REPUBLIQUE ALGERIENNE DEMOCRATIQUE ET POPULAIRE
MINISTER DE L'ENSEIGNEMENT SUPERIEUR ET DE LA RECHERCHE SCIENTIFIQUE

UNIVERSITE MOULOUD MAMMERI DE TIZI-OUZOU
FACULTE DE GENIE ELECTRIQUE ET DE L'INFORMATIQUE
DEPARTEMENT INFORMATIQUE

Mémoire

De fin d'études

En vue de l'obtention du diplôme de master en Informatique
Option: Systèmes informatiques

Thème

Un système de recommandation adapté à l'e-learning, basée sur l'analyse des sentiments et l'analyse des réseaux sociaux

Proposé et dirigé par :
Mr AHEMD-OUAMER Rachid

Réalisé par :
Melle LAZIB Lydia
Melle HACID Kahina

2012/2013

Table des matières

Table des figures

Liste des tableaux

Résumé

Pour faire une recommandation au sujet d'une entité cible donnée, un système de recommandation regroupe les avis ou opinions des utilisateurs sur Internet. En général, de bons avis donnent une recommandation positive et de mauvaises opinions donnent une recommandation négative.

Il s'agit ici de prendre en compte les avis des utilisateurs sur Internet pour aider l'apprenant à prendre sa décision. Par exemple, si un grand nombre d'utilisateurs a trouvé un service (une ressource pédagogique) intéressant, on peut s'attendre à ce que l'apprenant le trouve également intéressant.

Objectif

L'objectif du sujet est de développer un système de recommandation adapté à l'e-learning qui utilise l'analyse des sentiments et l'analyse des réseaux sociaux pour évaluer la recommandation d'une ressource pédagogique cible. Ce système de recommandation permettra de récupérer les opinions des apprenants sur plusieurs sites et prendra en considération les avis des apprenants sous forme quantitative et sous forme textuelle.

L'analyse des sentiments sera utilisée pour déterminer si une opinion donnée sous forme textuelle est positive ou négative. Le système de recommandation utilisera l'analyse des réseaux sociaux pour personnaliser les recommandations en fonction des préférences de l'apprenant et de ses activités.

Mots clés

système de recommandation, analyse des sentiments, réseaux sociaux, ontologies, e-learning, modèle elgve.

Introduction générale

L'essor connu ces dernières années par le web, ainsi que l'augmentation considérable du volume de ressources disponible sur la toile, font que les utilisateurs sont souvent confrontés à passer des heures à faire de longues recherches, et qui au final n'aboutissent pas au résultat escompté. Ce problème a mené à l'émergence d'un domaine de recherche consacré aux systèmes de recommandation.

Les solutions apportées consistent la plupart du temps à analyser le comportement des utilisateurs sur des sites web, afin de déduire leurs préférences et centres d'intérêts, pour ensuite leur recommander des ressources susceptibles de satisfaire leurs besoins, et ainsi leur éviter l'encombre des longues recherches sur le web.

L'objectif de notre travail est donc de développer un système de recommandation adapté à l'e-learning, qui utilisera l'analyse des sentiments et l'analyse des réseaux sociaux, afin de recommander des ressources pédagogiques. Ce système de recommandation permettra de récupérer les opinions des apprenants laissés à propos de ressources, en prenant leurs avis sous forme quantitative ou alors sous forme textuelle. L'analyse des sentiments sera utilisée pour déduire la note correspondante à une opinion donnée sous forme textuelle, pendant que l'analyse des réseaux sociaux sera utilisée pour personnaliser les recommandations en fonction des préférences et activités de l'apprenant.

Organisation du mémoire

Pour bien structurer notre travail, nous l'avons élaboré en trois chapitres, comme suit :

- **Le premier** chapitre présente un état de l'art des différents aspects abordés tout au long de notre travail, à savoir l'e-learning, les réseaux sociaux, le web sémantique, les ontologies, ainsi que les systèmes de recommandation.
- **Le second** chapitre, est consacré à l'analyse de différentes méthodes existantes uti-

lisées pour réaliser un système de recommandation basée sur une analyse de sentiments et une analyse de réseaux sociaux. Nous présentons, ensuite,l'approche finalement choisie pour la réalisation de notre application, accompagnée de sa conception.

– **Le troisième** et dernier chapitre concerne la présentation de l'implémentation de notre système de recommandation, illustrée par les interfaces de l'application développée.

Enfin, nous concluons ce travail en résumant nos principales contributions et les perspectives de cette étude.

Chapitre 1

Etat de l'art

1.1 Introduction

L'objectif de cet état de l'art est d'introduire les notions relatives aux différents aspects abordés tout au long de notre recherche, qui sont notamment, le e-learning, les réseaux sociaux, le web sémantique, les ontologies et les systèmes de recommandation. Nous ferons un tour d'horizon sur chacune de ces notions afin de pouvoir constituer une idée générale de l'approche à adopter.

1.2 E-learning

1.2.1 Définition

Terme anglophone pour e-formation ; E-learning désigne l'utilisation des nouvelles technologies multimédias et de l'Internet pour améliorer la qualité de l'apprentissage en facilitant l'accès à des ressources et des services, ainsi qu'aux échanges et à la collaboration à distance. L'e-learning définit également tout dispositif de formation qui utilise un réseau local, étendu ou l'Internet pour diffuser, interagir ou communiquer ; ceci inclut l'enseignement à distance en environnement distribué, l'accès à des ressources par téléchargement ou en consultation sur le net [13].

1.2.2 Modes et outils de communication en e-learning

Suivre une formation en e-learning ne signifie pas être seul face à son ordinateur, sans personne pour échanger sur les concepts abordés au cours de la formation ou pour nous apporter un support technique ou pédagogique. En effet, il existe de nombreuses possibilités de communication en e-learning, différentes de celles que l'on utilise en formation en présentiel.

Il existe deux modes de communications en e-learning : synchrone et asynchrone [8].

1. **Le mode synchrone** : permet de communiquer en temps réel avec le tuteur et/ou les autres participants de la formation, l'interactivité est aussi immédiate qu'en présentiel. Pour cela, il est possible d'utiliser entre autres, les éléments suivants : le chat, la Webcam, le micro, le partage d'applications, les fonctionnalités de prise en main à distance, le tableau blanc, le téléphone, etc.
2. **Le mode asynchrone** : permet de communiquer en temps différé avec le tuteur et/ou les autres participants de la formation. Pour cela, il est possible d'utiliser entre autres, les éléments suivants : le forum, l'E-mail, le partage de documents, etc.

1.2.3 Plates-formes e-learning

1.2.3.1 Définition

Une plate-forme d'e-learning est un logiciel qui assiste la conduite des formations ouvertes et à distance. Elle est basée sur des techniques de travail collaboratif et regroupe les outils nécessaires aux principaux acteurs de la formation. Elle fournit à chaque acteur un dispositif qui a pour première finalité l'accès à distance au contenu pédagogique, l'auto apprentissage, l'auto-évaluation et le télé-tutorat via l'utilisation de moyens de travail et de communication à plusieurs : visioconférence, e-mail, forums, chats, annotations, tableaux blancs partagés, etc. Le but est donc de combler la perte de la cohésion et de la stimulation de la salle que peut ressentir l'apprenant devant sa machine [5].

1.2.3.2 Les acteurs d'une plateforme e-learning et leurs rôles

La plate-forme d'e-learning comporte trois acteurs principaux : l'apprenant, le tuteur et l'administrateur. Un rôle différent est attribué à chacun de ces acteurs ; Ainsi, le tuteur crée des parcours de formation type, incorpore des ressources pédagogiques multimédias et

de suivi des activités des apprenants. L'apprenant, peut consulter en ligne ou télécharger les contenus pédagogiques qui lui sont recommandés, effectuer des exercices, s'auto-évaluer et transmettre des travaux à son tuteur pour les corriger. La communication entre apprenant et tuteur peut être individuelle ou en groupe. Il est possible de créer des thèmes de discussion et collaborer à des travaux communs en utilisant des moyens de travail et de communication à plusieurs [5].

L'administrateur, de son côté, assure l'installation et la maintenance du système, gère les droits d'accès, crée des liens vers d'autres systèmes et ressources externes. Ainsi, une plate-forme peut comporter des fonctionnalités relatives à la gestion des compétences, à la gestion des ressources pédagogiques, à la gestion de la qualité de la formation [5].

1.2.3.3 Architecture d'une plateforme e-learning

Les plates-formes e-learning intègrent des outils pour les différents acteurs, l'objectif étant de faciliter les rôles et les fonctions tenus par chacun des acteurs décrits précédemment.

La figure ci-dessous décrit un modèle de plate-forme e-learning avec quatre acteurs, sachant que les plates-formes ne prennent pas toutes en compte ce modèle, certaines fusionnent le rôle de l'auteur et du formateur pendant que d'autres au contraire intègrent d'autres acteurs pour compléter encore plus l'architecture [13].

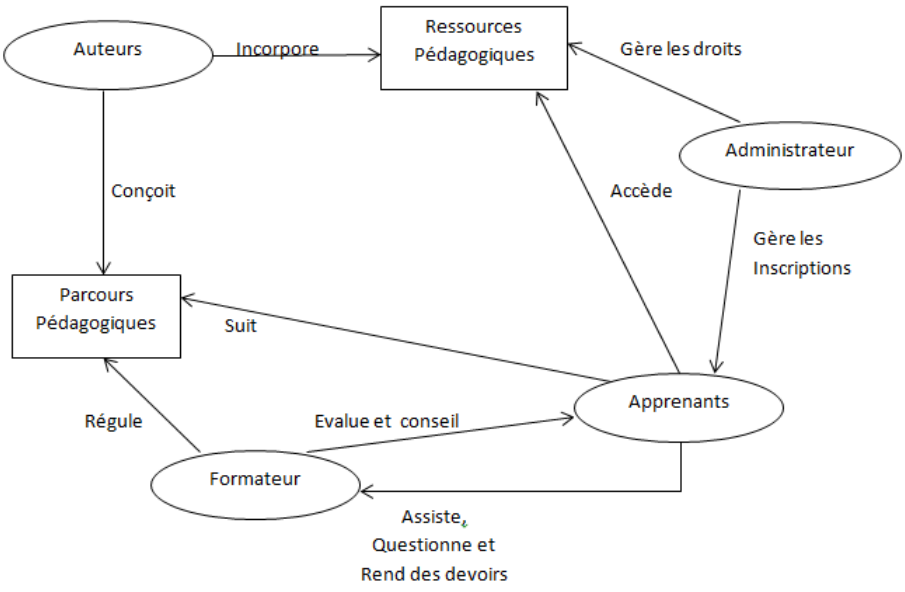

FIGURE 1.1: Exemple d'architecture de plateforme e-learning.

1.2.3.4 Quelques plates-formes

Avec l'évolution des techniques, des infrastructures de réseau et des normes, le nombre de plates-formes et environnements de formation a augmenté de manière significative et assez rapide. A chaque contexte de formation peut correspondre un ensemble de fonctionnalités adaptées et donc une plate-forme potentielle.

Le nombre de ces plates-formes dépasse les 500 de nos jours, les plus répandues sont [13] :

1. **Plates-formes propriétaires**
   - WebCT
   - Black Board
   - Top Class
   - Apex Learning
   - ANGEL Learning
2. **Plates-formes libres (Open Source)**
   - ATutor
   - Moodle
   - Dorkeos
   - Claroline
   - Sakaï Project

a) **La plate-forme Moodle**

Moodle est une plate-forme permettant la mise en place de cours et de sites web en ligne. C'est un projet bénéficiant d'un développement actif et conçu pour favoriser un cadre de formation socioconstructiviste. Moodle est mis à disposition gratuitement en tant que logiciel libre, suivant la licence GPL. Ses caractéristiques sont intéressantes tant pour l'auteur d'un cours que pour l'apprenant qui s'y inscrit. Moodle offre au tuteur des moyens :

- D'exploiter des activités d'évaluation tant formatrices que sommaires ;
- D'assurer un suivi individuel des apprenants inscrits ;

Cette plate-forme permet également, à l'apprenant :

– D'exploiter des activités d'autoévaluation associées au contenu du cours ;
– D'accéder à des travaux d'étudiants et à d'autres sites Web ;

Le point focal de Moodle est lié à l'interopérabilité et à la réutilisation de son contenu sur plusieurs plateformes ainsi qu'à sa conformité aux normes. La définition des parcours pédagogiques est bien élaborée dans Moodle. Ce dernier est aussi considéré comme une plate-forme de travail collaboratif.

1.2.3.5 Avantages et inconvénients de l'e-learning

Les avantages d'une formation e-learning sont nombreux que ce soit pour les décideurs de l'entreprise, les formateurs et les apprenants [1].

Avantages pour l'entreprise

L'e-learning reçoit un accueil favorable des entreprises, les arguments les plus fréquemment évoqués en facteur du e-learning sont les suivants :

– Le coût réduit des formations est l'un des plus gros avantages du e-learning par rapport à la formation classique, en effet, l'e-learning permet de réduire certains coûts associés à la formation, comme les frais de déplacement et d'hébergement. Comme il permet de réduire le coût lié à la disponibilité de l'apprenant grâce à l'optimisation de la gestion de son temps.
– L'e-learning permet notamment de dématérialiser le lieu de formation, cet avantage est d'autant plus intéressant pour les entreprises mondiales (les multinationales), dont les salariés sont répartis dans le monde entier, l'e-learning permet de s'assurer que le contenu de la formation et les messages transmis sont identiques quel que soit le pays.
– L'entreprise a la possibilité de former un effectif important d'employés, la formation de masse étant une des caractéristiques propre au e-learning.
– L'exploitation des nouvelles technologies dans la formation peut permettre à une entreprise d'améliorer son image au sein de son entourage, des bénéfices peuvent être tirés notamment lorsqu'il s'agit d'attirer des jeunes diplômés ou de retenir des employés grâce à des programmes plus orientés vers le développement durable des compétences.

Avantages pour l'apprenant

La formation en ligne (e-learning) permet d'accroître l'efficacité générale de l'effort de formation pour les raisons suivantes :

- L'e-learning est plus facilement accessible, l'apprenant pouvant se former au travail, à la maison ou de n'importe quel endroit disposant d'un accès Internet (dématérialisation du lieu),
- La gestion du e-learning est plus flexible, l'apprenant, pouvant se former quand il veut sans avoir de contraintes horaires, plus un gain de temps considérable.
- L'e-learning est facilement adaptable aux besoins spécifiques de chacun, grâce à l'interactivité de l'outil informatique, le contenu de l'apprentissage peut être plus facilement adapté aux besoins de l'apprenant en tenant compte de son niveau et de son rythme d'apprentissage.

Avantages pour le formateur

Le formateur n'est pas moins avantagé que l'entreprise ou l'apprenant, il bénéficie également des nombreux aspects du e-learning, ainsi :

- L'e-learning allège les contraintes logistiques que doit gérer le formateur, désormais dispensé de se déplacer sur le lieu de formation (gain de temps et d'argent) ou de réserver le matériel nécessaire par exemple.
- Le multimédia permet de créer des contenus interactifs qui apportent un aspect attractif et ludique à la formation et suscitent l'intérêt des apprenants.
- Une évaluation des prérequis plus facile grâce à l'utilisation de jeux ou de QCM interactifs, qui permettent d'établir un diagnostic du niveau de compétences.
- Suivi régulier des apprenants, il est ainsi possible de visualiser, en fonction de la plate-forme de formation retenue, la progression de l'apprenant dans le parcours pédagogique via un enregistrement automatique des scores obtenus aux exercices, ainsi que du nombre et du temps de connexion par apprenant ; c'est ce qu'on appelle le tracking.

Inconvénients de l'e-learning

L'enseignement à distance bien qu'ayant des avantages certains, il a tout de même des inconvénients : :

- L'apprentissage solitaire, qui ne convient pas à tout le monde, les forums, Internet et le questionnement par e-mail ne sont qu'une réponse imparfaite au besoin d'interaction et de confrontation d'idées et de perceptions,
- L'e-learning ne peut avoir de résultats bénéfiques que sur les formations théoriques, la pratique demande une formation sur le tas,
- L'effort motivationnel et d'apprentissage par l'e-learning est plus important que par la formation classique, en effet, l'apprenant ne peut être passif, il est acteur de sa formation. Pour cela, les conditions doivent être favorables pour le motiver,
- L'investissement en matériel informatique et en logiciels peut être conséquent si l'organisation n'est pas encore équipée pour faire du e-learning,
- De plus, l'e-learning peut se heurter à une résistance des salariées, il demande plus d'autonomie et d'initiative que les formations classiques, qui demande en parallèle l'implication de toute l'organisation de la base au sommet de la hiérarchie.

1.3 Réseaux sociaux

1.3.1 Définition d'un réseau social

Deux aspects se côtoient quand on parle de réseaux sociaux : d'un côté, l'aspect sociologique et communautaire et de l'autre l'aspect technologique et Internet.

D'un point de vue sociologique, selon Wasserman et Faust, auteurs de Social Network Analysis : Methods and Applications publié en 1994, un réseau social est un ensemble de relations entre des entités sociales (individus). Les contacts entre ces individus peuvent être, par exemple, des relations de collaboration, d'amitié, ou des citations bibliographiques.

D'un point de vue technologique, selon Esther Dyson, éditrice de la newsletter Release 1.0 , les réseaux sociaux fournissent des outils qui facilitent le processus de mise en relation d'individus autour d'un centre d'intérêt commun et permettent la prise de contact en ligne [24].

On peut également définir un réseau social comme étant une structure comportant un ensemble d'acteurs qui sont reliés ensemble par des interactions sociales. Un acteur est une entité sociale qui peut être une seule personne, un groupe, ou une organisation. Les acteurs sont connectés entre eux par des liens qui peuvent désigner une ou plusieurs

relations. Ces liens peuvent être de types différents, à savoir, des liens de famille, des liens d'amitié, des liens de collaboration, des liens d'affaires, etc [21].

1.3.2 Types de réseaux sociaux

Tout comme les réseaux informatiques, les réseaux sociaux en ligne peuvent être classés selon différentes typologies [24] :

1. **Les réseaux plateforme de partage**

   Les plates-formes permettent de diffuser du contenu, souvent multimédia (vidéo et son), aux internautes. La mise en ligne et le partage de vidéos par exemple deviennent plus faciles car accessibles par tous les internautes de la communauté. Exemples : YouTube, Dailymotion, etc.

2. **Les réseaux personnels et généralistes**

   Souvent orientés autour d'un centre d'intérêt (musique, lecture, etc.), le but de ce type de réseaux n'est autre que de faire partager ses passions au reste de la communauté. Les mises en relation directes sont rares sur ce type de réseaux. Exemples : MySpace, Skyblog, Friendster, etc.

3. **Les réseaux personnels et thématiques**

   Ils fonctionnent souvent sur le même principe que les réseaux généralistes mais sont orientés autour d'une thématique : les voitures, la musique, la cuisine, etc. Exemples : Boompa, EonsCom, etc.

4. **Les réseaux professionnels**

   Les réseaux professionnels sont les réseaux les plus aboutis. Ils offrent la possibilité de mise en relation ainsi que le partage d'informations (coordonnées, informations sur les entreprises, etc.). Exemples : 6nergies, Viaduc, LinkedIn, OpenBC, etc.

Chacun de ces réseaux peut être soit ouvert (accessible à tout le monde) soit fermé (accessible uniquement sur invitation).

La figure suivante illustre des exemples de réseaux sociaux selon leur typologie :

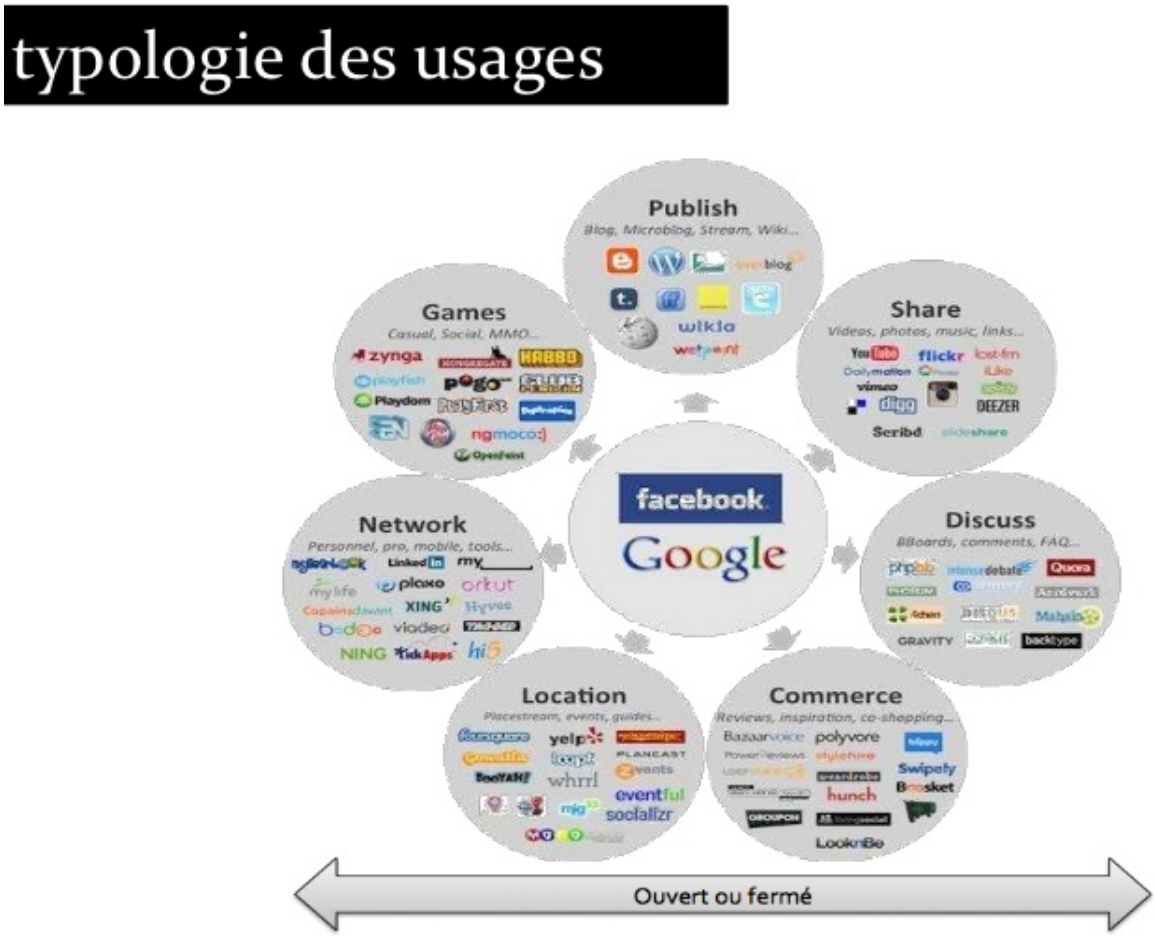

FIGURE 1.2: Typologies des réseaux sociaux.

1.3.3 Représentation des réseaux sociaux

La première personne à avoir représenté un réseau social est Jacob Levy Moreno au début des années 1930. Son objectif étant de visualiser graphiquement un réseau social, il a représenté les personnes par des points et une relation entre deux personnes par des flèches.

Cette représentation est depuis désignée par le terme sociogramme, mais on parlait également de toiles en raison de leur aspect en toile d'araignée. Les mathématiciens ont rapidement fait le rapprochement entre les représentations sociogrammes et la théorie des graphes au sens mathématique [10].

1.3.3.1 Approche basée sur la théorie des graphes

Les définitions suivantes listent quelques notions manipulées par la théorie des graphes pour les réseaux sociaux [10] :

- **Un sommet** est l'unité de base d'un réseau, il en représente une ressource. Dans un réseau social on parle d'acteur. Le terme nœud est également utilisé pour désigner un sommet.
- **Une arête** est une connexion entre deux sommets. On parle également d'arc ou de lien.
- **Une hyperarête** est une arête qui connecte 2 ou plusieurs sommets.
- **Une arête** est **orientée** si elle ne s'utilise que dans une seule direction.
  Inversement, on parle d'arête **non orientée** pour une arête qui s'utilise dans les deux directions.
- Une arête est **pondérée** lorsqu'on lui attribue un poids.
- Une arête est **étiquetée** lorsqu'on lui attribue un label.
- Un **graphe** est défini par un ensemble de sommets et un ensemble d'arêtes.
- Un **graphe orienté** désigne un graphe avec des arêtes orientées.
- Un **graphe pondéré** désigne un graphe avec des arêtes pondérées.
- Un **graphe étiqueté** désigne un graphe avec des arêtes étiquetées.
- Le **degré** d'un sommet est le nombre de ses arêtes adjacentes.
- Un **chemin** est une séquence d'arêtes qui relie deux sommets.
- Un chemin orienté est une séquence d'arêtes qui relie deux sommets en respectant l'orientation du parcours à chaque arrête.
- Le **diamètre** d'un graphe est le plus long chemin géodésique de ce graphe.
- Un **graphe** est complet lorsqu'il existe une arête entre toute paire de sommets.
- Un graphe est dit connexe lorsqu'il existe un chemin entre toute paire de sommets.

La figure ci-dessous est un exemple de représentation d'un réseau social à l'aide d'un graphe. Elle illustre une communauté qui s'est divisée en deux, suite à une divergence d'opinion, les membres de la première sont représentés par des ronds et les membres de la seconde par des carrés.

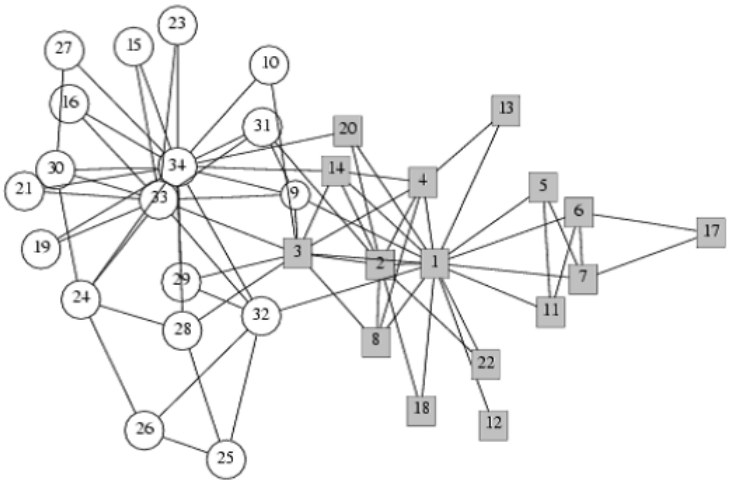

FIGURE 1.3: xemple de représentation d'un réseau social à l'aide d'un graphe.

1.3.3.2 Approche basée sur les matrices

La matrice est généralement l'objet mathématique le plus utilisé pour manipuler les concepts des réseaux sociaux, mais des approches ensemblistes ont également étaient proposées.

On distingue deux types de matrices dans un réseau social [10] :

– Les matrices d'incidence.

– Les matrices d'adjacence.

La matrice d'incidence

Les matrices d'incidence contiennent deux types de ressources, les lignes représentent un type et les colonnes un autre type.

Une matrice d'incidence est convertible en deux matrices d'adjacence, représentant chacune les ressources des lignes et des colonnes (tableau I.2 et I.3), les valeurs des cases contiennent les points communs entre les ressources correspondantes dans la matrice d'incidence, a_{ii} n'ayant pas de valeur.

Exemple :

	Projet 1	Projet 2	Projet 3	Projet 4
Employé 1	1	1	1	0
Employé 2	1	0	0	0
Employé 3	1	1	1	1
Employé 4	0	0	1	1

TABLE 1.1: Exemple de matrice d'incidence indiquant sur quel projet travaille chaque employé.

La matrice d'adjacence

On parle de matrice d'adjacence lorsqu'on a les mêmes ressources en ligne et en colonne, on obtient ainsi une matrice carrée avec la ligne i et les colonnes i représentant la même ressource.

Un graphe peut ainsi être représenté sous la forme d'une matrice M à n lignes et n colonnes représentant un tableau. Chaque case de ce tableau est notée a_{ij}avec i et j les numéros respectifs de ligne et de colonne de la case.

La valeur contenue dans la case a_{ij}est le poids de la relation entre les ressources v_i et v_j (égal à 1 dans le cas d'un graphe non pondéré), 0 correspond à une absence de relation.

Exemples :

	Employé 1	Employé 2	Employé 3	Employé 4
Employé 1	-	1	3	1
Employé 2	1	-	1	0
Employé 3	3	1	-	2
Employé 4	1	0	2	-

TABLE 1.2: Matrice d'adjacence des employés déduite du tableau II.1, chaque case représente le nombre de projets partagés entre les employés correspondants.

	Projet 1	Projet 2	Projet 3	Projet 4
Projet 1	1	1	1	0
Projet 2	1	0	0	0
Projet 3	1	1	1	1
Projet 4	0	0	1	1

TABLE 1.3: Matrice d'adjacence des projets déduite du tableau II.1, chaque case représente le nombre d'employés partagés entre les projets correspondants.

1.3.4 Représentation sémantique d'un réseau

Avec le caractère toujours plus participatif du web, le paysage de la toile est désormais le produit de ses utilisateurs, devenus une des ressources majeures du web. En réponse à ce phénomène social, la communauté du web sémantique propose des modèles ontologiques pour représenter et exploiter les profils des utilisateurs, leurs usages et leur réseau social [10].

1.3.4.1 Modèles ontologiques

Divers modèles ontologiques ont été proposés, on retrouve notamment [10] :

L'ontologie FOAF (Friend Of A Friend)

C'est l'initiative la plus célèbre et la plus adoptée, elle décrit les personnes, les liens entre elles et ce qu'elles créent et font. Un large ensemble de propriétés représentent la plupart des concepts nécessaires à la description d'un profil. Par exemple "family_name" et "interest" permettent respectivement de définir le nom de famille et un intérêt d'une personne. La propriété "knows" est ensuite utilisée pour connecter les profils entre eux et ainsi former le réseau social des profils FOAF.

Enfin FOAF modélise les usages des utilisateurs avec des classes pour représenter les ressources manipulées (OnlineAccount, Document, Group,etc.) et des propriétés pour les interactions des utilisateurs avec ces ressources (holdsOnlineAccount, weblog, member,etc.).

L'ontologie RELATIONSHIP FOAF

permet de décrire précisément les profils utilisateurs, la modélisation des relations entre utilisateurs et les usages, elle est très large. Ainsi, l'ontologie RELATIONSHIP a été proposée pour se spécialiser dans les relations dans le réseau social, en proposant un ensemble de propriétés étendant la propriété "knows" de FOAF. Elle modélise un grand nombre de liens entre les personnes comme les relations familiales, amicales ou encore professionnelles.

L'ontologie SIOC

Les activités en lignes principalement modélisées dans l'ontologie FOAF par la classe "OnlineAccount" et la propriété "holdsOnlineAccount" sont spécialisées dans l'ontologie SIOC, qui décrit l'information contenue explicitement et implicitement dans les moyens de communication d'internet. Pour cela, cette ontologie modélise les concepts issus des applications sociales du web, tels que les "Posts" des forums.

1.3.4.2 Social Tagging

Le social tagging consiste à partager des ressources et à les classifier avec des annotations sous forme de tags. Le fruit du social tagging est une classification de ressources librement établie par les utilisateurs, appelée folksonomie. L'adoption massive de cette pratique par les utilisateurs du web2.0 et la classification proposée par les folksonomies ont amené la communauté du web sémantique à s'intéresser de près à ces usages. Ainsi, le noyau d'une folksonomie, à savoir l'action de "tagging", est défini comme étant composé d'une ressource, d'un tag et d'un utilisateur.

L'ensemble des tags manipulés par une personne ou un groupe de personnes est appelé un nuage de tags. Le nuage de tags est l'une des alternatives pour naviguer au sein des ressources d'une folksonomie.

L'ontologie SCOT s'intéresse de près à ces nuages de tags et commence à s'imposer comme moyen de "représenter la structure et la sémantique des données du social tagging afin de les partager et de les réutiliser" [10].

1.3.4.3 Représentation sémantique de personnes et d'usages

Dans la représentation sémantique des personnes et des usages, il est important de mentionner les microformats. Cette initiative est importante dans la marche en avant vers un web sémantique qui doit passer par une sémantique légère avant d'atteindre le but attendu par la communauté.

Le principe des microformats est d'utiliser les attributs du HTML de manière consensuelle dans l'optique d'ajouter de la sémantique embarquée dans un document XHTML. Les règles mises en place permettent de s'abstenir de l'usage d'une ontologie et de mettre en place un mécanisme de sémantique légère, sans règles d'inférence ni relations de subsomption [10].

On retrouve ainsi un ensemble de microformats permettant de décrire des personnes, des ressources et des réseaux sociaux.

Exemples :

- Le microformat hCard pour représenter une carte de visite (nom, couriel, adresse, etc.).
- hResume pour la publication de CV et "XFN" (XTML Friends Network).

Grâce à leur facilité d'intégration, les microformats sont largement utilisés notamment dans l'optique de la portabilité des données mais aussi pour une exploitation directe des informations (import d'une carte de visite dans son répertoire, ajout d'un évènement dans son agenda, visualisation sur une carte d'un lieu, etc.) [10].

1.4 Ontologies et services web sémantique

1.4.1 Web sémantique

1.4.1.1 Qu'est-ce que le web sémantique ?

Le terme de Web sémantique a été proposé par Tim Berners-Lee en 2001 pour désigner une évolution du Web qui permettrait une collaboration entre humains et machines sur une base sémantique, de sorte à rendre les données disponibles sur le Web (contenus, liens, etc.) plus facilement localisables, utilisables et interprétables, et ce automatiquement, par des machines et des agents logiciels.

Le Web sémantique part du principe que les métadonnées (des données relatives à des

données) sont préalablement établies, pour qu'elles soient utilisées par les différentes techniques proposées, visant ainsi une meilleure exploitation des ressources par les machines. Le but du Web sémantique est donc de donner un sens aux informations disponibles, de telle sorte que les machines puissent les comprendre [18].

1.4.1.2 Architecture du Web sémantique

L'architecture du Web Sémantique se compose d'un ensemble de langages, généralement représentés sous la forme d'une pyramide. Chaque niveau repose sur les résultats définis au niveau inférieur, c'est-à-dire que chaque niveau est progressivement plus spécialisé et plus complexe que le niveau précédent. D'autre part, tout niveau est indépendant des niveaux supérieurs afin qu'il puisse être développé et rendu opérationnel de manière autonome par rapport aux développements des niveaux supérieurs. Cette pyramide des langages, proposée par Tim Berners-Lee, est représentée dans la Figure suivante [15] :

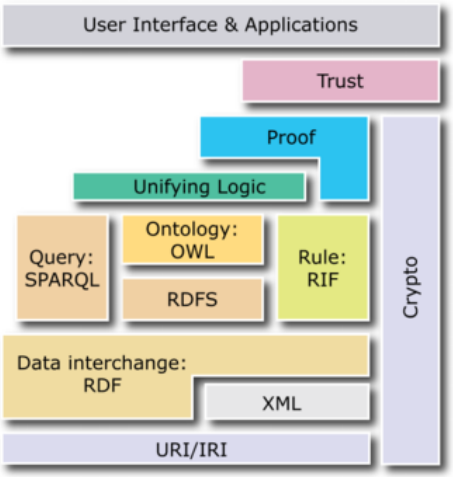

FIGURE 1.4: Architecture du Web sémantique proposée par Tim Berners-Lee.

Niveau d'adressage et de nommage

Les couches les plus basses, à savoir URI et XML, assurent l'interopérabilité syntaxique [15] :

L'**URI** (Uniform Resource Identifier - identifiant uniforme de ressource) est un protocole simple et extensible utilisé pour identifier, d'une manière unique et uniforme à l'aide

d'un nom ou d'une adresse (ou les deux) une ressource dans le Web, sans faire de distinction entre une ressource physique [1] ou une ressource abstraite [2]. Rappelons également que l'URL (Uniform Resource Locator), qui désigne un sous ensemble d'URI, est une chaîne courte de caractères qui est aussi utilisée pour identifier des ressources (physiques) par leur localisation, autrement dit, par leur mécanisme d'accès, plutôt que par un nom ou un autre attribut de cette ressource. Notons aussi que quelque chose qui peut être identifié avec un URI peut être décrit, ainsi le Web sémantique peut raisonner au sujet des personnes, des endroits, des idées, etc.

Unicode, qui est au même niveau que l'URI, est un encodage textuel universel pour échanger des symboles. Il permet à tous les langages humains d'être utilisés sur le Web avec un maximum d'interopérabilité.

Niveau syntaxique

Le niveau syntaxique est le niveau de la structuration des documents. La spécification de la structure logique des documents repose sur **XML** (eExtensible Markup Language), qui est le langage de base qui procure une syntaxe aux documents structurés. Il peut être vu comme la couche de transport syntaxique du Web sémantique, tous les autres langages étant exprimables et échangeables dans la syntaxe XML.

Avant de commencer à organiser les informations dans un document XML, il est impératif de définir la structure de ce dernier, afin de permettre notamment de vérifier sa validité. **DTD** (Document Type Definition) et **XML-S** (XML Schema) sont des langages de description de format de document XML. Cependant, XML-S, qui permet entre autre de définir des domaines de validité pour la valeur d'un champ, alors que cela n'est pas possible dans une DTD, est considéré comme la nouvelle recommandation de W3C et tend à remplacer la DTD [15].

Un fichier XML peut faire appel à plus d'un document DTD ou XML-S. Sa structure peut donc être organisée selon plusieurs grammaires. Afin d'éviter les conflits de noms, dus au fait que ces documents sont en général développés indépendamment les uns des autres, le W3C a mis en place un nouveau standard, baptisé « **Namespace** » (espace de nommage). Selon la définition de ce consortium, les espaces de nommage fournissent un

1. Sa représentation est récupérable via l'Internet telle qu'une page web, un service localisé sur un serveur, etc.
2. Qui ne fait pas partie du réseau, tel qu'un livre, une idée, une couleur, etc.

moyen simple pour qualifier les noms des éléments et des attributs dans le langage XML, en les associant avec des espaces de noms identifiés par URI [15].

Niveau sémantique

Après avoir référencé les ressources avec le protocole URI et structuré les informations avec le XML, l'étape suivante consiste à les annoter, afin de les doter d'un sens interprétable par la machine. C'est justement le rôle du niveau sémantique dans l'architecture du Web sémantique. Ce niveau est représenté d'une part par les langages de représentation d'ontologies **RDF/RDFS** et **OWL**, et d'autre part par les langages de règles, de logique, de preuves et de confiance [2].

RDF qui permet de décrire les métadonnées des documents sous la forme de triplets comme spécifié par RDF Schéma. Il permet d'implémenter un premier niveau d'ontologies, relativement simples.

Afin de représenter des ontologies plus complexes, le langage OWL qui définit des classes, attributs, relations et axiomes peut être combiné à un langage de règles permettant de raisonner sur les ressources et d'inférer de la nouvelle connaissance.

Le niveau Logique (Logic) est utilisé pour établir la cohérence des annotations et pouvoir inférer des conclusions non explicitement énoncées.

Le niveau Preuve (Proof) pourrait fournir les moyens pour tracer et expliciter les différentes étapes du raisonnement logique afin de pouvoir lui accorder un niveau de confiance.

Le dernier niveau (Trust) a pour objectif d'authentifier l'identité et la véracité des données et services disponibles sur le Web Sémantique.

1.4.1.3 Langages du Web sémantique

Dans le contexte du Web sémantique, plusieurs langages ont été développés. La plupart de ces langages reposent sur XML ou utilisent XML comme syntaxe.

Néanmoins, XML est limité, car il ne dispose pas d'une sémantique, ce qui nécessite le développement d'autres langages pour le web sémantique.

Dans la suite de cette section, nous étudierons plus particulièrement les formats XML, XML Schema, RDF, RDF Schema, OWL et SPARQL.

XML

XML (eXtensible Markup Language) est un langage de balisage extensible qui fournit une syntaxe pour les documents structurés, mais n'impose aucune contrainte sémantique à la signification de ces documents. Un langage à balises combine le texte et les informations supplémentaires (métadonnées) sur le texte. Les informations supplémentaires, telles que la structure, la police, la couleur, etc. des textes, sont exprimées en utilisant des balises, qui sont mélangées avec le texte à présenter. En plus, le langage XML permet aux utilisateurs de définir eux-mêmes leurs balises.

L'avantage de XML est la possibilité de personnaliser la présentation des documents en utilisant XSL (XML Stylesheet Language) qui permet de transformer automatiquement un fichier XML en une page HTML qui est consultable via un navigateur Internet. Cependant, son inconvénient est qu'il n'a pas de sémantique formelle permettant l'interprétation par la machine. XML décrit uniquement la structure de l'information, autrement dit, sa syntaxe.

Schéma XML

Un schéma XML est une description du type d'un document XML, qui contient un ensemble de règles (contraintes sur la structure et le contenu du document) auxquelles un document XML doit se conformer afin d'être considéré valide selon ce schéma.

DTD (Document Type Definition), **RELAX NG** (REgular LAnguage for XML Next Generation), **XML Schema** sont spécifiquement développés pour exprimer des schémas de XML [23].

RDF

Resource Description Framework (RDF) est une recommandation du W3C pour décrire les ressources du WWW (World Wide Web). Ces dernières sont représentées sous forme de graphes orientés et étiquetés. La principale entité en RDF est appelée ressource. RDF utilise le principe de l'URI pour identifier d'une manière unique les ressources. Le langage RDF consiste en des déclarations (statements) faites au sujet des ressources. Ces déclarations sont représentées sous la forme de triplets de la forme **<ressource, propriété, valeur>**, similaire au triplet **<sujet, prédicat, objet>**. Le sujet est la ressource à décrire. Le prédicat est un attribut ou une caractéristique de ce sujet, qui définit une relation binaire entre une ressource et une valeur. L'objet peut être soit une

autre ressource ou une valeur littérale de ce prédicat, cette valeur peut être une chaîne ou un entier [14].

Ce triplet peut être représenté graphiquement, comme le montre la figure suivante :

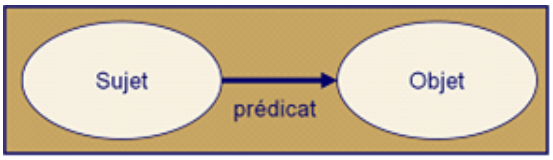

FIGURE 1.5: Représentation graphique d'une assertion RDF.

RDF utilise des types de données XML Schema pour indiquer le type de ces valeurs littérales.

RDFS

Afin d'apporter plus de sémantique à **RDF**, **RDFS** (RDF-Schema) a été développé dans le but de donner des moyens de définir les vocabulaires RDF, il a été décrit dans les termes du W3C comme étant un « langage de description de vocabulaires RDF ». Il permet également de créer des relations sémantiques entre classes de ressources. Ainsi, RDFS ajoute à RDF la possibilité de définir des hiérarchies de classes avec rdfs :subClassOf et de définir les genres et les propriétés des ressources avec rdfs :subPropertyOf. Les ressources peuvent être déclarées comme des instances de rdfs :Class via la propriété rdfs :type. RDFS permet également, aux concepteurs de définir les contraintes de domaine de définition (en anglais, domain) et le domaine de valeur (en anglais, range) des propriétés à l'aide des attributs rdfs :domain et rdfs :range [14].

Pour résumer, XML peut être vu comme la couche de transport syntaxique, RDF comme un langage relationnel de base et RDFS offre des primitives de représentation de structures ou primitives ontologiques.

OWL

En raison de la limitation de la capacité d'interprétation du contenu par le RDFS, les chercheurs dans le domaine du web sémantique ont travaillé sur l'extension de ce dernier pour représenter les ontologies du Web sémantique. Les fruits de recherches ont donné

naissance au langage de représentation d'ontologies du Web, **OWL**, qui est devenu un standard de W3C. Il a été conçu de sorte à être utilisé par les applications qui doivent traiter le contenu de l'information au lieu de simplement présenter des informations à l'utilisateur. OWL permet une plus grande interopérabilité entre les applications du Web comparée à celle offerte par XML, RDF et RDF Schema (RDFS) en fournissant un vocabulaire supplémentaire avec une sémantique formelle [14].

OWL est construit au-dessus de RDF / RDFS dans les couches du web sémantiques, ce qui signifie qu'une ontologie OWL peut comporter des déclarations faites sur les ressources. Notons que, parmi les apports du langage OWL : la possibilité d'importation de plusieurs ontologies au sein d'une même, la définition de plusieurs types de propriétés, la définition de cardinalités, la définition des caractéristiques de propriétés, la définition des restrictions, la définition de l'équivalence entre classes et individus [14].

OWL fournit trois sous langages de plus en plus expressifs conçus pour faire un compromis entre son pouvoir expressif et son pouvoir de raisonnement [15] :

1. **OWL Lite** : ce sous-langage ne contient qu'un sous-ensemble réduit des constructeurs disponibles. Il a la complexité formelle la plus basse et l'expressivité minimale dans la famille OWL, mais il permet tout de même d'exprimer la classification, qu'on appelle taxonomie, et les relations simples entre les classes.
2. **OWL DL** : ce sous-ensemble est fondé sur les caractéristiques de la logique descriptive. Il contient l'ensemble des constructeurs, mais avec des contraintes particulières sur leur utilisation, qui maintiennent toujours la décidabilité de la comparaison de types. Par contre, la grande complexité de ce langage semble rendre nécessaire une approche heuristique ;
3. **OWL Full** : ce sous-ensemble est la version la plus complexe d'OWL, mais également celle qui permet le plus haut niveau d'expressivité. OWL Full est destiné aux situations où il est plus important d'avoir un haut niveau de description, quitte à ne pas pouvoir garantir la complétude et la décidabilité des calculs liés à l'ontologie. Sans aucune contrainte, dans ce cas, le problème de comparaison de types est vraisemblablement indécidable.

Langage de requête SPARQL

Le langage de requêtes SPARQL offre des services puissants pour extraire des informations sur de grands ensembles de données RDF. Pour gérer les graphes RDF, les dessins et les implémentations de plusieurs langages de requêtes RDF ont été proposés.

En 2004, le Groupe de travail RDF Data Access, qui fait partie de l'activité web sémantique, a publié un premier projet de travail public d'un langage de requêtes pour RDF, appelé SPARQL. Depuis lors, SPARQL a été rapidement adopté comme le standard pour l'interrogation des données du Web sémantique. En janvier 2008, SPARQL est devenu une recommandation du W3C.

Les requêtes SPARQL sont des requêtes sur les triplets qui constituent un graphe de données RDF. La requête SPARQL officielle introduit quatre formes différentes :

- Requête de la forme SELECT, renvoie la valeur de la variable, qui peut être liée par un modèle de requête correspondant ;
- Requête de la forme ASK, renvoie vrai si la requête correspond aux données et faux sinon.
- Requête de la forme CONSTRUCT, renvoie un graphe RDF en remplaçant les valeurs dans les modèles de données ;
- Requête de la forme DESCRIBE, renvoie un graphe RDF qui définit la ressource correspondante.

1.4.1.4 Composants principaux du Web sémantique

Le Web sémantique s'articule autour de deux composants essentiels :

1. **Les ontologies** : technologie dorsale pour le Web sémantique et – plus généralement - pour le management des connaissances formalisées décrivant les ressources du Web. Elles fournissent la "sémantique" exploitable par machine des données et des sources d'informations qui peuvent être communiquées entre différents agents logiciels et humaines (ce point est détaillé dans la section suivante).
2. **Les annotations sémantiques** : décrivent les ressources en utilisant la "sémantique" définie dans l'ontologie. Les ressources annotées par les métadonnées faciliteront la recherche, l'extraction, l'interprétation et le traitement de l'information d'une manière plus efficace.

1.4.2 Ontologies

1.4.2.1 Définition

Le terme "ontologie" a été emprunté par les technologies de l'information à la philosophie, où il faisait référence à la science qui « étudie l'être en tant qu'être ». Ce terme a pris une toute autre tournure avec l'émergence de l'ingénierie des connaissances et du web sémantique, pour désigner la problématique de représentation et de manipulation des connaissances dans un système informatique. Les définitions sont souvent reformulées, mais dans celle donnée par Gruber une Ontologie désigne : « une spécification explicite d'une conceptualisation » [15].

Cette définition a été reprise et modifiée par Borst pour mettre l'accent sur l'importance du partage et la réutilisation des connaissances, sa définition est : « Une ontologie est une spécification formelle d'une conceptualisation partagée ».

1.4.2.2 Composants de l'ontologie

Les composants d'une ontologie sont toujours les mêmes, quelques soient les divergences qu'il peut y avoir relatives à sa structure (le degré de la formalisation). Les connaissances intégrées dans les ontologies sont formalisées en mettant en jeu cinq types de composants : concepts, relations, fonctions, axiomes, instances [3].

1. **Les concepts :** aussi appelés termes ou classes de l'ontologie, sont des notions (ou objets) permettant la description d'une tâche, d'une fonction, d'une action, etc. faisant parties de la réalité, retenues en fonction des objectifs qu'on se donne et de l'application envisagée pour l'ontologie. Ils peuvent être abstraits ou concrets, élémentaires ou composés, réels ou fictifs.
2. **Les relations :** Représentent un type d'interaction, ou bien des associations existantes entre les concepts d'un domaine. Elles se définissent formellement comme tout sous-ensemble d'un produit de n concepts : $R : C_1 \times C_2 \times \cdots \times C_{n-1} \rightarrow C_n$. Des exemples de relations binaires sont : sous-concept-de, sous-classe-de, associée-à, connecté-à, sorte-de, instance-de, etc.
3. **Les fonctions :** ce sont des cas particuliers de relations dans lesquelles le Nième élément de la relation est défini de manière unique à partir des n-1 premiers. For-

mellement, les fonctions sont définies ainsi, $F : C_1 \times C_2 \times \cdots \times C_{n-1}, C_n$. Comme exemples de fonctions binaires, nous avons la fonction mère de et carré.

4. **Les axiomes :** constituent des assertions, acceptées comme vraies, à propos des abstractions du domaine, traduites par l'ontologie. Ils ont pour objectif de définir la sémantique des concepts et des relations dans un langage logique. L'utilisation des axiomes sert à définir le sens des entités, mettre des restrictions sur la valeur des attributs, examiner la conformité des informations spécifiées ou en déduire de nouvelles.
5. **Les instances :** sont utilisées pour représenter des éléments et véhiculer l'information.

1.4.2.3 Langages de spécification d'ontologie

D'un point de vue opérationnel, on peut bâtir une ontologie grâce aux langages de programmation logique classique tels Prolog et Lisp. Mais, plus souvent, on utilise des modèles et langages spécialisés pour la construction d'ontologie tels que OKBC (Open Knowledge base Connectivity) et KIF (Knowledge Interchange format). On peut également utiliser des langages plus avancés tels que DAML+OIL ou les standards qui ont émergé auprès du W3C et qui sont utilisés par le Web sémantique tels que XML, RDF, RDFS et OWL (voir le point 3 dans la partie Web sémantique). Le choix du bon langage de développement dépend notamment du degré de nuance et de sophistication nécessaire pour répondre au besoin fonctionnel. De plus en plus, ces langages tendent à intégrer RDF comme technologie fondamentale pour intégrer les données présentes sur le Web.

1.4.2.4 Ontologies et web sémantique

Les ontologies jouent un rôle important pour faire communiquer les personnes et les machines dans le Web sémantique. En effet Une fois construite et adoptée par une communauté particulière, une ontologie doit traduire un certain consensus explicite et certain niveau de partages qui sont essentiels pour permettre l'exploitation de ressources sur le Web. La sémantique du Web est fondée sur des ontologies spécifiées explicitement dans un langage de représentation. Le W3C cherche à proposer un standard connu actuellement sous le nom d'OWL (Ontology Web Langage), qui s'appuie sur la logique de description.

1.4.3 Services web sémantique

1.4.3.1 Qu'est-ce qu'un service web ?

Un web service est un ensemble de protocoles et de normes informatiques utilisés pour échanger les données entre les applications. C'est un composant logiciel représentant une fonction applicative (ou un service applicatif). Il peut être accessible depuis une autre application (un client, un serveur ou un autre web service) à travers le réseau Internet en utilisant les protocoles de transports disponibles. Ce service applicatif peut être implémenté comme une application autonome ou comme un ensemble d'applications (liées ensemble par une infrastructure d'intégration) [9].

1.4.3.2 Problèmes existants dans le domaine des services web

Les actuels sujets de recherche dans le domaine des services web sont nombreux. Un nombre considérable d'études tournent autour de la découverte des services et ses sujets rattachés comme sont la sélection, la sémantique et la composition.

- **Problèmes de sélection :** Découvrir un service web qui nous intéresse est une chose, découvrir le service web le plus adéquat en est une autre. La qualité de service dans le cas des services web se mesure à l'aide de plusieurs métriques dont les métriques de performance et de fiabilité. Une recherche sur UDDI permet certainement de trouver plusieurs services web qui remplissent ces critères. Mais lequel sera le meilleur ? Il devient ainsi nécessaire de choisir les services web pertinents parmi ceux trouvés et de se fixer des critères pour choisir les meilleurs.
- **Problèmes de sémantique :** Tel que présentés précédemment, les services web sont décrits syntaxiquement et ne permettent en aucun cas l'interaction entre services, leur découverte dynamique ou automatique, ou encore leur composition sans une intervention humaine. Pour le permettre, il paraît alors nécessaire de se doter d'un mécanisme qui réglerait ce problème de sémantique.
- **Problèmes de composition de services web :** Les services web, tels qu'ils sont définis actuellement, sont limités à des fonctionnalités relativement simples. Toutefois, pour certains types d'applications, il est nécessaire de combiner un ensemble de services web simples en un service répondant à des exigences plus complexes.

1.4.3.3 Vers les services web sémantiques

Le besoin d'automatisation du processus de conception et de mise en œuvre des services web rejoint les préoccupations à l'origine du Web sémantique, à savoir comment décrire formellement les connaissances de manière à les rendre exploitables par des machines. En conséquence, les technologies et les outils développés dans le contexte du Web sémantique peuvent certainement compléter la technologie des Web services en vue d'apporter des réponses crédibles au problème de l'automatisation. C'est la naissance des services web sémantiques.

De manière générale, l'objectif visé par la notion de services web sémantiques est de créer un web sémantique de services dont les propriétés, les capacités, les interfaces et les effets sont décrits de manière non ambiguë et exploitable par des machines. La sémantique ainsi exprimée permet l'automatisation de plusieurs fonctionnalités qui sont nécessaires pour une collaboration inter-entreprises efficace, dont les principales sont les suivantes :

- **Découverte de services web** : Actuellement cette tâche doit être réalisée par un humain qui doit utiliser un moteur de recherche ou un annuaire pour trouver le service, lire la page Web qui décrit ce service, puis l'exécuter manuellement pour vérifier que celui-ci correspond bien aux attentes de l'utilisateur. Cette sémantique doit donc fournir une description déclarative des propriétés et des capacités du service web.
- **Invocation de services web** : L'invocation automatique d'un service signifie l'exécution du service par un programme informatique ou un agent logiciel. Cet agent doit être capable d'interpréter cette description sémantique afin de délivrer les données nécessaires à l'exécution du service web.
- **Composition de services web :** L'objectif qu'un utilisateur veut atteindre nécessite souvent l'utilisation de plusieurs services web. L'agent logiciel chargé d'atteindre cet objectif doit disposer de suffisamment de données afin de pouvoir sélectionner, composer et interopérer automatiquement ces services web. La description sémantique doit donc pouvoir fournir toutes ces informations.
- **Surveillance de l'exécution de services web :** un agent logiciel doit pouvoir connaître l'état d'avancement de sa requête. Cette description sémantique doit pouvoir fournir les informations nécessaires.

1.4.3.4 Langage de description sémantique de web services

Le WSDL-S est le langage WSDL augmenté d'un ensemble de fonctionnalités d'annotation sémantique pour les fichiers WSDL. Il définit un modèle sémantique pour capturer les termes et les concepts utilisés pour décrire et représenter la connaissance. La sémantique est ajoutée en deux étapes : la première consiste à faire référence, dans la partie définition WSDL, à une ontologie dédiée au service à publier ; la deuxième consiste à annoter les opérations de la définition WSDL de sémantique.

1.5 Systèmes de recommandation

1.5.1 Définition

Les systèmes de recommandation (SR) sont définis comme étant des outils et techniques logiciels fournissant à un utilisateur des suggestions d'items [3] susceptibles de l'intéresser. Ces suggestions se rapportent à des processus décisionnels variés, elles peuvent être des propositions de livre à acheter, de la musique à écouter, des films à regarder, des articles à lire, un restaurant à choisir, etc.

Autrement dit, un système de recommandation est une forme spécifique de filtrage de l'information qui cherche à prédire la valorisation ou la préférence qu'un utilisateur attribuerait à un item [19].

Un système de recommandation requiert généralement 3 étapes [19] :

1. La première consiste à recueillir de l'information sur l'utilisateur, appelée également la collecte de données.
2. La deuxième consiste à bâtir une matrice ou un modèle utilisateur contenant l'information recueillie, également appelé un profil utilisateur.
3. La troisième consiste à extraire à partir de cette matrice une liste de recommandations.

1.5.2 Formes de collecte de données

Pour pourvoir être pertinent, un système de recommandation doit pouvoir faire des prédictions sur les intérêts et goûts des utilisateurs. Il faut donc pouvoir collecter un

3. Item est le terme général utilisé pour dénoter ce que le système recommande aux utilisateurs.

certain nombre de données sur ceux-ci afin de pouvoir construire un profil pour chaque utilisateur.

On distingue donc deux (02) formes de collecte de données [11][19] :

1.5.2.1 Collecte de données explicite -Filtrage dit actif ou réactif

Ce type de collecte repose sur le fait que l'utilisateur indique explicitement ses intérêts suite à une demande du système. Par exemple : demander à un utilisateur de commenter, taguer/étiqueter, noter, liker ou encore ajouter comme favoris des contenus (objets, articles, etc.) qui l'intéressent.

L'avantage de ce genre de systèmes est qu'ils sont faciles à appliquer et ne requièrent aucune connaissance approfondie du domaine de la part de l'utilisateur. Toutefois, les critiques demeurent une arme à double tranchant. En effet, si elles représentent des informations explicites sur les appréciations, elles nécessitent un effort et un investissement de la part de l'utilisateur quant à l'expression de ses avis et appréciations. Les informations recueillies peuvent également contenir un **biais dit de déclaration** [4].

1.5.2.2 Collecte de données implicite – Filtrage dit passif ou proactif

Cette collecte repose sur une observation et une analyse des comportements de l'utilisateur effectués de façon implicite dans l'application qui embarque le système de recommandation, le tout se fait en "arrière-plan", sans rien demander à l'utilisateur.

Par exemples :

- Obtenir la liste des items que l'utilisateur a consultés, les musiques écoutées, les vidéos regardées ou les produits achetés en ligne.
- Analyser la fréquence de consultation d'un contenu par un utilisateur, le temps passé sur une page.
- Analyser son réseau social, etc.

L'avantage d'un système implicite est que l'utilisateur n'est plus sollicité pour fournir des informations ou des appréciations, toutes les informations sont collectées automatiquement, et les données récupérées sont a priori justes et ne contiennent pas de **biais de déclaration** [5]. Cependant, les données récupérées sont plus difficilement attribuables à un utilisateur et peuvent donc contenir des biais d'attribution , comme un utilisateur peut

4. Donner une fausse déclaration.
5. Utilisation commune d'un même compte par plusieurs utilisateurs.

ne pas aimer certains livres qu'il a achetés, ou il peut les avoir achetés pour quelqu'un d'autre.

1.5.3 Types de systèmes de recommandation

Il existe différentes approches de systèmes de recommandation, mais dans ce mémoire nous nous n'en aborderons que trois (03) dans ce mémoire : [6] :

- **Recommandation basée sur le contenu** (Content-Based filtering CB).
- **Recommandation basée sur le filtrage collaboratif** (Collaborative Filtering CF – Context Aware).
- **Recommandation Hybride**.

1.5.3.1 Recommandation basée sur le contenu (Content-Based filtering CB).

Egalement appelée **recommandation Objet**. Il s'agit de recommander des objets (ou contenus) en se basant sur les qualités et propriétés intrinsèques de l'objet lui-même et en les corrélant avec les préférences et intérêts de l'utilisateur. Ce type de système va donc extraire un certain nombre de caractéristiques et attributs propres à un contenu que l'utilisateur aura déjà consulté, afin de pouvoir lui recommander des contenus additionnels possédant des propriétés similaires. Cette méthode crée un profil pour chaque objet ou contenu, c'est-à-dire un ensemble d'attributs/propriétés qui caractérisent l'objet [6].

La recommandation se base sur des critères différents, en fonction du type de l'objet à recommander. S'il s'agit d'un document texte, il peut être représenté par les mots clés principaux de son contenu, ces derniers seront par la suite utilisés par le système de recommandation pour les comparer aux mots clés des documents que l'utilisateur a déjà consultés et appréciés. Dans le cas d'un site de vente de livres par exemple, on va se baser sur les caractéristiques du livre pour effectuer des recommandations, comme par exemple le sujet que traite l'ouvrage, son genre, son auteur, l'éditeur, etc [6].

La mesure de similarité

Les algorithmes de recommandation basée sur le contenu permettent de développer des modèles afin de trouver des patterns ou motifs semblables entre différentes données. Ils évaluent à quel point un contenu pas encore vu par l'utilisateur est similaire aux contenus

que celui-ci a évalués positivement dans le passé. Pour ce faire, on utilise la notion de similarité qui peut être mesurée de plusieurs manières :

1. Le système peut tout simplement vérifier si un livre, par exemple, se trouve dans la liste des genres préférés de l'utilisateur. Dans ce cas la similarité sera de 0 ou 1 (binaire/booléen).
2. Une autre façon serait de ne pas se baser sur le genre du livre, mais sur les mots-clés qui caractérisent l'ouvrage, et calculer la similarité de chevauchement entre les mots-clés du livre qui va éventuellement être suggéré avec les mots-clés préférés de l'utilisateur. Des indicateurs de mesure de similarité sont utilisés dans le cas d'un objet avec des propriétés multi-valeurs (cas des mots-clés), les plus utilisés sont :
   - Le produit scalaire
   - Le cosinus
   - Le coefficient de Dice.

   Si chaque document est décrit par un ensemble de mots-clés, représentés dans un espace vectoriel (matrice de tous les mots récurrents dans le document), alors ces indicateurs vont permettre de mesurer le degré de similarité entre 2 documents à partir de leur représentation vectorielle.

Avantages

Ce type de recommandation n'a pas besoin d'une large communauté d'utilisateurs pour pouvoir effectuer des recommandations. Une liste de recommandations peut être générée même s'il n'y a qu'un seul utilisateur.

Inconvénients

- Ce type de recommandation est difficilement applicable à des items dont les caractéristiques doivent être fournies manuellement, tels que les livres, les films, etc. ceci nécessite généralement beaucoup de temps, avec le risque d'introduire d'éventuelles erreurs.
- Les documents recommandés par ce genre de systèmes sont toujours similaires en termes de contenu aux documents déjà consultés par les utilisateurs. L'utilisateur est ainsi, toujours restreint à ses intérêts passés et l'empêchent l'exploration de thématiques nouvelles et différentes [11].

1.5.3.2 Recommandation basée sur le filtrage collaboratif (Collaborative Filtering CF – Context Aware)

C'est un système qui se base sur le comportement passé des utilisateurs similaires, en effectuant une corrélation entre des utilisateurs ayant des préférences et intérêts similaires [6].

Les méthodes utilisées collectent et analysent des données sur le comportement, les activités, les préférences des utilisateurs et des algorithmes tentent de prédire ce que l'utilisateur aimera en cherchant des utilisateurs qui ont les mêmes comportements que l'utilisateur à qui l'on souhaite faire des recommandations.

L'idée générale est que les personnes ayant apprécié les mêmes choses dans le passé, donc ayant les mêmes goûts, sont susceptibles de partager encore les mêmes intérêts dans le futur.

Les techniques de recommandation sociale :

Il existe deux (02) techniques de recommandation basée sur le filtrage collaboratif [6] :

– **Filtrage utilisateur-utilisateur (basé sur la mémoire ou user-centric)**

Dans cette approche, des utilisateurs sont identifiés et sélectionnés sur la base de la similarité de leurs intérêts et préférences avec l'utilisateur actif. On utilise alors principalement les ratings [6] de ces utilisateurs, qui sont également appelés "voisins", pour calculer des similarités avec l'utilisateur actif. Pour chaque produit p que l'utilisateur n'a pas encore vu, une prédiction est faite en se basant sur les ratings de p assignés par le panel d'utilisateurs voisins.

Le rating peut être limité seulement à une note donnée par l'utilisateur à un item, il peut également être exprimé par des like ou dislike, ou encore par un une appréciation textuelle, qu'il faudra interpréter par la suite en une note quantitavive, ou alors se baser sur des données plus implicites en observant le comportement de l'utilisateur sur le site. On peut observer par exemple quelle musique il a écouté, quel article il a lu, et on croise ses infos avec celles du reste des utilisateurs afin de lui proposer de nouvelles suggestions.

Le calcul de la mesure de similarité entre utilisateurs est calculée de différentes manières, les plus utilisées sont :

– Le coefficient de corrélation de pearson.

6. Une note attribuée par l'utilisateur à un item pour exprimer son appréciation pour ce dernier.

– La distance euclidienne.

Une comparaison entre ces deux méthodes sera faite dans la partie conception de ce travail.

La recommandation user-centric a cependant ses limites. Lorsqu'il s'agit d'un gros site qui gère des millions d'utilisateurs et des milliers d'items, il faut scanner un grand nombre de voisins potentiels, ce qui rend impossible la recommandation en temps réel (d'où le nom basé sur la mémoire). Pour pallier ce problème, les gros sites implémentent souvent une technique différente plus apte au traitement préalable des données hors-ligne (offline preprocessing), la recommandation item-centric (à ne pas confondre avec la recommandation basée sur le contenu).

– **Filtrage item-item (basé sur le modèle ou item-centic)**

Cette autre approche propose une inversion de l'approche user-centric. Au lieu de mesurer la corrélation entre des utilisateurs, les ratings sont utilisés pour mesurer la corrélation entre les contenus.

Les mêmes méthodes sont utilisées pour mesurer la similarité, à la différence qu'elles sont appliquées cette fois-ci au contenu.

Pour le dire autrement, l'approche item-centric propose de rechercher en premier lieu les contenus similaires en termes de ratings, et ensuite de faire une recommandation à l'utilisateur. Cette approche permet de faire un traitement préalable sur la matrice de ratings, pour déterminer les contenus similaires et ainsi pouvoir effectuer des prédictions en temps réel, contrairement à l'approche user-centric très gourmande en mémoire.

Avantages

L'approche du filtrage collaboratif ne requière aucune connaissance sur les contenus eux-mêmes. Par exemple, dans le cas d'un magasin de vente de livres en ligne, le système de recommandation collaboratif n'a pas besoin de savoir le type de contenu du livre, son genre, qui en est l'auteur, etc. La recommandation sociale est capable de recommander des contenus sans avoir besoin de comprendre le sens ou la sémantique du contenu lui-même. Les informations propres au livre n'ont pas besoin d'être introduite dans le système.

Inconvénients

- Scalability : souvent, les plates-formes sur lesquelles sont utilisés les filtres collaboratifs ont des millions d'utilisateurs, de produits et contenus. Ça demande donc beaucoup de puissance de calcul pour pouvoir proposer des suggestions aux utilisateurs.
- Cold Start : les systèmes de recommandation sociale ont besoin de beaucoup de données et beaucoup d'utilisateurs pour être performants. Le lancement d'un service de recommandation peut souffrir au début du manque d'utilisateurs et d'informations sur ceux-ci.
- Sparsity (Rareté) : le nombre de produits ou contenus est énorme sur certaines plates-formes, et même les utilisateurs les plus actifs auront noté ou valorisé qu'un tout petit sous-ensemble de toute la base de données. Donc, même l'article le plus populaire n'aura que très peu de bonnes notes. Dans une telle situation, deux utilisateurs auront peu d'articles valorisés en commun, ce qui rend plus difficile la tâche de corrélation.

1.5.3.3 Recommandation hybride

La recommdantion hybride est une combinaison des deux (02) approches citées ci-dessus. Elles sont de plus en plus utilisées, car elles permettent de résoudre des problèmes comme le cold start et la sparsity (rareté) qu'on retrouve dans une approche de recommandation sociale. D'autre part, si par exemple on considère 2 utilisateurs avec les mêmes goûts mais qui n'ont pas évalué ou "raté" des objets en commun, un filtrage collaboratif pur ne les considérera pas comme similaires ou voisins. Rappelons que la mesure de similarité standard ne prend en compte que les éléments pour lesquels l'utilisateur actif et l'utilisateur à comparer ont effectué un rating [6].

Autrement dit, pour les cas de rareté (sparsity), lorsque peu d'items ont été évalués par les utilisateurs et qu'un filtrage collaboratif n'est pas possible, ce qu'on fait, c'est qu'on assigne en premier lieu un pseudo-rating ou vote artificiel par défaut à l'utilisateur sur les contenus disponibles en utilisant préalablement un algorithme basé sur le contenu, puis on applique sur la matrice (contenant peu de vrais rating et beaucoup de pseudo-ratings) un filtrage collaboratif [6].

1.6 Conclusion

Dans ce chapitre nous avons dressé un état de l'art relatif à l'e-learning, aux réseaux sociaux, au web sémantique, aux ontologies, ainsi qu'aux systèmes de recommandation.

Cet état de l'art nous a donc permis de comprendre et de bien cerner les différentes notions dont nous aurons besoin pour la conception de notre approche, à savoir les différents types de systèmes de recommandation, les réseaux sociaux, l'e-learning, etc.

Chapitre 2

Analyse et Conception

2.1 Introduction

De nos jours la formation est un processus qui nous accompagne tout au long de notre vie. L'utilisation du e-Learning et plus particulièrement dans sa forme basée-Web gagne en popularité dans les institutions et entreprises issues des milieux académiques, des affaires et des gouvernements. Cependant, à mesure que cette solution se démocratise, les ressources pédagogiques mises à disposition des apprenants augmentent considérablement, rendant la tâche de découverte de données très difficile.

Nous proposons par le biais de notre approche, une solution pour ce problème crucial, afin d'aider les apprenants à mieux s'y retrouver dans les entrepôts de ressources mis à leur disposition, et ainsi avoir accès à l'information voulue en un temps minime, de manière personnalisée, et ce, selon les intérêts et préférences de chacun.

Les objectifs que nous voulons atteindre avec notre approche sont les suivants :

- Permettre aux apprenants de découvrir des ressources pédagogiques pouvant les intéresser en se basant sur leurs intérêts, leurs préférences et leurs activités, sans pour autant leur imposer d'innombrables questionnaires afin de cerner leurs attentes.
- Permettre aux apprenants d'évaluer des contenus et d'exprimer leurs opinions de manière quantitative ou textuelle.
- L'exploitation des activités des apprenants dans les réseaux sociaux afin de déduire leurs centres d'intérêts.

Nous ferons dans ce qui suit, une analyse ainsi qu'une étude de quelques approches exis-

tantes pour chacun des sous-systèmes qui composeront notre système de recommandation final, et nous finirons par une présentation de l'approche que nous avons finalement choisi d'adopter et d'une description de sa conception.

2.1.1 Approche générale

L'approche qu'on souhaite adopter dans notre application pour pouvoir atteindre l'objectif principal fixé, qui est de recommander des ressources susceptibles d'intéresser un apprenant, est de construire un système de recommandation basée sur les centres d'intérêts et le profil de chaque apprenant, ainsi que les appréciations d'autres utilisateurs jugés similaires à cet apprenant.

Notre approche n'est pas considérée comme étant hybride, du fait que les deux méthodes utilisées, le filtrage collaboratif et la recommandation basée sur le contenu, sont totalement indépendantes l'une de l'autre.

Les raisons pour lesquelles nous implémentons ces deux méthodes sont les suivantes :

- Le filtrage collaboratif souffre du problème de démarrage à froid (cold start), qui signifie que tout nouvel utilisateur, n'ayant jamais noté de ressources, n'aura pas accès à une recommandation basée sur les notes.
- Même problème pour les nouvelles ressources qui n'ont jamais été notées par des apprenants, et qui risquent donc de ne jamais apparaitre dans une liste de recommandation du fait qu'elles possèdent très peu ou pas de notes
- Les systèmes de recommandation basée sur le filtrage collaboratif se basent uniquement sur la notation de ressources, et ignorent ainsi les activités et centres d'intérêts des apprenants sur la plateforme de formation, ou sur d'autres sites.
- Les systèmes de recommandation basée sur le contenu ont tendance à recommander des ressources ayant toujours des contenus similaires, empêchant ainsi l'apprenant à découvrir des ressources de domaines et thématiques différentes.

2.1.2 Fonctionnement général

L'application que nous souhaitons développer est un système de recommandation de ressources pédagogiques, couplé à la plateforme de formation MOODLE. Elle est répartie en plusieurs modules, qui sont les suivants :

- **La plateforme MOODLE** : Tout nouvel utilisateur souhaitant s'inscrire sur la plateforme, aura à introduire une série d'informations, telles que son pseudonyme, son mot de passe, son rôle (enseignant, apprenant), ses centres d'intérêts, etc.
- **Un réseau social** : est généralement utilisé pour communiquer avec des amis ou des collègues ou même des inconnus, dans un but d'échange et de découverte de nouvelles ressources et informations.
- **Un système de recommandation** : Un apprenant enregistré sur la plateforme de formation pourra utiliser le système de recommandation pour voir les éventuelles ressources qu'on lui recommande.

Les différents composants que nous essayerons d'étudier dans cette section, et qui font partie intégrante de la conception et réalisation de notre approche sont les suivants :

1. Les systèmes de recommandations basée sur le filtrage collaboratif.
2. Les systèmes de recommandation basée sur le contenu.
3. L'analyse des réseaux sociaux.
4. L'analyse des sentiments.

2.2 Système de recommandation basée sur le filtrage collaboratif

Comme défini dans l'état de l'art, le filtrage collaboratif consiste à faire des recommandations de ressources à un apprenant en se basant sur le comportant passé des apprenants ayant des goûts et intérêts similaires à lui, cela signifie que le système se basera sur les appréciations de l'apprenant actif pour déduire l'ensemble des apprenants qui ont les mêmes goûts et préférences que lui, qu'on appellera "**voisins**" tout au long de notre travail.

Afin de concevoir un système capable de recommander des ressources à un apprenant, nous devons procéder par les étapes suivantes :

1. Collecter les préférences des différents apprenants de la plateforme d'apprentissage.
2. Trouver la similarité entre les différents apprenants afin d'établir l'ensemble des voisins de chaque apprenant.
3. Recommander les ressources ayant la plus grande probabilité de satisfaire les besoins des apprenants.

2.2.1 Collecte des préférences des apprenants

La première chose dont nous avons besoin dans la conception de notre système est de trouver un moyen pour représenter les différents apprenants de la plateforme et leurs préférences.

Dans notre cas, où nous utilisons la plateforme de formation MOODLE, nous utiliserons la base de données MOODLE pour collecter l'ensemble des ressources consultées par l'apprenant ainsi que les notes attribuées à chacune d'elles. Pour rappel, l'appréciation d'un apprenant au sujet d'une ressource peut être donnée de manière **quantitative** ou **textuelle**. Dans le cas d'une appréciation textuelle, une analyse de l'opinion est nécessaire pour déduire la note quantitative correspondante à cette opinion donnée sous forme textuelle.

Une fois ces données collectées, une matrice contenant toutes les données concernant tous les apprenants de la plateforme est construite, avec comme champs : Identifiant de l'utilisateur, Identifiant de la ressource et la Note attribuée. Pour les ressources non consultées par un apprenant on laisse la case vide.

2.2.2 Etablissement du voisinage des apprenants

Après avoir collecté les données concernant les appréciations des apprenants sur les différentes ressources de la plateforme, nous avons besoin de déterminer à quel degré les apprenants ont des goûts et appréciations similaires. Pour cela chaque apprenant est comparé aux autres apprenants en calculant **la mesure de similarité** entre eux. Il existe différentes manières pour calculer cette mesure, mais dans notre travail nous allons faire la comparaison des deux (02) méthodes les plus connues, et en choisir la meilleure pour la suite. Ces deux (02) méthodes sont : La distance Euclidienne et le coefficient de corrélation de Pearson.

2.2.2.1 La distance Euclidienne

Une manière très simple de calculer la mesure de similarité est d'utiliser la distance euclidienne, qui prend en entrée les items que des utilisateurs ont notés, et les utilise comme axes d'un graphe. On peut ensuite, placer les différents utilisateurs sur le graphe et voir à quel point ils sont rapprochés. Plus les utilisateurs sont rapprochés sur le graphe plus ils sont similaires, et vice-versa.

Comme exemples illustratifs de cette méthode et de celle du coefficient de Pearson, nous utiliserons les résultats obtenus dans [22].

[22] utilise comme exemple une petite base de données d'utilisateurs ayant noté des films qu'ils ont déjà vus, La matrice des préférences des utilisateurs sur les films se présente comme suit :

Utilisateurs\Films	Lady in the Water	Snakes on a Plane	Just My Luck	Superman Returns	You,You, Me and DupreeMe and Dupree	The Night Listener
Lisa Rose	2.5	3.5	3.0	3.5	2.5	3.0
Gene Seymour	3.0	3.5	1.5	5.0	3.5	3.0
Michael Phillips	2.5	3.0		3.5		4.0
Claudia Puig		3.5	3.0	4.0	2.5	4.5
Mick LaSalle	3.0	4.0	2.0	3.0	2.0	3.0
Jack Matthews	3.0	4.0		5.0	3.5	3.0
Toby		4.5		4.0	1.0	

TABLE 2.1: Matrice des notes attribuées par des utilisateurs à des films

La figure ci-dessous illustre l'espace de préférence des utilisateurs concertant les deux films "Snakes on a Plane" et "You, Me and Dupree" avec les notes représentées dans le tableau précédent :

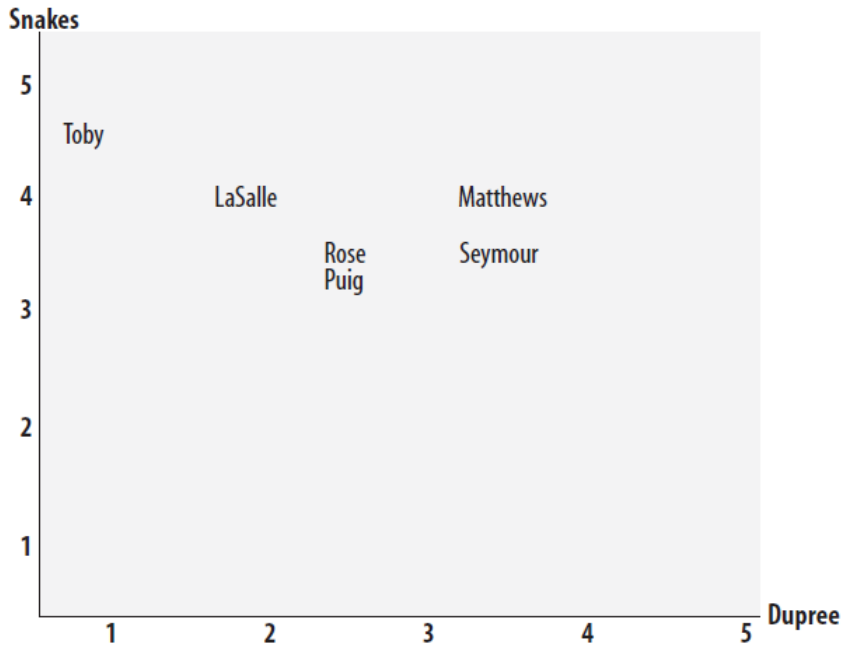

FIGURE 2.1: Des utilisateurs dans un espace de préférences.

On remarque dans ce graphe que : plus les personnes sont proches dans l'espace de préférences, plus leurs préférences sont similaires. Mais comme le graphe est bidimensionnel, on ne peut représenter que deux films à la fois, toutefois le principe reste le même avec tous les autres films.

Pour calculer mathématiquement la corrélation entre deux personnes avec la distance euclidienne, on utilise la formule suivante :

$$\sqrt{\sum (x_i - y_i)^2}$$

Où : x_i représente la note attribuée par l'utilisateur X à la ressources i

et y_i la note attribuée par l'utilisateur Y à la ressource i.

La valeur de similarité entre "Toby" et " Mick LaSalle" donnée par la formule de la distance euclidienne est 0.14 et celle de "Lisa Rose" et "Jack Matthews" est de 0.34.

2.2.2.2 Le coefficient de corrélation de Pearson

C'est une manière plus sophistiquée pour le calcul de la similarité entre les intérêts des utilisateurs. Le coefficient de corrélation est une mesure qui illustre la similarité de deux ensembles de données sur une ligne droite. La formule pour le calcul de ce coefficient

est plus compliquée que celle de la distance euclidienne, mais les résultats obtenus sont meilleurs, surtout dans le cas où les données ne sont pas bien normalisées, par exemple, lorsque les notes données par un utilisateur à des ressources sont plus sévères que la normale.

La formule suivante permet de calculer ce coefficient :

$$r = \frac{\sum_{i=1}^{n}(X_i - \bar{X})(Y_i - \bar{Y})}{\sqrt{\sum_{i=1}^{n}(X_i - \bar{X})^2}\sqrt{\sum_{i=1}^{n}(Y_i - \bar{Y})^2}}$$

Où : X_i et Y_i sont les notes de l'item i données par les utilisateurs X et Y respectivement.

$\bar{X}$ et $\bar{Y}$ sont les moyennes des notes attribuées par les utilisateurs X et Y respectivement.

Pour visualiser cette méthode, nous procédons à l'établissement d'un graphe qui contient en axe les notes de deux utilisateurs données à chaque film vu, comme illustré dans la figure ci-dessous, "Superman" a été noté 3 par "Mick LaSalle" et 5 par "Gene Seymour", donc il est placé à l'emplacement (3,5) sur le graphe :

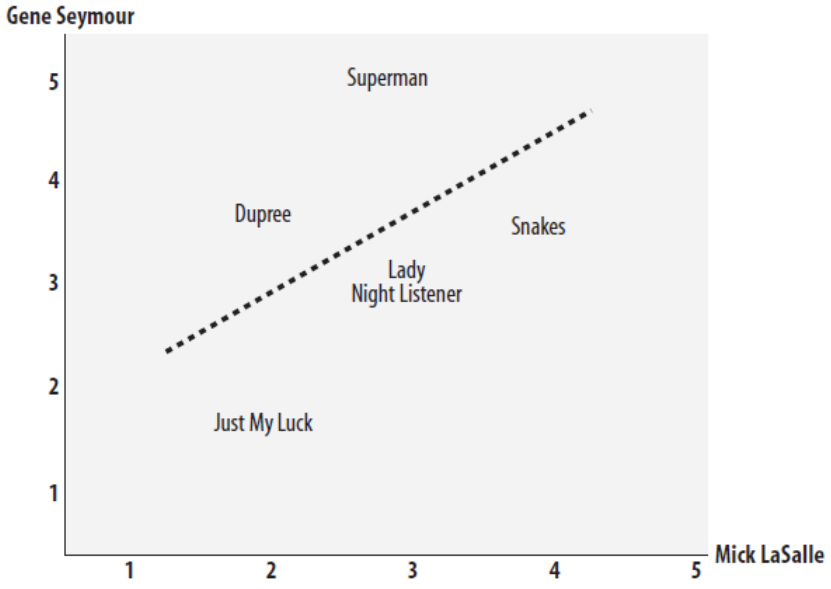

FIGURE 2.2: Comparaison des notes données par deux utilisateurs

Nous pourrons également remarquer sur ce graphe une ligne droite, appelée "the best-fit line" en français "la ligne de meilleur ajustement" car elle est placée le plus proche possible de tous les items notés. Si les notes attribuées par les deux utilisateurs sont identiques, la ligne serait diagonale et toucherait tous les items du graph, donnant une parfaite corrélation avec un coefficient de 1. Mais comme illustré dans la figure précédente, les notes ne sont pas tout à fait similaires, donnant un coefficient de corrélation de 0.4. La figure suivante montre une meilleure corrélation des notes données par les utilisateurs "Lisa Rose" et "Jack Matthews".

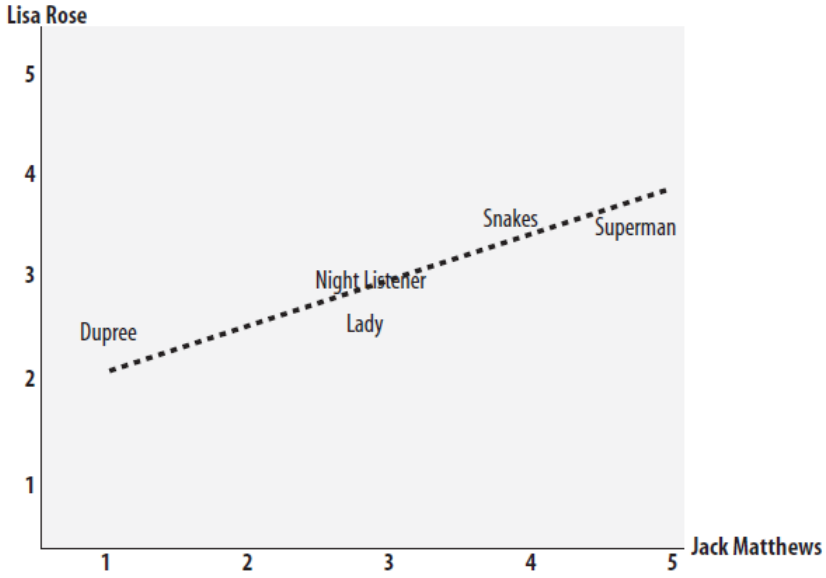

FIGURE 2.3: Illustration d'une forte corrélation entre les notes de deux utilisateurs

L'aspect intéressant dans le coefficient de corrélation de Pearson, qu'on remarque dans la figure précédente, est qu'il corrige les éventuelles inflations des notes. Comme on le remarque dans la figure "Jack Matthews" tend à donner des notes plus élevées que " Lisa Rose", qui a tendance à être plus sévère dans sa notation, mais cela n'empêche pas le fait que ces deux utilisateurs soient relativement similaires. La méthode de la distance euclidienne, décrite plus haut, aurait plus tendance à dire que ces deux utilisateurs ne sont pas similaires, car l'un d'eux a noté les items plus sévèrement que l'autre, même si ces deux utilisateurs ont des goûts très rapprochés.

2.2.3 Recommandation de ressources

Pour effectuer une recommandation de ressources qui pourraient intéresser un utilisateur, une note approximative est calculée. Cette note représente la note que pourrait attribuer l'utilisateur à une ressource qu'il n'a pas encore consultée. La méthode suivie pour le calcul de cette note est tirée de [22], elle consiste à :

- Multiplier chaque note, d'une ressource que l'utilisateur actif n'a pas encore consultée, par la mesure de similarité entre l'utilisateur actif et l'utilisateur ayant attribué cette note. Ce calcul est nécessaire pour valoriser les votes des utilisateurs qui sont similaires à l'utilisateur actif des autres qui sont différents.
- Sommere toutes ces nouvelles notes calculées. Cette somme aurait pu servir de note approximative de l'utilisateur actif, mais elle avantagerait plus les ressources ayant eu un nombre de vues importants. Pour corriger cela, on divise cette somme par la somme de toutes les mesures de similarité de l'utilisateur actif avec les autres utilisateurs ayant noté la ressource. Le résultat obtenu est la note approximative que pourrait donner l'utilisateur actif à la ressource concernée.

Le tableau suivant illustre ce processus pour une meilleure compréhension de la méthode suivie :

Critic	Similarity	Night	S.xNight	Lady	S.xLady	Luck	S.xLuck
Rose	0.99	3.0	2.97	2.5	2.48	3.0	2.97
Seymour	0.38	3.0	1.14	3.0	1.14	1.5	0.57
Puig	0.89	4.5	4.02			3.0	2.68
LaSalle	0.92	3.0	2.77	3.0	2.77	2.0	1.85
Matthews	0.66	3.0	1.99	3.0	1.99		
Total			12.89		8.38		8.07
Sim. Sum			3.84		2.95		3.18
Total/Sim. Sum			3.35		2.83		2.53

FIGURE 2.4: Création d'une recommandation pour l'utilisateur 'Toby'

Ce tableau montre la mesure de similarité de chaque utilisateur avec "Toby", et les notes attribuées aux films : The Night Listener, Lady in the Water et Just My Luck, que "Toby" n'a pas encore vus.

Les colonnes qui commencent par "S.x" représentent la similarité multipliée à la note. La ligne "Total" montre la somme de toutes ces notes multipliées. La ligne "Sim.Sum" est la somme des similarités, sachant que lorsqu'un utilisateur n'a pas noté un film, sa similarité avec "Toby" n'est pas prise en compte dans le calcul de la somme, et enfin la ligne "Total/Sim.Sum" est le résultat de la division de "Total" sur "Sim.Sum", qui est la note finale que pourrait attribuer "Toby" à chaque film qu'il n'a pas encore vu.

En fonction de l'application, on pourrait choisir d'afficher toutes ces notes à l'utilisateur et lui laisser libre choix de décider quelle ressource il souhaite consulter, ou alors imposer le seuil qu'une note ne doit pas dépasser afin que la ressource soit affichée dans les recommandations de l'utilisateur.

2.3 Système de recommandation basée 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.

La plupart des systèmes de recommandation basée sur le contenu identifient les ressources similaires aux ressources qu'un utilisateur donné a appréciées. Ainsi, quand de nouvelles ressources sont introduites dans le système, elles peuvent être recommandées directement sans que cela ne nécessite un temps d'intégration comme cela est le cas dans le cadre des systèmes de recommandation basée sur le filtrage collaboratif.

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.

Profils de ressource [7]

Les profils de ressource consistent en un ensemble d'attributs décrivant les ressources, la précision de cette approche est hautement dépendante de la nature des ressources : elle est beaucoup plus élevée pour des ressources textuelles 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 métadonnées sont utilisées. Par conséquent, la plupart des recherches sur la recommandation basée sur le contenu portent sur des données textuelles.

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 la lemmatisation, ou encore appelée le *stemming*. La lemmatisation 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 times inverse document-frequency* ou **TFIDF**.

Une fois cette étape finalisée, le système dispose d'une matrice terme-document, qui représente les mots les plus importants ou les plus informatifs de chaque ressource, avec comme valeurs les poids associés à chaque terme de chaque ressource.

Profils d'utilisateur [7]

Le profil d'un utilisateur définit ses centres d'intérêt. Un tel profil 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 consultées.

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, etc. Mais les centres d'intérêts des utilisateurs peuvent évoluer au cours du temps, ce qui impose une actualisation manuelle régulière, un autre 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 à partir 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, le coefficient de Dice ou encore le produit scalaire.

L'approche brièvement décrite ci-dessus est détaillée dans le reste de cette section.

2.3.1 Profils de ressource [11]

2.3.1.1 Collecte des descriptions des ressources

Dans notre approche, le système de recommandation se basera sur la description des ressources pédagogiques, pour ensuite les représenter en vecteur de termes, et ce au lieu du contenu de la ressource en en lui-même.

La collecte des descriptions de chaque ressource se fait à partir de la base de données de notre plateforme de formation MOODLE.

2.3.1.2 Lemmatisation

Nous avons expliqué brièvement plus haut ce qu'était la lemmatisation. Cette procédure est nécessaire pour capturer la similarité sémantique des mots, en les transformant en un lemme commun. Ainsi, nous devons regrouper et unifier la représentation des mots de la même famille (nom, pluriel, verbe a l'infinitif...) par la lemmatisation. Par exemple, les mots : "intelligente" et "intelligemment" seront tous les deux unifiés par le terme "intelligent".

2.3.1.3 Création de la matrice termes-documents et calcul du TFIDF

La procédure de génération des recommandations de notre système sera basée sur le calcul de similarité entre les ressources. Ce calcul nécessite premièrement la construction d'une matrice termes-documents contenant les fréquences brutes, puis d'une matrice TFIDF où les fréquences sont transformées en multipliant la fréquence par le poids (TFIDF).

Matrice termes-documents

Les descriptions sont représentées par un ensemble de termes lemmatisés. On crée ensuite une matrice termes-document dans laquelle chaque colonne correspond à un terme

unique et chaque ligne représente la description d'une ressource. Chaque cellule contient la fréquence d'apparition du terme dans la description.

La matrice doit contenir les fréquences d'apparition FA de chaque terme dans chaque ressource. Si nous avons un terme T2 qui apparait dans la description de la ressource C1 et C2 mais pas dans la description de la ressource C3, on écrit 1 dans les cellules qui associent les ressource C1, C2 avec le terme T2 et 0 dans la cellule qui associe C3 avec ce même terme. Comme illustré dans le tableau suivant :

	T1	T2	T3	T4	T5	T6	T7	T8	T9
C1	1	1	1	1	0	0	1	0	0
C2	1	1	0	0	0	1	1	1	1
C3	1	0	0	0	1	0	1	0	1

FIGURE 2.5: Matrice termes-documents.

Calcul du TFIDF

Pour une tâche de recherche d'information, tous les termes n'ont pas la même importance. En général, plus un terme est rare, plus il permettra d'identifier des documents spécifiques. Il s'avère donc pertinent d'attribuer un poids à chaque terme, qui exprime l'importance de chacun.

Le calcul du poids est basé sur l'idée qu'un terme qui n'apparaît pas fréquemment doit avoir un poids plus élevé qu'un terme qui apparaît souvent. Les poids des termes sont calculés en utilisant la fréquence inverse du document (IDF) qui correspond à un terme donné. Le calcul de la fréquence inverse de document pour un terme $t_i$correspond à cette formule :

$$idf_i = \log \frac{\mathrm{N}}{|\{d_j : t_i \in d_j\}|}$$

N : est le nombre de documents.

$|\{d_j : t_i \in d_j\}|$: est le nombre de documents qui contiennent le terme t_i.

Finalement, le poids s'obtient en multipliant les deux mesures :

$tf_i df_{ij} = tf_{ij}.idf_i$

où : tf_{ij} = la fréquence d'occurrences du terme t_i dans un document D_j.

Le calcul des similarités s'effectue ainsi sur les fréquences transformées TFIDF plutôt que sur les fréquences brutes, qu'il s'agisse du cosinus ou d'autres mesures de similarité.

2.3.2 Profils d'utilisateur

Les profils d'utilisateurs sont utilisés par de nombreux systèmes à des fins de recommandation. De tels profils consistent en différents types d'informations, ces informations peuvent être [17] :

- Un modèle des préférences de l'utilisateur, qui décrit les centres d'intérêts de l'utilisateur. Il y a plusieurs manières de représenter ce genre d'informations, mais la plus commune est de représenter les centres d'intérêt des utilisateurs par des vecteurs de termes à partir des vecteurs de termes représentant les ressources que l'utilisateur a appréciées.
- L'historique des interactions de l'utilisateur sur la plateforme de formation, ou sur un réseau social. Cela pourrait être une liste des ressources consultées par l'utilisateur, ou alors des ressources qu'il a notées. D'autres types d'historiques conservant les requêtes tapées par l'utilisateur.

Ces différentes informations seront utilisées par le système de recommandation pour créer un modèle utilisateur et ensuite se baser sur ce modèle pour créer des recommandations.

Différentes méthodes sont utilisées pour réunir l'ensemble des informations nécessaires à la création d'un modèle utilisateur ; des méthodes manuelles et automatiques :

1. La première méthode, *manuelle*, consiste en une interface dans le système qui permet à l'utilisateur de préciser ses intérêts et ses préférences. Cela peut être des cases à cocher (par exemple le genre de musique qu'il apprécie) ou alors un espace pour décrire par des mots ses intérêts. Une fois que l'utilisateur a spécifié ces informations, le système génère les ressources ayant des caractéristiques en commun avec ces dernières, et les affiche à l'utilisateur.
2. La seconde méthode, *automatique*, consiste à déduire les intérêts d'un utilisateur par ses activités et interactions, plus communément appelés son historique de navigation. Cela peut être fait par des retours explicites de l'utilisateur, en anglais feedbacks, qui peuvent être positifs (like) ou négatifs (dislike), ou alors des retours implicites, comme le téléchargement d'une ressource ou l'achat d'un article, qui peuvent être vus comme une appréciation positive implicite.

2.3.2.1 Méthodes d'extraction automatique de profils utilisateurs

De nombreuses méthodes d'extraction automatique de profils, dans le domaine de la recherche d'information, ont été proposées. Dans le cadre de la recommandation de ressources, des probabilités sont calculées selon lesquelles un utilisateur donné appréciera ou non une ressource. Cela est généralement considéré comme un problème de classification où chaque classe représente un niveau d'appréciation (ex. « 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 [7].

– **Arbres de décision**

Un arbre de décision est obtenu en séparant de façon récursive les ressources en sous-groupes homogènes relatifs 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.

– **Classificateur naïf de Bayes**

Le principe du classificateur naïf de Bayes est de déterminer la classe C pour laquelle la probabilité $P(C|\theta_1, \ldots, \theta_k)$ qu'une ressource r ayant pour attributs $(\theta, \ldots, \theta_k)$ appartienne à cette classe C soit maximale. Les attributs sont supposés indépendants, et maximiser $P(C|\theta_1, \ldots, \theta_k)$ revient à maximiser la formule suivante :

$$P(C)\prod_{i=1}^{k} P(\theta_i|C)$$

Les valeurs de $P(C)$ et de $P(\theta_i|C)$ sont estimées à partir d'un corpus d'apprentissage. Pour chaque ressource r, chaque valeur de la formule précédente est estimée pour chaque classe (ici ces classes sont les 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 avec 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.

– **Réseaux de neurones**

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

$F(x_1, \ldots, x_k; w_1, \ldots, w_k) = \lfloor \sum_{i=1}^{k} w_i x_i \rfloor$

où les variables correspondent à des poids à associer aux variables $x_1, \ldots, x_k$ qui sont déterminés à partir d'un corpus d'apprentissage. Les neurones sont groupés en réseau selon deux types d'architectures : 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 $x_1, \ldots, x_k$ 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 perception multi-couche. 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 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 a l'avantage de permettre un ajustement particulièrement fin. Selon le domaine d'application il peut s'avérer plus ou moins efficace que ses alternatives.

2.3.3 Recommandation de ressource sur la base du calcul de la similarité

Dans l'approche que nous adoptons, pour avoir une liste de ressources à recommander à un apprenant, nous calculons la similarité entre la matrice termes-documents des ressources disponibles dans la base de données, avec le profil des apprenants. Dans notre cas, nous nous limitons à la méthode manuelle pour l'extraction des profils des apprenants, en nous basant sur les centres d'intérêts explicités lors de leur inscription. On se chargera de mettre à jour, au fur et à mesure de leur navigation sur la plateforme de formation, leurs intérêts par l'extraction des vecteurs de termes qui représentent chaque ressource qu'ils auront appréciées.

2.3.3.1 Mesures de calcul de la similarité

Pour le calcul de similarité entre vecteurs de termes, plusieurs mesures existent, mais nous n'en retenons que trois (03) [11] :

– **Le produit scalaire**

Le produit scalaire entre la requête Q et un document D_j est défini comme :

$$SC(\boldsymbol{Q}, \boldsymbol{D}_j) = \sum_{i=1}^{t} tfq_i \times tf_{ij}$$

où : Q : le vecteur des termes de la requête.

D_j : vecteur des termes du document j.

tf_{ij} :la fréquence d'apparition du terme i dans le document j.

tfq_i : la fréquence du terme i dans le vecteur requête Q.

– **Le cosinus**

Le calcul de cosinus normalise le résultat par rapport à la longueur du document et de la requête. Alors que le produit scalaire tend à augmenter avec la longueur du document, le cosinus pénalise les documents longs en favorisant la proportion relative de termes communs. Autrement dit, plus l'angle est fermé, plus on est proche.

$$cos(\boldsymbol{Q}, \boldsymbol{D}_j) = \frac{\boldsymbol{SC}(\boldsymbol{Q}, \boldsymbol{D}_j)}{\sqrt{\sum_{j=1}^{t} tf_{ij}^2} \sqrt{\sum_{j=1}^{t} tfq_i^2}}$$

– **Le coefficient de Dice**

Ce coefficient peut être défini comme deux fois l'information partagée entre un document D et une requête Q sur la somme des cardinalités.

$$S = \frac{2\,|\boldsymbol{D} \cap \boldsymbol{Q}|}{|\boldsymbol{D}| + |\boldsymbol{Q}|}$$

Habituellement, lorsque nous avons des vecteurs dans l'espace très proches de celui de la requête Q et qu'ils contiennent presque les mêmes termes, les documents représentés par ces vecteurs sont les documents pertinents.

2.4 Analyse des réseaux sociaux

Etant donné l'essor connus ces dernières années par le web 2.0 et l'avènement des applications de réseaux sociaux qui ont considérablement accru la participation, les interactions et le partage entre les utilisateurs du web, l'analyse et la compréhension de telles applications suscitent de vifs intérêts au sein de plusieurs communautés scientifiques afin d'extraire les usages des utilisateurs sur ces réseaux, et en faire une analyse pour exploiter au mieux ces informations et améliorer l'utilisation du web et de ses ressources.

L'analyse des réseaux sociaux inspire depuis peu la recommandation de contenu. En effet, le web social contient aujourd'hui une grande quantité de données sous forme de textes générés par les utilisateurs. Ce sont les statuts, les commentaires, les tags, etc. qu'ils laissent sur les différentes plateformes des réseaux sociaux. Les tags laissés sur ces sites sont souvent utilisés comme une représentation des intérêts des internautes, des profils sont construits à partir de ces derniers, ou alors sont utilisés pour détecter les communautés qui se forment autour de sujets ou domaines particuliers, pour ensuite définir les différentes communautés auxquelles appartient un utilisateur.

Les commentaires sont eux généralement utilisés pour déduire l'appréciation ou l'opinion de l'utilisateur sur la ressource ou le sujet qu'il commente, ainsi, on peut également déduire ses différents centres d'intérêts à partir de ces commentaires.

2.4.1 Découverte de communautés par l'analyse des usages et des tags

Depuis quelques années, le Web s'est transformé en une plateforme d'échange générique, où tout utilisateur devient un fournisseur de contenu par le biais de technologies comme les commentaires, les blogs et les wikis. Ce nouveau Web collaboratif ou participatif (Web 2.0) permet de construire des réseaux sociaux selon ses relations professionnelles

ou selon ses intérêts. Cependant, ces sites exigent de chaque utilisateur une description explicite de son réseau social ou de son profil. De plus, seules les communautés ainsi explicitées sont identifiées.

Or un grand nombre de communautés d'utilisateurs existent de façon implicite dans de nombreux domaines. Découvrir et identifier précisément ces communautés implicites est un gain pour de nombreux acteurs : le propriétaire du site, les régies publicitaires en ligne et surtout, les utilisateurs du système.

Les algorithmes classiques de détection de communautés dans les réseaux sociaux utilisent l'information structurelle pour détecter des groupes, les plus utilisés sont ceux basés sur la topologie du graphe des relations de ces réseaux.

Dans cette section, nous proposons une méthode de détection de communautés générique, car elle ne s'appuie que sur une **annotation des ressource**s et sur l'utilisation de ces ressources par les utilisateurs.

2.4.1.1 Démarche suivie pour la détection de communautés

L'objectif de l'approche abordée dans cette partie est de scinder les utilisateurs en communautés distinctes, en se basant sur les groupes de tags qu'ils apprécient. Chaque ressource $r \in R$ est annotée par un ensemble de tags $t_j \in T$. Ces annotations proviennent des fournisseurs de ressources.

Les utilisateurs sont supposés émettre des votes sur des ressources du site, ces votes ne sont pas nécessairement explicites, et peuvent être obtenus en se basant sur les usages des utilisateurs (la musique qu'ils sélectionnent, les achats qu'ils effectuent, les ressources qu'ils annotent ou recommandent, etc.).

Etant donnés les votes des utilisateurs sur des ressources et les annotations de ces ressources, soit $A(u_i) \subseteq R$ l'ensemble des ressources intéressant l'utilisateur $u_i \in U$ et $A(u_i, t_j) \subseteq R$ l'ensemble des ressources intéressant l'utilisateur u_i et annotées par le tag t_j.

Pour pouvoir scinder les utilisateurs en communautés en se basant sur le groupe de tags qu'ils apprécient, il est nécessaire de savoir à quel degré un utilisateur manipule des documents annotés par les tags t_j. Ce degré est appelé le coefficient d'appartenance x_{ij} d'un utilisateur u_i à un tag t_j, et il est calculé de la manière suivante :

$x_{ij} = \frac{|A(u_i,t_j)|}{|A(u_i)|}$

Plus ce coefficient est proche de 1, plus l'utilisateur i manipule des tags de type j.

Ce coefficient sera utilisé à la fin de l'expérimentation, une fois que les différentes communautés existantes seront découvertes, pour savoir à quelle communauté appartient l'utilisateur.

Découverte des communautés de tags

Une communauté de tags consiste à rassembler l'ensemble des tags similaires dans un domaine, une thématique ou un genre en commun.

Dans cette approche, il est considéré que tout usage d'une ressource portant un tag donné, est vu comme la réalisation d'une variable aléatoire représentant ce tag. Les intérêts de chaque utilisateur sont alors autant de réalisations indépendantes des m variables représentant les m tags possibles.

Pour rassembler les tags similaires, une technique appelée "**L'analyse en composante principale (ACP)**" est utilisée. Son objectif est de trouver des combinaisons linéaires des variables représentants les tags pour expliquer au mieux les intérêts des utilisateurs.

Ainsi, à chaque utilisateur u_i, nous associons le vecteur de ses degrés d'appartenance à chaque tag comme suit :

$X_i = (x_{i1}, x_{i2}, \ldots, x_{im})$.

Ce vecteur représente le positionnement de l'utilisateur dans l'espace des tags, et l'ensemble de tous les vecteurs X_i donne ainsi un nuage de points dans l'espace des tags.

De la même manière, on peut associer à chaque tag t_j le vecteur V_j, correspondant à ses degrés d'appartenance chez les n utilisateurs :$V_j = (x_{1j}, x_{2j}, \ldots, x_{ij}, \ldots, x_{nj})$.

Ces nuages de points sont difficilement analysables, à cause des dimensions considérées (nombre de tags, nombre d'utilisateurs) et de la variabilité des observations. C'est pour remédier à ce problème que l'ACP a été utilisée dans cette approche, elle va ainsi permettre une projection du nuage de points utilisateurs (initialement exprimés dans un espace de dimension k) sur des plans à deux (02) dimensions, qui reconstituent au mieux la variabilité entre les utilisateurs.

Ainsi, l'ACP permet la représentation des nuages de tags dans un espace de dimension réduite, en identifiant des axes explicatifs, tout en minimisant la perte d'information

effectuée lors de cette simplification. Les axes choisis lors de cette représentation sont appelés "**composantes principales**".

La méthode de rassemblement des tags est alors la suivante :

L'ACP fournit les composantes principales pertinentes pour l'analyse des usages. Selon chacune de ces composantes, on ignore les tags situés dans la zone de faible corrélation (corrélation entre $-\alpha$ et $+\alpha$, pour un seuil $\alpha \in]0;1]$ fixé). Les tags restants, situés dans les zones de forte corrélation (inférieure à$-\alpha$ ou supérieur à$+\alpha$), sont rassemblés dans une même communauté de tags.

L'algorithme suivant, tiré de [16], résume cette méthod :

Algorithme 1 : Découverte

entrées : Vecteurs V_j, seuil de décision α
sorties : Communautés de tags $G_1, \ldots, G_K$

```
1 début
2   identifier les composantes principales C = ((c1, c2), (c3, c4)...), expliquant la plus
    grande proportion de la variabilité des données
3   tant que (il reste des composantes principales (c, c') dans C) faire
4     ignorer les tags non corrélés (|coordonnées selon c et c'| < α)
5     rassembler dans une même communauté les tags corrélés selon c (|coordonnées
      selon c| > α)
6     rassembler dans une autre communauté les tags corrélés selon c' (|coordonnées
      selon c'| > α)
7     supprimer ces tags
8   fin tant que
9 fin
```

Découverte de communautés d'utilisateurs

Une fois l'ensemble des tags T décomposé en K communautés de tags $G_1, \ldots, G_k$, on en déduit les communautés d'utilisateurs. Pour cela, pour un utilisateur u_i donné, on calcule son degré d'appartenance x'_{ij} à chaque communauté de tag G_j de la manière suivante :

$x'_{ij} = \sum_{t_k \in G_j} x_{ik}$

Sa communauté $C(u_i)$ est alors sa communauté de tag majoritaire, c'est à dire l'indice j tel que x'_{ij} soit maximal. Chaque utilisateur est alors associé à ce groupe de tags. Ce groupe aura comme intitulé l'ensemble des tags qui le constituent.

2.4.1.2 Exemple détaillé de l'approche

Comme exemple applicatif de cette méthode, nous utilisons les résultats obtenus dans [16]. La méthode a été testée sur la base de films MovieLens.

Cette base contient 100 000 votes pour 1 682 films appréciés par 943 utilisateurs.

La matrice M des votes de l'ensemble des utilisateurs U sur l'ensemble des films R a été construite, et le degré d'appartenance des utilisateurs aux différents tags a été calculé.

L'ensemble des tags utilisés sont : 1 : Aventure, 2 : Enfant, 3 : Animation, 4 : Mystère, 5 : Crime, 6 : Drame, 7 : Fiction, 8 : Filmnoir, 9 : Fantasy, 10 : Musical, 11 : Action, 12 : Thriller, 13 : Romance, 14 : Comédie, 15 : Horreur, 16 : Guerre, 17 : Documentaire, 18 : Western. Le seuil de décision α a été fixé à 0,6 de façon empirique.

A. Matrice de corrélation

La première étape de l'analyse consiste à établir l'ensemble des composantes principales, qui correspondent à des facteurs. Ces facteurs sont des combinaisons linéaires des variables initiales, les coefficients de ces combinaisons sont donnés par les coordonnées des vecteurs propres (changement de base).

Le résultat fondamental concernant les variables est le tableau des corrélations variables-facteurs. Il s'agit des coefficients de corrélation linéaire entre les variables initiales et les facteurs. Ce sont ces corrélations qui vont permettre de donner un sens aux facteurs (de les interpréter).

Le tableau de la figure ci-dessous représente la matrice de corrélation entre une partie des variables initiales et les 6 premiers facteurs (composantes principales) sélectionnés (la méthode de sélection des facteurs est expliquée dans le point suivant) :

Tag	1	2	3	4	5	6
Aventure	**,777**	,349	-,272	,081	,037	-,056
Enfant	**,675**	-,231	*,465*	,187	-,145	-,147
Animation	**,657**	-,200	*,391*	,311	-,052	-,218
Mystère	**-,657**	,258	*,367*	,173	-,254	-,057
Crime	**-,624**	,265	,094	,226	,237	-,016
Drame	**-,614**	-,561	-,230	,094	,016	-,112
Fiction	,535	,531	-,252	,080	,249	-,152
Filmnoir	-,512	,066	,209	**,490**	,083	,158
Fantasy	,479	-,108	,197	-,076	,208	-,022
Musical	,422	-,409	,380	,372	-,193	,096
Action	,451	**,746**	-,262	-,117	-,128	,028
Thriller	-,393	**,704**	,314	-,176	-,221	,011
Romance	-,139	**-,685**	-,221	**-,395**	-,231	-,023
Comédie	,265	**-,592**	,161	**-,373**	,225	,242
Horreur	-,122	,424	*,360*	-,170	,369	,179
Guerre	-,032	-,037	**-,633**	**,425**	-,331	-,103
Documentaire	-,204	-,263	-,166	,232	**,639**	-,400
Western	,209	-,105	-,262	,353	,142	**,780**

FIGURE 2.6: Corrélation entre les variables et les composantes.

À partir de la matrice de corrélation, on voit que :

- La 1re composante principale représente essentiellement les variables Aventure, Enfant, Animation, Mystère, Crime et Drame.
- La 2e composante principale représente essentiellement les variables Action, Thriller, Romance et Comédie.
- La 3e composante principale représente essentiellement la variable Guerre et à un moindre degré les variables Enfant, Animation, Mystère et Horreur.
- La 4e composante principale représente essentiellement les variables Filmnoir, Guerre d'une part, et Romance, Comédie d'autre part.
- La 5e composante principale représente essentiellement la variable Documentaire.

1. La 6e composante principale représente essentiellement la variable Western.

B. Sélection des composantes principales

La deuxième étape consiste à déterminer le nombre de facteurs à retenir, pour cela on tient compte des facteurs qui permettent d'extraire une quantité d'informations (valeur propre) > 1.

La figure suivante représente la variance expliquée par chaque composante principale (valeur propre) :

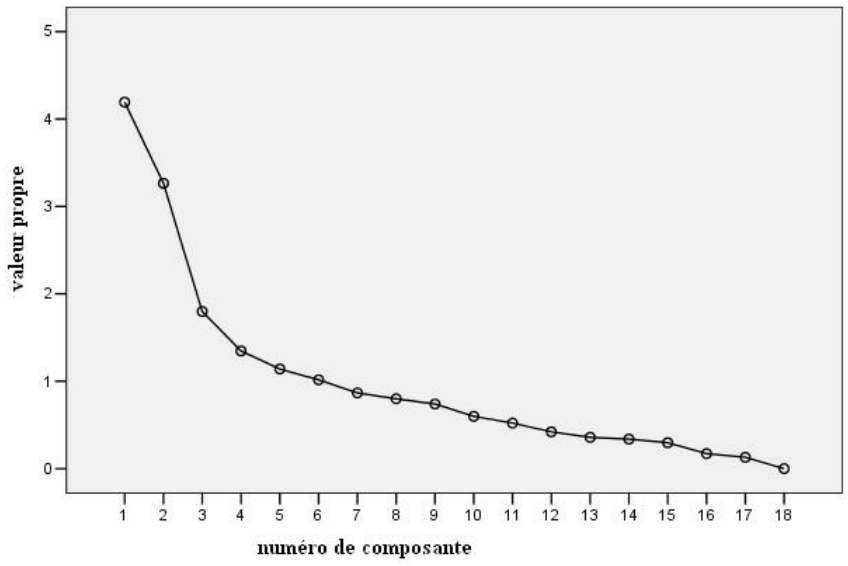

FIGURE 2.7: Variance expliquée par chaque composante principale.

Pour savoir combien de composantes principales utiliser, on recherche une rupture de pente sur le graphique. Cette rupture signifie que l'on passe d'un facteur représentant beaucoup d'information à un facteur en représentant moins. On s'arrête au facteur précédant cette rupture de pente. Dans cette expérimentation, on retient les 6 premières composantes dont la valeur propre est supérieure à 1.

C. Interprétation des axes

La dernière étape de l'expérimentation est l'interprétation des axes. On donne un sens à un axe à partir des coordonnées des variables. Le degré de corrélation entre une variable est un facteur doit être supérieur ou égale à α en valeur absolue (dans notre cas à 0.6), les tags situés dans la zone de faible corrélation sont alors ignorés. Les tags restants, situés dans les zones de forte corrélation, sont rassemblés dans une même communauté de tags.

Nous rapprochons les tags par les degrés d'appartenance des utilisateurs à ces tags en nous basant sur les graphiques générés lors de cette étape :

- Le 1er axe (figure 2.8) oppose les tags Animation, Enfant et Aventure aux tags Mystère, Crime et Drame. Ceci correspond à une interprétation naturelle : les personnes qui aiment le premier groupe de films n'aimant en général pas le second. Deux communautés sont ainsi créées.
- Le 2e axe oppose les films de Romance et de Comédie aux films Thriller et Action, en créant ainsi deux nouvelles communautés.
- Le 3e axe (figure 2.9) oppose les films de Guerre aux films étiquetés Enfant, d'Animation ou de Mystère.
- Le 4e axe oppose les films Filmnoir et les films de Guerre aux films de Romance et de Comédie.
- Le 5e axe oppose les films Documentaire aux films de Guerre.
- Le 6e axe oppose les films Western aux films Documentaire.

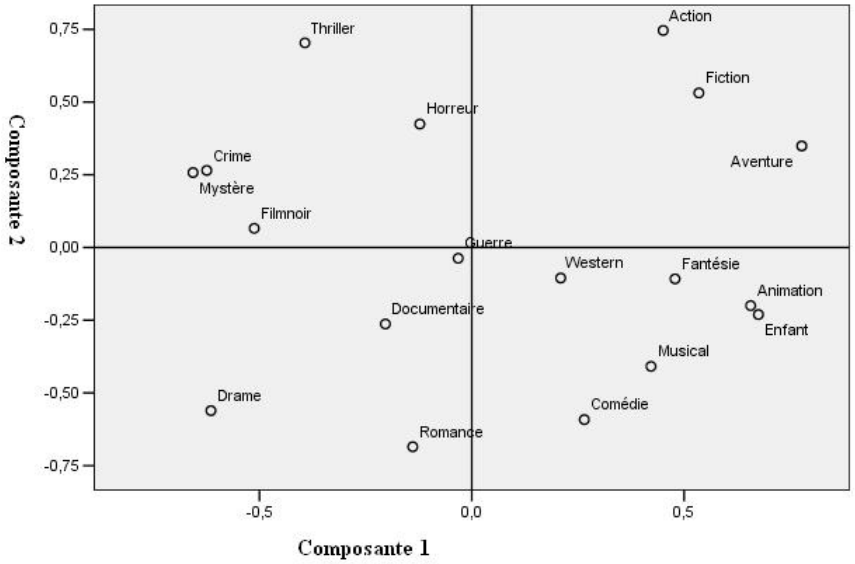

FIGURE 2.8: Composantes 1 et 2.

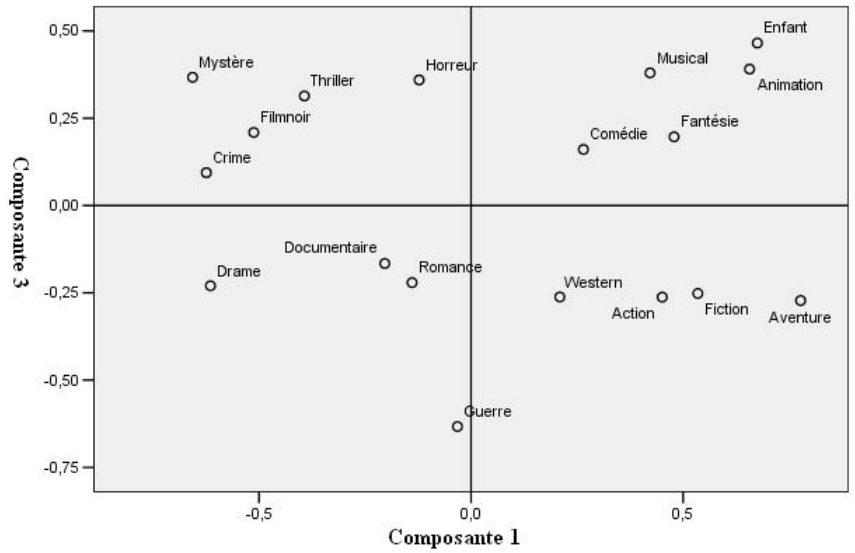

FIGURE 2.9: Composantes 1 et 3.

Cette interprétation nous donne 7 groupes de tags, comme indiqué au tableau 1. Les groupes qui sont disjoints sont 1 et 2, 3 et 4, 4 et 6 et enfin 6 et 7. Les utilisateurs sont

regroupés en fonction de ces communautés de tags. Les tags qui ne sont pas pris en compte par les axes sont expliqués par leur faible occurrence : par exemple le tag Fantasy n'est utilisé que 22 fois sur toute la collection des 1682 films.

communauté	tags associés
1	Aventure, Enfant, Animation
2	Mystère, Crime, Drame
3	Action, Thriller
4	Romance, Comédie
5	Western
6	Filmnoir, Guerre
7	Documentaire

FIGURE 2.10: Communautés de tags.

D. Définir la communauté de chaque utilisateur

Une fois l'ensemble des communautés de tags déduits, pour chaque utilisateur ui donné, on calcule son degré d'appartenance x'ij à chaque communauté de tags Gj. Sa communauté c(ui) est alors sa communauté de tag majoritaire, c'est à dire l'indice j tel que x'ij soit maximal.

Ces communautés pourront être utilisées ensuite dans différents but, comme dans notre cas, ces communautés nous seront utiles pour effectuer différentes recommandations de ressources à un utilisateur appartenant à une communauté donnée. Ces ressources seront des ressources que d'autres utilisateurs, appartenant à la même communauté que la sienne, ont appréciées et qu'il n'a pas encore consultées.

La méthode présentée ci-dessous est tirée de [16], elle se base essentiellement sur une analyse statistique des annotations de ressources manipulées par les utilisateurs. Cette méthode permet de représenter les données originelles (utilisateurs et annotations manipulées) dans un espace de dimension. La représentation des données dans cet espace de faible dimension facilite considérablement l'analyse et permet ainsi de regrouper ou d'opposer des communautés.

2.4.2 Découverte des intérêts par l'analyse des commentaires

Les traces laissées par les internautes sur le web sous forme textuelles sont souvent représentatives de leurs appréciations, opinions et préférences. Ainsi, l'analyse de telles traces sont nécéssaires pour établir leurs centres d'intérêts.

[20] analyse les commentaires textuels laissés par les internautes à propos de films à des fins de recommandation, en classifiant les opinions de chaque utilisateur pour ensuite déduire une note quantitative. Cette méthode rentre dans le cadre de l'analyse des sentiments (ou encore l'analyse des opinions) de notre approche, abordée dans la section 1.4 de ce chapitre. Une autre méthode bien connue qui se fait, est l'analyse de contenus textuels pour une extraction d'informations, comme il a été fait dans [**?**]. [**?**] utilise une ontologie de domaine pour déterminer les informations pertinentes et guider le processus d'extraction par l'identification des concepts présents dans le texte. D'autres études encore, font une analyse textuelle et sémantique des textes, mais celles-ci rentrent plus dans le domaine du Datamining et sortent du contexte de notre étude.

2.4.3 Réseaux sociaux et systèmes de recommandation

Comme précisé précédemment, les réseaux sociaux ont suscité l'attention des groupes de recherches scientifiques, qui considèrent aujourd'hui ces réseaux comme source d'information concernant les activités, intérêts et préférences des internautes.

2.4.3.1 Réseaux sociaux et filtrage collaboratif

Dans les réseaux sociaux, les utilisateurs sont souvent menés à Liker, partager ou encore commenter des contenus.

Les Likes des utilisateurs sont généralement exploités par les systèmes de recommandation basée sur le filtrage collaboratif pour déduire les utilisateurs voisins (qui ont des intérêts et goûts similaires) et ainsi recommander des ressources qui pourraient éventuellement intéresser l'utilisateur actif en se basant sur cette liste de voisinage.

Ainsi, les différentes études sur la classification des opinions classent les opinions en deux catégories (Like, Dislike), cette classification sera utilisée pour construire la matrice utilisateur-ressource-note (user-item-rating). Une fois cette matrice construite, le filtrage collaboratif entre en jeu pour le calcul de similarités. Cette méthode est expliquée plus en détails dans la section 1.1 (Système de recommandation basée sur le filtrage collaboratif).

2.4.3.2 Réseaux sociaux et filtrage basé sur le contenu

Dans les systèmes de recommandation basée sur le contenu, des profils d'apprenants sont généralement utilisés afin de pouvoir effectuer des recommandations. Ces profils contiennent des informations concernant les intérêts et les préférences des différents utilisateurs.

Ces informations peuvent être collectées de manière explicite, à travers différentes questions posées à l'utilisateur, comme ses centres d'intérêts, les livres déjà lus, le genre de musiques et de films qu'il aime, etc. Ou de manière implicite, et ce, en observant et analysant les différents comportements que peut avoir l'utilisateur, tels que son historique de navigation, la fréquence de consultation des ressources, le temps passé dessus, les ressources partagées, mises en favoris, ou alors qu'il recommande à ses amis. En considérant que toute ressource qu'un utilisateur recommande, ajoute en favoris ou partage dans son réseau social, lui est intéressante.

Une fois qu'un ensemble de ressources qui intéressent l'utilisateur est défini, le système procède à l'extraction des vecteurs de termes des différentes ressources, et les utilisera pour l'actualisation des préférences et centres d'intérêts des utilisateurs. Enfin, le profil de l'apprenant ainsi établi sera utilisé par le système de recommandation afin de calculer des similarités avec les ressources disponibles dans un but de recommandation. Cette méthode est également expliquée en détail dans la section 1.2 (Système de recommandation basée sur le contenu).

2.5 Analyse des sentiments

Avec la rapide augmentation de la quantité de textes subjectifs disponibles sur le web sous forme de blogs, de commentaires dans les forums, etc. les entreprises se tournent de plus en plus vers internet afin d'obtenir des informations plus subtiles et subjectives - opinions - sur leurs produits.

Dans la littérature, il existe généralement deux types d'approches pour la détection automatique d'opinion et de la polarité. Certaines sont basées sur le lexique, d'autres sur l'apprentissage.

2.5.1 Détection de l'opinion

Deux approches peuvent être utilisées pour la détection d'opinion : l'approche basée sur l'apprentissage machine et l'approche basée sur un lexique.

2.5.1.1 Approches basées sur l'apprentissage machine (Machine Learning)

Ces approches utilisent des classifieurs. Des données, qui représentent des phrases subjectives (ou des documents avec opinion), sont fournies au classifieur pour l'apprentissage.

Le classifieur génère un modèle, qui sera utilisé dans la partie test. Des « features » sont utilisées pour l'apprentissage tels que les bigrammes, les n-grammes, POS (étiquettes morphosyntaxiques) etc. Il existe plusieurs types de classifieurs, parmi lesquels [4] :

- **L'approche SVM** : repose sur la notion d'hyperplan séparateur et de marge maximale. Un hyperplan séparateur entre deux ensembles de points (ensemble de documents de polarité positive et l'ensemble de document de polarité négative) est la frontière entre ces deux ensembles. La marge représente la distance entre un de ces ensembles et cet hyperplan.
- **La régression logistique :** est une méthode statique permettant de produire un modèle pour décrire des relations entre une variable catégorielle et un ensemble de variables de prédiction.

2.5.1.2 Approches basées sur le lexique

Ces approches utilisent un lexique de mots qui contiennent un sentiment. Ce lexique est soit externe c'est-à-dire construit indépendamment de tout corpus, il peut être général (SentiWordNet, SUBJ lexique, General Inquiry, Wilson lexicon, WordNet-Affect) ou construit manuellement, soit généré automatiquement à partir du corpus (les mots qui contiennent une opinion sont extraits directement du corpus). À chaque mot du lexique est associé un ensemble de scores d'opinions. Ce score est traité différemment par les différentes approches pour le calcul du score d'opinion d'un document. La méthode la plus simple est de donner à un document un score d'opinion égal au nombre total de mots qui contiennent une opinion présents dans le document [4].

2.5.2 Classification de la polarité des opinions

La classification des sentiments est un raffinement de la détection d'opinions dans la mesure où elle permet de classifier les documents ayant une opinion sur un sujet en classes. Il existe deux types de classification : binaire ou multi-classes. La classification binaire définit deux classes : positive et négative. Par contre la classification multi-classes définit cinq classes : fortement positive, positive, neutre, négative, fortement négative. La plupart des travaux se sont focalisés sur la classification binaire mais la classification multi-classes – méthode que nous avons adoptée – peut être utile dans les applications où on veut faire une meilleure classification [4].

Plusieurs travaux importants ont abouti à la création de ressources lexicales pour améliorer les systèmes automatiques, en particulier pour traiter l'anglais. Nous pouvant citer comme exemple :

– **Lexique des sentiments**

Développé manuellement pour le français, le lexique des sentiments comporte un millier de mots - exclusivement des mots simples – exprimant des sentiments, des émotions et des états psychologiques. Ces mots sont répartis en 38 classes sémantiquement homogènes : 22 classes négatives (Peur, Tristesse, Irritation, etc), 14 classes positives (Amour, Intérêt, Passion, etc), 2 classes sans polarité (Étonnement et Indifférence). Chaque classe est nommée par le sentiment ou l'état psychologique décrit, comme la classe Peur qui contient les mots relatifs à un sentiment de peur (peur, crainte, frayeur, effrayer, effrayant, etc). Les classes sémantiques sont liées entre elles par des relations de sens, d'intensité et d'antonymie.

Bien qu'étant une ressource intéressante pour le français, ce lexique des sentiments est limité par sa taille et la nature de ses entrées. L'auteur relève l'importance du niveau pragmatique et énonciatif du texte pour mesurer la subjectivité d'un mot et précise que « la reconnaissance et l'interprétation du simple vocabulaire du domaine ne sont pas suffisantes le plus souvent, il faut prendre en compte d'autres éléments comme les expressions idiomatiques ou figées telle que être la prunelle de ses yeux » [25].

– **WordNet-Affect**

Wordnet-Affect est une ressource linguistique pour la représentation lexicale de connaissances sur les affects pour l'anglais. Un sous-ensemble de synsets de WordNet appropriés est choisi pour représenter des concepts affectifs. Des informations additionnelles sont

ajoutées aux synsets affectifs, en leur associant une ou plusieurs étiquettes qui précisent une signification affective [25].

WordNet-Affect est constitué de 1903 concepts (539 noms, 517 adjectifs, 238 verbes et 15 adverbes) directement ou indirectement liés à un état mental ou émotionnel. Par exemple, les concepts affectifs représentant un état émotif sont représentés par des synsets marqués par l'étiquette Emotion (Anger, Fear, etc.). WordNet-Affect a été développé semi-manuellement en deux étapes : l'identification manuelle d'un premier « noyau » de synsets affectifs, l'utilisation des relations de synonymie/antonymie présentes dans WordNet afin de propager les informations de ce noyau à son voisinage. Les entrées lexicales de Wordnet-Affect sont très majoritairement des mots simples.

– **SentiWordNet**

De façon complémentaire, SentiWordNet est une ressource dédiée aux systèmes de classification de textes d'opinions. SentiWordNet assigne à chaque synset de WordNet trois valeurs : Positivité, Négativité, Objectivité (en respectant l'égalité : Positivité + Négativité + Objectivité = 1). Chacune des valeurs a été déterminée par apprentissage supervisée sur des corpus dont les textes sont classés positif, négatif ou objectif en exploitant également les relations de synonymie/antonymie de WordNet par la suite. Des exemples de scores associés aux entrées de SentiWordNet sont présentés dans la figure suivante :

Category	WNT Number	pos	neg	Synonyms
A	01123148	0.875	0	good#1
A	00106020	0	0	good#2 full#6
A	01125429	0	0.625	bad#1
A	01510444	0.25	0.25	big#3 bad#2
N	03076708	0	0	trade_good#1 good#4 commodity#1
N	05144079	0	0.875	badness#1 bad#1

FIGURE 2.11: Exemple de scores associés aux entrées de SentiWordNet.

2.6 Approche adoptée

A l'issu de l'analyse des différentes approches existantes effectuée dans la partie précédente, nous avons fait des choix bien pensés sur les approches à adopter pour la mise en application de notre système.

Comme nous l'avons déjà précisé, notre système est réparti en quatre sous-systèmes, chacun est chargé d'effectuer une tâche bien précise. Dans ce qui suit, nous aborderons l'approche retenue pour chaque sous-système :

2.6.1 Système de recommandation basée sur le filtrage collaboratif (CF)

L'approche choisie pour construire un système de recommandation basé sur le filtrage collaboratif est la même que celle utilisée dans [22]. On aura donc à établir, à chaque fois qu'un apprenant utilisera notre système, la liste de ses voisins ainsi que les notes approximatives qu'il pourrait attribuer aux N ressources qu'il n'a pas encore consultées (N est le nombre de ressources que nous choisissons d'afficher), et ce, en utilisant le coefficient de corrélation de Pearson. Notre choix s'est porté sur cette mesure, car communément aux tests vu dans la partie analyse, les résultats obtenus lors du calcul de la similarité entre deux utilisateurs avec Pearson sont beaucoup plus concluant.

2.6.2 Système de recommandation basée sur le contenu (CB)

Comme expliqué dans la partie analyse de ce chapitre, pour pouvoir calculer une similarité entre un utilisateur et une ressource, il faut utiliser les profils de chacun d'eux.

Dans notre approche, pour établir le profil utilisateur, on utilisera seulement une méthode manuelle, dans laquelle la partie centre d'intérêts de la plateforme de formation MOODLE est utilisée pour déduire le vecteur de termes qui représente le profil de l'apprenant. Les informations de cette partie, centre d'intérêts, sont fournies par l'apprenant lorsqu'il s'inscrit sur la plateforme, puis sont mises à jour, automatiquement, au fur et à mesure qu'il consulte des ressources et leur attribues des notes estimées "bonnes". Pour chaque ressource que l'apprenant apprécie, on extrait les mots de son vecteur de termes qu'on ajoute à ses centres d'intérêts.

Pour intégrer l'aspect "traces d'activités" à l'élaboration du profil de l'apprenant, nous utiliseront les résultats obtenus lors d'une analyse des réseaux sociaux qui est expliquée dans le point .. de cette partie.

Pour ce qui est du profil de ressources, on se basera seulement sur la description de chaque ressource pour extraire les mots qui vont constituer son vecteur de termes.

Pour pouvoir recommander des ressources, on calculera la similarité entre le vecteur de termes du profil de l'apprenant, avec le vecteur de termes du profil de chaque ressource que l'apprenant n'a pas consultée.

Comme nous l'avons abordé dans la partie analyse, la similarité peut être calculée de différentes manières, avec le produit scalaire, la mesure du cosinus, ou alors le coefficient de Dice. Etant donné les résultats obtenus dans [11], qui démontrent que le coefficient de Dice donne de meilleurs résultats, alors nous choisissons d'adopter ce coefficient.

Les ressources ayant une forte similarité avec les préférences de l'apprenant sont données par ordre décroissant, où les ressources susceptibles de l'intéresser le plus sont mises en avant.

2.6.3 Analyse des réseaux sociaux

Pour pouvoir intégrer l'analyse de réseaux sociaux dans notre application, nous avons considéré le forum de la plateforme de formation MOODLE comme simulation. Ainsi, nous procédons à l'analyse des commentaires postés par les apprenants dans ce forum pour déduire leur préférences et centres d'intérêts, considérées ici comme leur "traces d'activités".

2.6.3.1 Méthode adoptée pour l'analyse des commentaires dans les réseaux sociaux

L'analyse des commentaires, postés par les apprenants dans le forum de la plateforme, sera faite en appliquant des méthodes d'analyse textuelles pour déduire les concepts qui intéressent ces apprenants.

Les méthodes d'analyse textuelles que nous allons implémenter se baseront essentiellement sur une analyse des sentiments d'un texte posté par l'apprenant, et une extraction de concepts à partir du même texte.

1. L'analyse des sentiments est abordée plus en détails dans la section 1.4 de ce chapitre. Mais contrairement à ce qui est expliqué dans cette section, l'analyse des sentiments des postes et commentaires des apprenants dans le forum, ne se fera pas dans le but de déduire une note quantitative, mais pour classer notre poste comme étant soit positif ou négatif.

2. Dans le cas où le poste est positif, une extraction de concepts se fait pour déduire les informations qui intéressent l'apprenant. Ici, on part du fait qu'un apprenant qui parle en bien (positivement) d'un sujet particulier, est intéressé par ce dernier.

Pour déterminer les informations pertinentes dans les postes ou commentaires laissés par l'apprenant dans le forum, nous procédons à l'extraction de concepts à l'aide d'une ontologie E-learning. Pour ce faire, on procède tout d'abord à la collecte des postes et commentaires de la base de données MOODLE, nous devrons ensuite lemmatiser tous les termes pour ne garder que le lemme du mot. Les lemmes ainsi obtenus seront utilisés pour interroger l'ontologie E-learning par des requêtes SPARQL. Si un concept est retourné, le lemme correspondant est considéré comme information pertinente, et est sauvegardé dans les centres d'intérêts de l'apprenant comme mot clé de ses préférences.

Ces préférences et centres d'intérêts seront utilisés par le système de recommandation basé sur le contenu dans le profil de l'apprenant comme étant ses traces d'activités, afin de recommander des ressources susceptibles de correspondre à ses attentes.

2.6.4 Analyse des sentiments

Nous avons décidé d'appliquer pour la partie analyse des sentiments une approche basée sur le lexique et avons choisi d'utiliser pour cela l'opinion lexicon SentiWordNet.

On peut retrouver dans la littérature trois principales méthodes d'exploitation de SentiWordNet [12] :

- **Méthode de comptage** : les scores positifs et négatifs d'un mot sont obtenus à partir de SentiWordNet en fonction de parties spécifiques d'un texte (tags). Une méthode de comptage de termes est ensuite appliquée pour classifier les commentaires comme positifs ou négatifs. Dans cette méthode, le lexicon est exploité pour compter le nombre de mots positifs et négatifs trouvés dans un commentaire. La polarité du sentiment est déterminée en se basant sur la classe qui a reçu le plus haut score. Cette technique ne prend cependant pas en considération les degrés de positivité, négativité et d'objectivité de chaque mot. Pour exemple, « good » et « acceptable » sont considérés comme des mots positifs sans prendre en compte leur degré de positivité.
- **Somme des scores des termes** : méthode dans laquelle la somme des scores positifs et négatifs des mots est calculée afin d'obtenir le score positif et négatif de tout

le commentaire. Le sentiment contenu dans le commentaire est ensuite déterminé en fonction du score le plus élevé.

– **Moyenne de la phrase et moyenne du commentaire** : Le score de positivité et de négativité de chaque phrase est calculé à partir de la moyenne des scores positifs et négatifs de tous les mots de celle-ci. Le calcul du score positif et négatif de tout le commentaire est ensuite fait à partir de la moyenne des scores positifs/négatifs de toutes les phrases. On peut ainsi déterminer la polarité du sentiment contenu dans la critique.

Voici un tableau comparatif des trois méthodes de calcul [12] :

Méthode	Degré de précision
Méthode de comptage	56.77%
Somme des scores des termes	67.00%
Moyenne de la phrase et moyenne du commentaire	70.00%

TABLE 2.2: Comparaison des méthodes de calcul d'un score.

Nous avons choisi d'adopter la méthode du « calcul de la moyenne de la phrase puis de la moyenne du commentaire » - troisième méthode - car, comme démontré dans le tableau ci-dessus, celle-ci présente le plus haut degré de précision.

Les résultats de cette méthode sont ensuite traités et manipulés afin d'obtenir un note (quantitative) équivalente au commentaire analysé.

Le schéma présenté ci-dessous illustre le fonctionnement général des différents sous-systèmes cités plus haut :

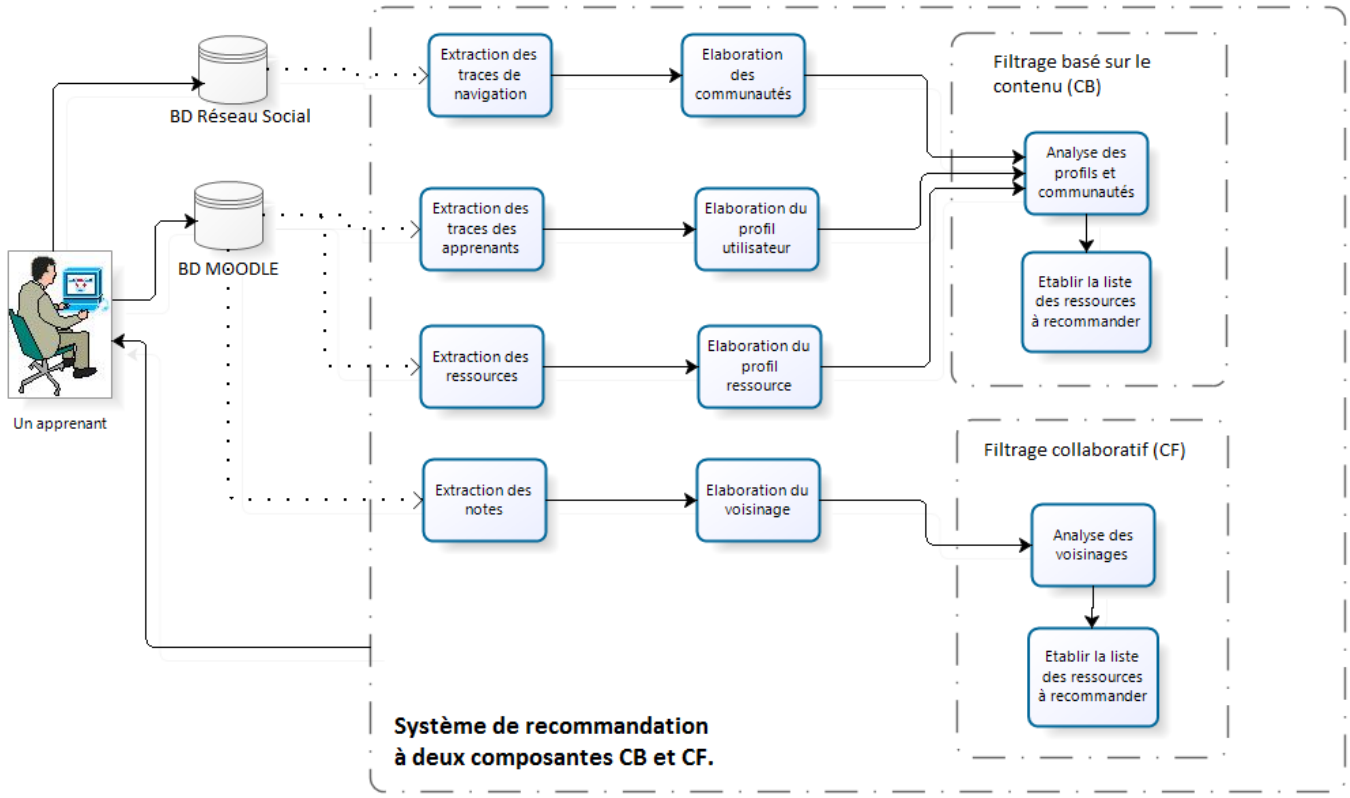

FIGURE 2.12: Schéma du fonctionnement général de l'approche adoptée.

2.7 Diagrammes représentatifs de l'approche

2.7.1 Diagramme de contexte de l'application

L'analyse effectuée ci-dessus, nous a permis de dégager le diagramme de contexte suivant :

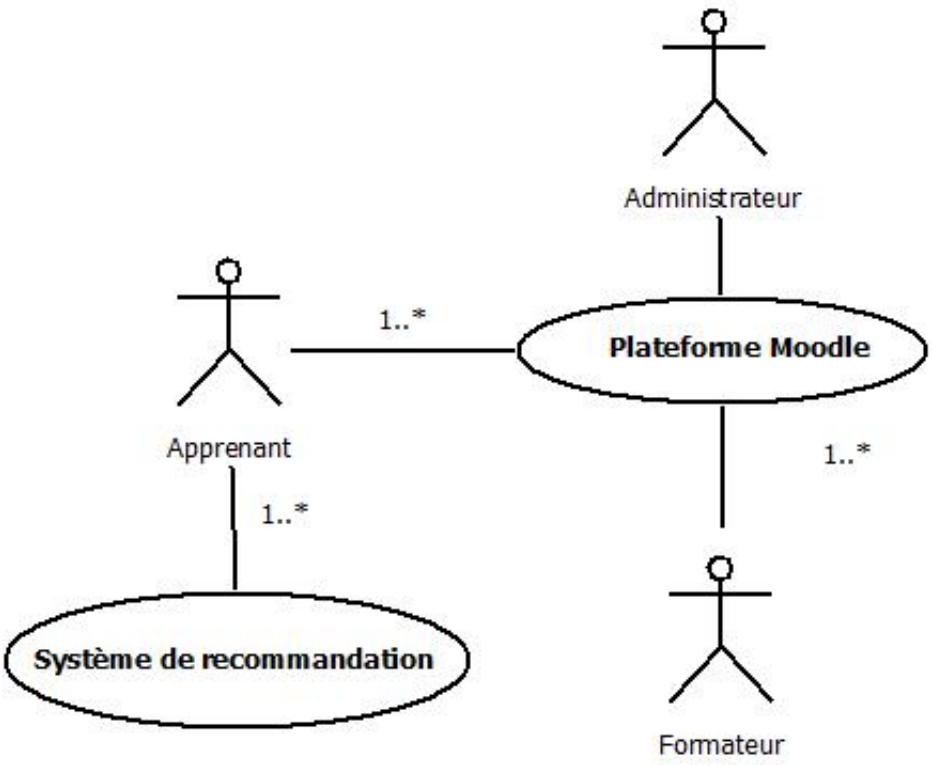

FIGURE 2.13: Diagramme de contexte.

2.7.2 Diagramme de cas d'utilisation d'un apprenant

Le diagramme de cas d'utilisation est un diagramme UML, utilisé pour donner une vision globale du comportement fonctionnel de notre application. Ce diagramme représente une unité discrète d'interaction en un utilisateur (homme ou machine) et un système.

Dans un diagramme de cas d'utilisation, les utilisateurs qui sont appelés acteurs (actors) intéragissent avec les cas d'utilisation (use case).

Les cas d'utilisation d'un apprenant dans notre système de recommandation sont :

- Identification.
- Accéder à la page de recommandation, avec possibilité de changer d'onglet, entre la recommandation basée sur le contenu ou la recommandation basée sur le filtrage collaboratif.
- Noter une ressource.
- Commenter une ressource.

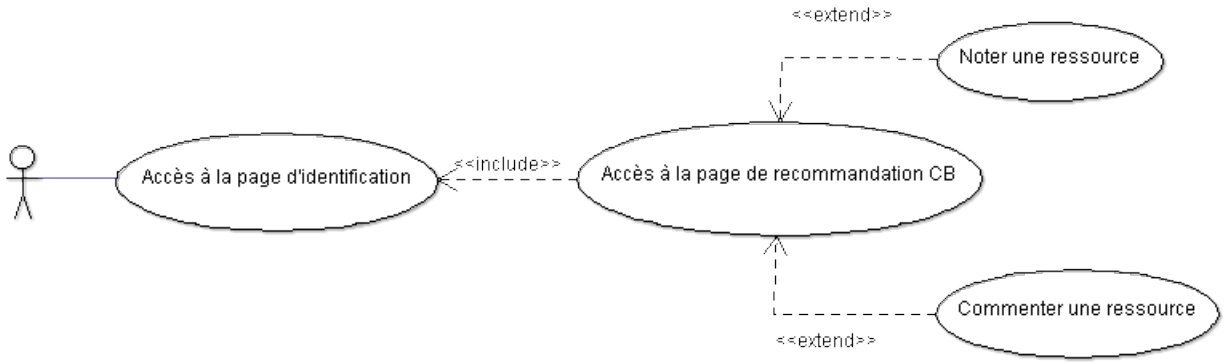

FIGURE 2.14: Diagramme de cas d'utilisation d'un apprenant.

2.7.3 Diagrammes de séquence

Les diagrammes de séquence offrent une vue dynamique du système par une spécification du comportement du système. Les diagrammes de séquence décrivent les interactions entre les objets. Ces interactions sont modélisées comme des échanges de messages, on se focalise essentiellement sur l'expression des interactions.

2.7.3.1 Diagramme de séquence du cas "noter une ressource dans le système CF"

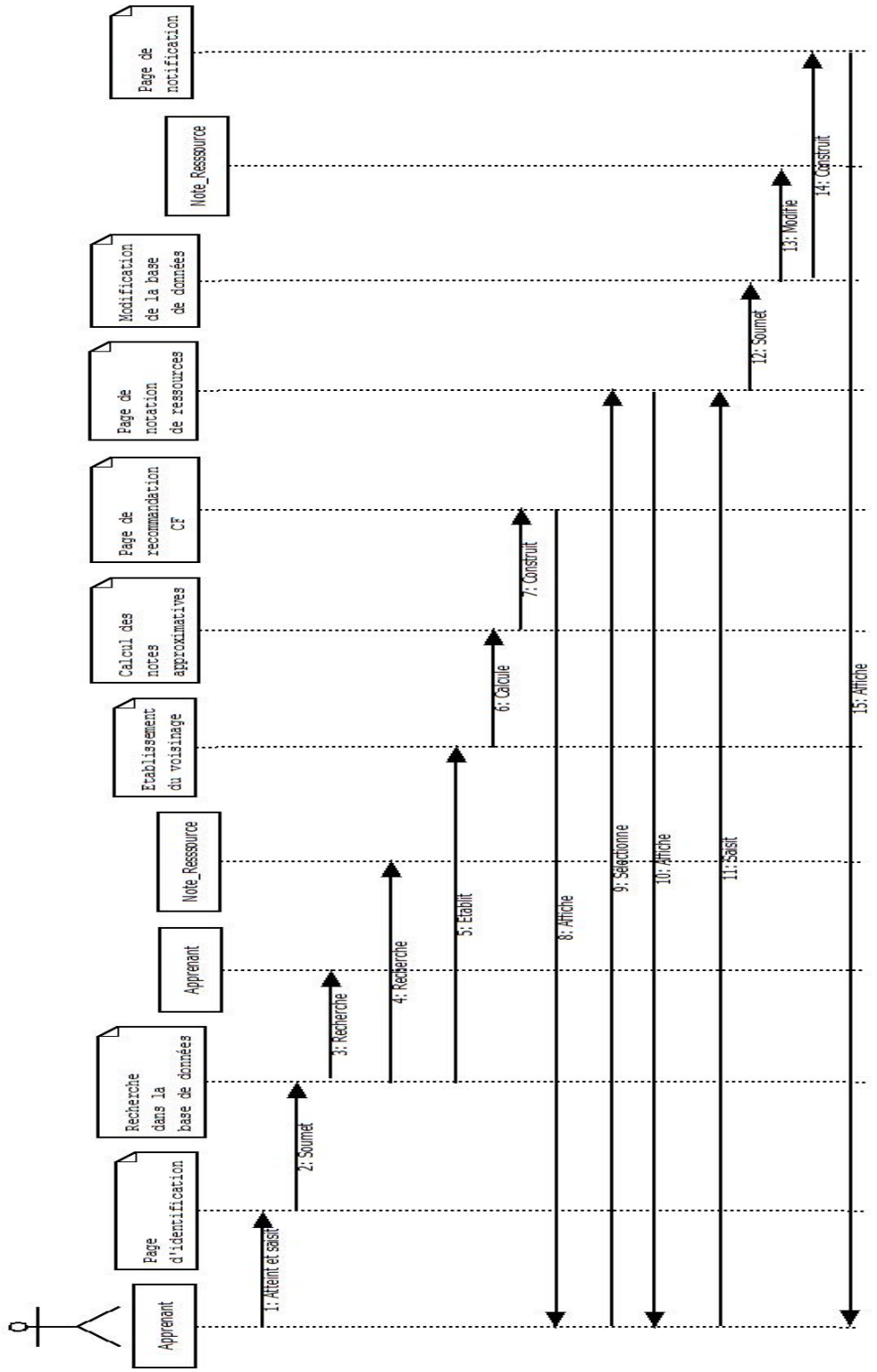

FIGURE 2.15: Diagramme de séquence du cas "commenter une ressource dans le système CF".

1 : l'apprenant atteint la page d'identification de l'application et saisit ses informations

2 : l'apprenant soumet ses informations au système pour avoir des recommandations de ressources

3 : le système recherche dans la base de données si l'apprenant existe

4 : dans le cas où l'apprenant existe, le système recherche les notes qu'il a attribuées aux ressources

5 : le système établit les voisins de l'apprenant

6 : le système calcule à partir du voisinage de l'apprenant, les notes approximatives qu'il pourrait attribuer aux ressources qu'il n'a pas consultées

7 : le système construit la liste des ressources que l'apprenant pourrait apprécier

8 : le système affiche cette liste à l'apprenant

9 : l'apprenant sélectionne une ressource à noter, après l'avoir consultée

10 : le système affiche la page de notation de la ressource

11 : l'apprenant saisit la note qu'il a attribuée à la ressource

12 : l'apprenant soumet cette note au système

13 : le système modifie la base de données pour ajouter cette note à la table Note_ressource

14 : le système construit la page de notification

15 : la page est affichée à l'apprenant, en lui confirmant la note qu'il a attribuée.

2.7.3.2 Diagramme de séquence du cas "commenter une ressource dans le système CB"

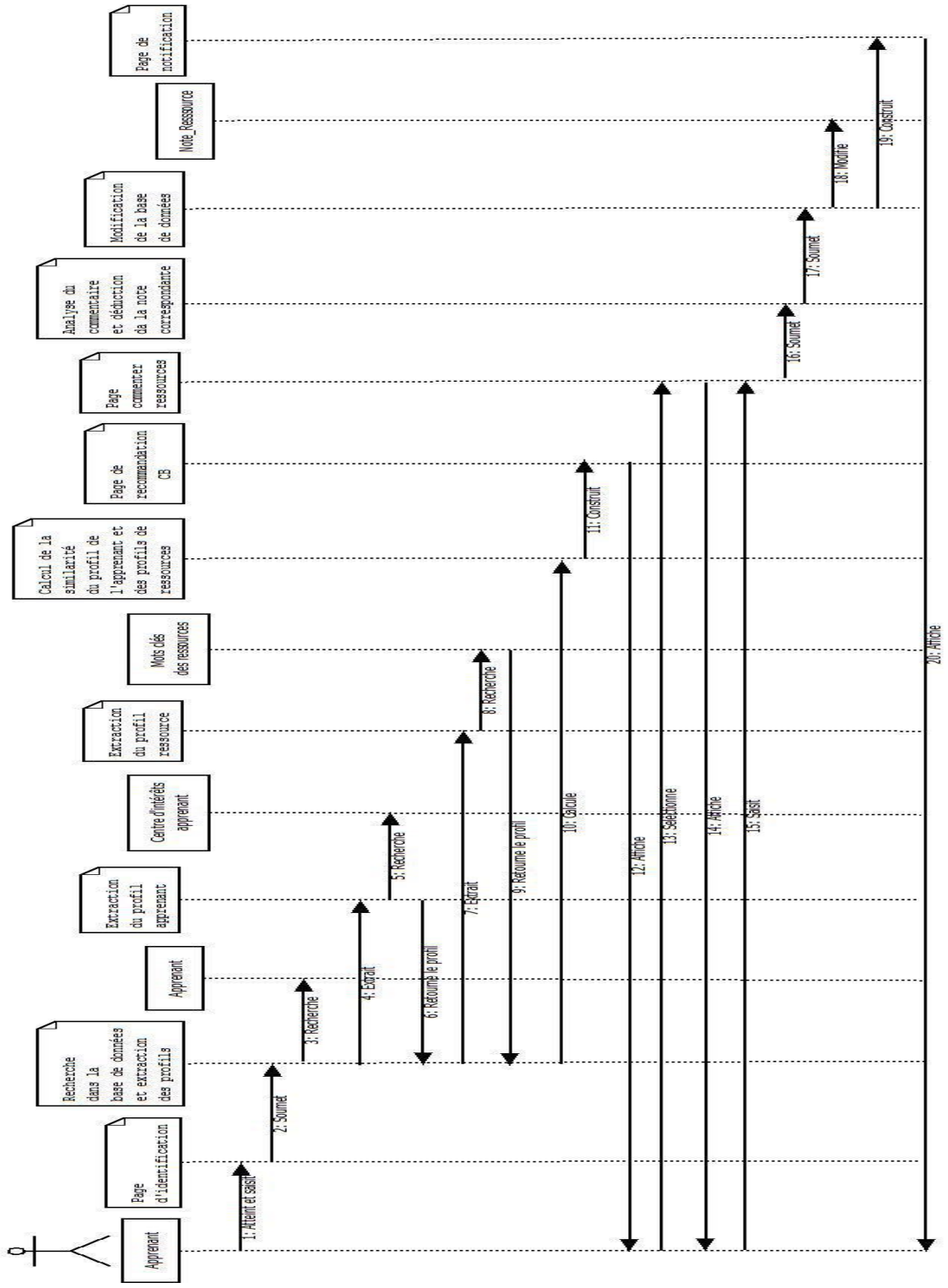

FIGURE 2.16: Diagramme de séquence du cas "commenter une ressource dans le système CB".

1 : l'apprenant atteint la page d'identification de l'application et saisit ses informations

2 : l'apprenant soumet ses informations au système pour avoir des recommandations de ressources

3 : le système recherche dans la base de données si l'apprenant existe

4 : dans le cas où l'apprenant existe, le système fait une extraction de son profil

5 : le système recherche dans la base de données les centres d'intérêts de l'apprenant pour construire son profil

6 : le système retourne le profil de l'apprenant 7 : le système fait ensuite une extraction des profils de ressources

8 : le système recherche dans la base de données des descriptions (mots clés) des ressources pour construire leur profil

9 : le système retourne les profils des ressources

10 : à partir des profils extraits, le système calcule la similarité entre le profil de l'apprenant et le profil de chaque ressource

11 : le système construit la liste des ressources similaires à son profil

12 : le système affiche cette liste à l'apprenant

13 : l'apprenant sélectionne une ressource à commenter, après l'avoir consultée

14 : le système affiche la page pour commenter la ressource

15 : l'apprenant saisit le commentaire pour donner son appréciation sur la ressource

16 : l'apprenant soumet ce commentaire au système

17 : le système analyse ce commentaire, par une analyse des sentiments, et déduit la note correspondante, puis la soumet à fin que la base de donnée soit modifiée

18 : le système modifie la base de données pour ajouter cette note à la table Note_ressource

19 : le système construit la page de notification

20 : la page est affichée à l'apprenant, en lui confirmant la note qu'il a attribuée.

2.7.3.3 Diagramme de classes

Les diagrammes de classes expriment de manière générale la structure statique d'un système en termes de classes et de relations entre ces classes. De même qu'une classe décrit un ensemble d'objets, une association décrit un ensemble de liens ; les objets instances de classes et les liens sont instances de relations.

L'organisation des différentes classes du système est donnée par la figure suivante, où chaque paquetage représente un module particulier de notre système de recommandation.

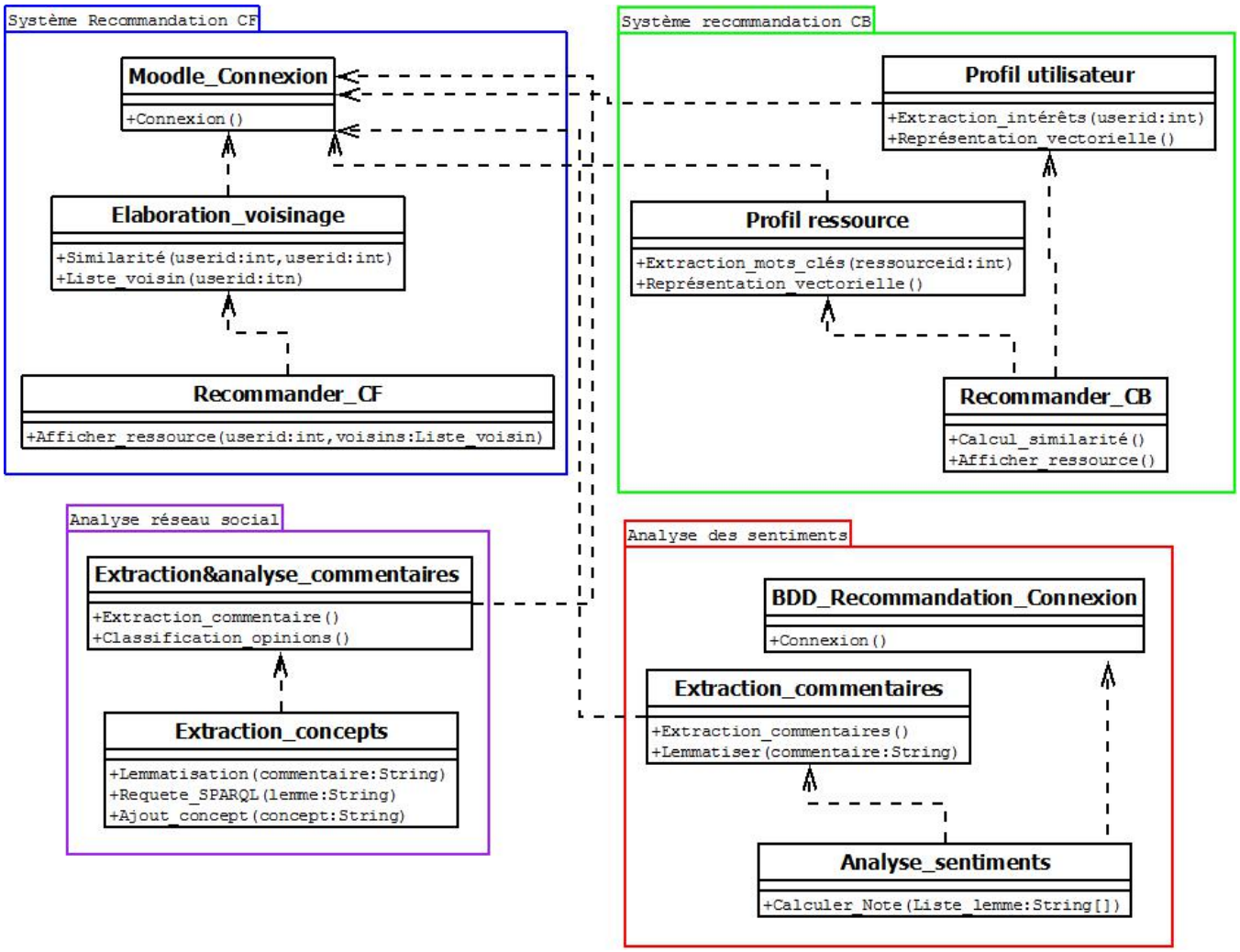

FIGURE 2.17: Diagramme de classes UML du système de recommandation.

2.8 Conclusion

A l'issu de ce chapitre, nous avons abordé les différentes étapes de notre application, ainsi que les approches sélectionnées, pour recommander des ressources ayant de fortes chances de satisfaire les besoins d'un apprenant, lui permettant un gain de temps considérable en lui évitant des recherches dans la vaste collection de ressources dont dispose le web aujourd'hui, et qui n'aboutissent parfois jamais.

Les différentes approches citées dans ce chapitre ont été mises en œuvre pour l'élaboration de notre application, dont la réalisation est détaillée dans le chapitre suivant.

Chapitre 3

Réalisation

3.1 Introduction

Une des étapes de la vie d'un projet, aussi importante que la conception est l'implémentation. Cette étape constitue la phase d'achèvement et d'aboutissement du projet. Pour accomplir cette tâche avec succès, il faut savoir utiliser les outils adéquats et nécessaires. Ce choix d'outils peut influencer sur la qualité du produit obtenu et donc nécessite une attention particulière et doit se baser sur les besoins du projet et le résultat escompté.

Ce chapitre présente alors l'environnement technique du travail, le choix pris en matière d'environnement logiciel, ainsi que quelques interfaces de notre application.

3.2 Environnement de développement

Un diagramme de déploiement spécifie un ensemble de constructions qui peut être utilisé pour définir l'architecture d'exécution des systèmes. Le diagramme de déploiement permet de représenter l'environnement de développement de notre application.

Notre solution se base sur une architecture client/serveur à trois tiers (trois niveaux) qui sont :

1. Le premier niveau de cette architecture qui est le niveau présentation est constitué d'un ordinateur disposant d'un navigateur Web, ainsi que de notre application.
2. Le deuxième est le niveau applicatif (logique applicative) qui est pris en charge par le serveur Apache, et qui se compose de scripts écrits en PHP.

3. Les langages de développementLe troisième niveau, qui fournit au niveau intermédiaire les données dont il a besoin, est pris en charge dans notre cas par le SGBD MySQL.

L'implémentation de notre système se base sur l'architecture illustrée dans la figure ci-dessous :

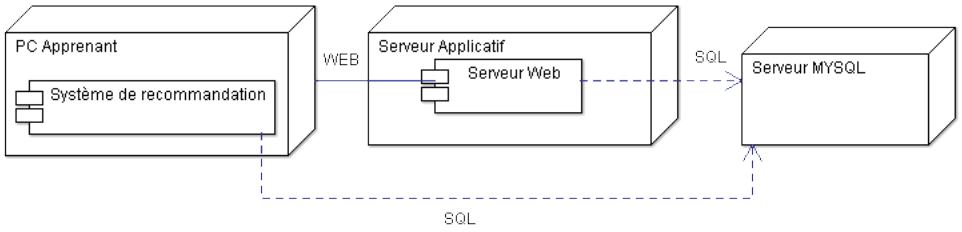

FIGURE 3.1: Diagramme de déploiement.

1. **Le serveur de base de données MySQL :** MySQL est un système de gestion de base de données SGBD. Son rôle est de stocker et de gérer une grande quantité de données en les organisant sous forme de tables, et de permettre la manipulation de ces données à travers le langage de requête SQL. Il est implémenté sur un mode client/serveur, avec du côté serveur : le langage MySQL, et du côté client : les différents programmes et librairies.
2. **Le serveur web Apache :** tout développement de site web requiert un serveur Web qui s'occupe du traitement des requêtes des clients, le transfert des pages HTML au browser et l'exécution des programmes sur la machine serveur.
3. **Interface phpMyAdmin :** phpMyAdmin est un outil d'administration de serveurs MySQL. Il est sous licence open source, et possède principalement les fonctionnalités suivantes :
   - création, modification et suppression de bases de données et de tables.
   - gestion des utilisateurs et de leurs privilèges.
   - exécution de requêtes SQL.

3.3 Langages de développement

1. **Langage java :** java est un langage de programmation orienté objet, assurant entre autre la portabilité des applications développées avec ce langage.
2. **Le langage de requêtes SQL :** SQL est un langage de gestion de bases de données relationnelles. Il permet :
   - l'interrogation de bases de données.
   - la manipulation des données (mises à jour, suppressions, etc.).
   - la définition des données (création de bases de données et de tables relationnelles).
3. **Le langage SPARQL :** SPARQL est un langage de requête et un protocole qui permet de rechercher, d'ajouter, de modifier ou de supprimer des données RDF disponibles à travers Internet. SPARQL est l'équivalent de SQL car comme en SQL, on accède aux données d'une base de données via ce langage de requête alors qu'avec SPARQL, on accède aux données du Web des données.

3.4 Outils de développement

1. **Eclipse IDE :** est un environnement de développement intégré libre, extensible, universel et polyvalent, permettant potentiellement de créer des projets de développement mettant en œuvre n'importe quel langage de programmation. Eclipse IDE est principalement écrit en Java (à l'aide de la bibliothèque graphique SWT, d'IBM), et ce langage, grâce à des bibliothèques spécifiques, est également utilisé pour écrire des extensions.
2. **Wamp Server :** Wamp Server est une application de développement Web, distribuée sous licence open source. Elle permet de faire fonctionner localement (sans se connecter à un serveur externe) des applications web. Wamp Server intègre les serveurs Apache et MySQL, le langage PHP, ainsi que l'outil phpMyAdmin.

3.5 Description de l'application

3.5.1 Description de la base de données

Le nombre de tables de la base de données varie selon les versions. Dans notre cas, il existe au total 300 tables.

Ci-dessous, nous présentons les tables ayant servi à l'élaboration de notre système de recommandation, nous ne garderons des tables que les champs qui ont été utilisés par notre système.

3.5.1.1 Table utilisée pour l'identification

Afin que l'utilisateur ait accès aux recommandations de ressources faites par notre système, une identification est nécessaire. Il doit fournir son pseudonyme, qui sera utilisé pour retrouver son identifiant dans la table *mdl_user*, et qui sera utilisé tout au long du processus de recommandation. Cette table se présente comme suit :

Champ	Type	Signification
id	bigint(10)	Identifiant de l'apprenant
username	varchar(100)	Pseudonyme
description	longtext	Description des centres d'intérêts de l'apprenant

3.5.1.2 Tables utilisées par le système de recommandation basée sur le filtrage collaboratif

Le filtrage collaboratif utilise les notes attribuées par des apprenants à des ressources pour effectuer des recommandations, ces notes sont stockées dans une table que nous avons ajoutées à la base de données de Moodle, et nous l'avons appelée *note_ressource* et se présente comme suit :

Champ	Type	Signification
id_ressource	bigint(10)	Identifiant de la ressource
id_user	bigint(10)	Identifiant de l'apprenant
note	float	Note attribuée par l'apprenant à la ressource

Ce type de filtrage utilisera également la table *mdl_resource* pour afficher les noms des ressources à recommander. Celle-ci se présente comme suit :

Champ	Type	Signification
id	bigint(10)	Identifiant de la ressource
course	bigint(10)	Identifiant du cours
name	varchar(255)	Nom de la ressource
into	longtext	Description de la ressource

3.5.1.3 Tables utilisées par le système de recommandation basée sur le filtrage basé sur le contenu

Le filtrage basé sur le contenu utilise également les tables *mdl_resource* et *mdl_user* pour établir les profils utilisateurs et les profils ressources, également nécessaires pour le processus de recommandation.

3.5.2 Présentation des interfaces de l'application

Notre système de recommandation fournit à l'apprenant une interface lui permettant à la fois d'avoir des recommandations basées sur le filtrage collaboratif, et des recommandations basées sur le contenu, tout en ayant la possibilité de noter et commenter des ressources.

Nous avons conçu cette interface sous forme d'une seule fenêtre principale, précédée par une fenêtre d'identification de l'apprenant (figure 3.2). Cette fenêtre principale a été prévue pour accueillir l'ensemble des opérations possibles de notre application, qui sont réparties en deux onglets différents :

1. **L'onglet CF (Filtrage collaboratif) :** comporte deux parties, la première affiche les voisins de l'apprenant, et se compose en deux colonnes, une qui indique l'identifiant de l'apprenant voisin et l'autre qui indique son nom. La seconde partie elle, se compose en trois colonnes, qui indiquent l'identifiant de la ressource recommandée, son nom et la note calculée, le tout accompagné de deux boutons "noter" et "commenter", qui permettent respectivement de noter et commenter les ressources.
2. **L'onglet CB (Filtrage basé sur le contenu) :** comporte deux colonnes, qui indiquent le nom de la ressource recommandée ainsi que son identifiant, et également

les deux boutons "noter" et "commenter".

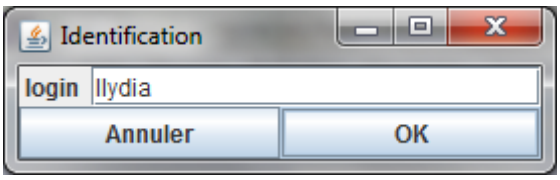

FIGURE 3.2: Interface d'identification.

3.5.2.1 Description de l'onglet CF

L'onglet CF permet d'afficher la liste des ressources à recommander à l'apprenant, qui s'est préalablement identifié, en utilisant la technique de recommandation basée sur le filtrage collaboratif.

En premier lieu, nous affichons à l'apprenant la liste de tous ses voisins, afin qu'il puisse avoir une idée des utilisateurs avec qui il partage les mêmes goûts et intérêts. Ensuite, une fois la liste des voisins construite, nous générons la liste des recommandations. Ainsi, nous affichons pour chaque ressource à recommander son identifiant "id", son nom "nom" et la note calculée "note", qui est une note approximative que pourrait attribuer l'apprenant à la ressource. Comme l'illustre la figure suivante :

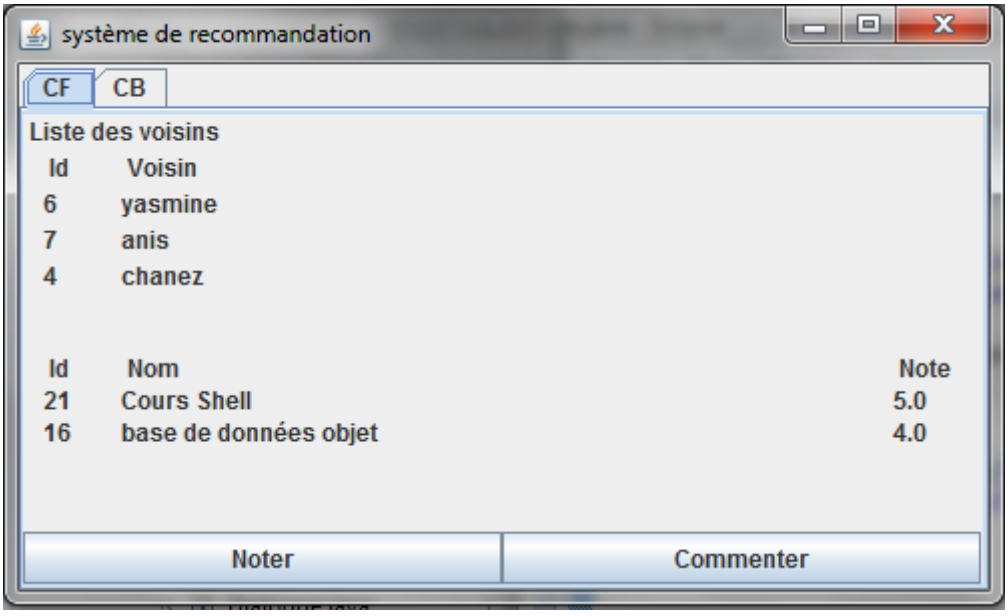

FIGURE 3.3: Interface Filtrage collaboratif.

L'apprenant peut ainsi choisir une ressource à consulter sur la plateforme de formation MOODLE, en se basant directement sur la note qu'il pourrait lui attribuer.

Une fois que l'apprenant ait consulté une ressource, on lui offre la possibilité de noter ou commenter la ressource en question, afin d'améliorer les recommandations faites par notre système. Cela se fait à l'aide des boutons "noter" ou "commenter".

Dans le cas où l'apprenant décide d'attribuer une note directement à la ressource, il appuie sur le bouton "noter", une fenêtre s'affiche avec comme champs obligatoires l'identifiant de la ressource "id" et la note à attribuer "note", comme le montre la figure suivante :

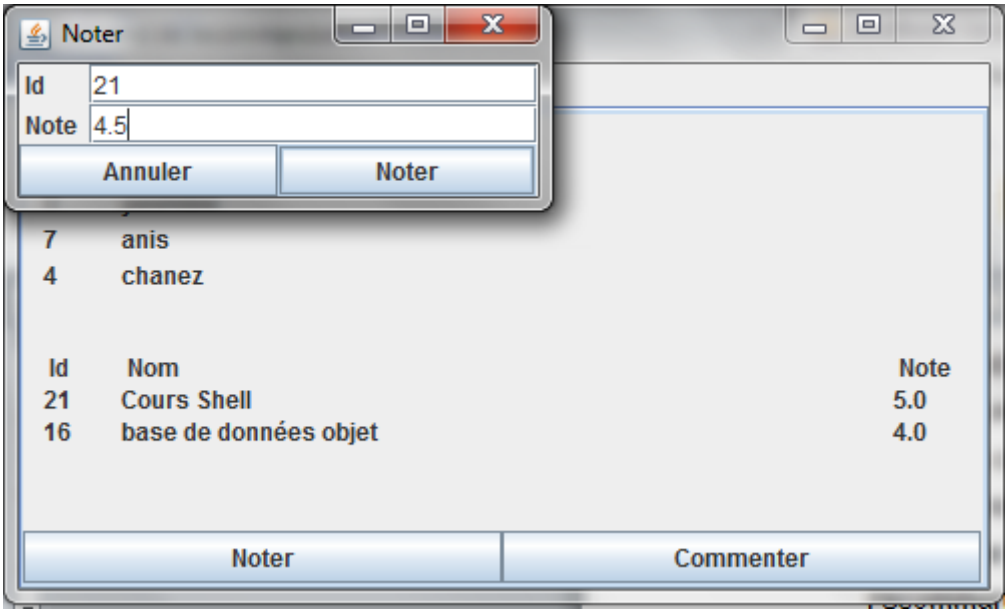

FIGURE 3.4: Interface Notation de ressource.

Une fois ces champs remplis, il ne reste plus qu'à valider afin que la note soit mise à jour dans la base de données de notre système, qui sera utilisée, à la prochaine ouverture du système de recommandation, pour améliorer le service de recommandation basée sur le filtrage collaboratif. Une notification est renvoyée à l'apprenant pour l'informer que la note a bien été ajoutée, comme l'illustre la figure suivante :

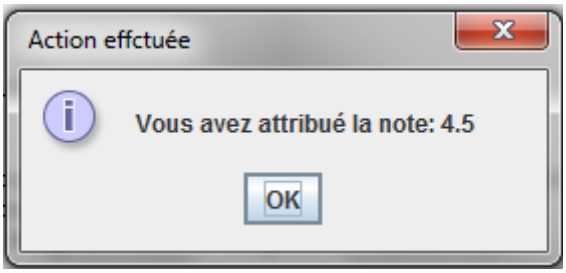

FIGURE 3.5: Interface confirmation de la note attribuée.

Si l'apprenant, au contraire, décide de laisser un commentaire à propos de la ressource, il appuie sur le bouton "commenter" qui lui affiche une fenêtre avec toujours comme champs obligatoires l'identifiant de la ressource, et le commentaire qu'il souhaite laisser. Ce commentaire a comme but d'exprimer l'avis de l'apprenant à propos de la ressource qu'il a consultée. La figure suivante montre cela :

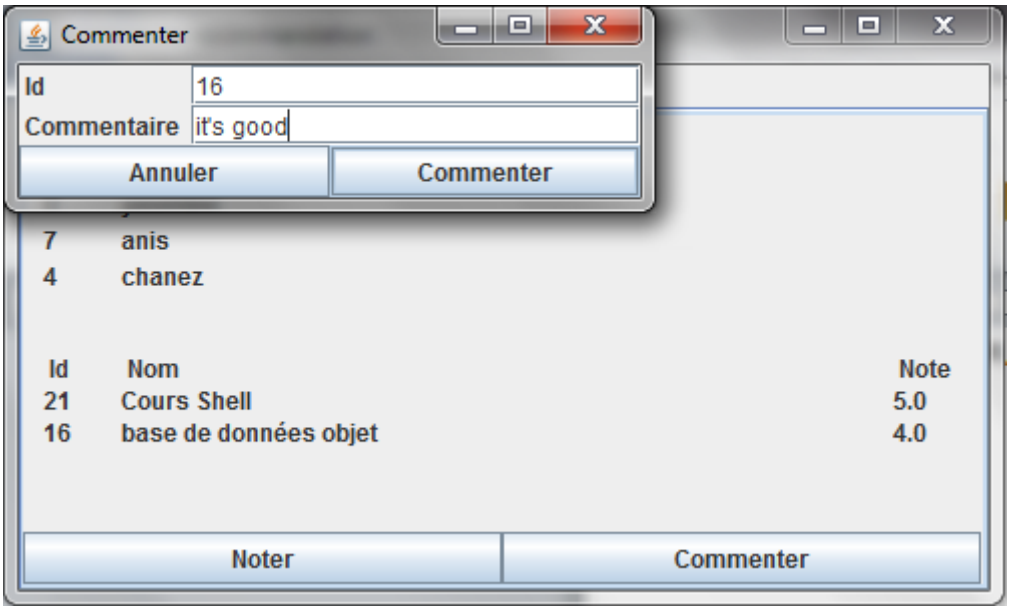

FIGURE 3.6: Interface Commenter ressource.

Une fois le commentaire validé par l'apprenant, il est directement soumis à une analyse des sentiments, qui se chargera de déduire une note équivalente au commentaire donné. Une fois cette note déduite, elle sera utilisée pour mettre à jour le système et ainsi donner

des résultats plus pertinents aux apprenants. Une fenêtre de dialogue s'affiche pour notifier l'apprenant et l'informer de la note déduite par le commentaire qu'il a fait, comme dans la figure suivante :

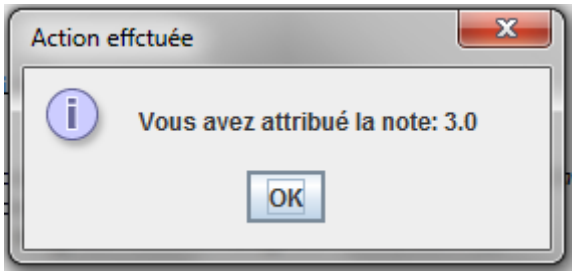

FIGURE 3.7: Interface confirmation de la note attribuée.

3.5.2.2 Description de l'onglet CB

Dans l'interface de notre système, nous pouvons également choisir de consulter l'onglet CB, qui affiche la liste des ressources recommandées à l'apprenant, en utilisant cette fois la technique de recommandation basée sur le contenu.

Les ressources sont affichées par ordre décroissant de pertinences, ainsi les ressources se trouvant en haut de la liste sont celles qui sont le plus susceptibles de satisfaire les besoins de l'apprenant. Pour chaque ressource, nous affichons son identifiant "id" et son nom "nom", comme illustré dans la figure suivante :

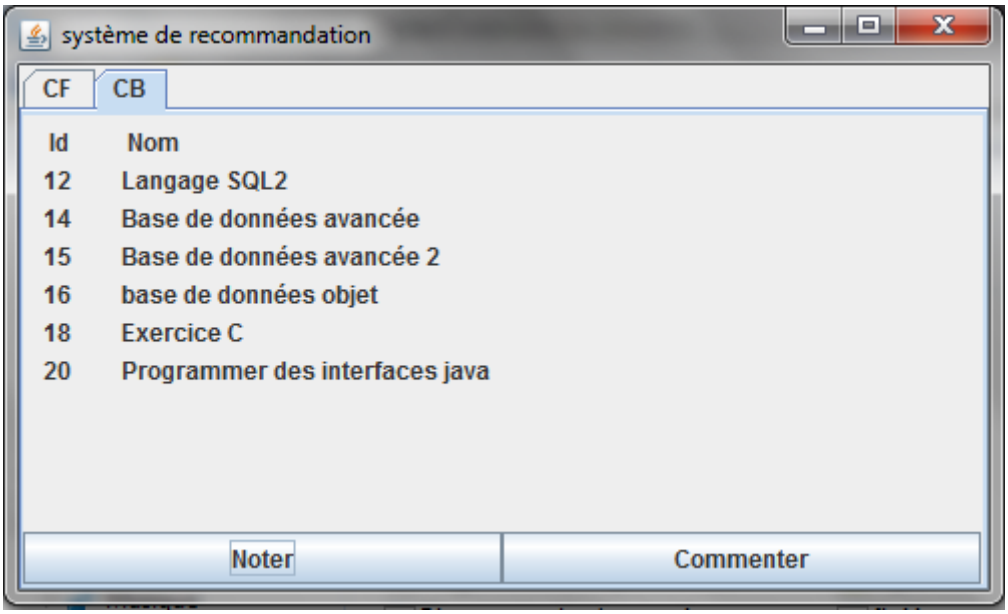

FIGURE 3.8: Interface filtrage basé sur le contenu.

Comme dans l'onglet CF, l'apprenant a le choix de noter ou alors commenter une ressource qu'il a consultée. Leur principe reste toujours le même.

3.5.2.3 Le forum de la plateforme de formation MOODLE

Dans notre application, le forum de la plateforme MOODLE est considéré comme une simulation d'un réseau social, pour lequel nous faisons une analyse régulière afin de détecter les activités de chaque apprenant. Les commentaires laissés sur cette plateforme sont analysés pour déduire les intérêts de chacun. Chaque cours de la plateforme dispose d'un espace "Forum de discussion" qui donne la possibilité aux apprenants de poser des questions, d'exprimer leurs requêtes, etc.

Comme illustré dans la figure 3.9, un apprenant a posté un commentaire dans le forum, ce dernier sera analysé afin de déduire l'opinion exprimée à travers. Dans le cas où l'on en parle positivement, une extraction de concepts est effectuée en utilisant une ontologie E-learning. Une fois le concept extrait, il est directement ajouté dans les centres d'intérêts de l'apprenant, comme le montre la figure 3.10, le concept BDD et son instance SQL ont bien été ajouté aux centres d'intérêts.

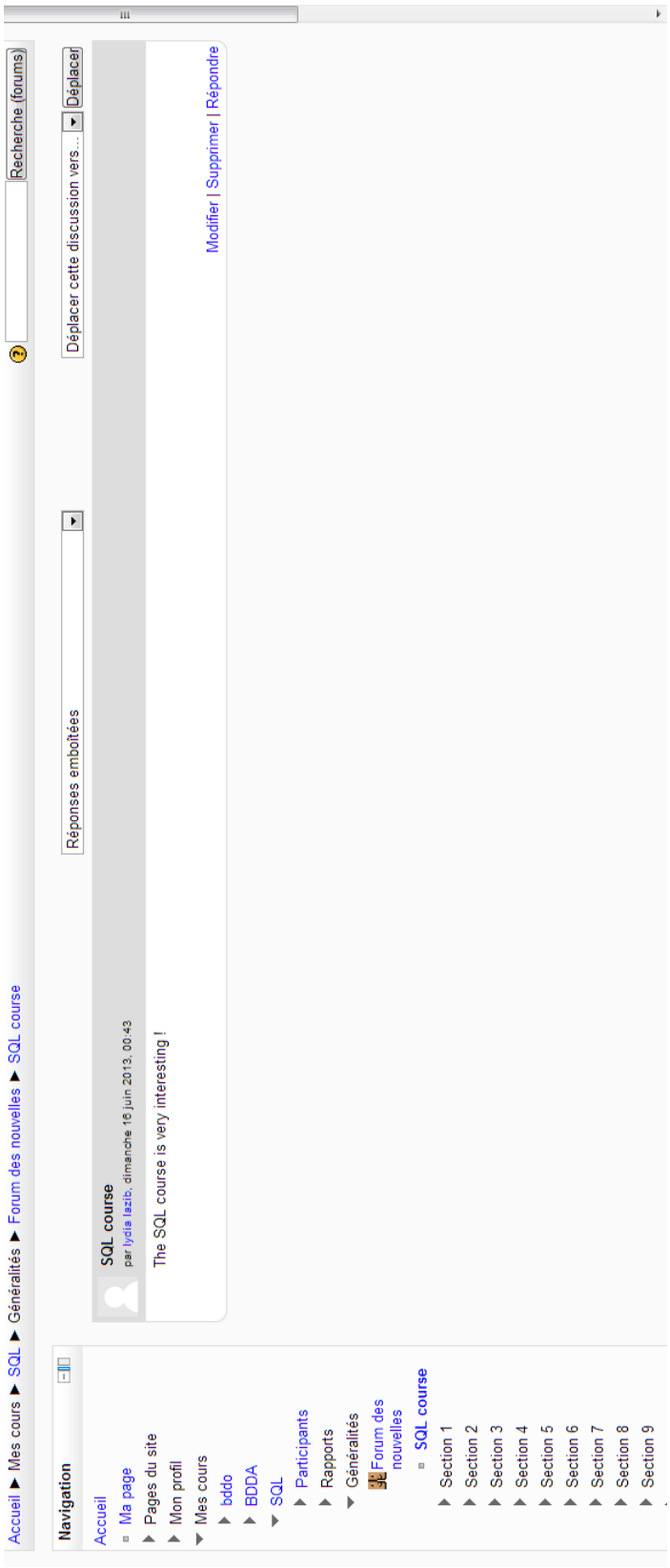

FIGURE 3.9: Commentaire posté dans le forum de MOODLE.

FIGURE 3.10: Description des centres d'intérêts de l'apprenant dans la base de données MOODLE.

3.6 Conclusion

Ce chapitre, consacré à l'implémentation de notre système de recommandation, nous a permis de montrer la faisabilité d'un tel système en adoptant les démarches abordées dans le chapitre précédent.

Nous avons présenté les différents outils utilisés pour son implémentation, nous avons également donné une description des différentes tables de la base de données MOODLE utilisées par notre système. Des exemples d'interfaces de l'application développée sont présentés.

Nous avons pu recommander des ressources à des apprenants en nous basant sur la méthode du filtrage collaboratif, ainsi que sur le filtrage basé sur le contenu. Ainsi, un apprenant aura le choix de consulter les ressources qui lui conviennent le plus.

Conclusion générale

Ce travail de recherche s'inscrit dans le cadre de l'e-learning, et se centre sur la recommandation de ressources pédagogiques à des apprenants en se basant essentiellement sur leurs préférences et centres d'intérêts.

Nous avons commencé par mener une recherche bibliographique, afin d'étudier les différentes méthodes de recommandation existantes, les différentes approches d'analyse de réseaux sociaux, ainsi que les méthodes d'analyse des sentiments.

Cette étude nous a permis d'adopter une approche adéquate à nos besoins. Ainsi, pour la partie recommandation de notre approche on s'est basé sur la méthode du filtrage collaboratif et sur la méthode du filtrage basé sur le contenu. Ce système utilise les données des apprenants laissées sur la plateforme de formation pour personnaliser les recommandations de ressources.

Afin de réaliser ce système, nous avons utilisé la plateforme de formation MOODLE, dans laquelle nous faisons une simulation d'un réseau social en utilisant le forum de la plateforme à cet effet. Ainsi, les commentaires laissés par les apprenants dans ce forum sont analysés pour en déduire leurs préférences.

Puis nous avons créé un certain nombre de ressources et simulé des activités d'apprenants sur cette plateforme, telles que des notes attribuées par des apprenants à des ressources, des commentaires laissés dans les forums, des ressources consultées, etc. afin qu'elles soient exploitées par notre système et pouvoir faire des recommandations de ressources. Ainsi, notre système exploite les notes dans son filtrage collaboratif, et les commentaires laissés dans le forum ainsi que les ressources consultées dans son filtrage basé sur le contenu.

En perspectives, pour des résultats plus pertinents nous proposons d'utiliser un site simulé comme réseau social, à la place du forum, afin de détecter les activités des apprenants avec leurs amis, et appliquer par la suite la méthode d'analyse des réseaux sociaux

basée sur "les usages et les tags" abordée dans le chapitre de conception, et ainsi, avoir des recommandations plus personnalisées, et plus proches des attentes des utilisateurs.

Bibliographie

[1] http ://www.allaboutelearning.lu.

[2] Florence AMARDEILH. *Web sémantique et informatique linguistique : propositions méthodologiques et réalisation d'une plateforme logicielle*. PhD thesis, Paris Nanterre, 2007.

[3] Djida BAHLOUL. *Une approche hybride de gestion des connaissances basée sur les ontologies : application aux incidents informatiques*. PhD thesis, L'Institut National des Sciences Appliquées de Lyon, 2006.

[4] Faiza BELBACHIR. Expérimentation de fonctions pour la détection d'opinions dans les blogs. Master's thesis, Université de Toulouse, 2010.

[5] Ahcene BENAYACHE. *CONSTRUCTION D'UNE MEMOIRE ORGANISATIONNELLE DE FORMATION ET EVALUATION DANS UN CONTEXTE ELEARNING : LE PROJET MEMORAE*. PhD thesis, Université de Technologie de Compiègne (UTC), 2005.

[6] Sihem Benlizidia. Loresa : Un système de recommandation d'objets d'apprentissage basé sur les annotations sémantiques. Master's thesis, Université de Montréal, 2007.

[7] Geoffrey BONNIN. *Vers des systèmes de recommandation robustes pour la navigation Web : inspiration de la modélisation statistique du langage.* PhD thesis, Université de Nancy, 2010.

[8] Farida BOUARAB. *Modélisation basée ontologie pour l'apprentissage intéractif - Application à l'évaluation des connaissances de l'apprenant.* PhD thesis, UMMTO, 2010.

[9] Fatiha BOUDALI. Publication et découverte des web services pour le domaine du e-learning. Master's thesis, Institut National de formation en Informatique (I.N.I), 2008.

[10] Guillaume Erétéo. Analyse des réseaux sociaux et web sémantique : un état de l'art. Agence Nationale de la Recherche ANR, 07 2009.

[11] Waad GASMI. Le filtrage basé sur le contenu pour la recommandation de cours (fcrc). Master's thesis, UNIVERSITÉ DE MONTRÉAL, 2011.

[12] Alaa HAMOUDA and Mohamed ROHAIM. Reviews classification using sentiwordnet lexicon. *The Online Journal on Computer Science and Information Technology (OJCSIT)*, 2.

[13] Walid Kassem, Ahmad Mounajed, and Nadia Saadoun. Etat de l'art du e-learning, 02 2004.

[14] Sonia LAJMI. *Annotation et recherche contextuelle des documents multimédias socio-personnels*. PhD thesis, INSA de Lyon, 2011.

[15] Phuc Hiep LUONG. *GESTION DE L'ÉVOLUTION D'UN WEB SÉMANTIQUE D'ENTREPRISE*. PhD thesis, Ecole des Mines de Paris, 2007.

[16] Alexandre PASSANT. De l'intérêt du web sémantique pour le web social, et réciproquement. In *Le Web Social 2010*, 2010.

[17] Michael PAZZANI and Daniel BILLSUS. Content-based recommendation systems. *THE ADAPTIVE WEB : METHODS AND STRATEGIES OF WEB PERSONALIZATION.*, 4321 :325–341, 2007.

[18] Quang Trung Tien PHAN. Ontologies et web services. Institut de la Francophonie pour l'Informatique, 2005.

[19] Romain Picot-Clémente. *Une architecture générique de Systèmes de recommandation de combinaison d'items. Application au domaine du tourisme*. PhD thesis, Université de Bourgogne, 2011.

[20] Damien POIRIER, Françoise FESSANT, and Isabelle TELLIER. Reducing the cold-start problem in content recommendation through opinion classification. In *Web Intelligence (2010)*, 2010.

[21] Elie RAAD. *Relationship discovery in social networks (Découverte des relations dans les réseaux sociaux).* PhD thesis, UNIVERSITÉ DE BOURGOGNE, 2011.

[22] Toby SEGARAN. *Programming Collective Intelligence.* O'REILLY, 2007.

[23] BACH Thành Lê. *Construction d'un Web sémantique multi-points de vue.* PhD thesis, L'École des Mines de Paris à Sophia Antipolis, 2006.

[24] Philippe Torloting. *Enjeux et perspectives des réseaux sociaux .* PhD thesis, Institut supérieur du commerce de Paris, Marketing, Management et Technologie de l'information., 2006.

[25] Matthieu Vernier and Laura Monceaux. Enrichissement d'un lexique de termes subjectifs à partir de tests sémantiques. Technical report, Université de Nante, 2009.