République **A**lgérienne **D**émocratique et **P**opulaire
Ministère de l'**E**nseignement **S**upérieur et de la **R**echerche **S**cientifique

UNIVERSITE **M**OULOUD **M**AMMERI DE **T**IZI-**O**UZOU

FACULTE DU **G**ENIE **E**LECTRIQUE ET D'INFORMATIQUE
DEPARTEMENT D'INFORMATIQUE

Mémoire de Fin d'Etudes
MASTER PROFESSIONNEL

Domaine : **Sciences et technologie**
Filière : **Informatique**

Option : **ISI**

Thème

Un système de recommandation personnalisé en recherche d'informations

Mémoire soutenu publiquement le 04 /11/2020 Devant le jury composé de :

Président : Mme S.FELLAG

Examinateur : Mme M.BENTAYEB

Encadreur : Mme ACHEMOUKH F.

Présenté par :

- ***Mr BENKHOUYA Badredine***
- ***Mr AIT ABDELMALEK Rafik***

Promotion : 2019/2020

Nous tenons à remercier en premier lieu les personnes sans qui ce travail n'aurait jamais vu le jour.

Tout d'abord nous remercions notre promotrice ACHEMOUKH Farida, qui nous a offert l'opportunité de réaliser ce mémoire et pour sa patience, sa disponibilité et surtout ses judicieux conseils qui ont contribué à alimenter nos réflexions.

Nous tenons à remercier également les membres de jury d'avoir accepté d'évaluer notre travail.

Un énorme remerciement pour nos parents, frères et sœurs ainsi qu'à nos amis.

Du profond de mon cœur je dédie ce modeste travail à tous ceux qui me sont chers,

A mes grand parents que Dieu très haut leurs accordes Santé et longue vie.

A mes chers parents,

A ma sœur,

A mon binôme Rafik,

Ainsi qu'à mes proches et mes amis(es).

Badredine.

Du profond de mon cœur je dédie ce modeste travail à tous ceux qui me sont chers,

A mes chers parents que Dieu très haut leurs accordes Santé et longue vie.

A mon frère,

A mon binôme Badredine,

Ainsi qu'à mes proches et mes amis(es).

Rafik.

Sommaire

Sommaire

CHAPITRE II : MODELISATION DU PROFIL EN SYSTEME DE RECOMMANDATION.

CHAPITRE III : SPECIFICATION D'UN MODELE PERSONALISÉ POUR LA RECOMMANDATION.

CHAPITRE IV : EVALUATION ET VALIDATION EXPERIMENTALE DE L'APPROCHE PROPOSE.

Liste des figures

LISTE DES FIGURES

Introduction Générale

Introduction générale :

De nos jours, nous sommes privilégiés dans un monde riche en information, dans lequel la plupart, si ce n'est la totalité, de l'information dont nous avons besoin est au bout de nos doigts et est prête à être exploitée. L'essor du Web, conforté par le rapide développement des nouvelles technologies de l'information et de la communication a conduit à la production d'un volume d'information sans précédent, et la quantité d'information manipulée dans le monde est élevée et sa gestion sans l'ordinateur n'est plus imaginable. Ce flux informationnel est ingérable avec les moyens classiques, ainsi plusieurs domaines permettant de trouvé des informations pertinente et qui rependent aux besoins des utilisateurs ont apparus.

Un des domaines de recherche principaux, relatifs à la problématique de la surcharge d'information est le domaine de la recherche d'information. Le principe général est d'élaborer des méthodes et des algorithmes afin de rechercher des ressources en fonction de requêtes formulées par des utilisateurs. Il n'est cependant pas toujours évident pour un utilisateur de savoir comment exprimer sa demande. De plus, sa requête correspond généralement à une quantité importante de ressources et il est difficile de savoir quels résultats lui présenter en premier, d'autant plus que d'un utilisateur à un autre, l'ordre de priorité peut changer.

Un autre domaine de recherche relatif à cette problématique est le domaine des systèmes de recommandation. Ces systèmes sont capables de fournir des recommandations adaptées aux préférences et aux besoins des utilisateurs. Ils se sont avérés être très satisfaisants pour aider les utilisateurs à accéder aux ressources désirées dans un temps limité. Initialement conçus pour la recommandation de ressources web, films, etc. les systèmes de recommandation sont devenus de plus en plus populaires et sont aujourd'hui un composant principal de beaucoup d'applications dans différents domaines. Un avantage très conséquent des systèmes de recommandation est que l'utilisateur n'a pas besoin de formuler de requêtes. Sa seule requête est implicite, elle peut se traduire par : "Quelles sont les ressources qui correspondent à mes préférences, mes besoins et mes contraintes ? ".

Mais le problème principal auquel sont confrontés les systèmes de recommandations est le problème de démarrage à froid, et cela parce que le système ne possède aucune information caractérisant l'utilisateur du système. Notre travail se situe dans ce contexte, notamment dans le cadre des systèmes de recommandation des documents. Nous adoptons une approche basée

sur le contenu lors du démarrage à froid, et sur le filtrage collaboratif dans d'autre cas. Ce type de recommandation repose en générale sur la contribution de l'utilisateur dans le système c'est-à-dire les notes et préférences attribuées par cet utilisateur aux différents documents qu'il a consulté, l'ensemble de ces informations sont appelés profils utilisateurs.

Une des difficultés majeures est la construction de ce profil, dont la pertinence vis-à-vis des besoins/intérêts de l'utilisateur, joue un rôle important dans la qualité des recommandations produites. De ce fait, le profil utilisateur devient central dans les systèmes de recommandation, cette problématique fera objet de notre mémoire.

Notre travail est réparti sur quatre chapitres :

Le premier chapitre présente une vue générale sur les systèmes de recommandation, nous définissons d'abord ce qu'est un système de recommandation. Ensuite nous détaillons ses différents types, ainsi les avantages et inconvénients de chaque approche.

Dans le deuxième chapitre nous verrons les modèles de représentation de profils des ressources ensuite les modèles de représentation de profils utilisateurs, les méthodes d'acquisition des informations des utilisateurs, les techniques de constructions et de mise à jour des profils.

Dans le chapitre 3, nous expliquons l'approche que nous avons proposée.

Et le dernier chapitre se base sur les détails d'implémentation et de mise en œuvre de notre approche, ainsi qu'à la présentation des résultats obtenus.

Nous terminons notre mémoire par une conclusion générale.

CHAPITRE I

Etat de l'art sur les systèmes de recommandation

1- INTRODUCTION :

Le développement du Web a créé un besoin de nouvelles techniques pour satisfaire les besoins des utilisateurs mais aussi pour faire savoir qu'une information existe. Ces techniques sont appelées les Systèmes de Recommandation.

Les préludes des systèmes de recommandation découlent de recherches menées dans la construction de modèles représentant les choix d'utilisateurs. Ces recherches sont issues de domaines distincts tels que la recherche documentaire, les sciences de gestion et marketing, les sciences cognitives et les théories d'approximation (Adomavicius et *al*., 2005).

La recommandation peut être comparée à un dialogue entre une personne experte d'un domaine et l'autre désireuse d'acquérir des informations dans ce domaine. Plus concrètement, un bibliothécaire va pouvoir, en fonction des gouts d'un de ses clients, proposer une liste d'ouvrages à ce dernier qui ne sera autre qu'une recommandation au sens des systèmes de recommandation.

En considérant cette analogie, le bibliothécaire, compte tenu de sa connaissance des différents ouvrages qu'il propose peut être vue comme la base de connaissances des items à recommander. Il connait ainsi les items de manière individuelle et est capable d'effectué des associations d'items suivant différents critères caractérisés par le profil d'un utilisateur. Ce dernier est en fait conscient de ses gouts et peut les soumettre au bibliothécaire. Nous pouvons même aller plus loin en supposant que le bibliothécaire connaisse les gouts des différents clients. Il serait alors en mesure de proposer des ouvrages à un client, qui ont été aimés par d'autres clients similaires. La notion de recommandation induite par cet exemple est sans doute à la base des principes de systèmes de recommandation.

Dans ce premier chapitre nous donnons une définition des systèmes de recommandation. Ensuite, nous présentons les techniques et approches de recommandation, ainsi que les avantages et les inconvénients de chacune d'entre elles.

2- DEFINITION DES SYSTEMES DE RECOMMANDATION

Un système de recommandation est une forme spécifique de filtrage de l'information qui a pour but de présenter à un utilisateur des éléments qui sont susceptibles de l'intéresser, et ce, en se basant sur ses préférences et son comportement.

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 prêtes attention.

3- CONCEPTS DE BASE, NOTATION ET NOTIONS LIEES

Nous définissons dans cette partie quelques concepts relatifs aux systèmes de recommandation, qui seront utilisés par la suite.

3.1- L'USAGER ET L'ITEM

Les deux entités de base qui apparaissent dans tous les systèmes de recommandations sont l'usager et l'item. L'« usager » est la personne qui utilise un système de recommandation, donne son opinion sur diverses items et reçoit les nouvelle recommandations du système. L'« Item » est le terme général utilisé pour désigner ce que le système recommande aux usagers.

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 ou par un triplé. 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. Une note peut être attribuée directement par un utilisateur à un item en donnant une valeur numérique ou binaire à travers l'interface du système appelée évaluation explicite (Burk R., 2002). En outre, les préférences de l'utilisateur peuvent être déduites par le système en utilisant des algorithmes et techniques spécifiques (Rendle et al, 2009) (Lee et al, 2008), et dans ce cas appelée évaluation implicite (Ouard et *al*, 1998) (Burk R., 2002) (Kelly et *al*, 2003).

3.3- FILTRAGE D'INFORMATION

Le filtrage d'information est l'expression utilisée pour décrire une variété de processus dédiés à la fourniture de l'information adéquate aux personnes qui en ont besoin (Bel et al, 2007). Son but est de sélectionner et suggérer aux utilisateurs, à partir de larges volumes d'informations générés dynamiquement, les informations jugées pertinentes pour eux. Par conséquent, le filtrage d'information peut être vu aussi comme étant le processus d'élimination de données indésirables sur un flux entrant, plutôt que la recherche de données spécifiques sur ce flux. Le filtrage commence donc après la définition du besoin de l'utilisateur, il permet d'éliminer les documents qui peuvent ne pas intéresser l'utilisateur. Le filtrage offre à l'utilisateur un gain d'effort et de temps.

3.4- MATRICE D'EVALUATION UTILISATEUR-ITEM

L'ensemble de tous les triplets du système 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 filtrage collaboratif, qui utilisent les préférences (votes) pour la génération des recommandations.

3.5- LA PREDICTION

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.

3.6- LA PERSONNALISATION VS LA RECOMMANDATION

La personnalisation est une notion proche de la notion de recommandation mais elle est moins générale et elle consiste à adapter un item aux goûts, aux besoins et parfois même au comportement de l'utilisateur. Tandis qu'une recommandation génère une liste d'items plus ou moins adaptée aux besoins de l'utilisateur (c.à.d. elle ne garantit pas une personnalisation totale parce que les listes recommandées peuvent contenir des items nouveaux pour l'utilisateur ou différents de ces préférences, pour améliorer la satisfaction).

4- LES TECHNIQUES DE RECOMMANDATION

Plusieurs facteurs entrent en considération afin de catégoriser les systèmes de recommandation.

1. La connaissance de l'utilisateur (c.-à-d. son profil en fonction de ses goûts).

2. Le positionnement d'un utilisateur par rapport aux autres (la notion de classes ou réseaux d'utilisateurs).

3. La connaissance des items à recommander.

4. La connaissance des différentes classes d'items à recommander.

De ces facteurs sont produits divers types de recommandations dont les plus utilisés dans la littérature sont le filtrage basé sur le contenu et le filtrage collaboratif. Ce document présente dans un premier temps ces deux approches ainsi que leur hybridation.

4.1- RECOMMANDATION BASEE SUR LE CONTENU

La recommandation basée sur le contenu consiste à analyser le contenu des ressources ou des descriptions de ces ressources afin de déterminer quelles ressources sont susceptibles d'être utiles ou intéressantes pour un utilisateur donné. Ce sous-domaine est fortement similaire au domaine de la recherche d'information. En effet, les mêmes techniques sont utilisées, la différence se trouvant essentiellement dans l'absence de requêtes explicites formulées par l'utilisateur. Par conséquent, beaucoup de concepts généraux de la recommandation basée sur le contenu proviennent de la recherche d'information.

La plupart des systèmes de recommandation basée sur le contenu identifient les ressources similaires aux ressources qu'un utilisateur donné à appréciées (Zhang et al., 2002 ; Adomavicius et Tuzhilin, 2005 ; Pazzani et Billsus, 2007). Ainsi, quand de nouvelles ressources sont introduites dans le système, elles peuvent être recommandées directement sans que cela ne nécessite un temps d'intégration.

Habituellement, la recommandation basée sur le contenu est séparée des autres formes de recommandation que nous présentons dans cette section : recommandation à partir de cas , recommandation basée sur la démographie, sur l'utilité et sur la connaissance. Dans cette section, nous nous intéressons donc dans un premier temps à la vision globale de la recommandation basée sur le contenu et aux approches habituellement présentées comme telles. Nous nous focalisons ensuite sur ce que nous considérons comme des formes particulières de recommandation basées sur le contenu.

4.1.1- Approche générale

Pour recommander des ressources en se basant sur le contenu, deux éléments doivent être constitués : les profils de ressource et les profils d'utilisateur. La notion de contenu ne se rapporte donc pas uniquement au contenu des ressources, mais également aux attributs descriptifs des utilisateurs.

a- Profils de ressource

Les profils de ressource consistent en un ensemble d'attributs décrivant les ressources, de façon analogue à l'index utilisé dans le domaine de la recherche d'information. Comme dans le domaine de la recherche d'information, la précision de cette approche est donc hautement dépendante de la nature des ressources : elle est beaucoup plus élevée pour des ressources de recommandation basée sur le contenu textuel que pour des ressources telles que les images, les vidéos ou les ressources audio, dont il est difficile d'extraire des attributs. En général, quand cette approche est employée pour des ressources non textuelles, des méta-données sont utilisées. Par conséquent, la plupart des recherches sur la recommandation basée sur le contenu porte sur des données textuelles (Adomavicius et Tuzhilin, 2005 ; Pazzani et Billsus, 2007).

Une étape importante de cette approche est la transformation des données textuelles sans restriction, c'est-à-dire écrites en langage naturel, en une représentation structurée. Une des techniques les plus répandues pour répondre à cette problématique est le stemming (Porter,1997). Le stemming consiste à effectuer une transformation systématique des mots relatifs à un même concept en un même terme qui les représente tous. Ensuite un poids est attribué à chacun de ces termes en fonction de leur importance dans la ressource textuelle. Une façon classique de calculer ce poids est l'utilisation de la formule term-frequency inverse document-frequency ou tf ·idf (Salton et Buckley, 1987). Une limitation de cette technique est qu'elle ne prend pas en compte le contexte des termes. Ainsi, l'application de cette technique à des textes contenant par exemple des tournures négatives ou ironiques peut aboutir à de mauvaises représentations.

Une fois cette étape finalisée, le système possède soit les listes des mots les plus importants ou les plus informatifs de chaque ressource, soit un ensemble de vecteurs de termes, c'est-à-dire un ensemble de poids associé à chaque terme de chaque ressource.

b- Profils d'utilisateur

Le profil d'un utilisateur consiste en un ensemble d'informations qui peuvent être entrées manuellement par l'utilisateur, ou extraites automatiquement à partir du contenu des ressources qu'il a consulté.

La première possibilité est donc de demander à l'utilisateur de fournir directement ses centres d'intérêts, à l'aide de formulaires, en lui demandant d'entrer une liste de termes. Si un nombre restreint d'informations est demandé, cette approche rendra le système opérationnel rapidement, mais ne pourra pas fournir de recommandations précises. À l'inverse, en demandant un grand nombre d'informations, les recommandations seront plus précises mais le système sera trop contraignant pour l'utilisateur. De plus, les centres d'intérêts des utilisateurs peuvent évoluer au cours du temps, et une telle approche impose une actualisation manuelle régulière, ce qui est également contraignant. Un dernier inconvénient, est que l'utilisateur peut ne pas remplir le formulaire honnêtement, auquel cas, les recommandations qui lui seront fournies ne pourront pas être pertinentes.

La seconde possibilité, l'extraction automatique à partir du contenu des ressources consultées par l'utilisateur, est donc souvent préférable. Une des méthodes les plus simples est de représenter les centres d'intérêt des utilisateurs par des vecteurs de termes représentant les ressources que l'utilisateur a appréciées. Les appréciations peuvent être obtenues de façon explicite en demandant directement aux utilisateurs de les fournir, ou implicitement en utilisant des algorithmes basés sur les usages pour calculer les recommandations, il suffit alors de calculer la similarité entre les profils de ressource et les profils d'utilisateur. Cela peut être effectué selon diverses méthodes, comme la mesure de similarité cosinus. C'est dans ce cadre que cette approche est la plus similaire aux approches de la recherche d'information.

Beaucoup d'autres méthodes d'extraction automatique de profils qui se démarquent davantage de la recherche d'information ont été proposées. Dans ce cadre, les recommandations sont calculées selon la probabilité qu'un utilisateur donné appréciera une ressource. Cela peut être considéré comme un problème de classification où chaque classe représente un niveau d'appréciation (e.g. « aime » et « n'aime pas »). Trois des algorithmes de classification célèbres souvent utilisés dans ce contexte sont présentés dans cette section : les arbres de décision, le classificateur naïf de Bayes et les réseaux de neurones.

1) Arbres de décision

Un arbre de décision est obtenu en séparant de façon récursive les ressources en sous-groupes homogènes relativement à des variables déterminées au préalable. Dans le cas de la recommandation de ressources textuelles, ces variables sont en principe des variables booléennes sur la présence ou l'absence de termes. Ensuite, pour chaque sous-groupe, la probabilité que l'utilisateur appréciera une ressource de ce sous-groupe est conservée.

Le problème principal de l'application de cette approche à la recommandation basée sur le contenu est que la précision obtenue est dépendante du nombre de variables manipulées. Cette approche est simple et performante dans le cadre de recommandations portant sur des ressources ayant un nombre d'attributs limité, mais n'est pas appropriée dès que ces attributs sont en nombre élevé, ce qui est le cas des ressources textuelles sans restriction.

2) Classificateur naïf de Bayes

Le principe du classificateur naïf de Bayes est de déterminer la classe **C** pour laquelle la probabilité ***P(C|θ1,...,θk)*** qu'une ressource r ayant pour attributs ***(θ1,...,θk)*** appartienne à cette classe ***C*** soit maximale. Les attributs sont supposés indépendants, et maximiser ***P(C|θ1,...,θk)*** revient à maximiser la formule suivante :

$$p(C)\prod_{i=1}^{k} p(\theta_i|C) \qquad (1.1)$$

Les valeurs de ***P(C)*** et de ***P(θi|C)*** sont estimées à partir d'un corpus d'apprentissage. Pour chaque ressourcer, chaque valeur de la formule (1.1) est estimée pour chaque classe (ici chaque niveau d'appréciation). r est alors placée dans la classe pour laquelle cette valeur est la plus élevée.

En dépit du fait que les attributs des ressources sont en réalité interdépendants, le classificateur naïf de Bayes s'avère fournir une grande précision et représente un algorithme simple et ayant un temps de calcul réduit. De plus, contrairement aux arbres de décision, il est applicable aussi bien sur des données ayant un nombre d'attributs limité que sur des données sans restriction.

3) Réseaux de neurones

Dans un réseau de neurones, un neurone est simplement une fonction non linéaire, de variables réelles et bornée. Cette fonction est généralement définie comme suit :

$$f(\mathrm{x1},\ldots,\mathrm{xk};\mathrm{w1},\ldots,\mathrm{wk}) = \left\lfloor \sum_{i=1}^{k} w \right\rfloor \qquad (1.2)$$

Où les variables ***w1,...,wk*** correspondent à des poids à associer aux variables ***x1,...,xk***, qui sont déterminés à partir d'un corpus d'apprentissage. La fonction tangente hyperbolique est une fonction sigmoïde qui a certaines propriétés particulièrement appropriées pour l'apprentissage de réseaux de neurones (Kalman et Kwasny, 1992). De tels neurones sont associés en réseau selon deux types d'architecture : les réseaux bouclés qui correspondent à des graphes orientés avec circuit et les réseaux non-bouclés qui correspondent à des graphes orientés sans circuit.

Dans le cadre de la recommandation basée sur le contenu, les variables ***x1,...,xk*** correspondent à la fréquence des termes utilisés pour caractériser les ressources (qui peut être normalisée par rapport à la longueur du texte). L'architecture la plus fréquemment adoptée est l'architecture en réseaux non bouclés avec une structure de perceptron multicouche (Hornik,1993). Plus précisément, cette structure consiste en général en k entrées (les ***k*** attributs d'une ressource), une couche d'un certain nombre de neurones cachés, et un certain nombre de neurones de sortie. Chaque neurone de sortie indique un score permettant de déterminer si une ressource appartient à la classe du niveau d'appréciation à laquelle il est associé. Un algorithme répandu pour effectuer l'apprentissage des poids s'est l'algorithme PLA (Perceptron Learning Algorithm) Il consiste à initialiser les variables de façon aléatoire et à les ajuster itérativement de façon à minimiser le nombre de ressources disposées dans de mauvaises classes.

En plus de permettre un apprentissage rapide, l'utilisation de réseaux de neurones à l'avantage de permettre un ajustement particulièrement fin grâce à l'utilisation de la fonction sigmoïde. Selon le domaine d'application il peut s'avérer plus ou moins efficace que ses alternatives (Pazzani et Billsus, 1997).

4.1.2- Formes particulières de recommandation basée sur le contenu

Nous présentons à présent quatre approches habituellement présentées comme sortant du cadre de la recommandation basée sur le contenu, mais considérées comme des points de vue particuliers de la recommandation basée sur le contenu. En outre, une approche de recommandation basée sur contenu peut relever de plusieurs de ces quatre approches.

a- Recommandation à partir de cas

La recommandation à partir de cas (Smyth, 2007) est basée sur le raisonnement à partir de cas (Althoff, 2001) qui consiste à adapter des solutions concrètes à des problèmes spécifiques rencontrés dans le passé, appelés cas, pour résoudre des problèmes similaires.

Pour effectuer des recommandations en utilisant le raisonnement à partir de cas, il suffit donc de considérer les ressources comme des cas et les recommandations

comme des solutions à ces problèmes. Les deux critères qui différencient la recommandation à partir de cas des autres formes de recommandation basée sur le contenu, sont la façon dont les ressources sont représentées et la façon dont le concept de similarité est appréhendé.

Les cas consistent en un ensemble d'attributs décrivant les ressources. Ils sont donc a priori similaires aux profils de ressource présentés dans la section 1.4.1.1 de ce chapitre. La différence est que dans le domaine de la recommandation basée sur le contenu, les attributs considérés consistent généralement en des termes extraits de données textuelles, alors que les cas contiennent généralement des attributs qui sortent de ce cadre (par exemple le prix d'un livre dans le cadre d'une vente en ligne).

Dans le cadre de la recommandation basée sur le contenu, la similarité entre deux ressources est généralement calculée en fonction du nombre de termes qu'elles ont en commun. Dans le cadre de la recommandation à partir de cas, la similarité entre deux cas c1 et c2 est généralement calculée selon une somme pondérée des similarités des attributs correspondants de c1 et c2 (Smyth, 2007). L'avantage est donc que des attributs de types différents peuvent être considérés, et que pour chaque attribut, il est possible de déterminer une mesure de similarité spécifique.

b- Recommandation basée sur les données démographiques

La recommandation basée sur les données démographiques consiste à répartir les utilisateurs en plusieurs classes en fonction d'informations démographiques leur étant associées, telles que le sexe, l'âge, la profession, la localisation, etc. L'hypothèse sur laquelle repose cette approche est que deux utilisateurs ayant évolué dans un environnement similaire ont des codes esthétiques communs et sont donc plus susceptibles d'avoir des goûts communs que deux utilisateurs ayant évolué dans des environnements différents et ne partageant donc pas les mêmes codes. Un des avantages principaux de cette technique est qu'elle est applicable dès que les informations nécessaires sont obtenues, et permet de fournir des recommandations relativement satisfaisantes dès qu'un utilisateur commence à utiliser le système (Nguyen et al., 2006).

c- Recommandation basée sur l'utilité

La recommandation basée sur l'utilité, parfois appelée recommandation basée sur les préférences, consiste à calculer les recommandations selon une fonction d'utilité pour l'utilisateur (Stolze et R., 2001). Toute la problématique est donc de définir une telle fonction d'utilité. Une façon de procéder est de demander aux utilisateurs de remplir des formulaires. Par exemple, dans le cadre de vente en ligne de micro-ordinateurs, il est possible de demander des renseignements sur l'usage qu'en fera le client.

d- Recommandation basée sur la connaissance

La recommandation basée sur la connaissance consiste à accumuler des informations relativement élaborées sur un utilisateur pour pouvoir ensuite lui recommander des ressources (Towle et Quinn, 2000). Une analogie avec la vie réelle serait par exemple une recommandation faite par un ami qui nous connaît bien et se serait basé sur des informations précises nous concernant, plutôt que sur nos préférences. Cette approche permet également d'expliciter des liens entre les ressources : par exemple que la cuisine chinoise est proche de la cuisine vietnamienne. Une forme de recommandation basée sur la connaissance est la recommandation à partir de cas avec des attributs se rapportant à ce genre d'informations (Schmitt et Bergmann, 1999).

4.1.3- Limitations de la recommandation basée sur le contenu

La principale limitation de la recommandation basée sur le contenu est qu'elle nécessite l'acquisition d'un nombre suffisant d'attributs décrivant les ressources. C'est pourquoi elle est appropriée dans le cadre de ressources textuelles ou quand des descriptions textuelles des ressources ont été entrées manuellement. Dans le cadre de ressources textuelles, une des limitations provient des méthodes de classification de texte utilisées : en effet, deux ressources peuvent être similaires du point de vue de leurs attributs, mais avoir une qualité ou une pertinence incomparable.

Une autre limitation est que ces modèles ne peuvent recommander que des ressources similaires à celles qu'un utilisateur donné a appréciées, ce qui empêche de recommander d'autres ressources que ce même utilisateur pourrait également

apprécier. Pour amoindrir ce problème, il est possible de fournir des recommandations aléatoires parmi les recommandations.

Enfin, une dernière limitation est qu'un nouvel utilisateur d'un tel système doit avoir consulté ou fourni des appréciations pour un certain nombre de ressources avant que le système ne puisse lui fournir des recommandations pertinentes. Ce problème est connu sous le nom de démarrage à froid.

Une façon de réduire ce problème est de demander un certain nombre d'informations à l'utilisateur au moment de son arrivée (en nombre limiter pour ne pas rendre le système trop contraignant) et d'utiliser un profil type correspondant aux informations qu'il a fourni.

4.2- RECOMMANDATION BASEE SUR LES USAGES

Les systèmes de recommandation basée sur les usages calculent les recommandations en se basant sur les usages passés que les utilisateurs ont fait du système. Cette approche ne nécessite pas de considérer le contenu des ressources, ce qui présente plusieurs avantages. Le premier avantage est que cela évite l'extraction de profils de ressource et d'utilisateur. Le terme « profil » est utilisé ici dans le même sens que celui employé pour la recommandation basée sur le contenu de la section précédente. En réalité des profils d'utilisateurs peuvent être définis dans le cadre de la recommandation basée sur les usages, mais se rapportent à une forme différente de profil. Le second avantage est que les approches basées sur les usages ne sont pas aussi dépendantes de la nature des données que les approches basées sur le contenu. En particulier, elles sont applicables aussi bien aux données graphiques que sonores, deux types de données pour lesquels les recommandations basées sur le contenu ont une efficacité très limitée.

Parmi les critères exploitables pour effectuer des recommandations basées sur les usages, les deux critères principaux sont les appréciations et les motifs. L'utilisation d'appréciations correspond au filtrage collaboratif, et celle des motifs à différentes approches issues du domaine de la fouille de données.

Dans cette section, nous présentons les approches principales exploitant ces deux critères ; puis nous présentons les limitations de ces approches.

4.2.1- Filtrage collaboratif

La première méthode de recommandation basée sur les usages que nous présentons est le filtrage collaboratif. Il exploite les appréciations des utilisateurs sur les ressources. La représentation des appréciations se fait en règle générale par des notes. Ces notes sont soit attribuées de façon explicite par les utilisateurs, ce qui représente une forme d'usage, soit de façon implicite à partir d'autres formes d'usage des utilisateurs. Le filtrage collaboratif est une des techniques les plus explorées du domaine de la recommandation (Das et al., 2007).

Le premier système de recommandation ayant été désigné comme étant un système de filtrage collaboratif est le système Tapestry (Goldberg et al., 1992). En réalité, ce système était à la fois un système de recommandation et un système de recherche d'information puisque les utilisateurs pouvaient accéder à des messages électroniques à la fois en fonction des appréciations des autres utilisateurs et en formulant des requêtes. Les auteurs avaient appelé cette approche « filtrage collaboratif » car les utilisateurs pouvaient collaborer afin de mettre de côté les messages électroniques indésirables. Cette expression a depuis beaucoup été reprise au sein de la communauté des chercheurs de ce domaine (Das et al., 2007 ; Abernethy et al., 2009 ; Koren, 2009 ; Huang et Jebara, 2010). Les systèmes dits de « filtrage collaboratif » d'aujourd'hui sont bien différents du système Tapestry. En effet, dans la représentation commune de la littérature actuelle, le filtrage collaboratif consiste à fournir des recommandations en exploitant exclusivement les notes attribuées par les utilisateurs pour regrouper ces derniers en fonction de leurs goûts communs. Ces systèmes de recommandation sélectionnent donc des ressources plutôt qu'ils n'en filtrent (filtrer consiste davantage à mettre de côté les ressources indésirables), et les utilisateurs soumettent des appréciations indépendamment les uns des autres et ne collaborent pas directement.

Plus précisément, le filtrage collaboratif utilise une matrice dont les lignes correspondent aux utilisateurs et les colonnes aux ressources. Chaque cellule de la matrice correspond alors à une note fournie par l'utilisateur correspondant pour la ressource correspondante. Le but est alors de prédire les notes que les utilisateurs attribueraient aux ressources pour lesquelles ils n'ont pas encore fourni de note, pour ensuite recommander les ressources ayant les meilleures notes prédites. Le filtrage

collaboratif est en général classé en deux approches : l'approche mémoire et l'approche modèle. Dans cette section, nous présentons tout d'abord la notation que nous utiliserons ; puis nous présentons les deux approches mémoire et modèle du filtrage collaboratif.

Notation

Soit $R = \{r_1, r_2, \dots, r_N\}$ l'ensemble des ressources, $U = \{u_1, u_2, \dots, u_M\}$ l'ensemble des utilisateurs. Nous désignons par u_a l'utilisateur actif pour lequel une recommandation doit être calculée. Chaque note de $u_i$pour chaque ressource r_j est une valeur numérique désignée par $a_{i,}$. La matrice de notes est alors la matrice suivante :

$$\begin{pmatrix} a_1 & a_1 & \cdots & a_1 \\ a_2 & a_2 & \cdots & a_2 \\ \vdots & \vdots & \ddots & \vdots \\ a_M & a_M & \cdots & a_M \end{pmatrix}$$

a- Approche mémoire

L'approche mémoire consiste à utiliser des algorithmes qui calculent chaque prédiction de note à partir de toute la matrice de notes. La méthode la plus couramment utilisée dans la littérature consiste à effectuer une somme pondérée de ces notes. Généralement, la formule utilisée pour le calcul de la prédiction de la note a_a^* d'une ressource r_j pour l'utilisateur actif $u_a$est la suivante :

$$a_a^* = \overline{a_a} + k \sum_{i=1}^{M} w(u_a, u_i).(a_{i,} - \overline{a_i}) \quad (1.3)$$

Où $\overline{a_i}$ désigne la moyenne des notes attribuées par l'utilisateur u_i, $w(u_a, u_i)$ est une fonction de pondération indiquant la similarité entre $u_a$et u_i, et K est un coefficient de normalisation tel que la somme des valeurs absolues des poids soit égale à 1. L'utilisation des $\overline{a_i}$ permet de recentrer les notes attribuées par les utilisateurs. En effet, il se peut que deux utilisateurs aient des goûts similaires, mais que l'un soit plus sévère que l'autre dans ses notations. Repositionner chaque note de chaque utilisateur relativement à la moyenne de ses notes permet de limiter les effets de ce phénomène. L'élément déterminant de cette formule est la fonction de pondération $w(u_a, u_i)$, qui permet d'accorder plus d'importance aux notes des utilisateurs les plus

similaires à l'utilisateur actif. Les valeurs de cette fonction sont en général calculées hors-ligne, et peuvent être obtenues de différentes façons. Parmi les fonctions de pondération les plus communes, on peut citer la similarité cosinus entre les vecteurs de notes (Breese et *al.*, 1998) :

$$w(u_a, u_i) = \cos(\overrightarrow{u_a}, \overrightarrow{u_i}) = \frac{\overrightarrow{u_a}.\overrightarrow{u_i}}{\|\overrightarrow{u_a}\|.\|\overrightarrow{u_i}\|} = \frac{\sum_j a_a . a_i}{\sqrt{\sum_j a_a^2}\sqrt{\sum_j a_i^2}} \quad (1.4)$$

Où les sommes sur j correspondent à la somme sur toutes les ressources notées à la fois par u_a et par u_i. C'est donc la même mesure de similarité que celle utilisée dans le cadre de la recommandation basée sur le contenu pour calculer la similarité entre les ressources (et entre le profil d'utilisateur et les ressources). La différence est que les dimensions du vecteur correspondent ici aux ressources et ont pour valeur les notes, alors que dans le domaine de la recommandation basée sur le contenu, les dimensions correspondent aux termes des ressources. Une autre fonction de pondération célèbre est le coefficient de corrélation de Pearson (Resnick et *al.*, 1994) :

$$w(u_a, u_i) = \frac{\sum_j (a_a - \overline{u_a})(a_i - \overline{u_i})}{\sqrt{\sum_j (a_a - \overline{u_a})^2}\sqrt{\sum_j (a_i - \overline{u_i})^2}} \quad (1.5)$$

La différence avec la mesure de la similarité cosinus est que les vecteurs $\overrightarrow{u_a}$ et $\overrightarrow{u_i}$des utilisateurs sont centrés relativement à leurs notes moyennes, comme dans la formule 1.2. Le résultat est donc une valeur de corrélation comprise entre -1 et 1. Si $w(u_a, u_i)$ vaut 1, alors les utilisateurs sont considérés comme très similaires ; si $w(u_a, u_i)$ vaut 0, alors les utilisateurs sont indépendants ; si $w(u_a, u_i)$ vaut -1, alors les utilisateurs sont fortement opposés (Castagnos, 2008). Une façon d'utiliser cette mesure avec la formule 1.2 est d'ignorer les utilisateurs ayant une valeur de corrélation négative avec u_a.

L'approche mémoire a l'avantage d'être à la fois simple et performante, et de permettre une adaptation dynamique au fur et à mesure que de nouvelles notes sont entrées dans la matrice ; cependant sa complexité est telle que son utilisation n'est possible que sur un espace de données relativement réduit. Cette dernière limitation est connue sous le nom de problème du passage à l'échelle.

b- Approche modèle

L'approche modèle répond à la problématique de la complexité de l'approche mémoire en utilisant des modèles d'utilisateur, de ressource, de communauté, de session, etc. L'avantage est que ces modèles peuvent être construits hors ligne à partir de corpus d'apprentissage et être utilisés rapidement en ligne pour calculer les recommandations. Par conséquent, la problématique de la complexité en temps de la construction de ces modèles peut être secondaire, selon le type d'application. Il est en effet souvent envisageable de construire un modèle en plusieurs heures voire en plusieurs jours, ce qui a l'avantage de permettre des approches de construction plus subtiles, mais l'inconvénient de rendre impossible une actualisation fréquente.

Dans le cadre de l'approche modèle, la prédiction d'une note peut être calculée de deux manières. La première manière consiste à construire un modèle probabiliste dans lequel sont stockées des estimations de probabilités. Ces probabilités peuvent alors être utilisées pour calculer l'espérance des notes ou de déterminer la note la plus probable :

1. en calculant l'espérance de la note (Breese et *al*., 1998) :

$$p_a = E(\alpha_{ii}) = \sum_{\alpha=\alpha_m}^{\alpha_m} \alpha . P(\alpha_{ii} = \alpha) \quad (1.6)$$

Où a représenté une valeur de note.

2. ou en recherchant la note la plus probable (Schafer et al., 2007) :

$$p_a = \arg\max P(\alpha_{ii} = \alpha) \quad (1.7)$$

La problématique est alors de construire un modèle à partir duquel il sera possible d'obtenir les probabilités de ces deux équations.

Une autre possibilité généralement classée dans les approches modèle consiste à regrouper les utilisateurs ou les ressources en sous-groupes homogènes, et d'appliquer les approches mémoire sur les sous-groupes ainsi obtenus. L'espace de données sur lequel sont appliqués les algorithmes peut ainsi être suffisamment réduit pour que les algorithmes de l'approche mémoire lui soient appliqués. De nombreuses

approches modèle ont été proposées, dont les principales sont présentées dans cette section.

1) Réseaux bayésiens

Un réseau bayésien, parfois appelé diagramme d'influence, consiste en un graphe orienté sans circuit ***G = (X, A)*** dans lequel les nœuds X = $\{x_1, \dots, x_N\}$ représentent des variables aléatoires et les arcs A les relations entre ces variables. Ces relations sont des relations de dépendance conditionnelle formulées sous forme de probabilités conditionnelles. Ces probabilités conditionnelles sont stockées sous forme de tableaux, chaque nœud du graphe possédant son propre tableau. La distribution jointe du réseau peut alors être formulée de la façon suivante :

$$P(x_1, \dots, x_N) = \prod_{i=1}^{N} P(x_i | P(x_i)) \quad (1.8)$$

Où Par (x_i) correspond à l'ensemble des valeurs des parents de xi dans le graphe.

Dans le cadre du filtrage collaboratif, chaque nœud est associé à une ressource et chaque valeur de nœud à une note. De cette manière, chaque ressource possède un ensemble de parents dont ils sont grandement dépendants. Il est ainsi possible de stocker des probabilités de façon compacte et de les utiliser pour effectuer les prédictions de notes.

Un des travaux les plus célèbres utilisant des réseaux bayésiens pour le filtrage collaboratif est celui de (Chickering et *al*., 1997). Dans ce travail, les auteurs proposent d'utiliser des arbres de décision pour représenter les dépendances conditionnelles de façon encore plus compacte. Dans (Breese et *al*., 1998), cette dernière approche est comparée empiriquement à d'autres modèles de filtrage collaboratif. Les résultats montrent qu'elle est plus performante que l'approche mémoire utilisant la similarité cosinus et aussi performante que l'approche mémoire utilisant la corrélation de Pearson. Les réseaux bayésiens ont une complexité en temps et en mémoire plus faible pour le calcul des prédictions de note, mais une complexité en temps très élevée pour la phase d'apprentissage (plusieurs heures, voire plusieurs jours). Par conséquent, ils sont préférables quand une actualisation en temps réel n'est pas obligatoire et que l'espace de données est très important.

2) Classifier naïf de Bayes

Le classificateur naïf de Bayes, est une forme particulière, et très simple, de réseau bayésien (Friedman et al., 1997). Il peut être représenté en tant que tel, et utilisé pour construire des communautés homogènes d'utilisateurs. Dans ce cadre, les attributs correspondent aux ressources, et leurs valeurs aux notes attribuées par l'utilisateur considéré. Par conséquent, au lieu de porter sur les descriptifs des ressources, l'hypothèse d'indépendance porte sur les notes attribuées par les utilisateurs, hypothèse qui n'est pas plus réaliste que la précédente. Relativement à cette considération, l'utilisation du classificateur naïf de Bayes fournit de bons résultats dans le cadre du filtrage collaboratif également, mais se révèle moins performante que les réseaux bayésiens dans leur version moins restreinte décrite ci-dessus (Breese et *al*., 1998).

Un des problèmes liés à l'utilisation d'un tel classificateur est que le nombre de classes et leurs paramètres doivent être déterminés à l'avance. Une solution est d'utiliser l'algorithme Expectation Maximization (Dempster et *al*., 1977) avec un nombre fixé de classes pour déterminer les paramètres qui fournissent le maximum de vraisemblance et la méthode de Cheeseman et Stutz (Cheeseman et Stutz, 1996) pour déterminer le nombre de classes qui fournit la meilleure vraisemblance marginale (Breese et *al*., 1998).

Plutôt que de classer les utilisateurs en communautés homogènes, il est possible de classer les ressources en fonction de leur niveau d'appréciation. Un des premiers travaux ayant utilisé cette approche (Miyahara et Pazzani, 2002) utilise un nombre de classes fixé à deux : « aime » et « n'aime pas » en utilisant une transposition relativement à une note seuil. Cette approche a depuis été expérimentée avec un plus grand nombre de classes, en particulier dans (Su et Khoshgoftaar, 2006). Les résultats montrent qu'une telle classification des ressources permet un meilleur passage à l'échelle que les approches mémoire, mais fournit de moins bons résultats.

3) Clustering

Une alternative à la classification est le clustering, qui a l'avantage de ne pas imposer de connaître à l'avance les paramètres des classes. En effet, le clustering consiste à regrouper des ressources similaires et/ou à séparer les ressources

dissimilaires (Han et Kamber, 2006), quand les classificateurs répartissent les ressources dans des classes dont le nombre et les paramètres sont prédéterminés.

Il existe trois catégories classiques de clustering :

1- Hiérarchique (l'algorithme BIRCH (Zhang et *al*., 1996)).
2- Par partitionnement (l'algorithme k-means (MacQueen, 1967)).
3- Basé sur la densité (l'algorithme DBSCAN (Ester et *al*., 1996)).

Généralement, quand le clustering est expérimenté pour le filtrage collaboratif, les résultats obtenus sont légèrement inférieurs aux résultats fournis par les approches mémoire. En effet, le clustering est utilisé pour séparer les utilisateurs en communautés homogènes sur lesquelles il sera possible d'appliquer les approches mémoire. Or, les communautés obtenues ne sont jamais parfaitement homogènes, et les utilisateurs appartenant à des communautés différentes pourraient apporter des informations pertinentes qui sont ignorées. Cependant, dans le cadre d'un espace de données important, le clustering permet un passage à l'échelle que ne permettent pas les approches mémoire.

4.2.2 Limitations du filtrage collaboratif

Une des principales limitations du filtrage collaboratif est le problème du manque de données. En effet, dans le cadre de notes explicites, le pourcentage moyen de ressources pour lesquelles les utilisateurs ont fourni une appréciation est très basse. Par exemple, une des bases de données fournies par MovieLens 6 consiste en 100 000 notes pour 1 642 films par 943 utilisateurs, soit 6, 3% de notes fournies. Dans un tel cadre, la similarité entre deux utilisateurs ne peut être calculée que s'ils ont noté un minimum de ressources communes.

Tout comme les approches basées sur le contenu, le filtrage collaboratif souffre du problème du démarrage à froid : avant que le système puisse fournir des recommandations pertinentes à un utilisateur, il faut que ce dernier ait fourni, implicitement ou explicitement, des appréciations pour un nombre suffisant de ressources. Un problème supplémentaire par rapport aux recommandations basées sur le contenu est que cette limitation s'applique également aux nouvelles ressources

introduites dans le système. Des solutions à ces problèmes se trouvent dans les approches hybrides, présentées dans la section 1.4.3.

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.

4.3.1 Hybridation pondérée

L'hybridation pondérée 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 calcules, ceux-ci sont classées par ordre décroissant avant d'être présentés à l'utilisateur Ce type d'hybridation nécessite que les systèmes de recommandations utilisés effectuent une tâche de prédiction de note.

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 (Daniel Billsus and Michael J. Pazzani,2000)

4.3.3 Hybridation mixée

L'hybridation mixée (Barry Smyth and Paul Cotter,2000) consiste à présenter à l'utilisateur des recommandations issues de plusieurs approches de recommandation.

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 (Chumki Basu and al ,1998). Par exemple, lors de l'utilisation d'une approche collaborative U2U, 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 a apprécié 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.

4.3.5 Hybridation en cascade

L'hybridation en cascade implique l'application successive des approches de recommandation du système (Robin Burke,2002). 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.

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 (Prem Melville and al,2002). 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.

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 (Michael J. Pazzani,1999). La deuxième approche de recommandation remplace complétement 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.

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 afin d'obtenir une approche hybride qui remédie aux limitations des approches simples. Cependant, malgré la multitude d'approches de recommandation existantes, on ne trouve pas d'approche infaillible qui convient à tous les utilisateurs. En effet, chaque utilisateur répond mieux à certaines approches plutôt que d'autres (Nicolas Ducheneaut and al ,2009). Utiliser une seule approche de recommandation, qu'elle soit simple ou hybride, ne pourra correspondre à tous les utilisateurs. L'approche de (Michael D. Ekstrand and al ,2015) propose aux utilisateurs la possibilité de choisir le processus de recommandation qui leur sera appliqué après leur avoir attribué un processus aléatoire. L'évaluation de cette étude a démontré que seuls 24.9% des utilisateurs ont exploré cette option. C'est pourquoi, nous proposons dans notre système de recommandation de détecter a priori à quel type d'approche de recommandation l'utilisateur répondra le mieux, en analysant son profil, afin de personnaliser au mieux le processus de recommandation qui lui sera appliqué.

5- CONCLUSION

Dans ce chapitre, nous avons défini la notion des systèmes de recommandation, et on a présentés les trois approches les plus utilisées par ces systèmes qui sont le filtrage collaboratif et filtrage basé contenu ainsi que leur hybridation et leurs limitations.

La mise en place de ses systèmes nécessite l'intégration d'un profil utilisateur que nous abordons dans le chapitre suivant, à savoir la modélisation du profil utilisateur en systèmes de recommandation.

CHAPITRE II

Modalisation du profil en systèmes de recommandation

1- INTRODUCTION

Le profil est l'élément essentiel dans les systèmes de recommandation. C'est une structure qui consiste à représenter et stoker les informations relatives à l'utilisateur et à la ressource pour pouvoir recommander à l'utilisateur un contenu pertinent en fonction de ses besoin et exigences.

Ce chapitre porte sur la modélisation du profil de la ressource, puis la modélisation du profil utilisateur, à savoir sa représentation, sa construction .

2- LA MODELISATION DES RESSOURCES

Le but des systèmes de recommandation et de fournir des recommandations permettant de guider l'utilisateur vers des ressources intéressantes et utiles au sein d'un espace de donnés important.

La manière la plus simple de décrire ces ressources est d'avoir une liste explicite des caractéristiques de chacune d'elle. Comme exemple, un livre est une ressource caractérisée par : le genre, le nom des auteurs, l'éditeur ou toute autre information relative au livre. Ces caractéristiques sont stockées dans une structure vectorielle.

Le tableau **(tableau 2.1)** suivant illustre l'exemple de représentation de ressources caractérisées par : Titre, Genre, Auteur, Prix, Mots-clés .

Titre	*Genre*	*Auteur*	*Prix*	*Mots-clés*
Shining	Thriller	S. King	19,50	*Alcoolisme, Colorado, Médium, Surnaturel, Hôtel, …*
Millenium	Policier	S. Larsson	23,20	Journalisme, Investigation, Meurtre, Suède, Politique, …
Le Journal de Bridget Jones	Romance	H. Fielding	8,50	Célibataire, Humour, Amour, Trentenaire, Journal intime, …

Tableau 2.1 : Exemple de modélisation de ressources

3-LA MODELISATION DES UTILISATEURS :

Le modèle utilisateur a pour objectif de profiler l'utilisateur en prenant compte ses goûts et ses préférences. Pour cela plusieurs approches ont été définies pour représenter le profil de l'utilisateur dans le cas des systèmes de recommandations. Ce profil peut contenir :

- **Données sur les usages** : la modélisation de l'utilisateur par les données issues de l'analyse des usages, et particulièrement les votes, est essentiellement utilisée par les systèmes de recommandation collaboratifs (lousan et al, 2009) .Les utilisateurs sont modélisés par la matrice des votes contenant l'ensemble de leurs votes et interaction avec les items recommandés.

- **Données démographiques** : le profil est modélisé par un ensemble de données démographiques telles que l'âge, le sexe, la profession, le niveau d'étude, le pays etc, décrivant le type de l'utilisateur. Les systèmes à filtrage démographique propose les mêmes recommandations aux utilisateurs de même type, c'est à dire ne fournissent pas de recommandations précise a un seul individu. (mastchoff, 2011).

- **Données sur le contenu des items :** le contenu sémantique des items est essentiellement exploité par les systèmes de recommandation basés sur le contenu.

  Plusieurs approches ont été utilisées pour représenter le profil utilisateur à partir du contenu de l'item (montaner et al, 2003) ; parmi lesquelles on peut citer le modèle vectoriel (VSM) dans lequel l'utilisateur est représenté par un vecteur de poids définit dans le même espace que celui représentant les items. Chaque poids mesurant l'importance du terme correspondant pour l'utilisateur.

3.1 Représentation du profil utilisateur

Il existe différentes représentations du profil utilisateurs dans la littérature. Nous allons présenter dans ce qui suit les modèles les plus connus en système de recommandation.

3.1.1 Représentation ensembliste

L'approche ensembliste consiste à représenter le profil de l'utilisateur par des paquets de termes pondérés. On parle plutôt d'une représentation vectorielle par analogie au modèle vectoriel de Salton **(salton, 1971)** sur laquelle elle se base. Ces paquets de termes, traduisant les centres d'intérêts de l'utilisateur, peuvent être regroupés différemment selon l'approche suivie pour considérer le profil de l'utilisateur.

On distingue dans la littérature trois grandes approches de représentation du profil utilisateur basées sur ce modèle :

- Par une liste de mots clés, où chaque mot correspond à un centre d'intérêt spécifique

  (freitag et al, 1995).

- Par un vecteur de termes pondérés pour chaque centre d'intérêt (Tebri et al, 2005).
- Par un ensemble de vecteurs de termes pondérés (ou non) indépendants, pour prendre en compte des centres d'intérêt multiples (Somlo et al, 2003) où chaque vecteur correspond à un domaine d'intérêt (pazzani et al, 1996).

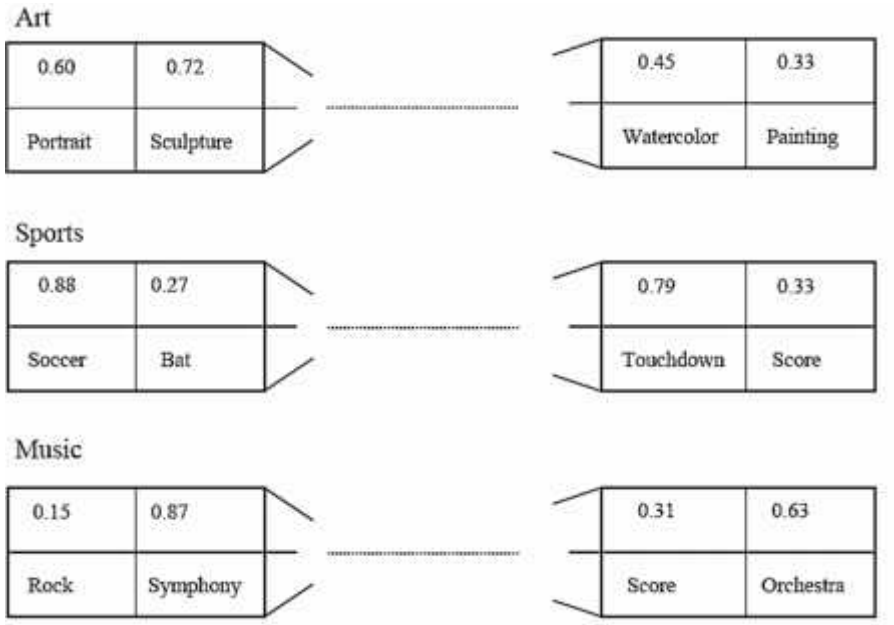


Figure 2.1 : Un exemple de profil représenté par des mots clés

La représentation ensembliste fut parmi les premiers modèles de profils utilisateur exploités plus précisément en approche basé sur le contenu. La pondération des termes est généralement basée sur un schéma de la forme $TF * IDF$ communément utilisé en recherche d'information (salton et al, 1973). Le poids associé à chaque terme permet de représenter son degré d'importance dans le profil de l'utilisateur. La **figure 2.1** donne un exemple de profil utilisateur représenté par des mots clés pondérés. Ce profil contient trois centres d'intérêts : *Art*, *Sports* et *Music*. Chaque centre est représenté par un ensemble de termes pondérés.

Music =< (*Rock*, 0,15), (*Symphony*, 0,87), · · ·> est un extrait du l'ensemble de termes pondérés représentant le centre *Music*.

3.1.2 REPRESENTATION SEMANTIQUE A BASE D'ONTOLOGIE :

C'est l'un des modèles les plus populaires en recommandation, où un profil est une hiérarchie de concepts pondérés qui sont représentés par des nœuds auxquels on attache un poids qui représente l'intérêt de l'utilisateur pour ce concept.

Un exemple de profil utilisateur représenté par le modèle à base d'ontologie est illustré dans la figure 2.2 (Micro speretta et al, 2004) ; (Vistu kanth, 2004) ; (stuarte et al, 2001) ; (ahu sieg et al, 2007).

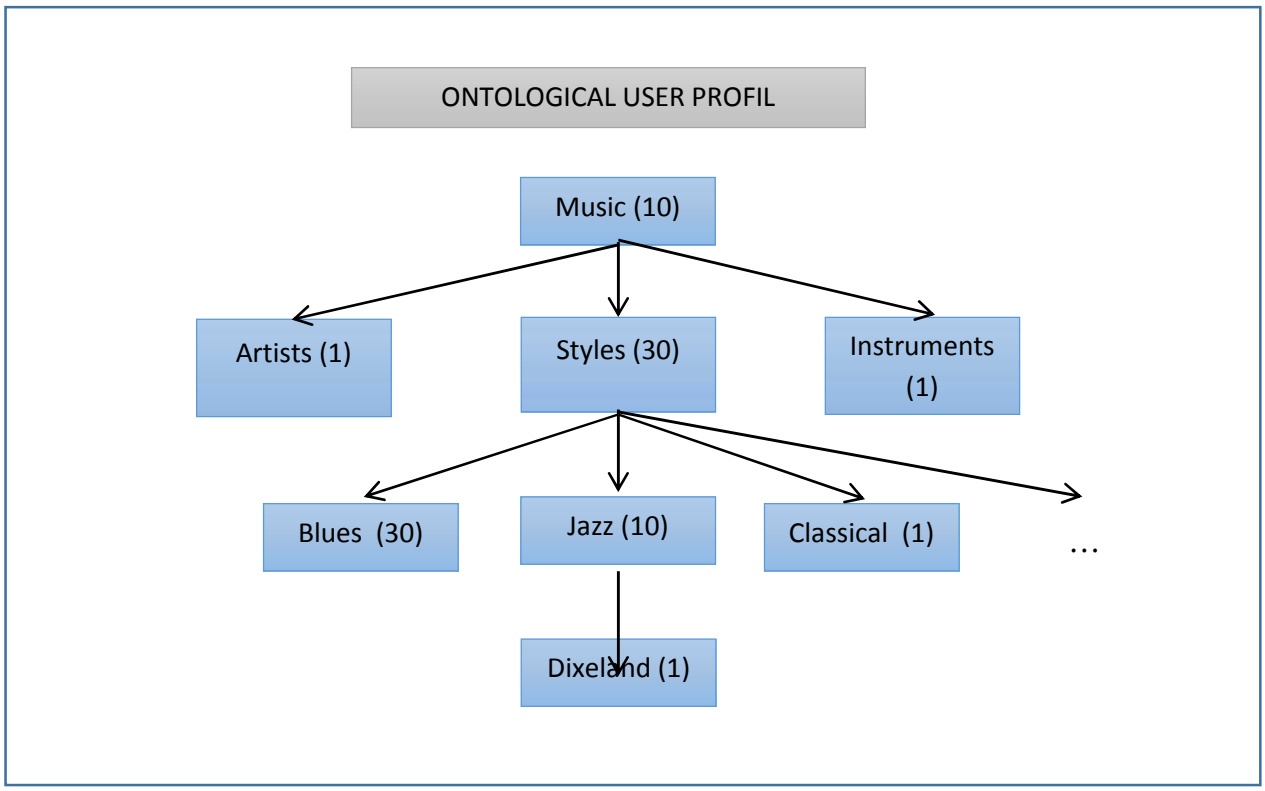


Figure 2.2 : Exemple du profil utilisateur représenté par le modèle d'ontologie. (ahu sieg et al, 2007)

Comparer à la représentation vectoriel ; la représentation sémantique du profil utilisateur peut aider à mieux connaître les intérêts des utilisateurs. Par exemple, si on utilise le modèle ontologie (comme illustré dans la **figure 2.2**) le concept " instruments " est un concept " fils " du concept " music ", alors on peut facilement savoir qu'il s'agit des instruments de musiques alors que si on représente un profil utilisateur par

un vecteur de termes pondérés qui contient le terme instrument, on ne sait pas exactement si ce terme concerne des instruments de musique ou d'autres types d'instruments.

De plus, un autre avantage du modèle d'ontologie est qu'on peut propager la valeur d'intérêt d'un concept vers les autres concepts reliés (par exemple, son concept "père") afin de trouver des nouveaux centres d'intérêts. Cette représentation est très souvent utilisé en recommandation nous citons comme exemple le système Foxtrotte Quicks tep (stuarte et al, 2001) ; (Struarte et al, 2003), qui utilise un modèle à base d'ontologie des articles scientifiques pour représenter les centres d'intérêts des utilisateurs.

3.1.3 REPRESENTATION MULTIDIMENTIONELLE

Cette représentation a pour objectif de capturer toutes les caractéristiques informationnelles de l'utilisateur. Différents travaux ont abordé cet aspect sans le couvrir dans son ensemble. Ainsi, les propositions de standards P3P pour la sécurisation des profils ont défini des classes distinguant les **attributs démographiques** des utilisateurs (*identité*, *données personnelles*), les **attributs professionnels** (*employeur*, *adresse*, *type*) et les **attributs de comportement** (*trace de navigation*).

Une autre proposition faite par (Amato et al, 1999) consiste à représenter le contenu du profil utilisateur par un modèle structuré de dimensions (ou catégories) prédéfinies :

- *catégorie de données personnelles*.
- *catégorie de données de la source.*
- *catégorie de données de livraison.*
- *catégorie de données de comportement.*
- *catégorie de données de sécurité.*

L'auteur a proposé ce modèle dans le cadre du développement d'un service avancé de librairie digitale (recherche et livraison personnalisées de l'information sur le *web*) : le système EUROgatherer.

Dans ce même cadre, Kostadinov (D. Kostadinov ,2003) a poursuivi cette classification en proposant un ensemble de dimensions ouvertes, pouvant contenir la

plupart des informations susceptibles de caractériser.

L'utilisateur. Dans sa représentation il distingue principalement huit dimensions décrites brièvement dans ce qui suit :

- *Les données personnelles*

Les données personnelles sont la partie statique du profil. Elles comprennent l'identité civile de l'utilisateur ainsi que ces données démographiques (âge, genre adresse, nom, prénom etc).

- *Le centre d'intérêt*

Le centre d'intérêt exprime le domaine d'expertise de l'utilisateur

- *L'ontologie du domaine*

L'ontologie du domaine complète la définition du centre d'intérêt en explicitant la sémantique de certains termes ou de certains opérateurs employés par l'utilisateur dans son profil ou dans ses requêtes.

- *La qualité attendue*

Permet d'exprimer des préférences extrinsèques comme l'origine de l'information, sa précision, sa fraîcheur, sa durée de validité, le temps nécessaire pour la produire ou la crédibilité de sa source. Les attributs de cette dimension expriment la qualité attendu, ou espérée par l'utilisateur.

- *La customisation*

La customisation concerne d'abord tout ce qui est lié aux modalités de présentation des résultats en fonction de la plateforme, de la nature et du volume des informations délivrées, des préférences esthétiques ou visuelles de l'utilisateur.

- *La sécurité*

La sécurité est une dimension fondamentale du profil. Elle peut concerner les données que l'on interroge ou modifie, les informations que l'on calcule, les requêtes utilisateurs elles-mêmes ou les autres dimensions du profil. La sécurité du processus exprime la volonté de l'utilisateur à cacher un traitement qu'il effectue.

- *Le retour de préférences*

On désigne par ces termes ce qu'on appelle communément le « feedback » de l'utilisateur. Cette dimension regroupe l'ensemble des informations collectées sur l'utilisateur.

- *Les informations diverses*

Certaines applications demandent des informations spécifiques ne pouvant être incluses dans aucune des dimensions précédentes comme par exemple la bande passante attribuée au gestionnaire du profil. Pour cette raison l'utilisateur a la possibilité de rajouter ce type de préférences dans la partie divers du profil et de décrire leurs utilisations.

Pour une application donnée, un utilisateur n'a pas besoin de toutes les dimensions ou sous dimensions ni de toutes les informations caractérisant une dimension. Un profil donné est donc une instanciation partielle de ce méta modèle en fonction des besoins de l'utilisateur, du type d'application et de l'environnement d'exécution de cette application.

3.1.4 REPRESENTATION PAR MATRICE UTILISATEURS_ITEMS

La représentation par matrice utilisateurs-items est souvent utilisé dans les systèmes de recommandation collaborative (voir **tableau 2.2)**. Chaque ligne de la matrice représente un utilisateur et chaque colonne représente un item. Une cellule [i,j] de la matrice contient le vote de l'utilisateur i pour sl'item j sur une échelle quelconque.

Dans cette exemple l'échelle est de (D. Kostadinov ,2003) (ou rien si utilisateur n'a pas voté cet item) . Dans ce type de représentation, le profil d'un utilisateur est considéré comme un vecteur des votes de cet utilisateur pour les items.

	Item1 2	Item2	Item3	Item4
Utilisateur1	-	6	9	2
Utilisateur2	2	-	6	8
Utilisateur3	8	2	-	-

Tableau 2.2 : Matrice utilisateurs items

3.2 Construction du profil utilisateur

La construction du profil traduit un processus qui permet d'instancier sa

représentation à partir de divers sources d'information **(Tamine, 2007)**.Elle consiste à collecter et exploiter les données et sources d'information pertinentes pour les représenter.

Nous présentons, tout d'abord, les techniques d'acquisition des sources d'informations et le prétraitement des données. Puis, nous présentons les techniques de construction du profil utilisateur.

3.2.1 Acquisitions des données

a- Acquisition des données explicites

Les données explicites sont celles qui sont fournies par les utilisateurs eux-mêmes.

La technique d'acquisition de ces données est une technique simple, qui consiste à interroger l'utilisateur, pour lui demander des informations personnelles, démographiques et/ou ses intérêts **(Brusilovsky & al., 07)**.

Cela peut se faire en demandant à l'utilisateur de remplir un formulaire d'informations personnelles pour construire son profil, des interviews, des questionnaires, introduction des mots clés,…etc. Aussi on peut considérer la réaction de l'utilisateur lors de son interaction avec le système (par exemple les programmes suivis ou les produits achetés sur Internet, ou encore son choix de recommander tels articles ou tels produits à d'autres utilisateurs).

Ces données sont parfois non fiables (lorsque les utilisateurs fournissent de fausses données) alors la pertinence du profil dépend du degré d'implication de l'utilisateur pour fournir des réponses exactes et complètes.

b- Acquisition des données implicites

L'acquisition des données explicites est trop limitée, ce qui a orienté les travaux vers des techniques d'acquisition des données implicites de l'utilisateur. De plus, le profil d'un utilisateur doit contenir des informations précises sur ses préférences, or souvent il y a un décalage entre l'intention de l'utilisateur et ce qu'il désire réellement. Donc il s'agit plus de demander à l'utilisateur de disposer ses données, mais de trouver d'autres sources permettant d'extraire des connaissances sur l'utilisateur et de construire son profil.

Le fonctionnement de base de cette approche est réalisé par l'établissement d'un dialogue entre le système et l'utilisateur ou mieux encore, en observant son

comportement à travers ses différentes interactions avec les systèmes pour récolter discrètement l'information nécessaire sur lui (Kelly & al., 03).

c- Acquisition hybride

Quelques systèmes ont choisi de combiner les deux précédentes méthodes pour obtenir une meilleure performance. Dans (young woo seo et al, 2000), un profil utilisateur contient des termes pondérés, chaque fois qu'un document est jugé pertinent, le poids d'un terme dans son profil est mis à jour en utilisant les paramètres suivants : le vote explicite, le temps utilisé pour lire ce document, le nombre de liens suivis et l'action sauvegarder dans les signets de ce document.

3.2.2 PRETRAITEMENT DES DONNEES

Les données issues de la sélection des données, comme expliqué précédemment, peuvent contenir de nombreuses inconsistances telles que : des données incomplètes (manque de valeurs ou attributs importants), des données biaisées (présence d'erreurs produites lors des saisies ou de la collection automatique de données), des incohérences (nommages ou codages différents dans les données, …). De plus, ces données brutes peuvent ne pas être conformes au modèle ou au format d'entrée de l'algorithme de construction du profil utilisateur.

Après l'étape d'acquisition de données, il est nécessaire de mettre en œuvre une étape de prétraitement avant l'étape finale de construction du profil utilisateur.

On utilise plusieurs types de prétraitement selon les données (Gracia et al, 2015); (Liu, 2007) .

Pour les données incomplètes, biaisées ou incohérentes, on peut appliquer des techniques de nettoyage de données, qui consistent à ignorer les données.

La discrétisation des données peut être appliquée pour convertir des attributs continus vers des attributs ordinaux.

La réduction des données peut être appliquée pour obtenir une représentation réduite du jeu de données, plus petite en volume, mais qui produit (ou presque) les mêmes résultats analytiques.

Après la collecte et le prétraitement des données, celles-ci sont utilisées en entrée de la phase de construction du profil utilisateur.

3.2.3 TECHNIQUES DE CONSTRUCTION DU PROFIL UTILISATEUR :

Le processus de construction consiste à organiser et extraire les éléments qui constituent le profil à partir des données de l'utilisateur collectées lors de l'étape précédente.

Selon le modèle de représentation du profil utilisateur, la construction s'appuie sur différentes techniques. On distingue trois principales techniques, détaillées dans les paragraphes suivants : l'*extraction des termes*, l'*extraction de réseaux de termes* et l'*extraction de concepts*.

a- Extraction d'ensemble de termes

L'extraction des termes est basée sur des techniques d'analyse statistique de mots clés.

L'idée principale consiste à analyser le contenu des documents utilisateur et d'en extraire des mots clés significatifs qui décrivent son contenu. Ces termes constituent les données d'entrée pour l'algorithme d'apprentissage du profil.

Dans le cas où le profil contient simplement que des mots-clés, ces termes vont être regroupés en paquets selon leur degré de similarité pour former les centres d'intérêts. Dans le cadre d'une approche vectorielle, les termes vont être pondérés pour former des vecteurs de termes représentant les centres d'intérêts. Le poids attribué à chaque mot clé permet de traduire son degré d'importance dans le profil. La fonction de pondération appliquée par la majorité des systèmes est issue du schéma $TF * IDF$ (G. Salton and M. McGill, 1983). Le nombre de termes extraits est souvent fixé selon un seuil de pondération de sorte que seuls les termes dépassant cette valeur contribuent à la construction du profil. Ceci permet d'obtenir des profils plus concis et plus représentatifs des centres d'intérêts de l'utilisateur.

La formule de pondération $TF * IDF$ est donnée par ce qui suit :

$$W_{t_i,d} = TF(t_i, d) * IDF(t_i) \qquad (2..1)$$

Tel que $IDF(t_i) = \log \frac{|D|}{df(t_i)}$ où $|D|$ le nombre total de documents, $df(t_i)$ est la fréquence d'apparition du terme dans le document, $TF(t_i, d)$ est la fréquence du terme t_i dans le document d.

b- Extraction de réseaux de termes

Similairement aux techniques de construction de profils ensemblistes ; les

termes sont extraits des documents jugés par l'utilisateur. Néanmoins, à la différence des approches précédentes, où les termes forment des vecteurs, les techniques de construction sémantique intègrent ces termes dans un réseau de nœuds.

La construction des profils nécessite l'exploitation de relations préexistantes entre les termes et les concepts, tels que WordNet dans le cas du système SiteIF (A. Stefani and C. Strappavara ,1998), ou manuellement construites tel que celui effectué par WIFS (A. Micarelli and F. Sciarrone, 2004).

Dans les approches élémentaires, chaque utilisateur est représenté par un seul réseau sémantique dans lequel chaque noeud contient un mot-clé unique. Lorsqu'un terme est présent dans le réseau, le poids de son nœud est augmenté ou diminué selon le feedback de l'utilisateur. Si le terme n'apparaît pas dans le réseau, un nouveau nœud est créé. Les poids dans le réseau sont périodiquement réévalués à chaque mise à jour dans le but de modéliser les changements des centres d'intérêts de l'utilisateur à long terme. En outre, les concepts qui ne sont plus d'actualité peuvent être supprimés du réseau.

c- Extraction de concepts

L'approche de construction s'effectue de manière générale comme suit :

- ✓ **identifier les concepts et niveaux d'ontologie exploitée**.

  L'objectif étant d'extraire un sous ensemble de concepts représentants le profil générale .Dans la plus part des travaux la ressource sémantique n'est pas exploité dans sa totalité. Certes, l'utilisation de tous les concepts de la hiérarchie (telle que c'est le cas dans le système *Persona* (F. Tanudjaja and L. Mui. Persona,2002) permet d'obtenir des profils utilisateurs assez précis, pouvant couvrir un grand nombre de centres d'intérêts. Cependant, la difficulté de cette approche, se situe à juste titre au niveau de la profondeur de la hiérarchie d'ODP et la richesse des concepts. En effet, le profil de l'utilisateur peut devenir très important, contenant trop de concepts proches. Lors de la sélection des documents, le système doit établir des relations de similarité utilisant les concepts du profil de différents niveaux, puis statuer sur le niveau à exploiter pour évaluer les documents candidats. De ce fait, la plus part des travaux extraient qu'un nombre réduit de concepts à partir des premiers niveaux de la hiérarchie (J. Chaffee and al,2000).

- ✓ **Extraire les centres d'intérêts de l'utilisateur par analogie aux concepts de l'ontologie.**

Cette phase correspond à la phase de construction proprement dite du profil. En ce sens où le profil de chaque utilisateur est instancié à partir du profil général (la ressource sémantique) sur la base des informations collectées de l'utilisateur.

4- CONCLUSION

Dans ce chapitre nous avons présenté la modélisation du profil de la ressource et du profil utilisateur à savoir sa représentation, les techniques de sa construction dans les systèmes de recommandations.

La performance d'un système de recommandation dépend des techniques et approches adoptées pour la modélisation du profil .D'où la question :

Qu'elle approche est plus performante pour définir un profil utilisateur qui soit pertinent pour notre système de recommandation ?

CHAPITRE III

Proposition d'un modèle personnalisé pour la recommandation

CHAPITRE III
Spécification d'un modèle personnalisé pour la recommandation

1- INTRODUCTION.

L'utilisateur est l'élément essentiel dans le processus d'un système da recommandation. De ce fait la modélisation de son profil est primordiale pour l'efficacité et la performance d'un tel système.

Notre travail consiste à construire un système de recommandation base sur l'hybridation mixée en utilisant un filtrage collaboratif et une recommandation basée contenu qu'ont vas détailler dans la suite du chapitre, tout en se basant sur le modèle vectoriel pour la représentation du profil utilisateur.

Dans ce chapitre nous allons définir notre approche. En premier temps on définira les différents modules qui la compose tout en prenant en considération le cas de démarrage à froid qu'on peut considérer comme un module à part dans un système de recommandation.

Problématique :

Dans les systèmes de filtrage d'information, les utilisateurs reçoivent des documents que leur recommande le système sur la base de leurs profils et/ou de leurs communautés. Le profil et les communautés d'un utilisateur évoluent au cours du temps grâce aux interactions entre celui-ci et le système, notamment grâce aux évaluations produites par cet utilisateur. Lorsqu'il s'inscrit et commence à utiliser le système, le problème du « démarrage à froid » se pose, car son profil est encore très pauvre, voire inexistant, et ses communautés sont encore inconnues. Par conséquent, le système ne peut pas lui fournir des recommandations pertinentes**.**

Dans notre travail on se base sur les évaluations faites par les utilisateurs pour les documents, en utilisant des acquisitions de données explicite et implicite pour construire le profil utilisateur .Le profil est utilisé pour recommander à l'utilisateur des documents selon ses préférences.

2- DESCRIPTION DE L'APPROCHE PROPOSÉ

2.1 ARCHITECHTUTE DE L'APPROCHE

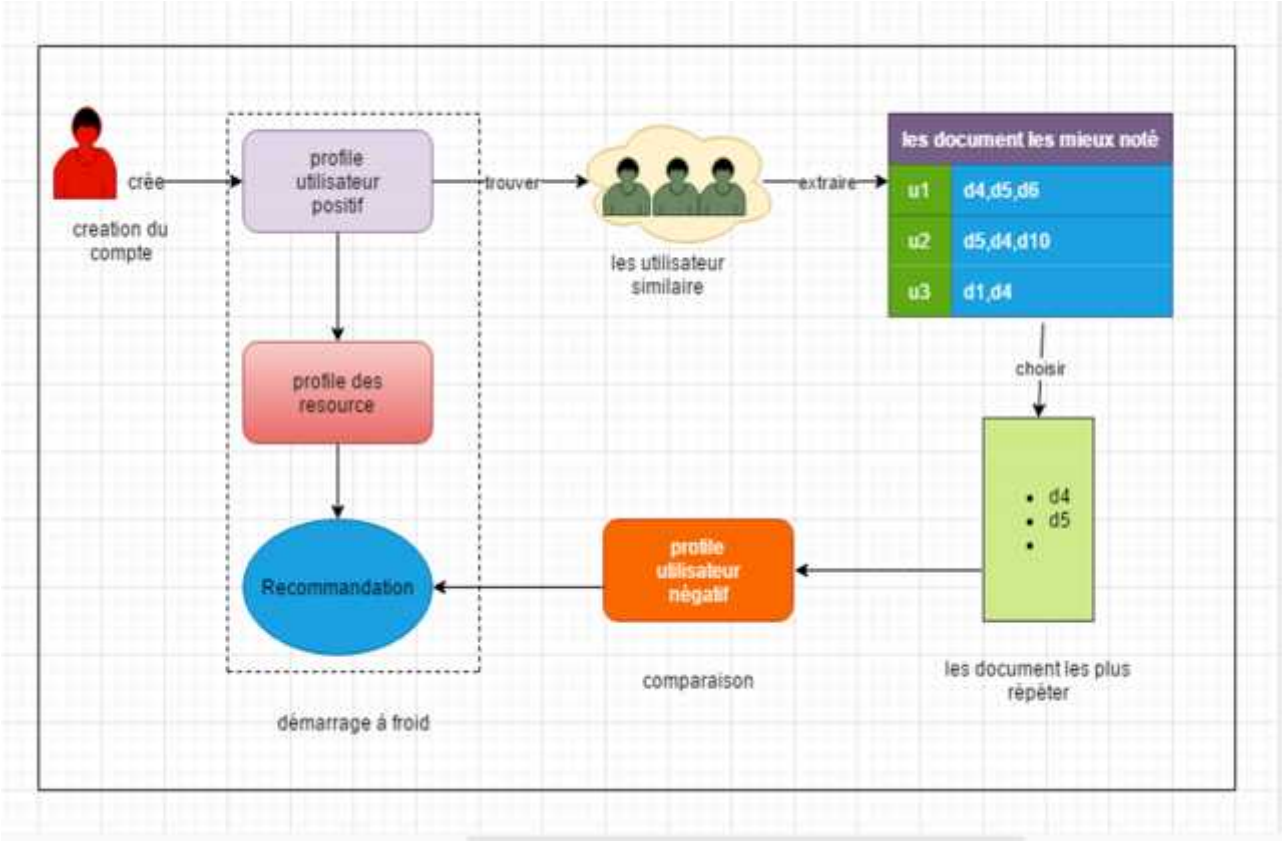


Figure 3.1 : architecture de notre approche.

L'architecture de notre approche se base sur différents modules qui consistent en :

1- le profil d'une ressource.
2- le profil utilisateur.
3- la phase de recherche des utilisateurs similaires qui est la phase d'exploitation du profil utilisateur pour la recommandation.
4- la phase de recommandation qui regroupe :
   - ✓ L'extraction des documents les mieux notés par chacun des utilisateurs similaires.
   - ✓ L'élimination des documents les moins répétés dans toutes ces listes
   - ✓ La comparaison avec le profil négatif de l'utilisateur.
5- Le démarrage à froid.

Dans ce qui suit nous allons représenter ces différents modules respectivement.

2.2 PROFIL D'UNE RESSOURCE.

Le profil d'une ressource dans notre cas est équivalent au profil d'un document et il est représenté par un vecteur de centres d'intérêts pondérés.

Le poids d'un centre d'intérêt j dans une ressource donnée i, noté $W_{i,j}$, est calculé par la formule TF*IDF (Salton et Buckley , 1987) lors de l'indexation des documents .

En plus de ce poids ; le profil d'une ressource contient le nombre d'utilisateurs qui ont apprécié la ressource i, noté **Na** et N_m , qui représente le maximum des valeurs Na .

2.3 PROFIL UTILISATEUR.

Pour la représentation du profil utilisateur nous avons opté pour la représentation vectorielle qui consiste à représenter l'utilisateur sous forme de vecteur de centres d'intérêts pondérés où un centre d'intérêts représente un terme qui as un poids élevé.

On distingue deux types de profils utilisateur dans notre approche :

a. Le profil positif contient les centres d'intérêts liées aux documents auxquels l'utilisateur a attribué des notes positives c'est-à-dire une note entre 3et 5.

b. Le profil négatif contient les centres d'intérêts liées aux documents auxquels l'utilisateur a attribué des notes négatives c'est-à-dire une note de 1 ou 2.

Donc pour la construction de ces profils on prend les notes (de 1 à 5) attribuées par l'utilisateur aux documents.

2.4 EXPLOITATION DU PROFIL EN RECOMMANDATION :

On calcule en premier temps le poids de chaque centre d'intérêt selon la formule suivante :

$$WC_{k=\sum_a} \quad w_{i,j*n_j} \qquad (3.1)$$

Avec :

WC_k^i : qui représente le poids du centre d'intérêt j pour un profil utilisateur donné

nj : Représente la note attribuée par l'utilisateur pour le document j

$w_{i,j}$: représente le poids du document j dans le centre i.

Aprés cette étape on divise chaque poids obtenu sur la somme de tous les poids des centres d'intérêts et le résultat sera stocké dans un vecteur de poids centres d'intérêts pondérés.

En se basant sur le résultat de cette formule, on classe les centres d'intérêts par ordre décroissant de leurs poids dans le profil utilisateur, puis on cherche les utilisateurs qui ont noté approximativement plus de K% (dépend de la puissance de la machine utiliser)de documents en communs avec cet utilisateur et cela pour les trois centres d'intérêts les mieux classés. Les profils de ces utilisateurs seront considérer comme similaires au profil de l'utilisateur en question.

2.5 LE PROCESSUS DE RECOMMANDATION.

2.5.1 EXTRACTION DES DOCUMENTS LES MIEUX NOTÉS.

Après avoir extrait la liste des utilisateurs similaires, une liste des documents les mieux notés pour chacun d'entre eux est extraite,

2.5.2 ELEMINATION DES DOCUMENT LES MOINS RĔPĔTĔS.

Apres avoir extrait la liste des documents les mieux notés pour chacun des utilisateurs similaires, on procède à un filtrage pour éliminer les documents les moins répétés dans l'ensemble des listes extraites.

2.5.3 COMPARAISON AVEC LE PROFIL NEGATIF :

Une fois le filtrage termine on obtient la liste des documents les plus répétés pour tous les utilisateurs et qui seront comparés au profil négatif de l'utilisateur, en utilisant le produit cartésien pour chaque document dj :

$$sim = \sum dj \in D\ wi,j * wi,k \qquad (3.2)$$

Où :

wi,j : représente le poids du centre d'intérêt dans le documents .

wi,k : représente le poids du centre d'intérêt dans le profil utilisateur .

Les documents en communs seront éliminés de la liste et le résultat sera recommandé à l'utilisateur.

2.6.DEMARRAGE A FROID.

a. Démarrage à froid de la ressource :

Lorsqu'un nouveau document est ajouté au système, on calcule avec la mesure de cosinus la liste des documents qui lui sont similaires et chaque utilisateur qui apprécie un document de cette liste lui sera recommandé.

b. Démarrage à froid de l'utilisateur :

Lorsqu'un nouvel utilisateur se connecte au système, aucune information le caractérisant n'est disponible. Pour cela on lui soumet un formulaire disposant d'une liste de centres d'intérêts pour lesquels il attribuera des notes afin d'exprimer ses gouts et préférences.

Puis on prend les trois centres d'intérêts auxquels l'utilisateur a attribué les meilleures notes. Pour chaque document caractérisé par l'un de ces trois centres d'intérêts on récupère les valeurs suivantes:

le poids d'un centre d'intérêt j dans la ressource donnée i, noté $w_{i,j}$ **,** cette valeur est calculée par la formule (2.1).

Le nombre d'utilisateurs qui ont apprécié la ressource i, noté Na, et le maximum des Na, noté N_m .

Puis on calcule une valeur N_f selon la formule :

$$N_f = \frac{N}{N_{ma}} . \qquad (3.3)$$

Pour chaque ressource on calcule une valeur R qui est la moyenne entre $w_{i,j}$ et N_f comme suite :

$$R = \frac{W + N_f}{2} \qquad (3.4)$$

Puis les ressources qui auront la plus grande valeur R, seront recommandées a l'utilisateur.

3- ILLUSTRATION DE L'APPROCHE PROPOSEE :

Pour illustré notre approche, nous avons utilisé une collection de 12 documents {D1,D2,D3,D4,D5,D6,D7,D8,D9,D10,D11,D12} ,avec un ensemble de 5 centres d'intérêts{ sport,tech, politics, entairtainement ,business }.

Le tableau ci-dessus (tableau 3.1) représente l'ensemble des documents auxquels l'utilisateur a attribué des notes positives ou négatives.

	Le document noté par l'utilisateur					
Document	**D1**	**D2**	**D3**	**D4**	**D5**	**D6**
Note de l'utilisateur n_j	3	5	2	1	4	3

Tableau 3.1: représentation des notes de l'utilisateur pour un ensemble de documents

Le tableau3.2 représente les poids des centres d'intérêts dans les documents, pour lesquels l'utilisateur a attribué une note, ces poids sont estimés selon la formule TF-IDF.

Les centres d'intérêt	**Le document noté par l'utilisateur**					
	D1	**D2**	**D3**	**D4**	**D5**	**D6**
Business	0.6	0.8	0	0	0	0
entertainment	0.1	0	0	0	0	0.6
Politics	0	0	0.1	0.8	0	0
Sport	0.15	0	0.6	0.1	0	0
Tech	0.1	0	0	0	0.9	0

Tableau 3.2 : le poids des centres d'intérêts dans les documents estimés selon la formule (2.6)

Le profil positif :

On récupère la liste des document {D1 ,D2,D5,D6 }apprécie par l'utilisateur et On calcule les poids des centres d'intérêts dans ces documents ,définissant ainsi le profil positif de l' utilisateur:

Le tableau 3.3 représente le poids des centres d'intérêts dans le profil positif de l'utilisateur et leurs pourcentages correspondants.

Centre	Poids	Pourcentage
business	5.8	47%
Tech	3.9	31%
Entertainment	2.1	17%
Sport	0.45	3%
politics	0	0%

Tableau 3.3: Poids et pourcentages des centres d'intérêts dans le profil positif.

Le profil négatif :

On récupère la liste des documents que l'utilisateur n'apprécie pas et on calcule le poids de chaque centre d'intérêt en suivant les mêmes étapes que le profil positif.

Le tableau 3.4 représente alors les poids des centres d'intérêts dans le profil négatif de l'utilisateur et leurs pourcentages correspondants.

Centre	Poids	Pourcentage
Sport	1.3	56%
Politics	1	43%
Business	0	0%
Tech	0	0%
Entertainment	0	0%

Tableau 3.4: poids et pourcentage des centres d'intérêts dans le profil négatif.

On ordonne les centres d'intérêts selon leurs pourcentages dans le profil positif, et on déroule l'approche de recommandation dans notre cas que pour le centre d'intérêt ayant le pourcentage le plus élevé. Le centre résultant correspond à 'business'.

On cherche les utilisateurs similaires dont le profil est lié à ce centre, puis on récupère les documents qui sont bien notés par ces utilisateurs.

Le tableau 3.5 représente ces documents avec les notes des utilisateurs.

	D10	**D11**	**D8**	**D12**	**D9**	**D7**
Utilisateur2	4	3	0	4	4	3
Utilisateur5	3	0	5	0	0	5
Utilisateur8	3	4	0	5	4	
Utilisateur11	0	3	0	4	0	5
Utilisateur9	4	0	0	4	0	5
Utilisateur3	5	0	0	4	0	3

Tableau 3.5 : liste des documents les mieux notés par des utilisateurs similaires et dont le profil correspond au centre d'intérêts business.

On récupère les document les plus répétés dans ses listes, à savoir D10,D12,D7.

Le tableau 3.6 représente les poids des centres d'intérêts dans les documents ces documents D10, D12,D7.

Les centres d'intérêt	**Le document retourné**		
	D10	**D12**	**D7**
Sport	0	0	0
Politics	0	0.1	0
Business	0.7	0.9	0.6
Tech	0.2	0	0
Entertainment	0	0	0.4

Tableau 3.6 : poids des centres d'intérêts dans les documents retournés.

On calcule la similarité entre le profile négatif et chacun des documents D10, D12, D7 avec la formule (3.2), les résultats obtenus montre qu'il n y a aucune similarité entre le profil négatif et le centre d'intérêts business.

On constate donc que les documents D7, D10, D12 ne sont pas similaires avec le profil négatif donc ils seront recommander à l'utilisateur.

3.1 Illustration dans le cas de démarrage à froid :

On propose un formulaire de centre d'intérêt pour l'utilisateur et on récupère les notes attribuées aux centres d'intérêts proposés .On calcule le pourcentage correspondant à ces notes et on déduit le profil utilisateur positif.

Le tableau 3.7 représente les notes et les pourcentages obtenus pour le profil positif.

Centre	Note	Pourcentage
Sport	1	07%
Politics	4	30%
Business	5	38%
Tech	2	15%
Entertainment	1	07%

Tableau 3.7: notes et pourcentage du profil positif .

Pour chaque centre d'intérêt, on calcule la valeur de Na_final qui représente et de R qui représente.

Les valeurs obtenues pour l'exemple de centre d'intérêts sport sont représenté dans le tableau 3.8 qui suit :

	Sport		
	Poids	**Na_final**	**R**
D4	0.11	0.22	0.165
D5	0.17	0.15	0.16
D9	0.7	0.16	0.43
D8	0.6	0.44	0.52
D1	0.65	0.45	0.55
D7	0.53	0.13	0.33
D6	0.8	0.16	0.48

Tableau 3.8: les valeurs du poids, Na_final et de R correspondantes aux documents liées au centre d'intérêt 'sport '.

On répète les mêmes étapes pour chacun des deux autres centres d'intérêts qui ont le pourcentage le plus élevé à savoir 'politics' et 'tech'. Puis on récupère les quatre documents qui ont la meilleure valeur de R, les documents D1, D6, D8 est le résultat de la recommandation pour le centre d'intérêts' sport'.

4- CONCLUSION.

Dans ce chapitre nous avons présenté notre approche de recommandation, tout d'abord nous avons illustré son architecture. Ensuite nous avons précisé les différents modules qui la composent.

Le chapitre suivant porte sur l'implémentation de notre approche.

CHAPITRE IV

Implémentation

CHAPITRE IV
Evaluation et validation expérimentale de l'aproche proposée

1- INTRODUCTION

Dans ce chapitre, nous présentons l'implémentation de notre approche ainsi que les différents outils que nous avons exploités et enfin nous présentons les résultats obtenus.

2- ENVIRONNEMENT DE DEVELOPPEMENT

Le langage Java

Le langage Java est un langage de programmation informatique orienté objet créé par James Gosling et Patrick Naughton, employés de Sun Microsystems, avec le soutien de Bill Joy (cofondateur de Sun Microsystems en 1982), présenté officiellement le 23 mai 1995 au SunWorld.

La société Sun a été ensuite rachetée en 2009 par la société Oracle qui détient et maintient désormais Java.

La particularité et l'objectif central de Java est que les logiciels écrits dans ce langage doivent être très facilement portables sur plusieurs systèmes d'exploitation tels que UNIX, Windows.

Java EE

Le terme « Java EE » signifie Java Enterprise Edition, et était anciennement raccourci en « J2EE ». Il fait quant à lui référence à une extension de la plate-forme standard. Autrement dit, la plate-forme Java EE est construite sur le langage Java et la plate-forme Java SE, et elle y ajoute un grand nombre de bibliothèques remplissant tout un tas de fonctionnalités que la plate-forme standard ne remplit pas d'origine. L'objectif majeur de Java EE est de faciliter le développement d'applications web robustes et distribuées, déployées et exécutées sur un serveur d'applications.

L'IDE Eclipse

C'est un outil puissant, gratuit, libre et multiplateforme. Les avantages d'un IDE dans le développement d'applications web Java EE sont multiples, et sans toutefois être exhaustif en voici une liste :

- Intégration des outils nécessaires au développement et au déploiement d'une application ;
- Paramétrage aisé et centralisé des composants d'une application ;
- Multiples moyens de visualisation de l'architecture d'une application ;
- Génération automatique de portions de code ;
- Assistance à la volée lors de l'écriture du code ;
- Outils de débogage…

Le serveur Tomcat :

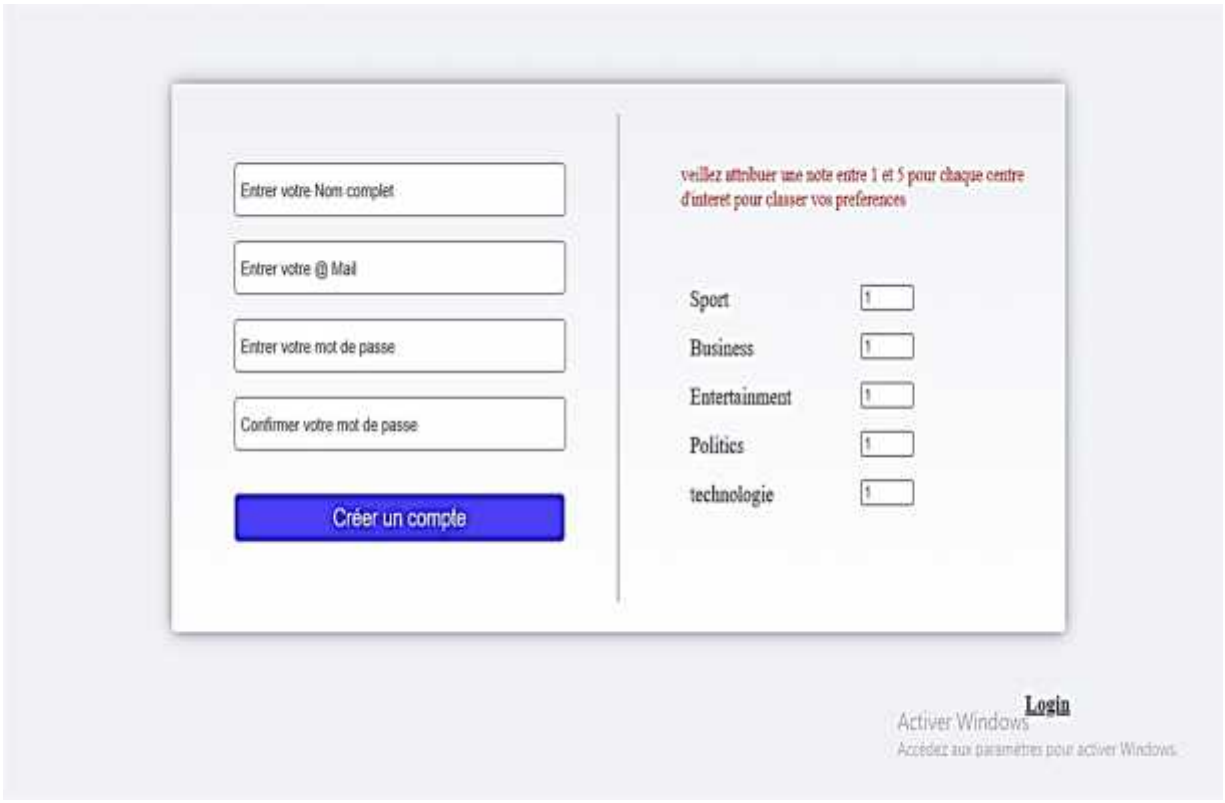
Pour faire fonctionner une application web Java EE, nous avons besoin de mettre en place un serveur d'applications. Il en existe beaucoup sur le marché : nous avons choisi d'utiliser Tomcat, car c'est un serveur léger, gratuit, libre, multiplateforme et assez complet pour ce que nous allons aborder. On le rencontre d'ailleurs très souvent dans des projets en entreprise, en phase de développement comme en production.

phpMyAdmin

Est un outil logiciel libre écrit en PHP , destiné à gérer l'administration de MySQL sur le Web. phpMyAdmin prend en charge un large éventail d'opérations sur MySQL et MariaDB. Les opérations fréquemment utilisées (gestion des bases de données, des tables, des colonnes, des relations, des index, des utilisateurs, des autorisations, etc.) peuvent être effectuées via l'interface utilisateur, tandis que vous avez toujours la possibilité d'exécuter directement toute instruction SQL.

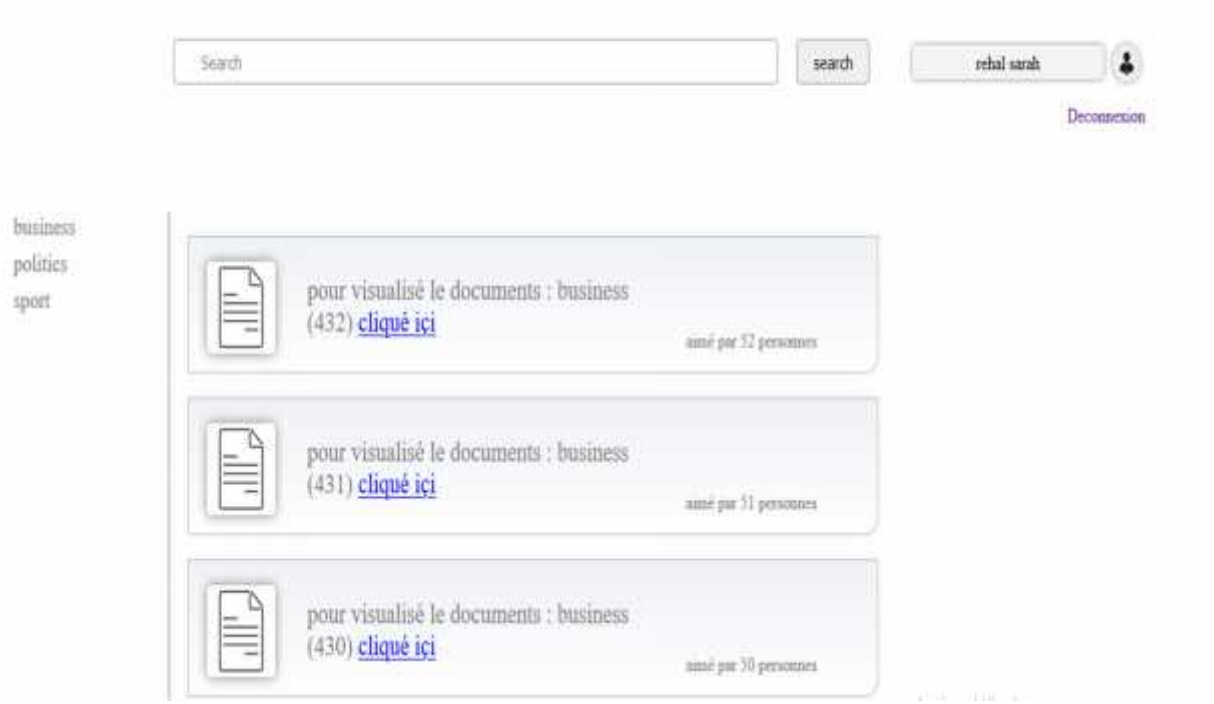
3- SCENARIOS D'EXECUTION AVEC CAPTURES D'ECRANS

Création d'un compte utilisateur

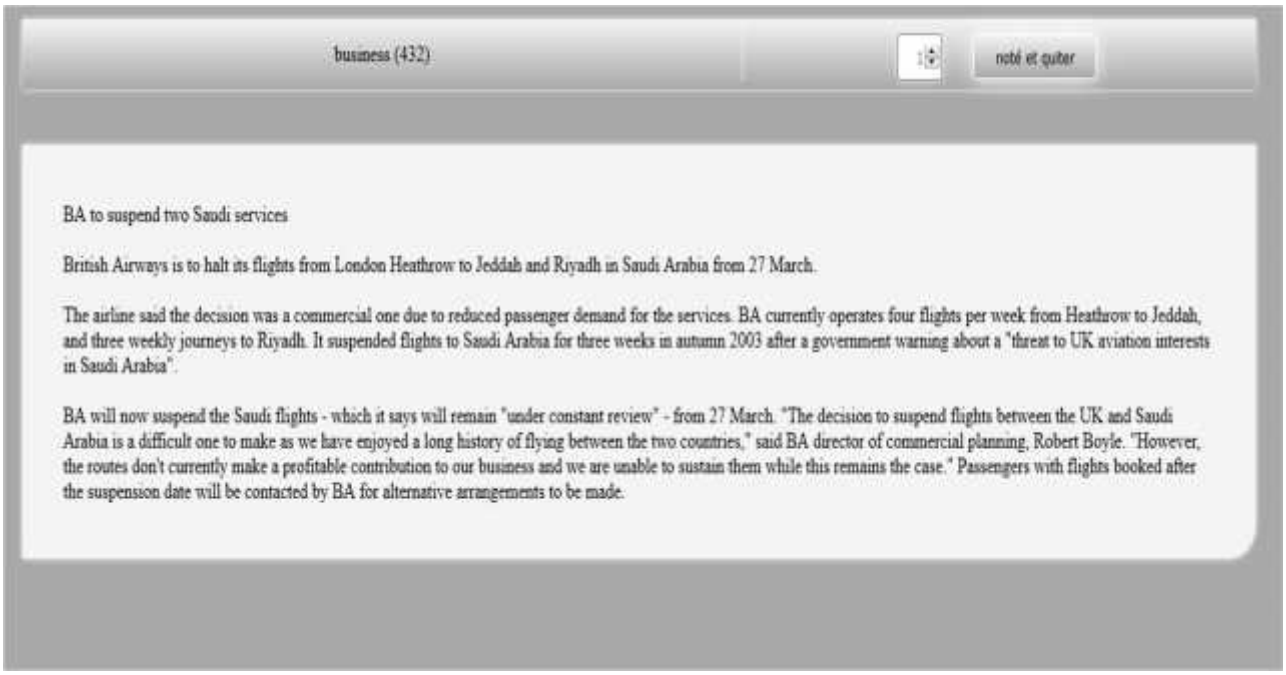


Connexion a un compte :

La liste des documents recommandés par le système lors du démarrage a froid pour un nouvel utilisateur



Visualisation du contenu d'un document



Interface administrateur pour l'ajout d'un nouveau document

Interface administrateur pour l'ajout d'un nouveau centre d'intérêt

4- CONCLUSION

Dans ce chapitre, nous avons présenté la réalisation de notre système en commençant par l'exploration des outils utilisés pour son implémentation, ensuite la description du système à travers différentes captures d'écran.

Conclusion générale :

Dans notre travail nous nous sommes intéressés à l'étape de construction du profil utilisateur pour le système de recommandation. L'objectif principal de ce système est d'optimisé le nombre de documents pertinents que l'on recommande à l'utilisateur.

Nous avons présenté les différentes étapes de modélisation de notre profil ainsi que son implémentation pour le système de recommandation et on a vu l'intérêt que l'on peut avoir en utilisant une hybridation mixée pour la recommandation de documents.

Comme perspectives à notre travail, nous envisageons :

- ✓ D'enrichir le profil utilisateur en utilisant le modèle multidimensionnel
- ✓ D'intégrer le facteur de temps dans la représentation du profil utilisateur
- ✓ De pouvoir traiter des flux d'informations plus important.
- ✓ D'étaler la recommandation sur d'autre type de documents (media, page web ….).

Conclusion Générale

Références bibliographiques

REFERENCES BIBLIOGRAPHIQUES

- ABERNETHY, F. BACH, T. EVGENIOU et J.-P. VERT : A New Approach to Collaborative Filtering : Operator Estimation with Spectral Regularization. *The Journal of Machine Learning*
  Research, 10:803–826, 2009. ISSN 1532-4435.
- Burk R. (2002). Burke, R. (2002).Hybrid recommender systems: Survey and experiments.
  User Modeling and User-Adapted Interaction,
- Rendle et al. (2009). Bayesian personalized ranking from implicit feedback. In Proc. of the 25th Conf. on UAI. pp. 452-461.
- Lee et al. (2008). Lee, T. Q., Park, Y., & Park, Y. T. (2008). A time-based approach to
  Effective recommender systems using implicit feedback. ESA, 34(4), pp. 3055-3062.
- Ouard et al. (1998). Oard, D. W., & Kim, J. (1998, July). Implicit feedback for recommender systems. In Proc. of the AAAI workshop on recommender systems, pp. 81-83.
- Kelly et all. (2003). implicit feedback for inferring user preference: a bibliography. In ACM SIGIR Forum, Vol. 37(2), pp. 18-28. ACM.
- Bel et al, B. (2007). Bell, R. M., & Koren, Y.Modeling Relationships at Multiple Scales to Improve Accuracy of Large Recommender Systems. In Proc. of the 13th ACM SIGKDD, Inter. Conf. On Knowledge.
- Adomavicius, G., Sankaranarayanan, R,Sen, S., and
  Tuzhilin, A. (2005). Incorporating Contextual Information in Recommender Systems Using a
  Multidimensional.
- Ahu Sieg, Bamshad Mobasher, and Robin Burke. Web search
  personalization with ontological user profiles. ALTHOFF : Case-Based Reasoning. volume 1, pages 549–588, 2001
- BREESE, D. HECKERMAN et C. KADIE : Empirical Analysis of Predictive Algorithms for
  Collaborative Filtering. pages 43–52, 1998
- CASTAGNOS : *Modélisation de comportements et apprentissage stochastique non supervisé de stratégies d'interactions sociales au sein de systèmes temps réel de recherche et d'accès à l'information*.
  Thèse de doctorat, Université Nancy 2, 2008

- CHEESEMAN et J. STUTZ : Bayesian Classification (AutoClass) : Theory and Results. pages
  153–180, 1996
- CHICKERING, D. HECKERMAN et C. MEEK : A Bayesian Approach to Learning Bayesian
  Networks with Local Structure. *In In Proceedings of Thirteenth Conference on Uncertainty in*
  Artificial Intelligence. Morgan Kaufmann, 1997
- Chumki Basu, Haym Hirsh, and William Cohen. Recommendation as classifi cation : Using social and content-based information in recommendation. In *Proceedings of the Fifteenth National Conference on Artificial Intelligence*, pages 714{720. AAAI Press, 1998
- D. Kelly and J. Teevan. Implicit feedback for inferring user preference : a bibliography. *SIGIR Forum*, 37(2) :18–28, 2003
- D. Kostadinov. La personnalisation de l'information,dØfinition de modŁle de profil utilisateur. rapport de dea. Master's thesis, UniversitØ de Versailles, France, 2003
- Daniel Billsus and Michael J. Pazzani. User modeling for adaptive news access. *User Modeling and User-Adapted Interaction*, 10(2-3) :147{180, 2000
- DAS, M. DATAR, A. GARG et S. RAJARAM : Google News Personalization : Scalable Online
  Collaborative Filtering. *WWW'07 : Proceedings of the 16th International Conference on World*
  Wide Web, pages 271–280, 2007
- DEMPSTER, N. LAIRD et D. RUBIN : Maximum Likelihood from Incomplete Data via the EM
  Algorithm. *Journal of the Royal Statistical Society. Series B (Methodological)*, 39(1):1–38, 1977
- ESTER, H. KRIEGEL, S. JÖRG et X. XU : A Density-Based Algorithm for Discovering Clusters in Large Spatial Databases with Noise. *In Proceedings of the International Conference on*
  Knowledge Discovery and Data Mining (KDD), pages 226–231, 1996.
- F. Tanudjaja and L. Mui. Persona : A contextualized and personalized web search. In *Proc 35th Hawaii International Conference on System Sciences*, page 53, Big Island, Hawaii, January 2002.

- freitag et al. (1995). D Freitag, R Armstrong, T Joachims, T Mitchell Web watcher: A learning apprentice for the word wide web
- Gracia et al. (2015). Data processing in data mining.
- G. Amato and U. Staraccia. User profile modelling and applications to digital librairies. In *Proceedings of the 3rd European Conference on Research and avanced technology for digital libraries*, pages 184–187, 1999
- G. Salton and M. McGill. *Introduction to Modern Information Retrieval*. McGrawHill, New York, 1983.
- GOLDBERG, D. NICHOLS, B. OKI et D. TERRY : Using Collaborative Filtering to Weave an
  Information Tapestry. *Communications of the ACM*, pages 61–70, 1992
- HAN et M. KAMBER : *Data Mining : Concepts and Techniques (Second Edition)*. Morgan Kaufmann, second édition, 2006
- HORNIK : Some New Results on Neural Network Approximation. *Neural Networks*, 6(8):
  1069–1072, 1993
- HUANG et T. JEBARA : Collaborative Filtering via Rating Concentration. *In Thirteenth International Conference on Artificial Intelligence and Statistics (AISTATS)*, pages 334–341, 2010
- J. Chaffee and S. Gauch. Personal ontologies for web navigation. In *Proceedings of the ninth international conference on Information and knowledge management*, pages 227–234, McLean, Virginia, United States, 2000. ACM Press
- KALMAN et S. KWASNY : Why tanh : Choosing a Sigmoidal Function. *In Neural Networks*,
  1992. IJCNN., International Joint Conference on, volume 4, pages 578–581, 1992
- Kobsa. Privacy-enhanced web personalization. In *The Adaptive Web : Methods and Strategies of Web Personalization, Lecture Notes in Computer Science*, volume 4321, Berlin Heidelberg New York, 2007. In Brusilovsky, Kobsa, P. and Nejdl, A. W. (eds.), Springer-Verlag
- KOREN : Collaborative Filtering with Temporal Dynamics. *In KDD '09 : Proceedings of the 15th*
  ACM SIGKDD international conference on Knowledge discovery and data mining, pages 447–456,
  New York, NY, USA, 2009. ACM. ISBN 978-1-60558-495-9

- lousan et al. (2009). Fabián P. Lousame and Eduardo Sánchez, a taxonomy of collaborative
  based recommender system in web personalisation in intelligent environnement.
  mastchoff. (2011). Combining individual models
- M PAZZANI et D. BILLSUS : Content-Based Recommendation Systems. *The Adaptive Web*, pages
  325–341, 2007.
- MACQUEEN : Some Methods for Classification and Analysis of MultiVariate Observations.
  In L. M. Le CAM et J. NEYMAN, éditeurs : *Proc. of the fifth Berkeley Symposium on Mathematical*
  Statistics and Probability, volume 1, pages 281–297. University of California Press, 1967.
- Micarelli and F. Sciarrone. Anatomy and empirical evaluation of an adaptive webbased information filtering system. *User Modeling and User-Adapted Interaction*,
  14(2-3) :159–200, June 2004
- Micarelli and F. Sciarrone. Anatomy and empirical evaluation of an adaptive webbased information filtering system. *User Modeling and User-Adapted Interaction*,
  14(2-3) :159–200, June 2004.
- Michael D. Ekstrand, Daniel Kluver, F. Maxwell Harper, and Joseph A. Konstan. Letting users choose recommender algorithms : An experimental study. In *Proceedings of the 9th ACM Conference on Recommender Systems*, RecSys ’15, pages
  11{18, New York, NY, USA, 2015. ACM
- Michael J. Pazzani. A framework for collaborative, content-based and demographic filtering. *Artif. Intell. Rev.*, 13(5-6) :393-408, 1999
- Micro speretta et al. (2004). Mirco Speretta and Susan Gauch. personalizing search based on
  user search histories
- MIYAHARA et M. PAZZANI : Improvement of Collaborative Filtering with the Simple Bayesian Classifier. *Transactions of Information Processing Society of Japan*,
  43(11):3429–3437,
  2002.
- montaner et al. (2003). *Miquel Montaner, Beatriz Lòpez, and JosepLluís de la Rosa, A taxonomy of recommander agents on the internet*

- NGUYEN, N. DENOS et C. BERRUT : Exploitation des données disponibles à froid pour
  améliorer le démarrage à froid dans les systèmes de filtrage d'information. *In INFormatique*
  des ORganisations et Systèmes d'Information et de Décision, pages 81–95, 2006
- Nicolas Ducheneaut, Kurt Partridge, Qingfeng Huang, Bob Price, Mike Roberts, Ed H. Chi, Victoria Bellotti, and Bo Begole. Collaborative filtering is not enough ? experiments with a mixed-model recommender for leisure activities. In *Proceedings of the 17th International Conference on User Modeling, Adaptation, and Personalization (UMAP)*, pages 295{306, 2009}.
- Pazzani et al. (1996). M Pazzani, J Muramatsu, and D Billsus Syskill & Webert : Identifyinginteresting web sites.
- Pazzani et al. (2007). Pazzani, M. J. and Billsus, D. (2007).Content-based Recommandation
  systems.In the adaptative web
- PORTER : An Algorithm for Suffix Stripping. pages 313–316, 1997
- Prem Melville, Raymond J. Mooney, and Ramadass Nagarajan. Content-boosted collaborative filtering for improved recommendations. In *Eighteenth National Conference on Artificial Intelligence*, pages 187-192, 2002
- RESNICK, N. IACOVOU, M. SUCHAK, P. BERGSTROM et J. RIEDL : GroupLens : An Open
  Architecture for Collaborative Filtering of Netnews. *In CSCW '94 : Proceedings of the 1994*
  ACM conference on Computer supported cooperative work, pages 175–186, New York, NY, USA,
  1994. ACM
- Robin Burke. Hybrid recommender systems : Survey and experiments. *User Modeling and User-Adapted Interaction*, 12(4) :331{370, 2002}
- salton et al. (1973) . G. Salton and C. Yan, On the specification of termes values in automatic
  indexing.
- SALTON et C. BUCKLEY : Term Weighting Approaches in Automatic Text Retrieval. Rapport
  technique, Ithaca, NY, USA, 1987

- salton. (1971). The SMAT retrieval system: expirements in automatic document processing
- SCHAFER, D. FRANKOWSKI, J. HERLOCKER et S. SEN : *Collaborative Filtering Recommender*
  Systems, chapitre 9, pages 291–324. Peter Brusilovsky, Alfred Kobsa and W Nejdl. Springer,
  2007
- SCHMITT et R. BERGMANN : Applying Case-Based Reasoning Technology for Product Selection and Customization in Electronic Commerce Environments. *In 12th Bled Electronic*
  Commerce Conference, pages 1–15, 1999
- SMYTH : Case-Based Recommendation. *The Adaptive Web*, pages 342–376, 2007
- Somlo et al. (2003). G. Somlo and A. Howe, Using web helper agent profiles in query generation l.
- Stefani and C. Strappavara. Personalizing access to web sites : The siteif project. In *Proceedings of the 2nd Workshop on Adaptive Hypertext and Hypermedia*, Pittsburgh, June 20-24 1998
- STOLZE et Walid R. : Towards Scalable Scoring for Preference-based Item Recommendation.
  IEEE Data Engineering Bulletin, 24:42–49, 2001
- Struarte et al. (2003). Stuart E. Middleton, Nigel R. Shadbolt ,and David C. De Roure, Capturing knowledge of user preferences.
- stuarte et al. (2001). Stuart E. Middleton, David C. De Roure, and Nigel R. Shadbolt, Ontologies in recommender systems.
- stuarte et al. (2001). Stuart E. Middleton, David C. De Roure, and Nigel R. Shadbolt, Ontologies in recommender systems.
- SU et T. KHOSHGOFTAAR : Collaborative Filtering for Multi-class Data Using Belief Nets
  Algorithms. *In ICTAI '06 : Proceedings of the 18th IEEE International Conference on Tools with*
  Artificial Intelligence, pages 497–504, Washington, DC, USA, 2006. IEEE Computer Society

- Tebri et al. (2005). H. Tebri, M. Boughanem, and C. Chrisment.Incremantal profile learning
  based on a reinforcement method.
- TOWLE et C. QUINN : Knowledge Based Recommender Systems Using Explicit User Models.
  pages 74–77, 2000
- Tuzhilin, A. (2005). Incorporating Contextual Information in Recommender Systems Usinga
  Multidimensional.
- Vistu kanth. (2004). Vishnu Kanth Reddy Challam, Contextual information retreival using
  based user profiles.
- W. N. Zemirli, L. Tamine, and M. Boughanem. PrØsentation et Øvaluation d'un modŁle d'accŁs personnalisØ à l'information basØ sur les diagrammes d'influence. In *CongrŁs Informatique des Organisations et SystŁmes d'Information et de DØcision (INFORSID 2007)*, pages 75–86, mai 2007
- Zhang et al. (2002). Zhang, Y., Callan, J., and Minka, T. (2002).Novelty and redundancy -detection in adaptive filtering .In Processings of the 25th annual international ACM SIGR
  conference on Research and development in information retrievel
- ZHANG, R. RAMAKRISHNAN et M. LIVNY : BIRCH : An Efficient Data Clustering Method
  for Very Large Databases. *In ACM SIGMOD International Conference on Management of Data*,
  pages 103–114, 1996