du fait que plus le système apprend les préférences des utilisateurs, plus le risque de stabiliser leurs profils augmente, ce qui rendra le changement de ces profils très difficile [1].

Une solution pour ce problème est de considérer que les nouvelles évaluations fournies par les utilisateurs sont plus fiables que les anciennes [43][44], en donnant un poids suffisant pour ces évaluations afin de virer le processus de recommandation vers les nouvelles préférences [45].Ainsi, les évaluations les plus récentes deviennent plus "importantes" pour les algorithmes d'apprentissage, en supposant qu'elles représentent mieux les intérêts des utilisateurs que les plus anciennes.

Une autre solution pour ce problème vise à apprendre les préférences des utilisateurs à propos de leurs intérêts courants, à partir d'une fenêtre de temps qui comprend les dernières observations pertinentes seulement [46].Toutefois, si les intérêts de l'utilisateur changent souvent, un profil utilisateur précis ne peut pas être appris à partir d'une petite série d'observations récentes uniquement. Par conséquent, le système peut rechercher les dernières épisodes, où l'utilisateur a démontré un même ensemble d'intérêts, et essaye d'apprendre une description plus précise des intérêts de l'utilisateur en se souvenant des observations pertinentes et oubliant les observations non pertinentes [46].

I.7.3 Les Systèmes Statiques

Les systèmes statiques sont des systèmes qui n'apprennent pas les préférences des utilisateurs au fur et à mesure de leurs connexions avec le système, mais ils considèrent juste les informations qu'ils fournissent instantanément dans le processus de recommandation.

Ce type de systèmes peut être considéré comme idéal pour les systèmes à usage occasionnel tels que les systèmes de recommandation des restaurants et des musées. Cependant, si les clients utilisent le système d'une manière fréquente ça va être ennuyeux pour eux de re-saisir leurs besoins à chaque utilisation. Ce problème constitue l'inconvénient majeur des approches qui ne sont pas basées sur la mémoire, dont l'approche basée sur la connaissance est la plus concernée par ce problème malgré qu'il ne soit pas global pour cette approche car on peut imaginer des systèmes qui adaptent leurs connaissances en fonction du retour de pertinence des utilisateurs, et génèrent des recommandations qui sont de plus en plus automatisées.

Beaucoup de ces systèmes utilisent les méthodes d'apprentissage automatique pour l'acquisition des profils intéressants [47]. Toutefois, les intérêts des utilisateurs peuvent changer durant le temps, certains de ces systèmes sont fournis avec des mécanismes qui sont en mesure de suivre les dérivatives des intérêts des utilisateurs [48][49] [50].

I.7.4 Le problème de la parcimonie de données (Sparcityproblem)

Ce problème est spécifique aux systèmes collaboratifs et il survient quand l'espace des évaluations est dispersé : peu d'utilisateurs ont évalué les mêmes items [1]. Dans les applications réelles, la matrice des évaluations comprend des milliers d'items et des millions d'utilisateurs. Toutefois, un utilisateur donné évalue seulement un petit nombre d'items. En conséquence, la matrice des évaluations présente de nombreuses cellules vides, ce qui conduit

à avoir des difficultés pour calculer la similarité entre les utilisateurs, surtout pour la Corrélation de Pearson, et réduit énormément la précision des prédictions.

Plusieurs travaux ont été présentés dans la littérature afin de minimiser l'effet de la clairsemé de données, tels que l'algorithme de recommandation utilisant des pas aléatoires sur l'espace des items pour détecter les probabilités de transition entre les items, afin d'améliorer la matrice de similarité utilisée dans la phase de prédiction [51]. De plus, la clairsemé peut être atténuée grandement par les méthodes de réduction de la dimension de l'espace de données d'entrée, qui ont prouvé leur efficacité à travers la littérature, telles que la méthode de factorisation matricielle [52][53],et la méthode de décomposition en valeurs singulières [54].

I.7.5 Le problème de l'évolutivité de données

Le nombre des utilisateurs et des items dans les systèmes de recommandation peuvent être assez importants (souvent des millions d'utilisateurs et des milliers d'items). Cela risque de ralentir la procédure de recommandation de manière significative depuis les algorithmes basés sur la mémoire pour sélectionner les plus proches voisins, ce qui exigera trop de calculs [55].

I.7.6 Le problème de lasur-spécialisation

La sur-spécialisation appelée aussi (*The Portfolio effect*) dans [1], caractérise l'approche collaborative et surtout l'approche basée sur le contenu, du fait que des doublons de recommandations peuvent se produire lorsque des items distincts désignent un même contenu.

Pour l'approche collaborative, cet effet ne se surmontera que si les profils des utilisateurs au niveau d'un même groupe convergent vers des préférences proches et plus en plus spécialisées. Dans ce cas, la recommandation pour un utilisateur se restreint aux items similaires à ceux déjà appréciés par lui.

Problèmes / Approche	Nouvel utilisateur	Nouvel Item	Nouveau système	Stabilité Vs. Dynamicité	Système statique	Dispersion	Evolutivité	Sur-spécialisation
FiltrageBaséContenu	×	-	-	×	-	×	-	×
FiltrageCollaboratif	×	×	×	×	-	×	×	-
FiltrageBaséConnaissance	-	-	-	-	×	-	-	-

Tableau I.3.*Bilan des approches et des problèmes inhérents aux SRs*

I.8 Conclusion

Nous avons listé dans ce chapitre les principes des approches de recommandation simples existantes : basées sur le contenu, collaboratives, démographiques, sociales, basées sur la connaissance et sur l'utilité, ainsi que les différentes manières, existantes à ce jour, de combiner ces approches fin d'obtenir une approche hybride qui remédie aux limitations des approches simples.

La tendance actuelle des systèmes de recommandation est plutôt axée sur des méthodes nouvelles, multicritères, multidimensionnelles ou encore se fondant sur des notions psychologiques comme les émotions, les opinions. Notons cependant qu'un système de recommandation doit avant tout s'adapter aux données, celles-là mêmes que l'on proposera à un utilisateur. Ainsi, le choix d'une méthode de recommandation doit en premier lieu être dirige par ce critère.

Chapitre II

Les ontologies et les relations sémantiques

II.1 Introduction

De nos jours, les systèmes de recommandation sont intégrés dans les sites de commerce électronique, les plateformes de chansons et de vidéos, la presse, les publications scientifiques, etc. Ces domaines sont larges et variés, si bien que les approches de recommandation utilisées sont nombreuses et différentes.

Les techniques existantes sont dédiées à des domaines spécifiques et précis. De ce fait, elles ne peuvent pas être appliquées à des domaines différents sans y apporter des modifications majeures. Nous avons donc voulu mettre en avant l'indépendance du domaine dans le système de recommandation défini. L'indépendance du domaine permet d'avoir une base opérationnelle pour la recommandation et sa mise en place a été assurée par la description de la connaissance du domaine dans une ontologie de domaine donnée en entrée au système. Cela a permis de proposer une approche générique reposant uniquement sur les concepts et propriétés de l'ontologie qui définissent la sémantique des données.

L'identification de la similarité dans les ontologies est un concept fondamental qui est adopté par plusieurs techniques telles que le regroupement, la fouille de données (datamining), le Web sémantique et en particulier, le domaine de la recherche de l'information. Ce dernier repose largement sur des mesures pour l'identification de la similarité entre les documents [56] [57].

Dans cette partie nous allons parler de l'ontologie et les mesures de similarité en générale, et conclure par la recommandation basée sur la sémantique.

II.2 La notion d'ontologie

Le terme «ontologie» est cependant usité en philosophie depuis le XIXème siècle. Dans ce domaine, l'ontologie est une étude de l'être en tant qu'être, c'est-à-dire, une étude des propriétés générales de ce qui existe. En informatique, la littérature fournit un tas de définitions du mot ontologie. Ces définitions, dans leur diversité, offrent des points de vues à la fois différents et complémentaires [58].

II.3 Définition d'Ontologie

En 1993, Gruber a proposé une définition à cette notion qui est la plus célèbre couramment citée dans la littérature [59].

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

Il a introduit la notion conceptualisation qui réfère a un modèle abstrait d'un certain domaine du monde réel en identifiant les concepts pertinents décrivant ce domaine explicite signifie que les concepts utilisé ou les contraintes sont réellement définis d'une manière claire et précise.

II.3.1 *Les composantes de l'ontologie*

Les ontologies produisent un vocabulaire commun d'un domaine et définissent, de façon plus ou moins formelle, la signification des termes et des relations entre eux. Les connaissances intégrées dans les ontologies sont formalisées en mettant en jeu cinq types de composants : concepts, relations, fonctions, axiomes, instances [58].

II.3.1.1 *Concepts*

Ils sont appelés aussi termes ou classes de l'ontologie. Un concept est un constituant de la pensée (un principe, une idée, une notion abstraite) sémantiquement évaluable et communicable. Un concept peut être divisé en trois parties : un terme (ou plusieurs), une notion et un ensemble d'objets.

- Le terme (ou bien label) d'un concept est l'expression linguistique utilisée couramment pour y faire référence.
- La notion désigne ce qui est appelé, au sens de la représentation des connaissances, l'intension du concept. Elle contient sa sémantique qui est définie à l'aide de propriétés (relations et attributs), de règles et de contraintes.
- L'ensemble d'objets définis par le concept forme ce qui est appelé l'extension du concept. Il s'agit des objets auxquels le concept fait référence, autrement dit, de ses instances.

Selon [Gomez-Pérez ,99], ces concepts peuvent être classifiés selon plusieurs dimensions :

- Niveau d'abstraction (concret ou abstrait).
- Atomicité (élémentaire ou composée).
- Niveau de réalité (réel ou fictif).

En résumé, un concept peut être tout ce qui peut être évoqué et, partant, peut consister en la description d'une tache, d'une fonction, d'une action, d'une stratégie ou d'un processus de raisonnement, etc.

II.3.1.2 *Relations*

Représentent un type d'interaction, ou bien des associations existant entre les concepts d'un domaine. Formellement, elles sont définies comme étant tout sous ensemble d'un produit de n concepts, c'est-à-dire : R ⊂C1∗C2∗…Cn.

Sous-classe-de (Spécialisation, généralisation), partie de (agrégation ou composition), associée-à, instance-de sont des exemples de relations binaires. Voici quelques relations les plus courantes dans la littérature :

1) L'équivalence : une relation R est une relation d'équivalence si et seulement si : R est symétrique, réflexive et transitive. On écrit :

(R est une relation d'équivalence) ⇔ ((R symétrique) ^ (R réflexive) ^ (R transitive))

2) La cardinalité : c'est le nombre possible de relations de ce type entre les mêmes concepts (ou instances de concept). Les relations portant une cardinalité représentent souvent des attributs. Exemple : une pièce a au moins une porte, un humain a entre zéro et deux jambes.
3) L'incompatibilité : Deux relations sont incompatibles si elles ne peuvent lier les mêmes instances de concepts. Exemples : les relations «être rouge» et «être vert» sont incompatibles.
4) L'inverse : Deux relations binaires sont inverses l'une de l'autre si, quand l'une lie deux instances I1 et I2, l'autre lie I2 et I1. Exemple : les relations «a pour père »et «a pour enfant» sont inverses l'une de l'autre.
5) L'exclusivité : Deux relations sont exclusives si, quand l'une lie des instances de concepts, l'autre ne lie pas ces instances, et vice-versa. L'exclusivité entraine l'incompatibilité. Exemple : l'appartenance et la non appartenance sont exclusives. [58]

Et bien d'autres relations…

II.3.1.3 *Fonctions*

Ce sont des cas particuliers de relations dans lesquelles le Nième élément de la relation est défini de manière unique à partir des n-1 premiers. Formellement, les fonctions sont définies ainsi :

$$F : C1 \times C2 \times \ldots \times C_{n-1} \times C_n \qquad \text{II.1}$$

Comme exemple de fonctions binaires, les fonctions mères de et le carré [58].

II.3.1.4 *Axiomes*

Constituent des assertions, acceptées comme vraies, à propos des abstractions du domaine, traduites par l'ontologie. Ils ont pour objectif de représenter des concepts et des relations dans un langage logique permettant de représenter leur sémantique. Ils représentent les intentions des concepts et des relations du domaine et, de manière générale, les connaissances n'ayant pas un caractère strictement terminologique. L'utilisation des axiomes sert à définir le sens des entités, mettre des restrictions sur la valeur des attributs, examiner la conformité des informations spécifiées ou en déduire de nouvelles [58].

II.3.1.5*Instances (ou individu)*

Elles constituent la définition extensionnelle de l'ontologie. Ils représentent les éléments singuliers véhiculant les connaissances à propos du domaine [58].

II.4 Différents types d'ontologies

Les ontologies possèdent plusieurs types comme présenté dans la figure **II.1.**

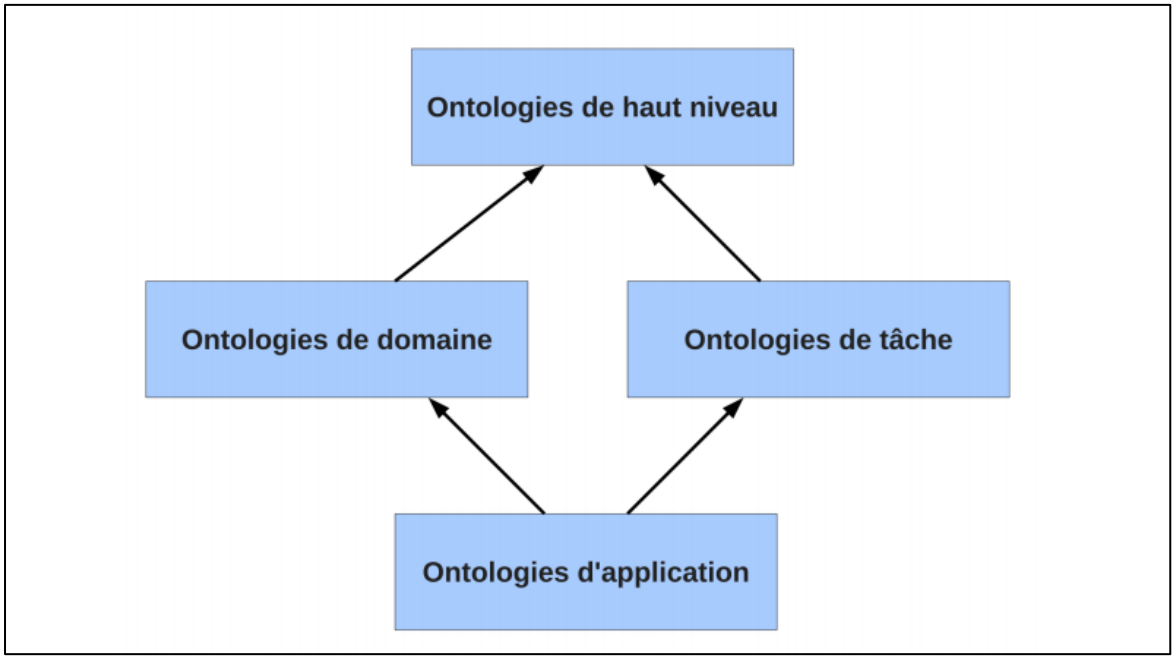


Figure II.1. *Types d'ontologie*

II.4.1 ontologies de haut niveau

Concepts très généraux indépendants du problème exemple : DOLCE, Wordnet [60].

II.4.2 ontologies de domaine

Concepts spécifiques à un domaine exemple : UMLS (Unified Medical Language System) Dublin Core[1].

II.4.3 ontologie de tâche

Concepts spécifiques à une application exemple : ONTOLINGUA[2].

II.4.4 ontologie d'application

Concepts très spécifiques à un domaine et une tâche particulière[3].

Exemple d'une hiérarchie d'une ontologie :

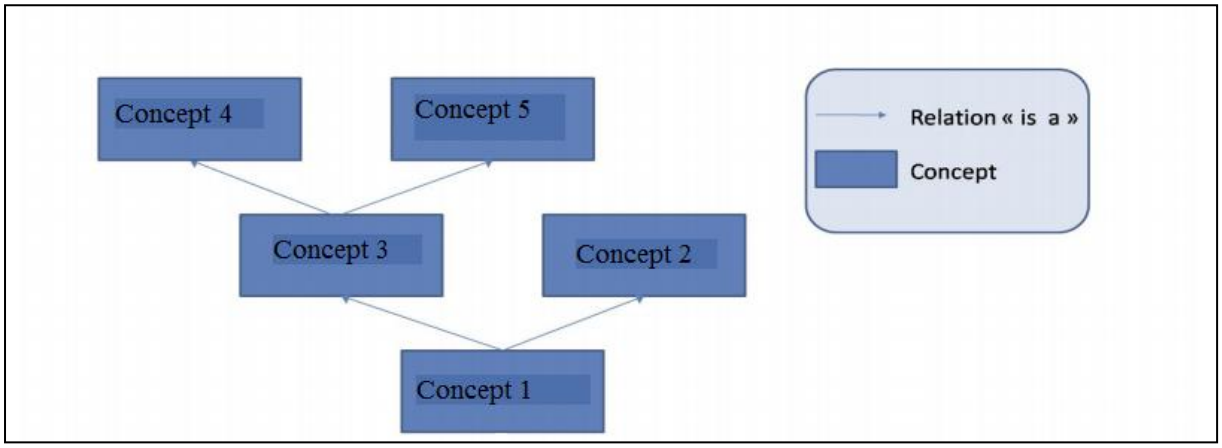


Figure II.2. *Exemple d'une hiérarchie des concepts d'une ontologie*

II.5 WordNet

Wordnet est une ontologie qui organise plus de 100,000 concepts généraux, qui sont sémantiquement structurés. Elle regroupe les concepts en ensembles de synonymes appelés synsets. Wordnet fournit des définitions générales et courtes, et enregistre les différentes relations sémantiques entre ces ensembles de synonymes. Elle est particulièrement bien

[1]. http://dublincore.org
[2]. http://ksl.stanford.edu/software/ontolingua/
[3]. http://odile.papini.perso.luminy.univ-amu.fr/sources/WEBSEM/cours-intro-websem-2.pdf

adaptée pour de mesures de similarité, car elle organise des noms et des verbes en hiérarchie de relation « est-un ».

II.6 La mesure de similarité sémantique

Dans la communauté du web sémantique, les mesures de similarité sémantique sont le plus souvent utilisées dans l'alignement d'ontologies. Elles se basent sur la structure matching [61] [62] [63], le string matching [64] et le lexical matching qui utilise des mesures telles que le coefficient de Jaccard [65], Dice [66] ou overlap. Ces mesures de similarité ont été définies pour l'alignement d'ontologies, leur but étant de retrouver les concepts similaires dans deux ontologies. De ce fait, elles ne conviennent pas au domaine de la recommandation car celui-ci a pour but de retrouver les instances similaires d'une ontologie.

La mesure de similarité va être appliquée aux instances d'une ontologie afin de calculer les instances similaires. Avec une mesure de similarité classique et non sémantique, deux instances sont considérées comme étant similaires si elles sont reliées à un certain nombre commun d'instances. Par exemple, deux films sont similaires s'ils ont des acteurs en commun. La figure 1 représente deux instances a et b qui ne sont en relation avec aucune instance commune. De ce fait, avec une mesure de similarité classique, a et b ne sont pas similaires.

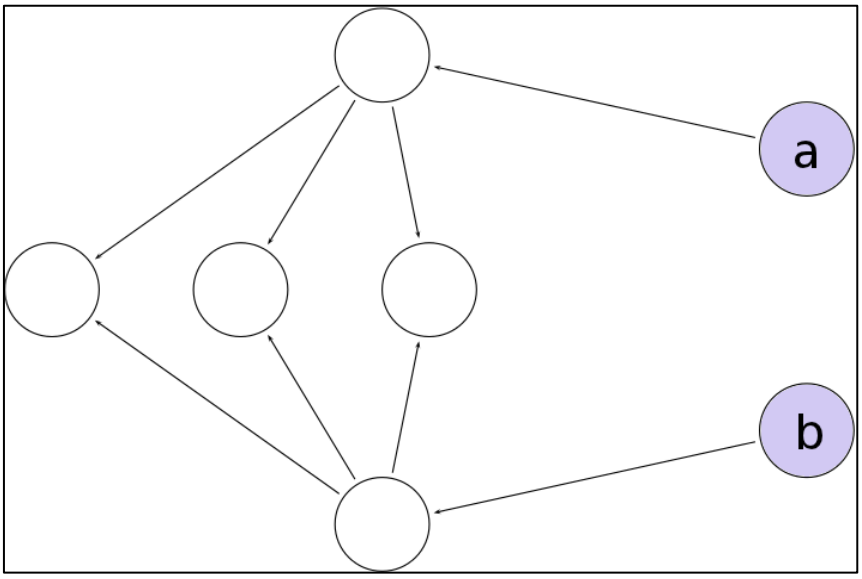


Figure II.3.*Deux instances a et b non similaires avec une mesure de similarité classique*

La définition des mesures classiques de façon que deux instances soient similaires si elles sont reliées à des instances communes, mais également si elles sont reliées à des instances qui sont elles-mêmes similaires. Ceci renforce la similarité calculée et découvre de

nouvelles instances similaires non considérées par une mesure classique. Ainsi, deux films sont similaires, si leurs acteurs sont similaires : s'ils ont joué dans les mêmes films par exemple. Les instances concernées par le calcul sont sélectionnées par l'expert. La figure II.2 représente les deux instances a et b de la figure1 qui vont être considérées comme similaires grâce à notre mesure de similarité sémantique car elles sont en relation avec des instances elles-mêmes similaires.

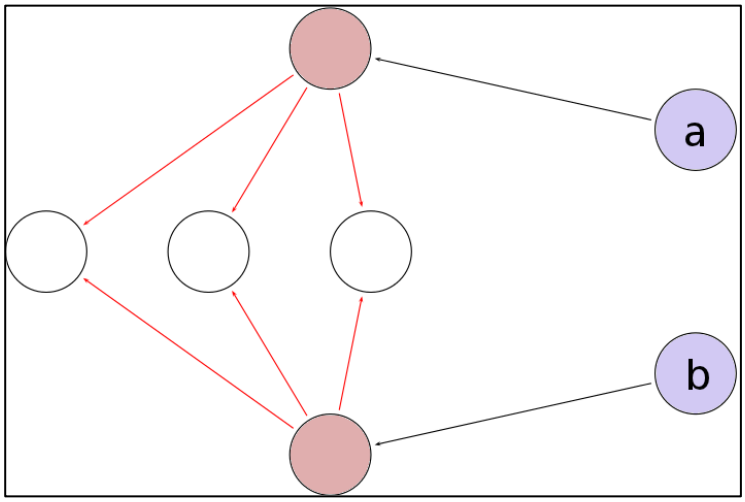
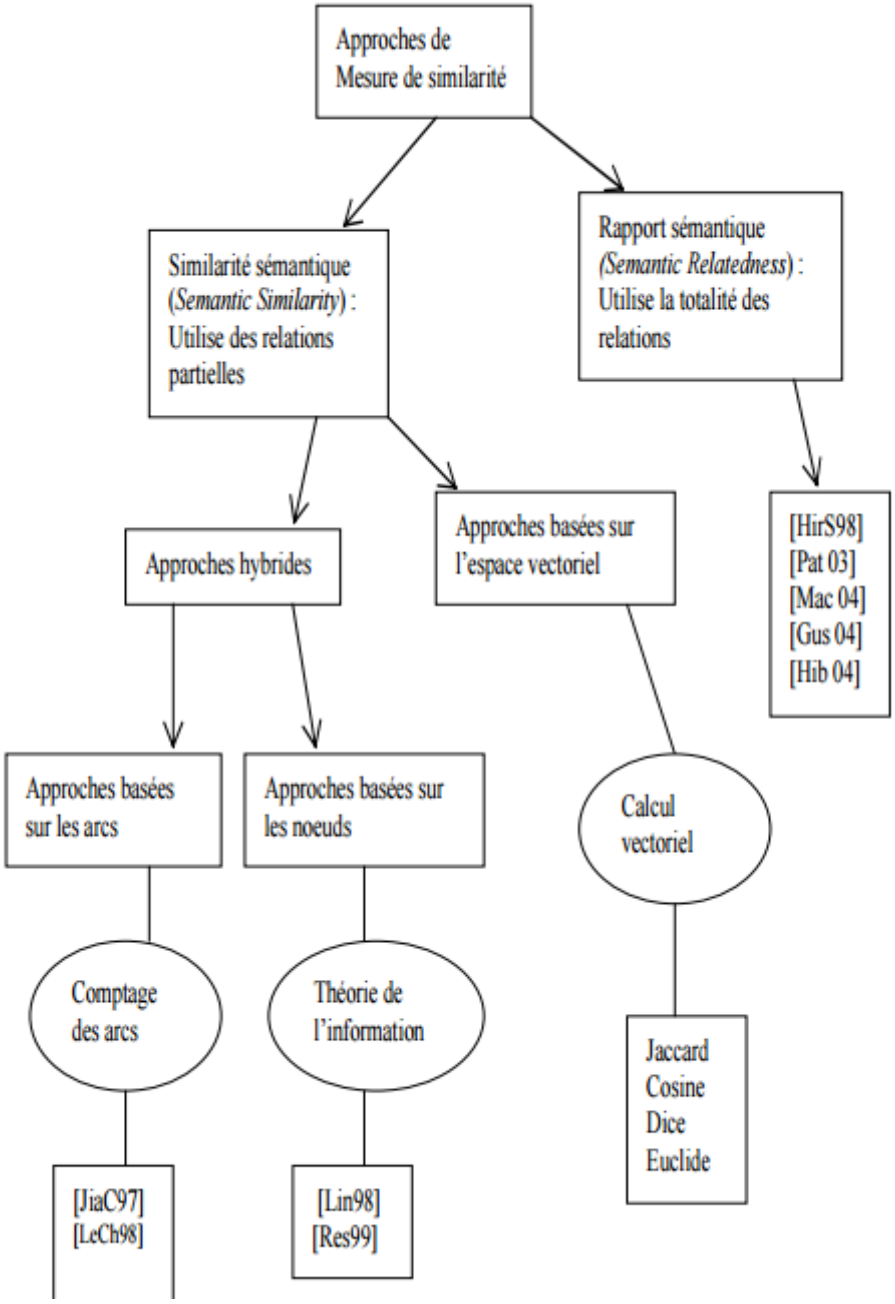


Figure II.4 : *Deux instances a et b similaires avec notre mesure de similarité sémantique*

II.7 Les approches de mesure de similarité

Les approches principales de la mesure de similarité. La figure suivante montre une partie de notre arrangement de classification ainsi que quelques approches d'échantillon.


Approches de Mesure de similarité
Similarité sémantique (*Semantic Similarity*) : Utilise des relations partielles
Rapport sémantique (*Semantic Relatedness*) : Utilise la totalité des relations
Approches hybrides
Approches basées sur l'espace vectoriel
[HirS98] [Pat 03] [Mac 04] [Gus 04] [Hib 04]
Approches basées sur les arcs
Approches basées sur les noeuds
Calcul vectoriel
Comptage des arcs
Théorie de l'information
Jaccard Cosine Dice Euclide
[JiaC97] [LeCh98]
[Lin98] [Res99]

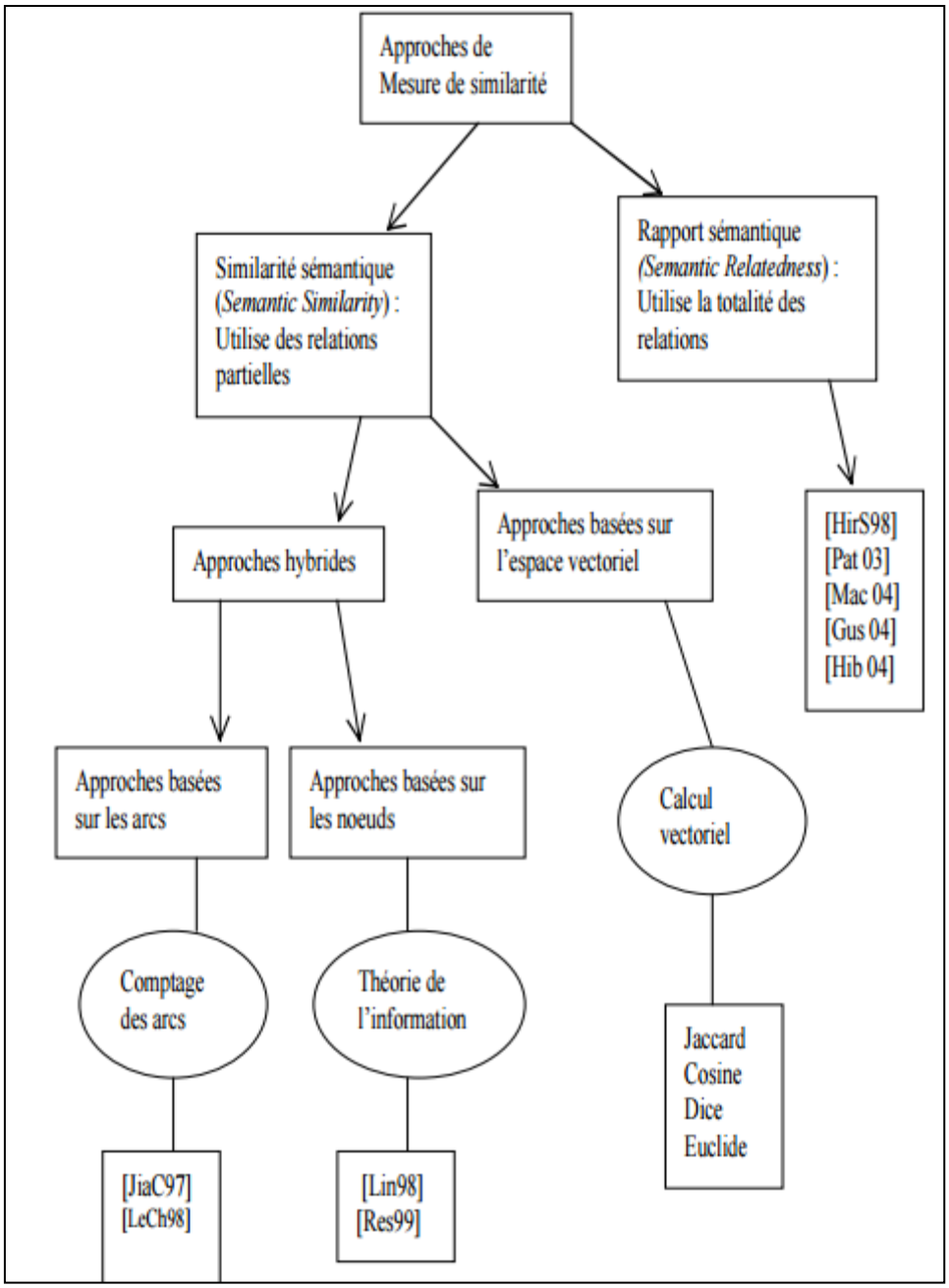


Figure II.5. *Taxonomie des approches de mesure de similarité*

II.7.1 *Approches basées sur l'espace vectoriel*

Dans le domaine de la recherche de l'information, les modèles de l'espace vectoriel sont largement adoptés [56] [57]. Ces approches utilisent un vecteur caractéristique, dans un espace dimensionnel, pour représenter chaque objet et calculent la similarité en se basant sur la mesure de cosine ou la distance euclidienne. Le modèle de l'espace vectoriel est employé pour un arrangement des objets complexes en les représentant comme des vecteurs de k-dimensions.

La définition de la similarité entre deux vecteurs d'objets est obtenue par leurs contenus internes. Parmi les approches citées dans la littérature on peut citer :

II.7.1.1*Similarité de Cosine*

Cette mesure utilise la représentation vectorielle complète, c'est-à-dire la fréquence des objets (mots). Deux objets (documents) sont similaires si leurs vecteurs sont confondus. Si deux objets ne sont pas similaires, leurs vecteurs forment un angle (X, Y) dont le cosinus représente la valeur de la similarité. La formule est définie par le rapport du produit scalaire des vecteurs x et y et le produit de la norme de x et de y.

$$sim(X,Y) = \cos(X,Y) = \frac{X * Y}{\|X\|_2 * \|Y\|_2} \qquad \text{II.2}$$

La mesure de Cosine [67] est donc la similarité numérique entre les deux vecteurs, comme le cosinus de l'angle entre les deux vecteurs.

II.7.1.2*Similarité de Pearson*

La mesure de similarité de Pearson est basée sur le calcul de la corrélation. Pour connaître le coefficient de corrélation liant deux séries X(x_1, x_2,…,x_n) et Y(y_1, y_2, …,y_n).

$$sim(X,Y) = \tau_p = \frac{\sum_{i=1}^{N}(x_i - \bar{x})\cdot(y_i - \bar{y})}{\sqrt{\sum_{i=1}^{N}(x_i - \bar{x})^2}\cdot\sqrt{\sum_{i=1}^{N}(y_i - \bar{y})^2}} \qquad \text{II.3}$$

Si r vaut 0, les deux courbes ne sont pas corrélées et donc ne sont pas similaire. Les deux courbes sont d'autant mieux corrélées que r est loin de 0 (proche de -1 ou 1). Avec :

$$\bar{x} = \frac{1}{N}\sum_{i=1}^{N} x_i \text{ : est la moyenne de x} \qquad \text{II.4}$$

$$\bar{y} = \frac{1}{N}\sum_{i=1}^{N} y_i \text{ : est la moyenne de y} \qquad \text{II.5}$$

II.7.1.3*Similarité euclidienne*

La similarité euclidienne est basée sur le ratio de la distance euclidienne augmenté de La distance euclidienne est définie par la formule suivante :

$$dE = \|x - y\|^2 \qquad \text{II.6}$$

La mesure de similarité est donc définie par :

$$sime(X,Y) = \frac{1}{1+dE} \qquad \text{II.7}$$

II.7.1.4 *Similarité de Jaccard*

La mesure de similarité de Jaccard est définie par le nombre des objets communs divisé par le nombre total des objets moins le nombre d'objets communs :

$$simj = \frac{x,y}{\|x\|_2^2 + \|y\|_2^2 - x.y} \qquad \text{II.8}$$

Tels que x et y sont des vecteurs extraits à partir des concepts X et Y.

II.7.1.5*Similarité de Dice*

La similarité de Dice [68] est définie par le nombre des objets communs multipliés par 2 sur le nombre total d'objets. La mesure de Dice est donc par la formule suivante :

$$simd(X,Y) = \frac{2x \cdot y}{\|x\|_2^2 + \|y\|_2^2} \qquad \text{II.9}$$

II.7.2 *Approches basées sur les arcs*

La mesure de similarité la plus intuitive des objets dans une ontologie est leurs distances [69] [70] [61]. Evidemment, un objet X est plus similaire à un objet Y qu'un objet Z. Cette similarité est évaluée par la distance qui sépare les objets dans l'ontologie.

Ces mesures se servent de la structure hiérarchique de l'ontologie pour déterminer la similarité sémantique entre les concepts. Le calcul des distances dans l'ontologie est basé sur un graphe de spécialisation des objets. Dans chaque graphe, la distance de l'ontologie doit être caractérisée par le plus court chemin qui fait intervenir un ancêtre commun ou le plus

petit généralisant, connectant potentiellement deux objets à travers des descendants communs. Parmi les travaux classifiés sous cette bannière on peut citer :

II.7.2.1 *Mesure de Wu & Palmer (1994)*

La mesure de similarité de [61] est basée sur le principe suivant : Etant donnée une ontologie formée d'un ensemble de nœud et un nœud racine R (Figure 4 Soit X et Y deux éléments de l'ontologie dont on calculer la similarité. Le principe de calcul de similarité est basé sur les distances (N1 et N2) qui séparent les nœuds X et Y du nœud racine et la distance qui sépare le concept subsumant (CS) de X et de Y du nœud R.

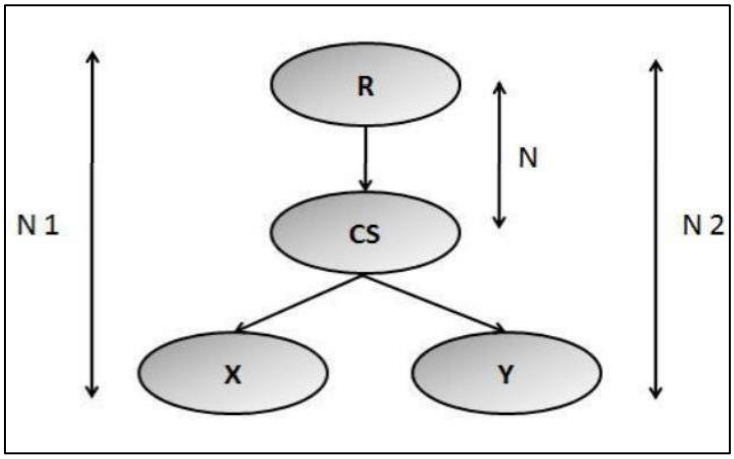


Figure II.6.*Exemple d'un extrait d'une ontologie*

La mesure de Wu et Palmer est définie par la formule suivante :

$$sim(X,Y) = \frac{2 * N}{N1 + N2} \qquad \text{II.10}$$

II.7.2.2 *Mesure de Rada et al (1989)*

Cette mesure est adoptée dans un réseau sémantique et elle est fondée sur le fait qu'on peut calculer la similarité en se basant sur les liens hiérarchiques «is-a». Pour calculer la similarité de concepts dans une ontologie, on doit calculer le nombre des arcs minimums qui les séparent. Cette mesure, est basée sur le calcul de distance entre les nœuds par le chemin le plus court, présente un moyen des plus évidents pour évaluer la similarité sémantique dans une ontologie hiérarchique. Il présente ainsi une mesure utilisant une métrique, distance(c1, c2), qui indique le nombre d'arcs minimum à parcourir pour aller d'un concept c1 à un concept c2.

$$sim(c1,c2) = \frac{1}{1 + distance(c1,c2)} \qquad \text{II.11}$$

II.7.2.3 Mesure de Ehrig et al

Un travail de mesure de similarité pour les ontologies a été introduit par [71]. Ce travail introduit trois couches : les données, l'ontologie et le contexte. La similarité des entités est mesurée au niveau des données en considérant les valeurs de données de type simple ou complexe (entiers, caractères). Les relations sémantiques entre les entités sont mesurées au niveau de la couche de l'ontologie. Finalement la couche du contexte spécifie comment les entités de l'ontologie sont utilisées dans un certain contexte externe, plus spécifiquement, le contexte de l'application.

II.7.2.4 *Approches basées sur les noeuds*

Ces approches adoptent une nouvelle mesure en termes de la mesure entropique (Contenu informationnel) de la théorie de l'information. La probabilité P pour l'identification de l'utilisateur d'une classe ou de ses descendants dans un corpus désigne l'information de la classe.

On définit l'entropie d'une classe par la formule suivante :

$$E(c) = \log(Pc) \qquad \text{II.11}$$

Où *P* est la probabilité de trouver une instance du concept *c*. La probabilité d'un concept c est calculée en divisant le nombre des instances de c par nombre total des instances.

En associant les probabilités aux concepts d'une taxonomie, il est possible d'éviter le manque de fiabilité des distances des arcs [58].

Parmi les travaux, recensés dans la littérature, sous cette catégorie on peut citer :

II.7.2.5 *Mesure de Resnik*

La notion du contenu informationnel (CI) a été initialement introduite par [62] qui a prouvé qu'un objet (mot) est défini par le nombre des classes spécifiées et que la similarité sémantique entre deux concepts est mesurée par la quantité de l'information qu'ils partagent. Pour évaluer la pertinence d'un objetil faut calculer le contenu informationnel. Le contenu informationnel est obtenu en calculant la fréquence de l'objet dans le corpus (Wordnet). La formule proposée par Resnik est définie par:

$$Sim(c1,c2) = Max\left[CI\left(PPG(c1,c2)\right)\right] \text{ où } CI = -\log\left(P(c)\right) \qquad \text{II.12}$$

II.7.2.6 *Mesure de Lin (1998)*

Lin a défini la similarité de concepts comme le rapport entre la quantité d'informations nécessaires pour indiquer le point commun entre ces deux concepts et les informations nécessaires pour les décrire [72]. Cette mesure est légèrement différente de celle de Resnik:

$$Sim(a,b) = \frac{2 * CI\left(PPG(a,b)\right)}{CI(a) + CI(b)} \qquad \text{II.13}$$

Cette mesure utilise une approche hybride qui combine deux sources de connaissances différentes (Thesaurus, corpus). En plus, elle représente la similarité comme degré probabiliste de chevauchement des concepts descendants de a et b. Les travaux de [73] ont évalué cette mesure à travers une expérience qui utilise des sujets humains pour évaluer la similarité entre 30 paires de noms, il en ressort que cette méthode offre une amélioration significative.

II.7.2.7 *Mesure de Hirst et Onge*

L'idée de cette mesure est que deux concepts lexicalisés sont sémantiquement étroits si leurs ensembles synonymes (synsets) de WordNet sont reliés par un chemin qui n'est pas trop long et qui "n'échange pas la direction trop souvent". Avec cette mesure, toutes les relations contenues dans un réseau Wordnet sont prises en considération. Dans le travail de [74], les auteurs ont classé la direction des liens en lien haut (super-classe), lien bas (sous-classe)et lien horizontal (antonyme). Le calcul de la similarité, avec cette méthode, s'effectue entre objets (mots) par le poids du plus court chemin allant d'un terme à un autre, en plus des classifications qui indiquent les changements de direction. La force du rapport est donnée par :

$$Simht = T - PCC - K * nd \qquad \text{II.13}$$

Où *T* et *K* sont des constantes, *PCC* est la distance du plus court chemin en nombre d'arc et nd le nombre de changements de direction.

II.7.3 *Approches hybrides*

Ces approches sont fondées sur un modèle qui combine entre les approches basées sur les arcs (Distances) en plus du contenu informationnel qui est considéré comme facteur de décision. [75]

II.7.3.1 *Mesure de Jiang et Conrath (1997)*

Pour remédier au problème présenté au niveau de la mesure de Resnik, Jiang et Conrath [76]ont apporté une nouvelle formule qui consiste à combiner l'entropie (contenu informationnel) du concept spécifique à ceux des concepts dont on cherche la similarité (combine entre les techniques basées sur les arcs et les techniques basées sur les nœuds qui consistent à compter les arcs afin d'améliorer les résultats par les calculs basés sur les nœuds. Notons que cette formule est définie par l'inverse de la distance sémantique.

$$sim(c1,c2) = \frac{1}{distance(c1,c2)} \qquad \text{II.14}$$

Sachant que la distance entre c1 et c2 est calculée par la formule suivante :

$$distance(c1,c2) = CI(c1) + CI(c2) - \left(2.CI\left(PPG(c1,c2)\right)\right) \qquad \text{II.15}$$

II.7.3.2 *Mesure de Leacock et Chodorow*

Une autre méthode présentée combine la méthode de comptage des arcs et la méthode du contenu informationnel [77]. La mesure proposée par Leacock et Chodorow est basée sur la longueur du plus court chemin entre deux synsets de Wordnet. Les auteurs ont limité leur attention à des liens hiérarchiques «is-a» ainsi que la longueur de chemin par la profondeur globale de la taxonomie. La formule est définie par :

$$Sim(X,Y) = -Log\left(\frac{CD(X,Y)}{2*M}\right) \qquad \text{II.16}$$

Où M est la longueur du chemin le plus long qui sépare le concept racine, de l'ontologie, du concept le plus en bas (la profondeur globale). On dénote par CD(X, Y) la longueur du chemin le plus court qui sépare X de Y.

II.7.4 *Quelques approches de la littérature*

Dans la table II.1 montre quelques branches de similarité qui accordent les critères de classification présentés dans la section précédente :

	[78] [79] [61]	[80] [77]	[75] [62] [68]	[81]
Type de similarité	Lexicale	Lexicale	Lexicale	Lexicale
Source de données	Dictionnaire monolingue	Wordnet	Wordnet et corpus	Thesaurus
Performance	Assez bien	très bien	très bien	très bien
Langues d'utilisation	Plusieurs Langues	Quelques Langues	Quelques Langues	Quelques Langues
Type de données	Non Structuré	structuré	structuré	Structuré

Tableau II.1 : Classification des caractéristiques de quelques travaux de similarités.

Les méthodes qui utilisent Wordnet ou Thesaurus peuvent fournir des résultats excellents, puisque Wordnet et thesaurus sont manuellement crées et les relations entre les mots, dans les ressources, sont bien explicites. Par conséquent, la similarité entre deux mots dépend simplement de leur position relative dans la ressource.

II.8 Recommandation basée sur la sémantique

La sémantique peut être définie succinctement comme l'étude du sens des mots. Son application dans le domaine de l'informatique ouvre de nouvelles perspectives au traitement des données.

En effet, les technologies d'analyse sémantique s'attachent à rendre exploitables les données textuelles en étudiant leur sens et en les structurant : cela permet, entre autres, d'identifier la langue d'écriture d'un texte, d'extraire des relations entre les informations, de les classifier et de les prioriser automatiquement. Il est ensuite possible d'exploiter ces informations pour mener à bien des projets aussi différents qu'étudier la « voix du client », pour adapter ces campagnes publicitaires, ou encore mettre en place un moteur de recherche sémantique pour valoriser ses archives.

La sémantique a été introduite par plusieurs méthodes dans le processus de recommandation. Les systèmes de recommandation basés sur la sémantique évoluent au rythme des méthodes et outils proposés dans le domaine du Web sémantique.

- **SiteIF** a été le premier système à adopter une représentation basée sur le sens des documents pour construire un modèle des intérêts de l'utilisateur SiteIF est un agent personnel pour un site Web de nouvelles multilingues. La source externe de connaissance impliquée dans le processus de représentation est MultiWordNet (une base de données lexicale multilingue).
- **ITR** (ITerm Recommender) est un système capable de fournir des recommandations d'items dans plusieurs domaines (films, musique, livres), à condition que les descriptions d'articles soient disponibles sous forme de documents texte
- **SEWeP** (Semantic Enhancement for Web Personalization) est un système de personnalisation Web qui utilise à la fois les logs d'utilisation et la sémantique du contenu du site Web dans le but de le personnaliser. Une taxonomie des catégories spécifiques au domaine a été utilisée pour annoter sémantiquement les pages Web, afin d'avoir un vocabulaire uniforme et consistant.
- **Quickstep** est un système de recommandation d'articles de recherche académique. Le système adopte une ontologie d'articles de recherche basée sur la classification scientifique du projet DMOZ open directory (DMOZ open directory project) (27 classes utilisées).
- **Informed Recommender** utilise les avis des utilisateurs sur les produits pour faire des recommandations. Le système convertit les opinions des clients dans une forme structurée en utilisant une ontologie de traduction, qui est exploitée pour la représentation et le partage de connaissance. Ces méthodes ci-dessus ont donné des résultats pertinentes et plus précis comparés aux méthodes traditionnelles basées sur le contenu. [12]

II.9 Conclusion

Dans ce chapitre, nous avons d'abord, parlé sur l'ontologie avec ses composantes et types, ensuite, la similarité sémantique et quelque approche enrichi avec les lois qui convient et nous avons également passé en revue la recommandation basée sur la sémantique pour pouvoir mieux expliquer le chapitre suivant.

PARTIE II :
Modélisation & Experimentation

Chapitre III

Modélisation

III.1 Introduction

Il existe, à ce jour, plusieurs approches de recommandation, chacune ayant ses avantages et ses limitations. C'est pourquoi, l'idée de combiner différentes approches, afin de bénéficier de leurs avantages respectifs et de pallier leurs limitations, a vite été envisagée. Plusieurs techniques d'hybridation ont été définies afin de répondre à différentes attentes. La première étape dans la définition d'un tel système hybride, consiste à définir les modules de recommandation utilisés lors de l'hybridation.

Dans cette partie de ce mémoire nous allons présenter un système de recommandation hybride qui consiste à appliquer séparément le filtrage collaboratif et autres techniques de filtrage basé sur le contenu pour générer des recommandations candidates, et ensuite combiner ces ensembles de recommandations selon une méthode. En premier temps nous décrivons l'architecture que nous proposons et qui est composée de : module collaboratif et module sémantique qui seront bien détaillé par la suite.

Dans la communauté de la sémantique, les mesures de similarité sémantique sont le plus souvent utilisées dans l'alignement d'ontologies, leur but étant de retrouver les concepts similaires. Les plus connus sont ceux qui utilisent une ontologie comme WordNet. C'est dans ce contexte qui s'inscrit notre travail.

III.2 L'architecture du système

L'architecture du système de recommandation est illustrée sur la Figure III.1. Cette architecture permet à la fois d'avoir deux systèmes hybridés, l'un est un module collaboratif, et l'autre un module sémantique.

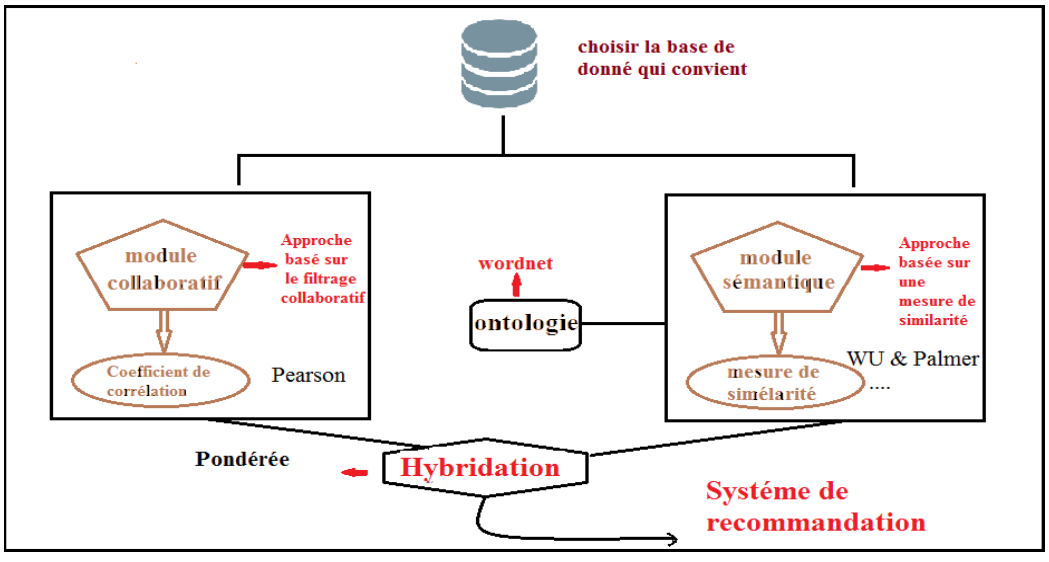


Figure III.1.*l'architecture du système de recommandation.*

Nous allons dans un premier temps choisir les données sur lesquelles nous allons travailler, ensuite, nous abordons deux parties appliqué séparément qu'on va les hybrider par la suite selon une méthode d'hybridation appelé hybridation pondérée, la première partie est un système collaboratif standard qui se focalise sur un coefficient de corrélation (Pearson), contrairement à la deuxième partie qui utilise des règles que nous avons étendus grâce a la sémantique des données afin de générer de nouvelles recommandation.

III.3 Module collaboratif

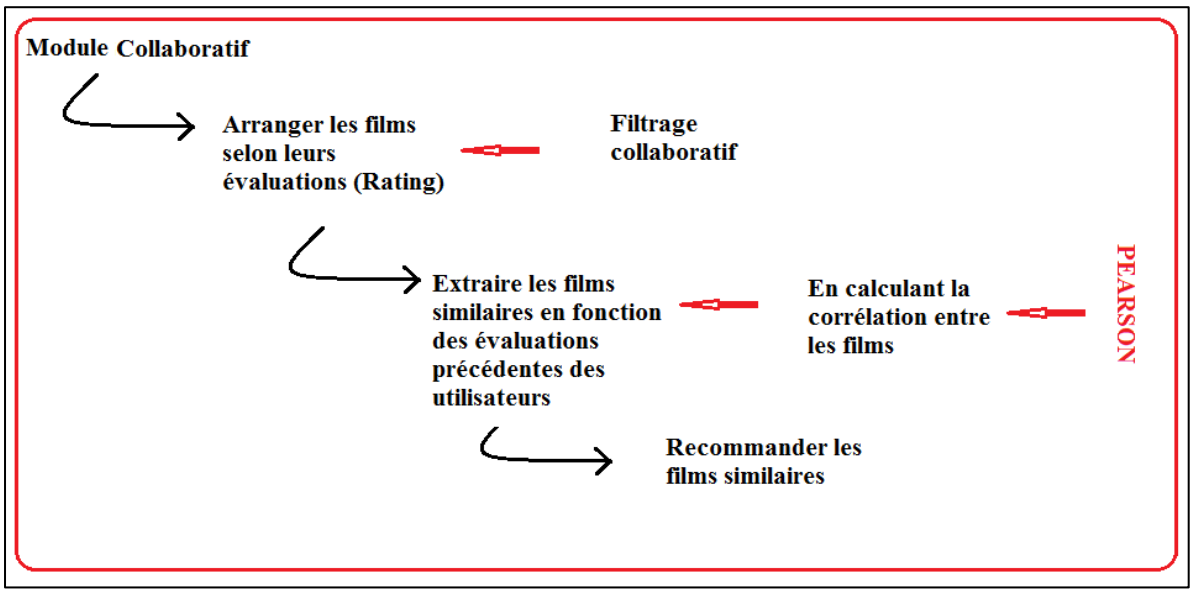


Figure III.2. *L'architecture du module collaboratif*

III.3.1 *Principe*

Le filtrage collaboratif (*Collaborative Filtering*) a pour principe d'exploiter les évaluations que des utilisateurs ont faites sur certains documents, afin de recommander ces mêmes documents à d'autres utilisateurs, et sans qu'il soit nécessaire d'analyser le contenu des documents [82]. Par exemple, dans la Figure III.3, supposons que l'on a des communautés formées par la proximité des évaluations des utilisateurs. Le document *d* sera recommandé à l'utilisateur *u*, car ce document est apprécié dans la communauté G où se trouve l'utilisateur.

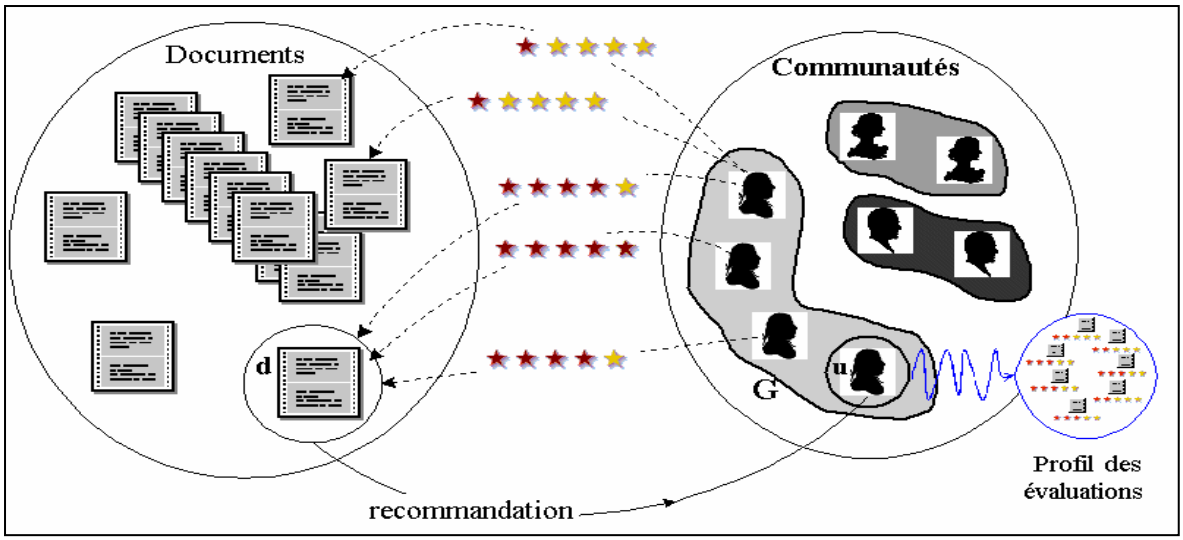


Figure III.3. *Principe général du filtrage collaboratif*

Il existe deux types de modèles collaboratifs, comme déjà expliqué dans l'état de l'art. Les méthodes basées sur la mémoire et les méthodes basées sur les modèles, notre premier système est basé sur les modèles.

Notre système identifie des articles (films) similaires en fonction des évaluations précédentes des utilisateurs. Par exemple, si les utilisateurs A, B et C attribuent un classement de 5 étoiles aux filmes X et Y, lorsqu'un utilisateur D regarde le film Y, il reçoit également une recommandation du film X car le système identifie les films X et Y comme similaires en fonction des évaluations des utilisateurs A, B et C. C'est exactement comme notre modèle de travail, nous allons utiliser ces évaluations pour calculer la corrélation entre les films.

III.3.2 *Coefficient de corrélation*

On utilise le coefficient de corrélation qui est une mesure numérique (statistique) d'un type de corrélation, ce qui signifie une relation statistique entredeux variables. Il existe plusieurs types de mesure, on a choisis le coefficient de corrélation de Pearson.

III.3.3 *Le coefficient de corrélation de Pearson*

Ce nombre sera compris entre -1 et 1. 1 indique une corrélation linéaire positive tandis que -1 indique une corrélation négative. 0 n'indique aucune corrélation linéaire. Par conséquent, les films avec une corrélation nulle ne sont pas du tout similaires.

A l'aide d'une matrice avec les titres de films en tant que ligne, user_id en tant que colonnes et les évaluations en tant que valeurs.

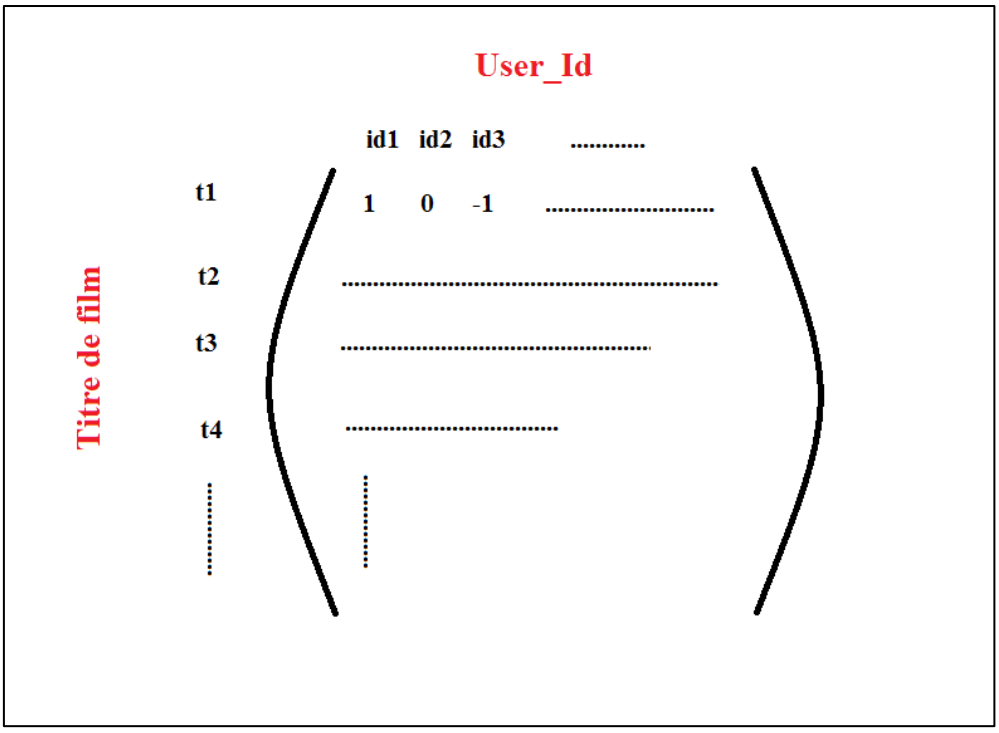


Figure III.4. La matrice obtenue dans le module collaboratif

III.4 Module sémantique

La recommandation, dans le module sémantique, consiste à présenter des films sémantiquement similaires à ceux regardé par l'utilisateur.

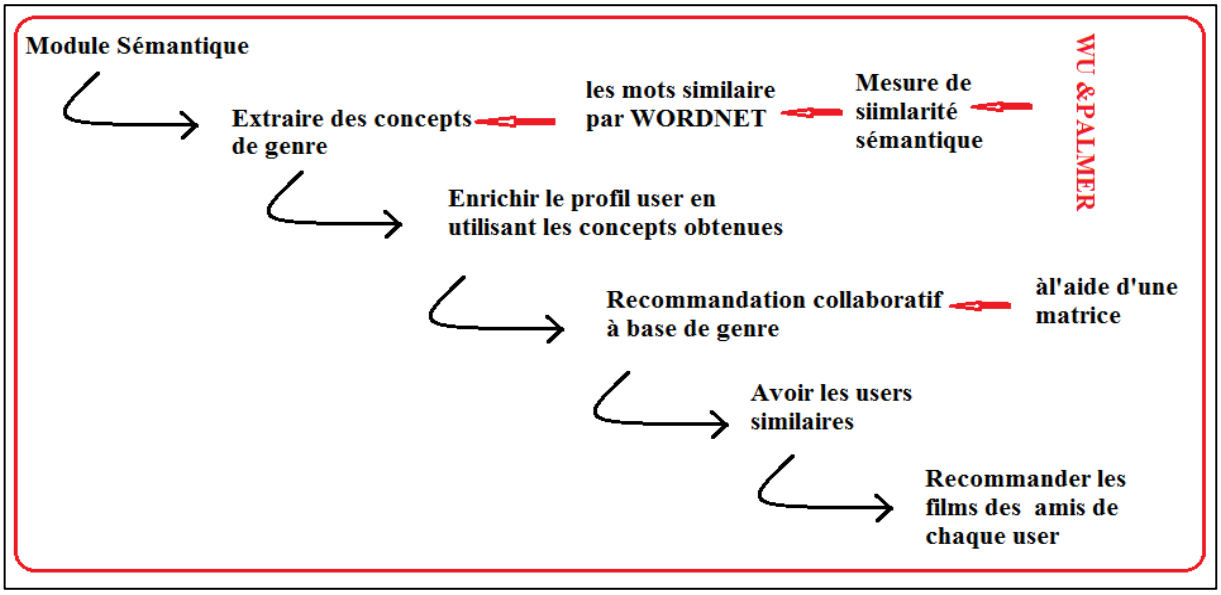


Figure III.5.*L'architecture du module sémantique*

Les nouvelles techniques de recommandation utilisent récemment des ontologies pour définir une typologie des items à recommander qui améliorera les performances du système.

Dans notre cas nous allons recommander à l'utilisateur des films selon le genre de son film vu, à base d'ontologie. Dans notre approche, l'ontologie est intégrée dans le processus de calcul de similarité sémantique.

III.4.1 *Similarité sémantique à base d'ontologie*

Les ontologies ont été largement exploitées dans les systèmes à base de connaissance, et plus précisément, pour le calcul de la similarité sémantique [83].

Notre démarche consiste donc à utiliser une ontologie, qui offre une structure plus riche sémantiquement tel que le WordNet afin de désambiguïser le sens des ressources et de filtrer l'information pertinente.

III.4.2 *La mesure de similarité WUP*

Nous allons dans un premier temps extraire les concepts de genre en utilisant la mesure de similarité WUP (Wu & Palmer).

$$sim(X,Y) = \frac{2 * N}{N1 + N2} \qquad \text{III.1}$$

A partir des genres des films nous allons les réduire pour avoir moins de genres qu'on les appelle concepts, afin de tracer la hiérarchie de notre ontologie, après avoir enrichir le profil de chaque user par les nouveaux genres (concepts) nous utilisons une matrice avec les User_Id en tant que lignes, les nouveaux genres obtenus en tant que colonnes, et remplir la matrice par calcule de la somme des évaluations des films vus par chaque utilisateur, c'est-à-dire calcule par poids et classement, afin d'avoir les utilisateurs similaires pour recommander les films de l'un et autre.

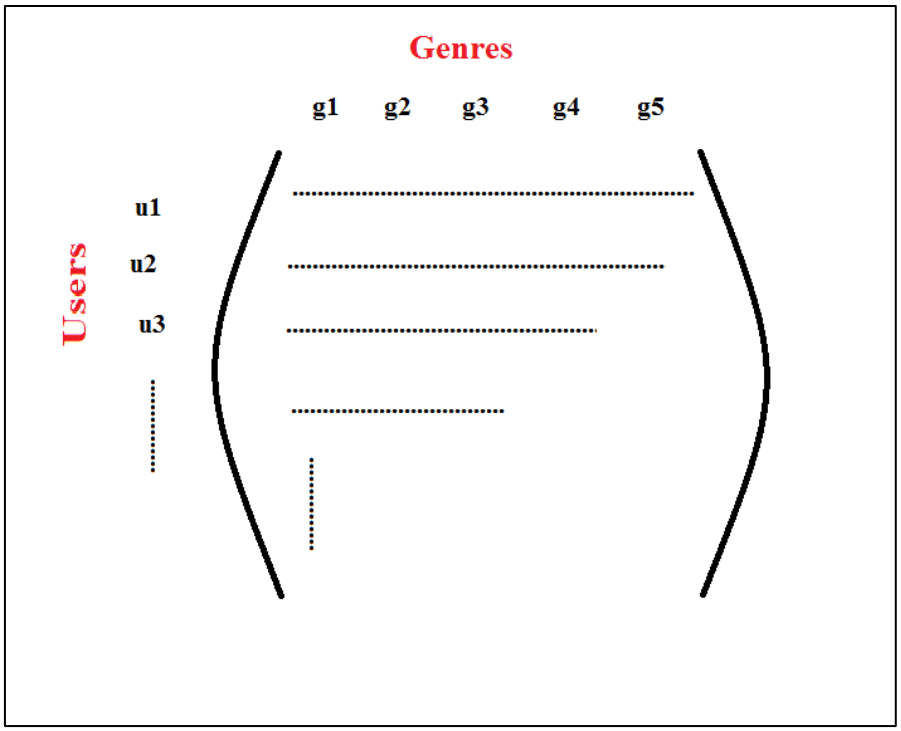


Figure III.6.*la matrice obtenue dans le module sémantique*

Nous obtenons les utilisateurs similaires a partir de la matrice en calculons la similarité entre un utilisateur U et le reste des utilisateurs Ur par :

$$sim(Ur,U) = \sum\sqrt{(Ur - U)} \qquad \text{III.2}$$

III.5 L'hybridation

En vue d'améliorer la qualité des recommandations et d'atteindre la satisfaction des utilisateurs, cette partie présente l'approche de recommandation hybride qu'on a utilisé visant à rendre les systèmes de recommandation plus adaptatifs et capables de considérer les différents choix et goûts des utilisateurs.

Nous allons utiliser la méthode d'hybridation pondérée suivante :

III.5.1 Hybridation pondérée

En effectuant la somme pondérée du score de l'item dans chaque approche de recommandation déjà présenté précédemment, en appliquant la règle suivante :

$$\alpha x + \beta y \qquad \text{III.3}$$

Où *x* est les degrés/évaluations des résultats du premier système (module collaboratif), y est les degrés/évaluations des résultats du deuxième système (module sémantique) et α et β varient entre 0 et 1 avec $\alpha+\beta=1$.

III.6 Conclusion

Dans ce chapitre nous avons décrit comment notre système de recommandation hybride marche, et comment hybrider les deux modules proposés:

- Module collaboratif : approche basée sur le filtrage collaboratif.
- Module sémantique : approche basée sur une mesure de similarité.

Ces deux modules peuvent être utilisés indépendamment ou combinés afin d'obtenir un module hybride.

Le filtrage collaboratif se base sur l'évaluation(Rating) des utilisateurs. La mesure de similarité sémantique calcule les instances similaires de l'ontologie, telles que deux instances sont considérées comme similaires si elles sont reliées à des instances communes ou à des instances elles-mêmes similaires. L'évaluation et l'expérimentation seront présentées dans le chapitre suivant.

Chapitre IV

Expérimentation et Evaluation

IV. Introduction

Nous décrivons dans cette partie les expérimentations et évaluations menées afin de valider le système de recommandation que nous proposons.

Ce chapitre est consacré à la partie expérimentation et réalisation de notre application, qui consiste à recommander des films aux utilisateurs en combinant deux systèmes de recommandation : collaboratif et sémantique. Nous présentons, également l'architecture de notre système, l'ontologie utilisée, l'environnement de travail choisi tout en présentant les langages et les outils utilisés ainsi que les copies d'écran de chaque étape.

Nous allons détailler le déroulement de l'expérimentation et de l'évaluation dans le domaine des films. Enfin, nous analysons les résultats obtenus.

IV.1Base de données

Les différentes approches proposées au niveau du chapitre II de la première partie sont établies et testées sur des ensembles de données des films, nous allons utiliser le MovieLens Dataset. Cet ensemble de données a été mis en place par le groupe de recherche Grouplens de l'Université du Minnesota. Il contient 1, 10 et 20 millions d'évaluations. Movielens possède également un site Web sur lequel nous pouvant contribuer des critiques et obtenir des recommandations de films. Nous pouvons trouver davantage de jeux de données pour diverses tâches de science des données à partir des ressources de données de Dataquest. Nous avons utilisé celle de 20 millions d'évaluations sur notre expérimentation.

IV.2 L'environnement de développement

Nous avons utilisé le langage Python[1] utilisé dans notre projet en raisons, de sa rapidité de développement (qualité propre aux langages interprétés), la grande quantité de modules fournis dans la distribution de base.

Nous avons ajouté des bibliothèques de Python qui nous aide à lire et écrire dans les fichiers Excel, ajouter des fonctions mathématiques, ou même de faire des graphes.

Le tableau suivant cite les déférentes bibliothèques de Python utilisé dans notre projet:

[1]https://www.python.org/.

La bibliothèque	Son but
Xlrd	lire un fichier Excel
Xlwt	écrire dans un fichier Excel
matplotlib.pyplot	pour faire les déférents diagrammes qui explique les résultats de notre travail
Math	pour utiliser des fonctions de calcule prédéfini

Tableau IV.1. *Les bibliothèques utilisées dans notre projet*

Comme nous avons utilisé les outils :

- *Pandas* [1] (fournissant des structures de données hautes performances et faciles à utiliser, ainsi que des outils d'analyse des données) ;
- *Numpy* [2] (pour le calcul scientifique) ;
- *NLTK* [3] (interface pour la création de programmes*) ;*
- Ainsi que l'Ontologie *WordNet.*

IV.3 Module collaboratif

Nous avons utilisé les movie-lens pour construire un système de recommandation simple basé sur la similarité d'articles.

Nous avons fais un prétraitement sur la base de donnée qu'on a utilisé, c'est fusionner les titres avec les ID pour voir les titres du film au lieu de s'occuper uniquement des ID. Les colonnes item_id étant identiques, nous pouvons fusionner ces ensembles de données sur cette colonne.

Notre table contient :

- user_id - l'ID de l'utilisateur qui a évalué le film.
- item_id - l'identifiant du film.
- rating - La note attribuée par l'utilisateur au film, entre 1 et 5.
- timestamp - Le moment où le film a été évalué.
- title - Le titre du film.

[1]. https://pandas.pydata.org/
[2]. https://www.numpy.org/
[3]https://www.nltk.org/

Nous pouvons dire que la note moyenne est de 3,52 et le maximum est 5. Nous constatons également que le jeu de données compte 100003 enregistrements.

Créons maintenant un cadre de données avec la note moyenne pour chaque film et le nombre de notes. Nous allons utiliser ces évaluations pour calculer la corrélation entre les films plus tard. La corrélation est une mesure statistique qui indique dans quelle mesure deux variables ou plus fluctuent ensemble. Les films qui ont un coefficient de corrélation élevé sont les films qui se ressemblent le plus. Dans notre cas, nous utiliserons le coefficient de corrélation de Pearson . Ce nombre sera compris entre -1 et 1. 1 indique une corrélation linéaire positive tandis que -1 indique une corrélation négative. 0 n'indique aucune corrélation linéaire. Par conséquent, les films avec une corrélation nulle ne sont pas du tout similaires. Afin de créer cette base de données, nous utilisons la groupby fonctionnalité pandas. Nous regroupons le jeu de données par le title colonne et calculez sa moyenne pour obtenir la note moyenne de chaque film.

```
ratings = pd.DataFrame(df.groupby('title')['rating'].mean())
```

Ensuite, nous aimerions voir le nombre d'évaluations pour chaque film. Nous faisons cela en créant une colonne'number_of_ratings'. Ceci est important pour que nous puissions voir la relation entre la note moyenne d'un film et le nombre de notes qu'il a obtenues. Il est très possible qu'un film 5 étoiles ait été évalué par une seule personne. Il est donc statistiquement incorrect de classer ce film comme étant un film 5 étoiles. Nous devrons donc définir un seuil pour le nombre minimal d'évaluations lors de la création du système de recommandation. Afin de créer cette nouvelle colonne, nous utilisons l' groupby utilitaire pandas. Nous regroupons par title colonne et utilisons ensuite la count fonction pour calculer le nombre d'évaluations obtenues par chaque film.

Traçons maintenant un histogramme en utilisant la fonctionnalité de traçage de panda pour visualiser la répartition des notes. Nous pouvons voir que la plupart des films sont classés entre 2,5 et 4. Ensuite, visualisons la colonne'number_of_ratings'de manière similaire.
Et traçons aussi un digramme pour voir la relation entre le classement d'un film et le nombre de classements.

Nous devons convertir notre ensemble de données en une matrice avec les titres de films comme colonnes, user_id l'index et les évaluations comme valeurs. Nous allons utiliser cette matrice pour calculer la corrélation entre les votes d'un seul film et le reste des films de la matrice. Ensuite, nous allons organiser les films parmi les mieux notés.

Supposons qu'un utilisateur a regardé Air Force One (1997) et Contact (1997). Nous voudrions recommander des films à cet utilisateur sur la base de cet historique. L'objectif est de rechercher des films similaires à Contact (1997) et Air Force One (1997) que nous recommanderons à cet utilisateur. Nous pouvons y parvenir en calculant la corrélation entre les cotes de ces deux films et celles du reste de la population. La première étape consiste à créer un cadre de données avec les évaluations de ces films de notre movie_matrix.

Nous avons maintenant les cadres de données montrant le user_idet la note qu'ils ont donnée aux deux films.

Afin de calculer la corrélation entre ces deux films, nous utilisons les corwith fonctionnalités de pandas. Corrwith calcule la corrélation par paire de lignes ou de colonnes de deux objets de base de données. Utilisons cette fonctionnalité pour obtenir la corrélation entre le classement de chaque film et le classement du film Air Force One.

Nous allons maintenant obtenir les films qui ressemblent le plus à Air Force One (1997) & Contact (1997).Nous les trions ensuite afficher les 10 premiers.

IV.3.1 Résultats

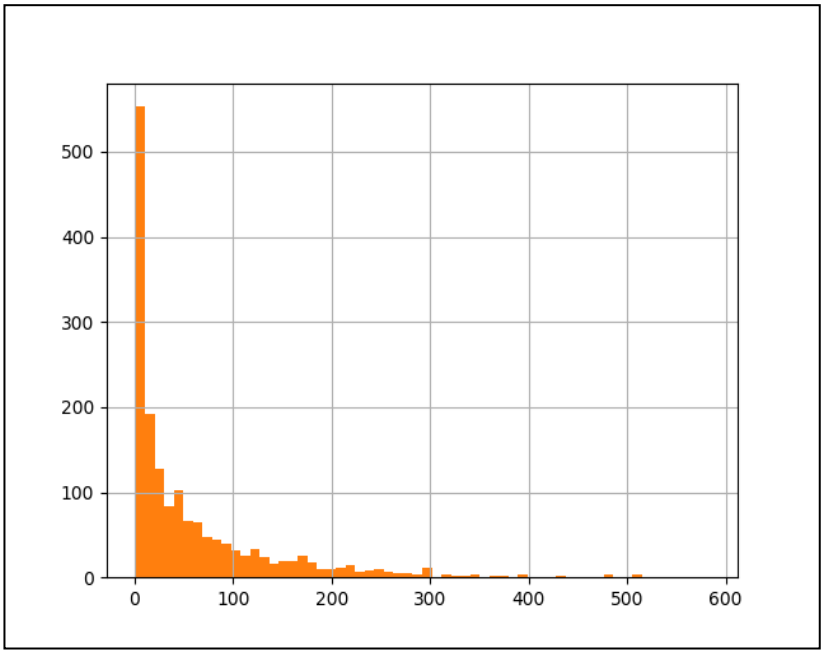


Figure IV.1.*Histogramme de la distribution des notes*

Il ressort clairement de l'histogramme ci-dessus que la plupart des films ont peu de notes. Les films ayant le plus d'audience sont ceux qui sont les plus célèbres.

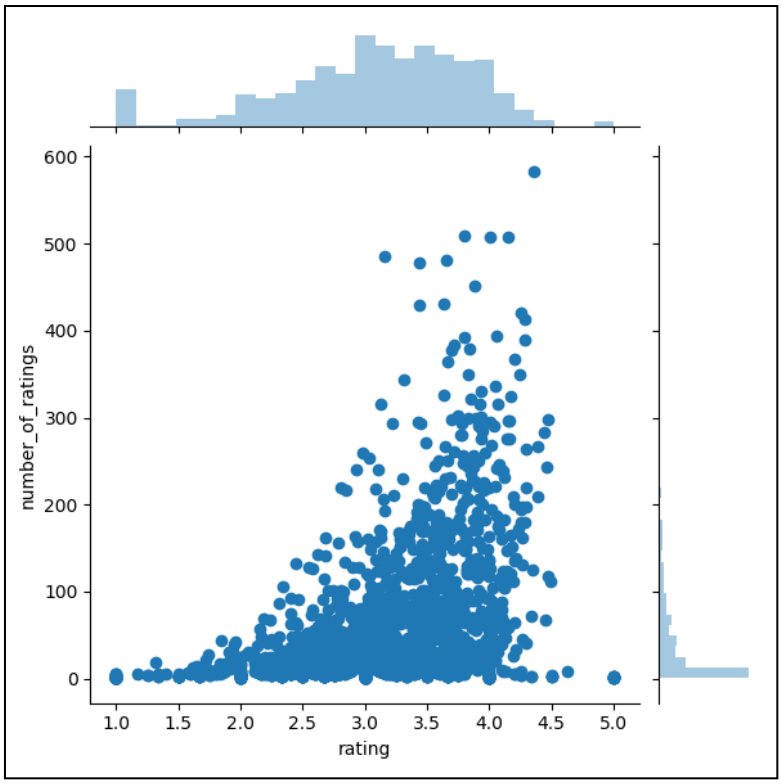


Figure IV.2. *La relation entre le classement d'un film et le nombre de classements*

Le diagramme montre qu'il s'agit d'une relation positive entre la note moyenne d'un film et le nombre de notes. Le graphique indique que plus un film obtient de notes, plus sa note moyenne est élevée. Il est important de noter cela en particulier lorsque vous choisissez le seuil pour le nombre de classements par film.

title	correlation	number_of_ratings
Air Force One (1997)	1.000000	431
Hunt for Red October, The (1990)	0.554383	227
Firm, The (1993)	0.526743	151
Murder at 1600 (1997)	0.514906	218
Eraser (1996)	0.500606	206
Absolute Power (1997)	0.497411	127
Rock, The (1996)	0.493542	378
Long Kiss Goodnight, The (1996)	0.490233	185
Crimson Tide (1995)	0.481205	154
My Fair Lady (1964)	0.474003	125

title	Correlation	number_of_ratings
Contact (1997)	1.000000	509
Philadelphia (1993)	0.446509	137
Mask, The (1994)	0.418328	129
Young Guns (1988)	0.388839	101
Sling Blade (1996)	0.384840	136
Sneakers (1992)	0.377275	150
Firm, The (1993)	0.376987	151
Arsenic and Old Lace (1944)	0.373759	115
Outbreak (1995)	0.358015	104
Little Women (1994)	0.352081	102

Figure IV.3. *Résultat des films à recommander du premier système*

IV.4.Module sémantique

Nous allons utiliser la même base de données (movielens) pour construire ce système de recommandation qui est basé sur une mesure de similarité. La première chose que nous allons faire est de partitionner notre travail en trois parties :

La première partie consiste à réduire le nombre de genres, à partir de 19 genres qu'on a sur la base, nous allons travailler sur 5 genres qui sont : **Action, Adventure, Children, Comedy, Science-Fiction**. Nous avons choisis de travailler par ses cinq concepts car ils sont les genres les plus dominants dans la base de données.

A l'aide de l'implémentation de la fonction de similarité Wu & Palmer WordNet (wup_similarity) nous avons pu calculer la similarité entre les genres pour les arranger dans un tableau comme suit :

Concepts :	**Action** ↓	**Children** ↓	**Sci– fic** ↓	**Adventure** ↓	**Comedy** ↓
	War	Documentary	Thriller	Crime	Sci-fic
	Crime	Musical	Fantay	Action	Romance
	Horror	Western	Romance	War	Thriler
	Black	Animation	Comedy	Drama	Action
	Drama	Action	Action	Mystery	
	Adventure				

Tableau IV.1. *Résultat de la mesure de similarité*

Il est très possible qu'un concept possède 6 termes et d'autre possède 4 seulement, c'est parce que les valeurs de similarité se varient, et le choix est basé sur un seuil qui est égale à 0.056 ce seuil qui est la moyenne des valeurs de la similarité résultante.

Ensuite, on nous avons enrichi le profil de chaque utilisateur par ses cinq nouveaux concepts seulement, nous avons ajouté un champ à chaque utilisateur nommé nouveau genre rempli par le genre qui convient selon le tableau précédent, par exemple si un film est de genre horror donc son nouveaux genre est Action.

Mais, si nous remarquons les résultats, les termes peuvent se répéter donc l'enrichissement sera basé sur un score, par exemple : si un film est de genre **War,** et **War** appartient à **Action** et **Adventure,** mais le score de **war** dans **action** est élevé que le score de **war** dans **adventure.**

Donc, comme cela, les genres sont réduis, et nous allons travailler par la suite seulement sur ses cinq genres. Cela nous a aidé à construire la matrice présenté dans la figure suivante qui est la deuxième partie de notre module sémantique.

La construction de la matrice :

- les User_Id en colonnes.
- les nouveaux genres obtenus comme indexe.

- Remplir la matrice par calcule de la somme des évaluations des films vus par chaque utilisateur, par exemple :

  Si user1 a vu Films A, B et C qui sont évalué 4,4.5 et 3.5 respectivement, et bien sûr chaque film a son nouveau genre (les 5 concepts obtenus) Film A → genre 2 , Film B → genre 4 , Film C → genre 2.

  D'après ses informations on fait la relation entre l'utilisateur et le nouveau genre :

  User1:

  Genre1= 0

  Genre2= film A + film B= (4*10) + (3.5*10)=70

  Genre3= 0

  Genre4= film B = 4.5*10 =45

Remarque : chaque évaluation est fois 10 pour éviter les virgules et les erreurs sur Python.

	Action	Children	Comedy	Sci-Fi	Adventure
User 1	825	525	660	75	2925
User 2	400	30	0	0	1070
User 3	1170	700	390	30	3270
User 4	270	110	110	0	410
User 5	300	260	350	0	1330
User 6	0	40	60	0	630
User 7	940	650	480	50	4510
User 8	590	230	170	0	1160
User 9	60	190	30	0	330
User 10	340	170	120	40	510
User 11	4300	2345	1670	85	7585
User 12	180	130	120	0	740
User 13	550	160	160	0	870
User 14	255	1070	1405	15	4545
User 15	160	300	140	0	640
User 16	300	0	130	0	1650
User 17	490	0	30	0	320
User 18	195	800	155	45	1445
User 19	160	160	200	0	870
User 20	160	195	35	0	430
User 21	590	1370	260	90	1870
User 22	1800	890	150	0	1360
User 23	720	1030	110	50	950
User 24	2740	1940	580	40	6790
User 25	525	630	345	0	2040
User 26	370	280	180	30	870

Figure IV.4.*La matrice obtenue (user_id / genre)*

Cette matrice nous a aidé de faire une recommandation collaboratif à base de genre, ce que permet d'avoir les utilisateurs similaires de chaque utilisateurs afin de lui recommander les films de ses amis (utilisateurs similaires).

Les utilisateurs similaires :

A partir de la matrice précédente, nous obtenons les utilisateurs similaires, si on prend un nouvel utilisateur:

	Action	Children	Comedy	Sci-Fi	Adventure
User 1	825	525	660	75	2925
User 2	400	30	0	0	1070
User 3	1170	700	390	30	3270

	Action	Children	Comedy	Sci-Fi	Adventure
New User	50	0	0	0	45

Figure IV.5.*Exemple de calculer les utilisateurs similaires*

La similarité se calcule comme suit :

sim (1,new) = abs (825-50) + abs (525-0) + abs (660-0) + abs (75-0) + abs (2925-45)

sim (2,new) = abs (400-50) + abs (30-0) + abs (0-0) + abs (0-0) + abs (1070-45)

sim (3,new) = abs (1170-50) + abs (700-0) + abs (390-0) + abs (30-0) + abs (3270-45)

IV.4.1 Résultat

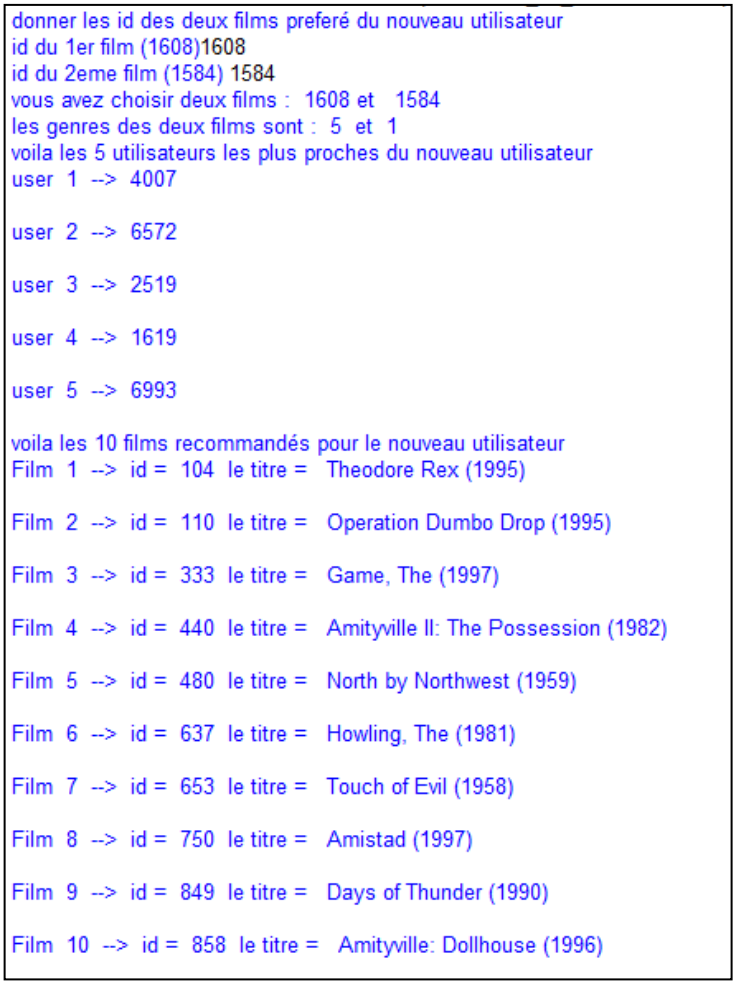


Figure IV.6.*Résultat des films à recommander du deuxième système*

IV.5 L'hybridation

Comme déjà présenté dans le chapitre 3, nous allons utiliser cette formule pour hybrider nos deux systèmes :

$$\alpha x + \beta y \qquad \text{II.1}$$

Le premier système nous a donné 20 films à recommander mais nous avons choisis les 10 meilleurs afin de les combiner avec les 10 films obtenus du deuxième système selon un score intégré dans le processus de classement.

II.5.1 Résultat

Nous jouons sur α et β qui se varient entre 0 et 1 et $\alpha+\beta=1$.

Si $\alpha=0.3$ alors $\beta=0.7$, nous avons comme les résultatsuivants :



Figure IV.7.*Résultat de l'hybridation 1*

Par contre si on change les valeurs de α et β nous aurons autre résultats, Si α=0.5 alors β=0.5, nous avons les résultats suivants :

```
donner la mésure Alpha qui represente le coefficient du 1er systeme ( entre 0 et
 1 )
0.5
coefficient du 1er systeme =  0.5
coefficient du 2eme systeme =  0.5
1     ---->   Willy Wonka and the Chocolate Factory (1971)
2     ---->   Theodore Rex (1995)
3     ---->   Operation Dumbo Drop (1995)
4     ---->   Game, The (1997)
5     ---->   Amityville II: The Possession (1982)
6     ---->   Star Trek VI: The Undiscovered Country (1991)
7     ---->   Cape Fear (1991)
8     ---->   Akira (1988)
9     ---->   Godfather, The (1972)
10     ---->   Miracle on 34th Street (1994)
11     ---->   Psycho (1960)
12     ---->   North by Northwest (1959)
13     ---->   Monty Python's Life of Brian (1979)
14     ---->   Phenomenon (1996)
15     ---->   Howling, The (1981)
16     ---->   Big Night (1996)
17     ---->   Touch of Evil (1958)
18     ---->   Amistad (1997)
19     ---->   Days of Thunder (1990)
20     ---->   Amityville: Dollhouse (1996)
```

Figure IV.8.*Résultat de l'hybridation2*

IV.6 Discussion des résultats

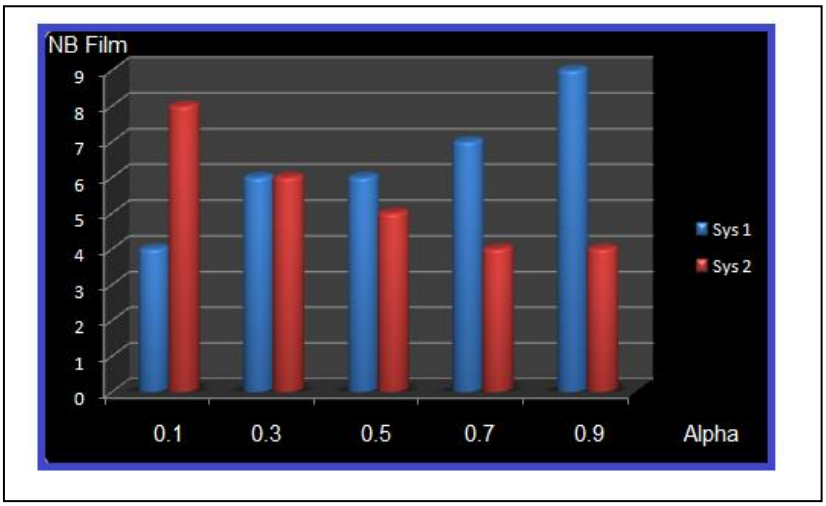


Figure IV.9.*Graphe de nombre de films recommandés par les deux systèmes en jouant sur alpha et beta*

En regardant le graphique, il nous montre que le nombre des films de système 1 est plus élevé que le système 2 sauf dans le cas ou α=0.1 c'est le système 2 qui est plus élevé, et dans le cas ou α=0.3 les deux systèmes sont équivalents.

index dans système	id movie	degré	système	
1	104	0.742	2	Theodore Rex (1995)
2	110	0.7124	2	Operation Dumbo Drop (1995)
9	125	0.474	1	Phenomenon (1996)
5	127	0.4974	1	Godfather, The (1972)
10	137	0.4465	1	Big Night (1996)
1	151	0.9036	1	Willy Wonka and the Chocolate Factory (1971)
8	154	0.4812	1	Monty Python's Life of Brian (1979)
7	185	0.4902	1	Psycho (1960)
4	206	0.5006	1	Akira (1988)
3	218	0.5149	1	Cape Fear (1991)
2	227	0.5543	1	Star Trek VI: The Undiscovered Country (1991)
3	333	0.6599	2	Game, The (1997)
6	378	0.4935	1	Miracle on 34th Street (1994)
4	440	0.623	2	Amityville II: The Possession (1982)
5	480	0.4842	2	North by Northwest (1959)
6	637	0.4615	2	Howling, The (1981)
7	653	0.4122	2	Touch of Evil (1958)
8	750	0.3741	2	Amistad (1997)
9	849	0.3385	2	Days of Thunder (1990)
10	858	0.3266	2	Amityville: Dollhouse (1996)

Figure IV.10. *Résultat final de notre système*

La figure représente un fichier qui contient les résultats obtenus de l'hybridation pondérée, nous avons comme résultats 20 films à recommander, dix (10) films appartient au premier système, et dix (10) autres au deuxième système.

Les résultats nous donnent le titre du film, le système qui appartient, son score, son Id, et le classement (index) dans son système. Par exemple : Nous avons le premier film titré 'Theodore Rex (1995)' il appartient au système deux (2), il a comme score 0.742, son ID 104 et classé le premier (1) dans le système deux (2).

IV.7 Conclusion

Dans ce chapitre nous avons parlé de l'implémentation et l'évaluation de notre système de recommandation que nous avons mis en place, nous avons obtenus un résultat hybridé de deux systèmes,

Nous avons présenté dans ce chapitre l'environnement de développement qui est le langage Pyton, et les bibliothèques que nous a aidé, et cité les différents outils. Ensuite, nous avons expliqué le fonctionnement des trois parties de notre travail (premier système,deuxième système et l'hybridation).

Finalement nous avons discuté sur les résultats obtenus dans ce travail en jouant sur α et β, en illustrant un graphique qui explique les résultats.

Conclusion générale

Conclusion générale

Les systèmes de recommandation ont reconnu une grande importance dans les différentes applications web pour aider les utilisateurs à sélectionner les items et produits qui correspondent à leurs besoins. Comme tout autre domaine, les systèmes de recommandation ont leurs problèmes qui réduisent leur efficacité, où l'un de ces problèmes est le problème de la stabilité de ces systèmes par rapport au profil de l'utilisateur, ce qui conduit à une stabilisation dans les contenus recommandés aux utilisateurs et par la suite à des recommandations similaires conduisant à ennuyer les utilisateurs. Du moment où, l'hybridation dans les listes de recommandation est devenue un critère important pour juger la qualité des recommandations et la faisabilité du système en termes de satisfaction des utilisateurs.

Dans notre projet nous avons obtenu nos objectifs présentés dans l'introduction générale qui sont l'adaptation et l'hybridation :

- On a pu développer un système de recommandation capable à s'adapter aux utilisateurs en fonction de leurs comportements.
- On a pu hybrider deux approches de recommandation afin d'avoir un résultat meilleur.

Dans ce mémoire, nous nous sommes intéressées au problème de la recommandation, Nous avons fixé dans un premier temps deux approches pour le système à mettre en place. Premièrement, une approche standard collaborative qui utilise les évaluations des utilisateurs. Deuxièment, une approche basée sur une ontologie dont nous avons défini une mesure de similarité sémantique qui propage la similarité entre les instances, ce qui permet de détecter des instances similaires non détectées par des mesures de similarité classiques.

L'évaluation du système a été effectuée dans le domaine des films et les résultats étaient satisfaisants. Nous avons appris dans ce travail beaucoup de points positives, nous avons développé nos capacités au domaine de la recherche d'information, nous avons rencontré par plusieurs travaux de recherche sous format d'article ou des thèses de doctorat dans le domaine. Et les résultats obtenus permettent aux chercheures de faire des tests et faire des comparaisons avec nos résultats, et aussi c'est une expérimentation qui aide les études aux futures

A la fin, nous pouvons dire que les différentes approches proposées dans ce mémoire ont achevé les buts visés pour lesquels ont été conçues, en recommandant des items divers et pertinents pour les utilisateurs qui correspondent à leurs évaluation et genre des films vus, et de rendre le système plus adaptatif à leurs besoins. Cependant, nos approches restent un peu restreintes ce qui ouvre la voie vers un nombre de perspectives afin de les perfectionner encore plus.

Perspectives

Les perspectives de travail dans le domaine de la recommandation sont vastes. Il serait, par exemple, intéressant d'ajouter d'autres modules de recommandation à notre approche et d'affiner les catégories des utilisateurs afin de mieux cibler leurs attentes.

Un module collaboratif item-to-item (I2I) serait une idée de module à ajouter. En effet, les approches I2I permettent un meilleur passage à l'échelle. Les expérimentations ayant été menées sur le jeu de données 100K de MovieLens, le problème du passage à l'échelle n'a pas été évalué avec l'approche actuelle bien que la plupart des calculs soient effectués hors ligne.

Références bibliographiques

Références

[1] Robin Burke.Hybrid recommender systems:Survey andexperiments. User Modeling and User -Adapted Interaction, 12(4):331–370,2002.

[2]BelkinN.J.,CroftW.B.(1992,December).Information filtering and information retrieval :twosidesofthesamecolin,*CommunicationoftheACM*,5(12),pp.29-38.

[3]CroftW.B.(1993).Knowledge-based and Statistical approaches to Text Retrieval. *IEEEEXPERT*, 8(2), pp. 8-12.

[4]Nicolas Béchet, « Etat de l'art sur les Systèmes de Recommandation, » Projet AxIS de l'INRIA. [en ligne]

[5] van Rijsbergen, C.andD, P. (1979). Information Retrieval.

[6]Malone,T.,Brobst,S.,Cohen,S.,Grant, K.,and Turbak,F.(1987). Intelligent information des systèmes de partage. In *Communications of the ACM*, volume30, pages 390–402.

[7]Goldberg, D., Nichols, D., Oki, B.M.andTerry, D.(1992a). Using collaborative filtering oweaveaninformation tapestry.Commun.ACM 35, 61–70.

[8]Resnick, P.andVarian, H.R.(1997).Recommender systems. Commun.ACM 40, 56–58.

[9]Maes,P. and Shardanand, U.(1995). Social information filtering: algorithms for automating "word of mouth". In *the SIGCHI conference on Human factors in computing systems*, Denver, Colorado, UnitedStates. ACMPress/Addison-WesleyPublishing Co.

[10]Conner, M.and Herlocker,J.(1999). Clustering items for collaborative filtering. In *the ACM SIGIRWorkshop on RecommenderSystems*.

[11]J.S.Breese,D.Heckerman, and C. Kadie. Empirical analysis of predictive algorithms for collaborative filtering. In Proceeding of the Fourteenth Conference on Uncer-tainty in Artificial Intelligence (UAI), pages43–52, Morgan Kaufmann, San Francisco, 1998.Madison, Wisconsin.

[12]KARAOUZENE Meryem, "Système de recommandation desservices web sémantiques,» mémoire, Université Abou Bakr Belkaid, Tlemcen, Algerie, pages, 2015. [En ligne]

[13]Majda MAATALLAH, "Une Technique Hybride pour les Systèmes de Recommandation, "THESE, Université Badji Mokhtar, Annaba, Algerie, pages, 2016. [en ligne]

[14]Shi, Y, Larson, M., Hanjalic, A.(2014).Collaborative filtering beyond the user-item matrix: A survey of the state of the art and future challenges. *ACMComputing Surveys(CSUR)*, 47(1), pp. 3.

[15]Cho Y. H, KimS.H.,KimJ. K. (2002). Apersonalized recommender system based on web usage miningand decisiontree induction, *ESA*, 23(3), pp. 329-342.

[16]Shapira B, ShovalP, HannaniU, (1997). Stereo types in Information Filtring Systems. *Information Proceedings Management*, 33(3), pp. 273-287.

[17] Rijsbergen C.J.V, (1979). Information Retrieval. *Second edition, Butter works.*

[18]WengL.T, XuY., LiY., NayakR., (2008). Exploiting Item Taxonomy for Solving Cold-Start Problem in Recommendation Making. In *the 20th IEEEInter. Conf.*, Vol.2, pp. 113-120.

[19]SchaferJ.B., KonstanJ. , and Reid lJ.,(1999).Recommender Systems in Ecommerce. *Proc. ofthe1 st ACM Conf. on ElectronicCommerce*, Denver, CO, pp. 158-166.

[20]Montaner M., LópezB., De La Rosa J. L., (2003). A Taxonomy of Recommender A gentson the Internet ,*Artificial Intelligence Review*, Vol. 19, pp.285-330.

[99]Mooney R. J. and RoyL. (1999). Content-based book recommendation using learning for text categorization. In *Proc; of the SIGIR Work shop on Recommender Systems: Algorithms and Evaluation.*

[21] Salton, G.and Buckley, C. (1988). Term-weighting approaches in automatic text retrieval. In INFORMATION PROCESSING AND MANAGEMENT pp.513–523.

[22]Rewari, A., Adler, M., Anick, P.G., Billmers, M.A., Carifio, M., Gunderson, A., Pundit, N. and Swart out, M.W. (1992). AIResearch and Applications in Digital's Service Organization. AI Magazine 13,63–75.

[23] Gediminas Adomavicius and Alexander Tuzhilin. Toward the next generation of recommender systems: Asurvey of the state-of-the-artand possible extensions. IEEE Trans. Knowl. Data Eng., 17(6):734–749,2005.

[24]Ken Goldberg, Theresa Roeder, Dhruv Gupta, and Chris Perkins. Eigentaste: A constant time collaborative filtering algorithm. Inf. Retr., 4(2): 133–151, July 2001.

[25]Robin Burke. Knowledge-based recommender systems. In Encyclopedia Of Library And Information Systems.Marcel Dekker, 2000.

[26]B. Sheth and P. Maes. Evolvingagents for personalized information filtering. In IEEE Conf. on AIf or applications. IEEE, 1993.

[27]Badrul Sarwar, George Karypis, Joseph Konstan, and John Riedl. Item-based collaborative filtering recommendation algorithms. In Proceedings of the 10th International Conference on World Wide Web, WWW '01, pages285–295, New York, NY, USA, 2001. ACM.

[28]Robin D. Burke. Hybrid web recommender systems. In The Adaptive Web, Lecture Notesin Computer Science, pages377–408. Springer Berlin Heidelberg, 2007.

[29]Bamshad Mobasher, XinJin, and Yanzan Zhou. Semantical lyenhanced collaborative filtering on the web. In Proceedings of the First European Web Mining Forum–EWMF 2003, pages 57–76.Springer, 2003.

[30]Daniel Billsusand Michael J. Pazzani. User modeling for adaptive news access. User Modeling and User-Adapted Interaction,10(2-3):147–180,2000.

[31]Barry Smyth and Paul Cotter. A personalized tvlistings service for the digital tvage. Knowl - Based Syst., 13(2-3):53–59,2000.

[32]Chumki Basu, Haym Hirsh, and William Cohen. Recommendation as classification: Using socialand content-based information in recommendation. In Proceedings of the Fifteenth National Conference on Artificial Intelligence, pages714–720.AAAIPress, 1998.

[33]Prem Mel ville, Raymond J. Mooney, and Ramadass Nagarajan. Content-boosted collaborative filtering for improved recommendations. In Eighteenth National Conference on Artificial Intelligence, pages187–192, 2002.

[34]MichaelJ. Pazzani. A frame work for collaborative, content-based and demographic filterin. Artif.Intell.Rev.,13 (5-6):393–408,1999.

[35]Cai-Nicolas Zieglerand Georg Lausen. Propagation models for trustand distrust insocial networks. Information Systems Frontiers, 7 (4-5):337–358, 2005.

[36]GuyShani. Tutorial on evaluating recommender systems. In Proceedings of the fourth ACM conference on Recommender systems, RecSys'10,pages1–1.ACM,2010.

[37]Michael D. Ekstrand, Daniel Kluver, F. Max well Harper, andJosephA. Konstan. Let tingusers choose recommender algorithms: An experimental study. In Proceedings of the 9th ACM Conference on Recommender Systems, RecSys '15, pages11–18, NewYork, NY, USA, 2015. ACM.

[38]RozaLémdani, "Système Hybride d'Adaptation dans les Systèmes de Recommandation," Thèse de Doctorat, l'Université Paris-Saclay, Paris, France, pages, 2016. [en ligne]

[39]MiddletonS. E.,AlaniH.,ShadboltN.R.,DeRoureC.,(2002).Exploiting Synergy Between Ontologies and Recommender Systems. *Proc.ofthe11 thInter.WWW Conf., Workshop on theSemanticWeb*.

[40]SchwabI.,KobsaA.,KoychevI.,(2001).LearningUserIntereststhrough Positive Examples Using Content Analysis and Collaborative Filtering.*Internal Memo, GMD, Germany.*

[41]GoodN.,Schafer J. B.,KonstanJ. A.,BorchersA.,SarwarB.,HerlockerJ.,RiedlJ., (1999). Combiningcollaborativefilteringwith personal agentsfor better recommendations, *In Proc. of the6thNat. Conf. on AI, Orlando, USA*, pp. 439-446.

[42] Nguyen A. and DenosN.and Berrut C., (2006). Modèle d'espaces de communautés basé sur la théorie des ensembles d'approximation dans un système de filtrage hybride,*Conf.EnRechercheInformationetApplications(CORIA*),Lyon,France,pp.303-314.

[43]AllanJ.(1996)IncrementalRelevanceFeedbackforInformationRetrieval.*Proc.ofSIGIR'96 Zurich, Switzerland, ACMPress*, pp. 270-278.

[44]Lam W., MukhopadhyayS., MostafaJ. and PalakalM. (1996). Detection of shifts inuserinterests for personalized in formation filtering.*Proc.OfSIGIR'96, Switzerland*, pp. 317-325.

[45]Schein A. I., Popescul A., UngarL. H. (2001). Generative Models for Cold-Start Recommendations. *In Proc. of the SIGIR Workshop on Recommender Systems*, USA.

[46]Koychev, I., Schwab,I. (2000,May).Adaptation to drifting user's interests. *In Proceedings of ECML2000 Work shop: Machine Learningin New Information Age*(pp. 39-46).

[47]Webb, G. Pazzani, M. and Billsus, D. (2001). Machine Learning for usermodeling". *UserModeling and User-AdaptiveInteraction*, Vol. 11, pp. 19-29.

[48]Mitchell,T., Caruana, R., Freitag, D., Mc Dermott, J. and Zabowski, D. (1994). Experience with a Learning Personal Assistant. *Communications of the ACM*, 37(7), pp. 81-91.

[49]Grab tree, I. and Soltysiak, S. (1998).Identifying and Tracking Changing In terests. *International Journal of Digital Libraries*, Vol. 2, pp. 38-53.

[50]Billsus, D., and Pazzani, M. J.(1999).A Hybrid User Model for News Classification, *In Kay J. ,User Modeling Proc. of the 7thInter.Conf.,Springer-Verlag*,pp.99-108.

[51]YildirimH., S. Krishnamoorthy M. S.,(2008).A random walk method for alleviating the sparsity problem in collaborative filtering, in*Proc .OfRecSys'2008, pp.*131-138.

[52]Chen G., Wang F., Zhang C., (2009). Collaborative Filtering Using Orthogonal Nonnegative Matrix Tri-factorization. *JIPM*, Vol. 45.

[53]Yehuda Koren. 1 the bellkor solution to the Netflix grand prize, 2009.

[54]KorenY., (2008). Factorization Meets the Neighbour hood: a Multifaceted Collaborative Filtering Model.In*Proc.Ofthe14th ACMSIGKDD Inter. Conf. on KDDM*,pp. 424-436.

[55]SarwarB.M., KarypisG., KonstanA., and RiedlJ., (2001). Item-based collaborative filtering recommendation algorithms. *In Proc. Of the 10 th Inter. Conf. onWWW*, pp. 285–295.

[56]R.Baeza-Yates et B.Ribeiro-Neto. Modern Information Retrieval. ACM Press; Addison-Wesley:New York; Harlow, England; Reading, Mass., 1999.

[57] G.Salton et M. J.McGill, Introduction to modern information retrieval. McGraw-Hill.New York, 1983.

[58] Ait Ahmed Nora, Idris Khodja Asma, “Système de Recommandation de Cours à Base d’Ontologie,” mémoire, UNIVERSITE Dr. TAHAR MOULAY, SAIDA, Algerie, pages, 2018. [en ligne]

[59] T. R. Gruber, « Toward principles for the design of ontologies used for knowledge sharing? », International journal of human-computer studies, vol. 43, no 5, p. 907–928, 1995.

[60] Samuel, Szoniecky (2017). Vers des écosystèmes de connaissance: Intelligence collective et archives numériques.GREAT BRITAIN ISTE: LTD,237 pages.

[61] http ://dublincore.org

[62] http://ksl.stanford.edu/software/ontolingua/

[63]http://odile.papini.perso.luminy.univ-amu.fr/sources/WEBSEM/cours-intro-websem-2.pdf

[64]Zhibiao Wu and Martha Palmer. Verbs semantics and lexical selection. In Proceedings of the 32nd annual meeting on Association for Computational Linguistics,ACL '94, pages 133{138, 1994.

[65] Philip Resnik. Using information content to evaluate semantic similarity in a taxonomy. In IJCAI, pages 448{453, 1995.

[66]Prasanna Ganesan, Hector Garcia- Molina, and Jennifer Widom. Exploitinghierarchical domain structur et ocompute similarity. ACMTrans. Inf. Syst., 21(1):64–93,2003.

[67]William E.Winkler. The state of record link a geand current research problems. Technical report, Statistical Research Division, U.S. Bureau of the Census, 1999.

[68]P. Jaccard. Distribution de la florealpine dans lebassin des dranses et dans quelques régions voisines. Bulletin de la Société Vaudoise des Sciences Naturelles, 37:241–272,1901.

[69]L.R.Dice. Measures of the amount of ecologic association betweenspecies. Ecology, 26(3):297–302,July 1945.

[70]T. Slimani, B. BenYaghlane etK. Mellouli. Une extensionde mesure de similarité entre les concepts d’une ontologie.4thInternational Conference: Sciences of Electronic, Technologies of Informationand Telecommunications. Mars 2007.

[71] D. Lin. An Information-Theoretic Definition of similarity. In Proceedings of the Fifteenth International Conference on Machine Learning (ICML'98). Morgan- Kaufmann: Madison, WI, 1998.

[72] R. Rada, H. Mili, E. Bichnellet M. Blettner, Development and application of a metric on semantic nets. IEEE Transaction on Systems, Man, and Cybernetics: pp 17-30. 1989.

[73] J.H.Lee, M.H.KimetY.J.Lee. Information Retrieval Based on Conceptual Distance in IS-A Hierarchy. Journal of Documentation 49, pp. 188-207, 1993.

[74] M.Ehrig, P.Haase, M.HefkeetN.Stojanovic. Similarity for ontologya comprehensive framework. In Workshop Enterprise Modelling and Ontology: Ingredients for Interoperability, 2004.

[75] D. Lin.AnInformation-Theoretic Definition of similarity. In Proceedings of the Fifteenth International Conferenceon Machine Learning.Pages 296 -304.1998.

[76] G. A. Miller, R. Beckwith, C. Fellbaum, D. Gross, et K. Miller. Introduction to WordNet: An On-line Lexical Database. Cognitive Science Laboratory, Princeton University, Princeton, Technical Report 1993.

[77] G.Hirst et D.StOnge. Lexical chains as representations of context for the detection and correction of malapropisms. In Christiane Fellbaum (editor), WordNet: An electronic lexical database, Cambridge, MA:The MIT Press .1998.

[78]ThabetSlimani, Boutheina Ben Yaghlane, KhaledMellouli, "Une extension de mesure de similarité entre les concepts d'une ontologie," 4rth International Conference: Sciences of Electronic, Technologies of Information and Telecommunications, Tunisia, pages, 2007. [En ligne]

[79]J. Jiang et D. Conrath. Semantic similarity based on corpus statistics and lexical taxonomy. In Proceedings of the 10thInternational Conferenceon ResearchinComputational Linguistics, Taiwan.1998.

[80]C. Leacocket M. Chodorow. Combining Local Context and WordNet Similarity for Word Sense Identification. InWordNet:An ElectronicLexical Database,C.Fellbaum,MITPress,1998.

[81] H. Kozima et T. Furugori. Similarity between words computed by spreading activation on an English Dictionary.Proceedings of EACL-93 (Utrecht), pp. 232- 239, 1993.

[82] H. Kozimaet A. Ito, Context-Sensitive Word Distance by Adaptive Scaling of a Semantic Space. RuslanMitkov and Nicolas Nicolov (editors.), Recent Advances in Natural Language Processing (a serie of "Contemporary Issues in Linguistic Theory" 136), John Benjamins, Amsterdam/Philadelphia, pp. 111- 124, 1997.

[83] H. N. Diep, Similarité de mots et extraction automatique de synonymes. University of Louvain.Internship Report, Belgium 2002.

[84] M. Okumura et T. Honda, Word Sense Disambiguation and Text Segmentation Based on Lexical Cohesion. In Proceedings of Fifteenth International Conference on Computational Linguistics (COLINGS-94), vol.2, Kyoto, Japan, pp.755-761, 1994.

ANNEXE

1 ***Les langages***

1.1*Python*

Le choix du langage Python[1]utilisé dans le projet n'est pas une coïncidence car le Python est connu mondialement et notre choix est bien expliqué dans les prochaines lignes.

- Définition de Python

Python est un puissant langage de programmation objet, il a été créé en 1991 comme une première version[2] et il a connu un développement énorme, il détermine implicitement les types.

Python est un langage de script de haut niveau, structuré et open source. Il est multi-paradigme et multi-usage. Développé à l'origine par Guido van Rossum en 1989, il est, comme la plupart des applications et outils open source, maintenu par une équipe de développeurs un peu partout dans le monde.

Conçu pour être orienté objet, il n'en dispose pas moins d'outils permettant de se livrer à la programmation fonctionnelle ou impérative ; c'est d'ailleurs une des raisons qui lui vaut son appellation de « langage agile ».

Parmi les autres raisons, citons la rapidité de développement (qualité propre aux langages interprétés), la grande quantité de modules fournis dans la distribution de base ainsi que le nombre d'interfaces disponibles avec des bibliothèques écrites en C, C++ou Fortran. Il est également apprécié pour la clarté de sa syntaxe, ce qui l'oppose au langage Perl.

Nous avons opté pour le langage de programmation Python en raison de ces caractéristiques, Python est sous licence open source, il nous offre une utilisation gratuite et libre et il est indépendant de toute plateforme [84].

- Bibliothèques de Python utilisés

Nous avons ajouté des bibliothèques de Python qui nous aide à lire et écrire dans les fichiers Excel, ajouter des fonctions mathématiques, ou même de faire des graphes.

[1]https://www.python.org/.
[2]https://www.python.org/downloads/release/python-364/:Lien d'installation du python

Le tableau suivant cite les déférentes bibliothèques de Python utilisé dans notre projet:

La bibliothèque	Son but
Xlrd	lire un fichier Excel
Xlwt	écrire dans un fichier Excel
matplotlib.pyplot	pour faire les déférents diagrammes qui explique les résultats de notre travail
Math	pour utiliser des fonctions de calcule prédéfini

2 *Les outils*

2.1 *Pandas*

Pandas[3] est une bibliothèque open source sous licence BSD fournissant des structures de données hautes performances et faciles à utiliser, ainsi que des outils d'analyse des données pour le langage de programmation Python.

Pandas est un projet sponsorisé par NumFOCUS. Cela contribuera au succès du développement de pandas en tant que projet open source de classe mondiale et permettra de faire un don au projet.

2.2*Numpy*

NumPy[4] est le paquet fondamental du calcul scientifique avec Python. Il contient entre autres:

- un puissant objet tableau N-dimensionnel
- fonctions sophistiquées (diffusion)
- outils d'intégration de code C / C ++ et Fortran
- Algèbre linéaire utile, transformée de Fourier et capacités de nombres aléatoires

Outre ses utilisations scientifiques évidentes, NumPy peut également être utilisé comme un conteneur multidimensionnel efficace de données génériques. Des types de données arbitraires peuvent être définis. Cela permet à NumPy de s'intégrer de manière transparente et rapide à une grande variété de bases de données.
NumPy est sous licence BSD, ce qui permet sa réutilisation avec peu de restrictions.

[3] https://pandas.pydata.org/
[4] https://www.numpy.org/

2.3*NLTK*

NLTK[5] est une plate-forme de premier plan pour la création de programmes Python utilisant des données en langage humain. Il fournit des interfaces faciles à utiliser avec plus de 50 corpus et ressources lexicales tels que WordNet, ainsi qu'une suite de bibliothèques de traitement de texte pour la classification, la création de jetons, le suivi, le balisage, l'analyse et le raisonnement sémantique. et un forum de discussion actif.

Grâce à un guide pratique qui présente les bases de la programmation, des thèmes informatiques, et une documentation complète sur les API, NLTK convient aux linguistes, ingénieurs, étudiants, enseignants, chercheurs et utilisateurs du secteur. NLTK est disponible pour Windows, Mac OS X et Linux. Mieux encore, NLTK est un projet gratuit, à code source ouvert et piloté par la communauté.

NLTK a été qualifié de «formidable outil pour enseigner et travailler dans les langages informatiques en utilisant Python», et comme «une bibliothèque incroyable pour jouer avec le langage naturel».

Le traitement automatique du langage avec Python constitue une introduction pratique à la programmation pour le traitement du langage. Écrit par les créateurs de NLTK, il guide le lecteur à travers les bases de l'écriture de programmes Python, du travail avec les corpus, de la catégorisation de texte, de l'analyse de la structure linguistique.

3 *Ontologie*

3.1 *WordNet*

Wordnet est une ontologie qui organise plus de 100,000 concepts généraux, qui sont sémantiquement structurés. Elle regroupe les concepts en ensembles de synonymes appelés synsets. Wordnet fournit des définitions générales et courtes, et enregistre les différentes relations sémantiques entre ces ensembles de synonymes. Elle est particulièrement bien adaptée pour de mesures de similarité, car elle organise des noms et des verbes en hiérarchie de relation « est-un ».

[5]https://www.nltk.org/