



République Algérienne Démocratique et Populaire



Ministère de l'Enseignement Supérieur et de la Recherche Scientifique

Université Akli Mohand Oulhadj de Bouira

Faculté des Sciences et des Sciences Appliquées

Département d'Informatique

Mémoire de Master

en Informatique

Spécialité : ISIL

Thème

Développement d'une application mobile pour la
visite personnalisée de sites culturels

Encadré par

— M.CHOUIREF Zahira

Réalisé par

— MOHAMMEDI Mouna

— ZAMOUM Hafidha

2018/2019

Remerciements

Louange A Dieu, le miséricordieux, sans lui rien de tout cela n'aurait pu être.

On tient à remercier vivement notre encadreur Dr. CHOUIREF Zahira pour avoir encadré et dirigé notre travail, pour sa disponibilité, ses orientations, ses précieux conseils et ses encouragements qui nous ont permis de mener à bien ce travail.

On tiens à remercier tous les enseignants de département informatique de nous avoir faire cette formation et permis d'acquérir des nouvelles connaissances.

Nous tenons à exprimer notre gratitude aux membres de jurys qui nous ont honoré en acceptant de présider notre soutenance. Nous remercions nos amis qui ont contribué à la finalisation des quelques tâches dans ce mémoire.

Nous tenons à remercier tous ceux qui ont contribué à ce travail parfois sans le savoir ou du moins sans mesurer la portée de leurs influences. Un énorme merci à nos familles pour leurs éternel soutient et la confiance qu'ils ont en nos capacité.

Dédicaces

Je dédie ce travail qui n'aura jamais pu voir le jour sans les soutiens indéfectibles et sans limite de mes chers parents qui ne cessent de me donner avec amour le nécessaire pour que je puisse arriver à ce que je suis aujourd'hui. Que dieux vous protège et que la réussite soit toujours à ma portée pour que je puisse vous combler de bonheur.

Je dédie aussi ce travail à :

- Mon marie et ma belle famille.
- Ma grand-mère.
- Mes frères, ma sœur et leur famille.
- Mes oncles, mes tantes et leur famille.
- Tous mes cousins et cousines.
- Tous mes amis, mes collègues et tous ceux qui m'estiment.

MOHAMMEDI Mouna.

Dédicaces

je dédie ce travail :

À mes Très Chers Parents Hocine et Dalila

Tous les mots du monde ne sauraient exprimer l'immense amour que je vous porte, ni la profonde gratitude que je vous témoigne pour tous les efforts et les sacrifices que vous n'avez jamais cessé de consentir pour notre instruction et notre bien-être. Que dieu vous protège et vous garde pour nous.

À mon mari amine.

À mes chers frères Riadh et abderrahman et leurs familles et mes précieuses soeurs assia et serine et leurs familles, les mots ne peuvent résumer notre reconnaissance et notre amour à votre égard.

À tous les membres de ma famille.

À tous les membres de ma belle famille À mes adorables amies, pour votre fidélité et votre soutien.

Que toute personne nous ayant aidé de près ou de loin, trouve ici l'expression de notre reconnaissance.

Finalement à tous ceux qui me portent dans leurs coeurs

ZAMOUM Hafidha.

Table des matières

Table des matières	i
Table des figures	v
Liste des tableaux	vii
Liste des abréviations	ix
Introduction générale	1
1 Systèmes de recommandation et apprentissage automatique	4
1.1 Introduction	4
1.2 Historique	4
1.3 Définition des systèmes de recommandation	6
1.4 Classification des systèmes de recommandation	8
1.4.1 La classification classique :	8
1.4.2 La classification de [Rao and Talwar, 2008] :	8
1.4.3 La Classification de [Su et al, 2009] :	8
1.5 Différents types de recommandation	9
1.5.1 Recommandation démographique :	9
1.5.2 Recommandations à base de connaissances	10
1.5.3 Recommandation communautaire (Social)	11
1.5.4 Recommandation basé sur le contenu	12
1.5.5 Recommandation basé sur le filtrage collaboratif	15
1.5.6 Filtrage hybride	21

1.6	Les objectifs de la recommandation :	23
1.7	Avantages et inconvénients des systèmes de recommandation	24
1.8	Systèmes de recommandation sensibles au contexte	26
1.9	L'apprentissage automatique :	26
1.9.1	Les algorithmes d'apprentissage automatique :	26
1.9.2	Types d'apprentissage automatique :	27
1.10	Conclusion	30
2	Contexte, Internet des objets	32
2.1	Introduction	32
2.2	Contexte	32
2.2.1	La notion de contexte	32
2.2.2	Classification des informations de contexte	35
2.2.3	Catégories du contexte	35
2.2.4	Caractéristiques du contexte	42
2.3	Internet des objets	43
2.3.1	Définition de l' IoT	43
2.3.2	Histoire de l' IoT	44
2.3.3	Quelques domaines d' IoT	45
2.3.4	Quelques exemples d'IoT	46
2.4	Conclusion	47
3	Approche proposée	48
3.1	Introduction	48
3.1.1	L'approche basées sur le filtrage collaboratif	48
3.2	La classification automatique supervisée	49
3.3	Les projets de recommandation en relation avec l'apprentissage automa- tique	49
3.3.1	Recommandation pour le secteur de l'hôtellerie à la dernière minute	49
3.3.2	Moteur de recommandation des cadeaux	50
3.3.3	BlaBlaCar :	52
3.4	Architecture générale de notre application	53
3.5	Base de données	55

3.6	K plus proches voisins	55
3.6.1	Architecture de l’algorithme KPP-v	55
3.6.2	Application de l’algorithme K Plus Proche voisin :	56
3.7	Notre algorithme proposé ”Algorithme CPP-s”	65
3.7.1	Application de l’algorithme CPP-s	71
3.8	Conclusion	78
4	Expérimentation et Test	79
4.1	Introduction	79
4.2	Objectif de l’expérimentation	79
4.3	Environnement informatique utilisé	80
4.3.1	Environnement materiel	80
4.3.2	Environnement logiciel	80
4.4	Analyse des résultats	81
4.4.1	Le résultat de l’algorithme KPP-v	81
4.4.2	résultat de l’algorithme CPP-s	83
4.5	Métrique dévaluation	85
4.6	Expérimentations et résultats	86
4.6.1	Le calcule de Rappel, Précision et F-mesure de l’algorithme KPP-v	87
4.6.2	Le calcule de Rappel, Précision et F-mesure de l’algorithme CPP-s	90
4.7	Comparaison des résultats obtenus	92
4.8	Partie application mobile	93
4.8.1	Environnement de développement	93
4.9	Langages de développement	94
4.9.1	JAVA	94
4.9.2	PHP	94
4.9.3	SQL	94
4.9.4	XML	94
4.9.5	JSON	95
4.9.6	les APIs utilisé	95
4.10	L’application ”Tour-App”	96
4.11	les interfaces de l’application ”Tour-App”	97
4.11.1	Enjoy with traveling	98

4.11.2 Visit the nearest place	98
4.11.3 adding and share place	98
4.11.4 sing in	98
4.12 Conclusion	101
Conclusion générale	102
Bibliographie	103
Annexe 1	113
Annexe 2	114

Table des figures

1.1	Schéma général du filtrage d'information	7
1.2	Classification principale des systèmes de recommandations	9
1.3	Schéma général du filtrage d'information	10
1.4	Recommandations à base de connaissances	11
1.5	Recommandations communautaire	12
1.6	Recommandation basé sur le contenu	14
1.7	Recommandation basé sur le filtrage collaboratif	16
1.8	Exemple de recommandation basé sur le filtrage collaboratif.	17
1.9	Les méthodes de filtrage collaboratif	18
1.10	Les algorithmes d'apprentissage supervisé et non-supervisé.	30
2.1	Les différents éléments du contexte	37
2.2	Graphe du nombre existé et estimé des objets connecter.	44
2.3	Historique de la technologie de la connectivité des objets	45
3.1	Exemple de processus de la recommandation des cadeaux.	51
3.2	Exemple de l'application BlaBlaCar	52
3.3	Architcture de notre application	54
3.4	Architecture de l'algorithme KPP-v	56
3.5	Shéma représentant distribution des scores selon l'algorithme CPP-s	67
3.6	Matrice obtenue pour le sexe	68
3.7	Matrice obtenue pour la distance	68
3.8	Matrice obtenue pour le type des places	69
3.9	Matrice obtenue pour les likes	70

3.10 La matrice globale User/place	70
4.1 Les top 10 places recommandées pour les utilisateurs selon le KPP-v	82
4.2 Les top 10 places recommandées pour le nouvel utilisateur selon le KPP-v	83
4.3 Les top 10 place recommandées pour les utilisateurs selon l'algorithme CPP-s	84
4.4 Les top 10 places recommandées pour le nouvel utilisateur selon l'algo- rithme CPP-s	85
4.5 Trigonométrie des pourcentages de recommandation de l'algorithme KPP- v	88
4.6 Graphe de Rappel-Précision de l'algorithme KPP-v	89
4.7 Trigonométrie des pourcentages de recommandation de l'algorithme CPP- s	91
4.8 Graphe de les résultats de l'algorithme Score	92
4.9 Histogramme comparatif des deux algorithmes	92
4.10 Interface d'accueil	97
4.11 Liste de fonctionnalités	97
4.12 Liste de fonctionnalités	98
4.13 Page Main	99
4.14 Chercher endroit	99
4.15 Liste des préférences	100
4.16 Liste de recommandations	100

Liste des tableaux

1.1	Les avantages et les inconvénients des techniques de recommandations. . .	25
2.1	Classification des informations de contexte	36
3.1	base d'apprentissage KPP (place)	57
3.2	base de test KPP(user)	57
3.3	V min et V max des attributs	58
3.4	Base d'apprentissage normalisée KPP-v(places)	59
3.5	Base de test normalisé KPP(user)	59
3.6	base d'apprentissage normalisé + distance KPP(place)	61
3.7	base d'apprentissage normalisé kpp2(place)	62
3.8	Base d'apprentissage KPP(user)	63
3.9	Base de teste KPP (new user)	63
3.10	V min et V max des attributs	63
3.11	Base d'apprentissage normalisée KPP(user)	64
3.12	Base de teste normalisée KPP(new user)	64
3.13	Base d'apprentissage normalisée + distance KPP-v (user)	64
3.14	La table des places	72
3.15	La table des utilisateurs	72
3.16	La table des place + Score (CPP-s)	74
3.17	la base d'apprentissage normalisée CPP-s(place)	75
3.18	la table des utilisateurs (user)	76
3.19	la table (new user)	76
3.20	la table des utilisateurs + Score CPP-s (user)	77

4.1	Les bibliothèques de Python utilisées dans notre projet	81
4.2	La moyenne de recommandation selon algorithme KPP-v	87
4.3	Tableau de rappel, précision et f-mesure de l'algorithme KPP-V	89
4.4	Nb de recommandation pour chaque utilisateur selon l'algorithme CPP-s .	90
4.5	Tableau de rappel, précision et f-mesure de l'algorithme CPP-s	91

Liste des abréviations

API	Application Programming Interface
CPP-s	Contexte Profil Préférence Score
FBC	Filtrage à Base de Contenu
FC	Filtrage Collaboratif
FD	Filtrage Démographique
GPS	Global Positioning System
IoT	L'internet des objets
JSON	JavaScript Object Notation
KPP-V	K Plus Proche Voisin
MAE	Mean Absolute Error
MS	Short Message Service
NFC	Near Field Communication
PC	Personal Computer
PDA	Personal Digital Assistance
PHP	Hypertext Preprocessor
RFID	Radio Frequency IDentification
RMSE	Root Mean Squared Error
RVM	Relevance Vector Machine
SQL	Structured Query Language
SVM	Séparateurs à Vastes Marges
XML	eXtensible Markup Language
W3C	World Wide Web Consortium

Introduction générale

Le cadre de ce travail concerne l'aide à la visite culturelle, Ce domaine est particulièrement touché par le problème de la surcharge d'information car ils renferment un volume d'information assez conséquent. De ce fait, les visiteurs sont confrontés à plusieurs problèmes : i) ils sont submergés par le nombre très important de choix possibles dans l'espace qu'ils explorent, ii) les visiteurs ne savent pas forcément ce qu'ils devraient voir ou ce qu'ils pourraient apprécier, iii) ils se limitent à voir les items les plus populaires comme dans la plupart des visites guidées. En conséquence, ils peuvent perdre du temps en regardant des œuvres ou en visitant des points d'intérêt. Qui ne les intéressent pas. Inversement, ils peuvent manquer des œuvres ou des points d'intérêt qui auraient pu les intéresser.

Un des domaines de recherche principaux relatifs à la problématique de la surcharge d'information est le domaine des systèmes de recommandation. Ces systèmes sont capables de fournir des recommandations adaptées aux préférences et aux besoins des utilisateurs. Initialement conçus pour la recommandation de ressources Web, films, etc. Les systèmes de recommandations ont devenus de plus en plus populaires et sont aujourd'hui un composant principal de beaucoup d'applications dans différents domaines. Un avantage très conséquent des systèmes de recommandation est que l'utilisateur n'a pas besoin de formuler de requêtes. Sa seule requête est implicite, elle peut se traduire par : " Quelles sont les ressources qui correspondent à mes préférences, mes besoins et mes contraintes? ". Les systèmes de recommandation peuvent être classés en plusieurs types d'approches. Nous nous intéressons sur l'approche basée sur le filtrage collaboratif.

Les approches basées sur le filtrage collaboratif, permet de fournir des recommandations à un utilisateur sans forcément considérer le contenu des ressources, mais en se basant sur l'analyse du comportement et/ou des appréciations de l'utilisateur à fin de recommander les ressources qui ont été appréciées par d'autres utilisateurs ayant des goûts similaires.

Nos travaux de recherche concernent les systèmes de recommandation dans le but d'offrir aux visiteurs des points d'intérêt personnalisés, en se basant sur les systèmes de recommandation basées sur le filtrage collaboratif.

Problématique

Notre travail vise à concevoir un systèmes, sur dispositif mobile, afin d'aider à améliorer l'expérience d'un visiteur. Il s'agit d'adapter et personnaliser le visite de l'utilisateur en fonction de ses préférences, son contexte et ses contraintes. Nos travaux s'orientent donc vers les systèmes de recommandation qui se sont avérés être très efficaces pour aider les utilisateurs à accéder aux ressources par lesquels il seraient potentiellement intéressés.

Une problématique supplémentaire pour le tourisme est que les points d'intérêt à recommander sont généralement de différents types, par exemple : Musée ,montagne, loisir, etc.

La nature hétérogène de ces points d'intérêt rend inexploitable les systèmes de recommandation classiques qui fournissent à l'utilisateur des recommandations sous forme de listes triées. Là aussi, d'autres contraintes entrent en jeu, comme la distance, les points d'intérêt qui doivent être proche du visiteur, etc. Il faut donc être en mesure de pouvoir recommander à l'utilisateur des parcours respectant ses contraintes.

Objectifs

Le travail réalisé vise à répondre aux objectifs suivants :

- Le système proposé capable de prendre en compte le profil, les préférence du demandeur de service ainsi que son contexte spatio-temporel. Pour cela nous allons proposer une approche qui se base sur notre algorithme CPP-s (Contexte- Profil-Préférence-Score) .

- L'évaluation expérimentale du système que nous allons proposer, en utilisant un data-set que nous avons généré (vue l'absence d'un data-set réel) démontre sa qualité et sa capacité à améliorer à la fois la précision et la diversité des résultats.

Nous avons organisé ce travail comme suit :

le premier chapitre est consacré à la présentation des concepts de base des systèmes de recommandation, les principales techniques sont décrites ainsi que leurs avantages et inconvénients. Il est consacré aussi à la présentation de base de l'apprentissage automatique.

Le deuxième chapitre décrit dans un premier temps le contexte ainsi que ses catégories et ses caractéristiques. Et dans le deuxième temps les objets connectés (l'IoT).

le troisième chapitre décrit notre système de recommandation collaboratif. Qui se base sur deux approches KPP-v et CPP-v.

La quatrième chapitre est consacré à l'expérimentation et l'évaluation et aussi la représentation de l'application.

Enfin, nous terminerons ce mémoire par une conclusion générale qui résume le travail effectué et quelques perspectives.

Systèmes de recommandation et apprentissage automatique

1.1 Introduction

Avec l'avènement d'internet, nous assistons aujourd'hui à une surcharge d'information. Par exemple, une personne qui désire lire un cours se retrouve devant un volume très grand de propositions de cours. Ce qui rend la tâche de choix d'un cours très difficile. Les systèmes de recommandation sont apparus pour remédier à ce problème.

Dans ce chapitre, nous commençons par définir ce qu'un système de recommandation. Ensuite, nous présentons les approches de filtrages qui permettent la recommandation. Nous enchaînons en présentant les différentes mesures de similarité qui permettent aux systèmes de recommandation de faire l'appariement entre les concepts. En fin, nous terminons en citant les limites et inconvénients des systèmes de recommandation,

1.2 Historique

La capacité des ordinateurs pour faire des recommandations à des utilisateurs a été reconnue assez tôt dans l'histoire de l'informatique. Grundy [1], un système bibliothécaire, était une première étape vers des systèmes de recommandation automatiques.

Ce système était assez primitif. Il classait les utilisateurs en "stéréotypes" en se basant sur une courte interview, et utilisait ces stéréotypes pour produire des recommandations

de livres.

Ce travail constituait une première tentative intéressante dans le domaine des systèmes de recommandation. Cependant, son utilisation est restée très limitée. Au début des années 1990, le filtrage collaboratif apparaît comme une solution pour faire face à la surcharge d'information.

L'année 1992 voit l'apparition du système de recommandation de documents Tapestry [2], ainsi que la création du laboratoire de recherche "GroupLens", qui travaille explicitement sur le problème de la recommandation automatique dans le cadre des forums de news de Usenet. Tapestry avait pour but de recommander à des groupes d'utilisateurs des documents issus des newsgroups susceptibles de les intéresser. L'approche utilisée était de type "plus proches voisins" à partir de l'historique de l'utilisateur. On parle alors de filtrage collaboratif manuel, comme une réponse au besoin d'outils pour le filtrage de l'information énoncé à la même époque.

La recommandation résulte d'une action collaborative des utilisateurs qui recommandent à d'autres utilisateurs des documents en leur attribuant des notes d'intérêt selon certains critères. Les systèmes de filtrage collaboratif automatiques apparaissent ensuite. "GroupLens" [3] utilise cette technique pour identifier les articles de Usenet susceptibles d'être intéressants pour un utilisateur donné. Les utilisateurs doivent seulement attribuer des notes ou effectuer d'autres opérations observables (par exemple, lire un article) ; le système combine alors ces données avec les notes ou les actions d'autres utilisateurs pour fournir des résultats personnalisés. Avec ces systèmes, les utilisateurs n'ont aucune connaissance directe des opinions des autres utilisateurs, ni des articles présents dans le système. Au cours de ces dernières années, les systèmes de recommandation deviennent un sujet d'un intérêt croissant dans les domaines de l'interaction homme-machine, de l'apprentissage automatique ainsi que la recherche d'information.

En 1995 apparaissent successivement Ringo [4], un système de recommandation de musique, basé sur les appréciations des utilisateurs et Bellcore [5], un système de recommandation de vidéos. La même année, "GroupLens" crée la société Net Perceptions dont le premier client a été Amazon. De nos jours, les systèmes de recommandation sont devenus des composantes incontournables pour la plupart des sites du e-commerce.

1.3 Définition des systèmes de recommandation

Les systèmes de recommandation peuvent être définis de plusieurs façons, vue la diversité des classifications proposées pour ces systèmes, mais il existe une définition générale de Robin Burke [6] qui les définit comme suit : *"Des systèmes capable de fournir des recommandations personnalisées permettant de guider l'utilisateur vers des ressources intéressantes et utiles au sein d'un espace de données important"*.

Un système de filtrage d'information, ou un système de recommandation (recommender system)[7], est un filtre de flux entrant d'information de façon personnalisée pour chaque acteur. Autrement dit, dans un but de personnaliser la recherche d'information dans un domaine d'application particulier, un système de filtrage collecte, sélectionne, classe et suggère à l'utilisateur les informations qui répondent vraisemblablement à ses intérêts à long termes.

Les deux entités de base qui apparaissent dans tous les systèmes de recommandations sont l'utilisateur et l'item. L'«utilisateur» est la personne qui utilise un système de recommandation, donne son opinion sur divers items et reçoit les nouvelles recommandations du système. L'«Item» est le terme général utilisé pour désigner ce que le système recommande aux utilisateurs. Les données d'entrée pour un système de recommandation dépendent du type de l'algorithme de filtrage employé. Généralement, elles appartiennent à l'une des catégories suivantes :

- **Les estimations** : (également appelées les votes), expriment l'opinion des utilisateurs sur les articles.
- **Les données démographiques** : se réfèrent à des informations telles que l'âge, le sexe, le pays et l'éducation des utilisateurs. Ce type de données est généralement difficile à obtenir et est normalement collecté explicitement.
- **Les données de contenu** : qui sont fondées sur une analyse textuelle des documents liés aux éléments évalués par l'utilisateur. Les caractéristiques extraites de cette analyse sont utilisées comme entrées dans l'algorithme de filtrage afin d'en déduire un profil d'utilisateur [8].

Pour réaliser le filtrage, le système de recommandation (SR) utilise les profils représentant des préférences relativement stables des utilisateurs pour calculer des recommandations. Ce calcul se fait par la prédiction des scores qu'un utilisateur est susceptible d'attribuer aux contenus. Le SR adapte ce profil au cours du temps en exploitant au mieux le retour

de pertinence que les utilisateurs fournissent sur les informations (documents) reçues. Par exemple, dans la figure 1.1, la fonction de décision du système traite le flux entrant de document pour suggérer à l'utilisateur, en consultant son profil, les documents qu'il préfère. A son tour, l'utilisateur doit fournir des évaluations c'est-à-dire évaluer fréquemment les recommandations, pour que le système comprenne mieux ses besoins en information, et lui fournisse par conséquent de meilleures nouvelles recommandations.

Les trois parties suivantes constituent un système de recommandation :

- **Les producteurs** : Ce sont ceux qui vont permettre de faire les recommandations, ils "fourniront" les données.
- **Le module de calcul** : Il s'agit de l'algorithme en lui même. En entrée il y a toutes les données et la demande et en sortie les différentes recommandations.
- **Le consommateur** : C'est celui qui demande la recommandation.

La figure 4.16 présente Schéma général du filtrage d'information

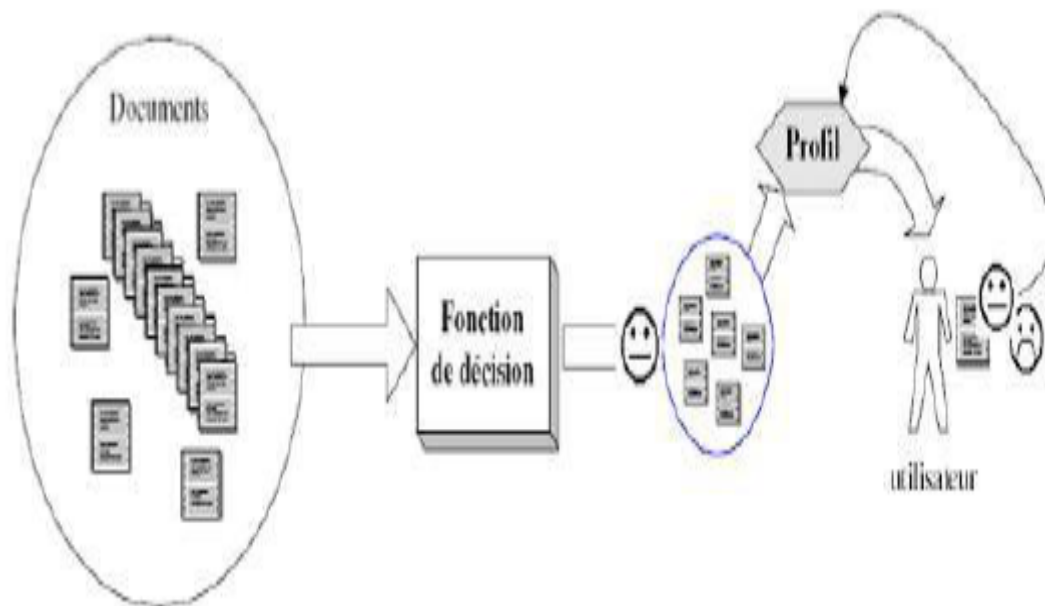


FIGURE 1.1 – Schéma général du filtrage d'information

1.4 Classification des systèmes de recommandation

Il existe plusieurs classifications des systèmes de recommandations qui sont résumées dans la figure 1.2 :

1.4.1 La classification classique :

cette classification de [9] est reconnue par trois types de filtrage ; un filtrage collaboratif(CF), un filtrage basé sur le contenu(CBF) et le filtrage hybride.

1.4.2 La classification de [Rao and Talwar, 2008] :

c'est une classification en fonction de la source d'information utilisée .

1.4.3 La Classification de [Su et al, 2009] :

elle est utilisée dans les systèmes de collaboration. Ils proposent une sous-classification qui comprend les techniques hybrides les classer dans les méthodes de collaboration hybrides. [Su et al, 2009] classent le filtrage collaboratif en trois catégories :

- Approches FC à base de mémoire : pour K-plus proches voisins.
- Approches FC basé sur un modèle englobant une variété de techniques telles que : clustering, les réseaux bayésiens, factorisation de matrices, les processus de décision de Markov.
- FC hybride qui combine une technique de recommandation CF avec un ou plusieurs autres méthodes.

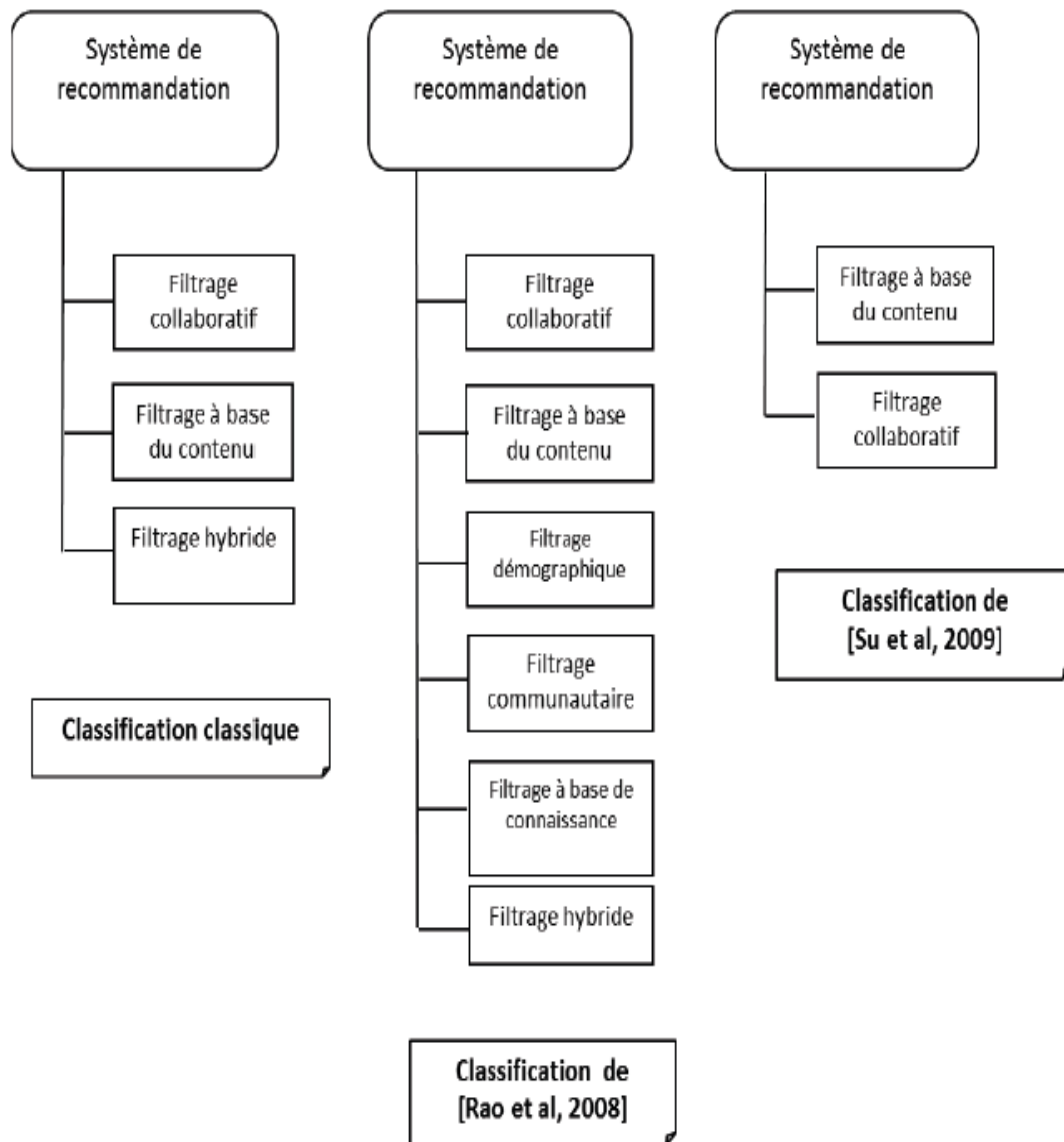


FIGURE 1.2 – Classification principale des systèmes de recommandations

1.5 Différents types de recommandation

1.5.1 Recommandation démographique :

C'est une recommandation simple qui propose des items par rapport au profil démographique d'utilisateur (figure 2.2). Elle comporte à partager les usagers en plusieurs classes ou groupes par rapport aux informations démographiques telles que le sexe, l'âge, la profession, la localisation, la langue, le pays, etc.

Le principe de cette approche est que deux utilisateurs ayant évolué dans un environnement similaire partagent des goûts communs que deux utilisateurs ayant évolué dans des

environnements différents et ne partageant donc pas les mêmes codes [10].

De nombreux sites utilisent cette solution simple à proposer une offre de contenu "personnalisé". Par exemple, les utilisateurs sont redirigés vers un site Web particulier en fonction de leur langue ou de pays. Ces approches ont été très populaires dans la littérature de marketing, mais ont reçu peu d'attention dans le domaine des algorithmes de recommandation.

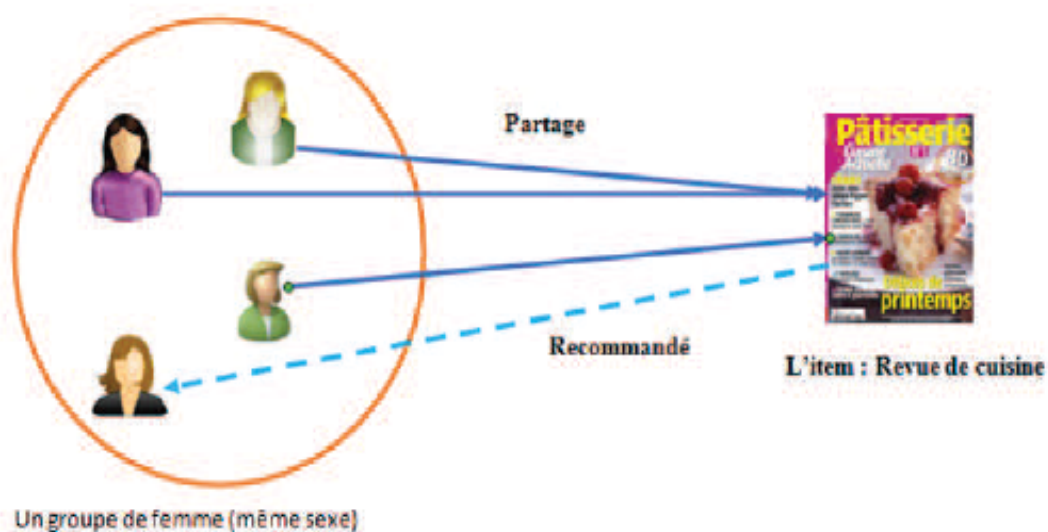


FIGURE 1.3 – Schéma général du filtrage d'information

1.5.2 Recommandations à base de connaissances

Les recommandations sont générées en utilisant des connaissances spécifiques dont certaines caractéristiques d'items répondent aux préférences de l'utilisateur (figure 2.3). Généralement les Systèmes à base de connaissances plus fiable par rapport à d'autres types de recommandation si les données limitées sont disponibles, à savoir, si le système ne peut pas compter sur l'existence d'un historique de l'utilisateur.

Il existe deux types de raisonnement :

- Le raisonnement à base de cas.
- Le raisonnement à base de contraintes.

Le raisonnement à base des cas

Le raisonnement à base des cas tire parti de la régularité du monde réel afin de résoudre des problèmes en recherchant la solution d'un cas semblable rencontré et résolu dans le passé [11].

Dans [11] les auteurs ont utilisé cette approche dans les systèmes de recommandation, ils estiment combien les besoins ou les préférences (description de problème) de l'utilisateur correspondent aux recommandations possibles (solutions du problème) en se basant sur le comportement de consommation précédente (cas précédents).

Le raisonnement à base de contraintes

Une recommandation à base de contraintes est un autre type de systèmes à base de connaissances. La recommandation à base de contraintes exploite des bases de connaissances prédéfinies qui contiennent des règles explicites sur la façon de relier les exigences des clients avec des fonctionnalités d'item. Par exemple, un utilisateur peut être intéressé à acheter des produits avec un certain ensemble de caractéristiques et dans une gamme de prix spécifique.

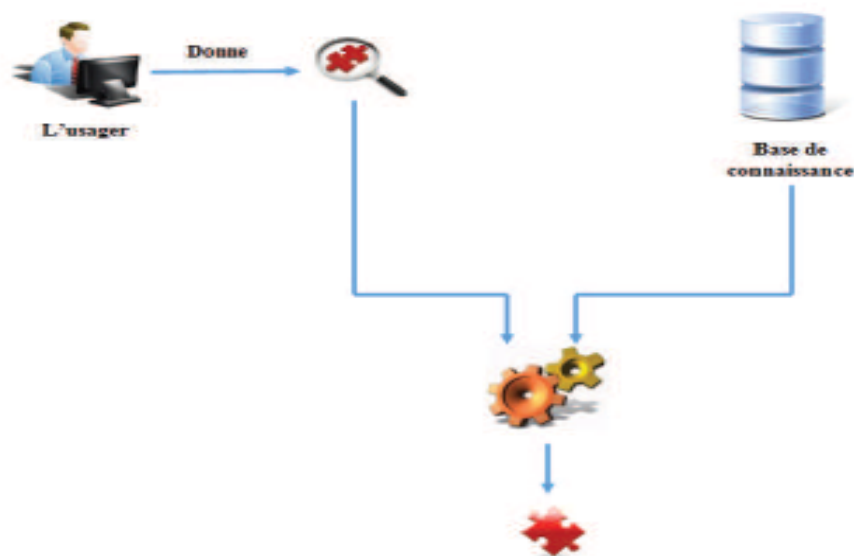


FIGURE 1.4 – Recommandations à base de connaissances

1.5.3 Recommandation communautaire (Social)

La plupart des réseaux sociaux (facebook, twitter, etc) s'appuient sur cette classification dans leurs recommandations (Figure 1.5).

L'idée de base est donc de dire que si des utilisateurs ont partagés des mêmes intérêts dans le passé, il y a de fortes chances qu'ils partagent aussi les mêmes goûts dans le futur. Le système propose des recommandations à partir des relations de l'usager avec ces amis dans le réseau social, et parfois cette recommandation dépend aussi de la valeur de confiance d'usager dans chacun de ses amis, l'exemple le plus connu de cette recommandation est la section des pages et des groupes qui apparait dans la partie droite d'une page facebook. L'importance décisionnelle du bouton « I Like » de Facebook a donné un succès croissant dont 55 des utilisateurs sont influencés par leurs amis.

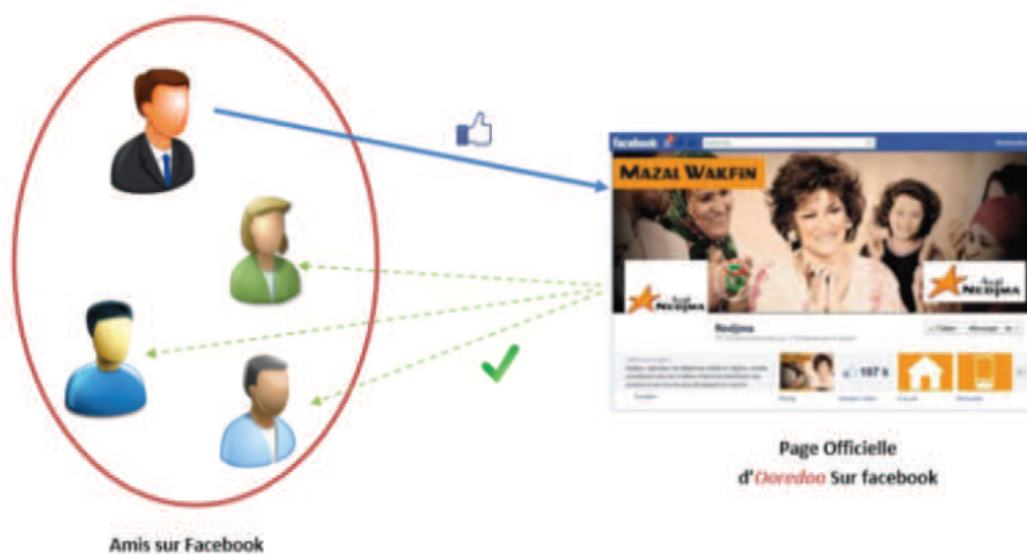


FIGURE 1.5 – Recommandations communautaire

1.5.4 Recommandation basé sur le contenu

La recommandation basée sur le contenu peut être considéré comme un système de recherche d'information exploitant le profil d'utilisateur (figure 2.5). Le profil utilisateur, composé de centres d'intérêts, sert à trouver des contenus présentant des méta-données en adéquation. Cette technique est fondée sur l'analyse des similarités de contenu entre les items précédemment consultés par les utilisateurs, ce système utilise également les informations de retour fournis par l'utilisateur (feedback) pour mettre à jour son profil. Ce dernier permet d'améliorer la qualité des recommandations au cours du temps. L'avantage des systèmes de filtrage cognitifs, basés contenu est qu'ils permettent d'associer

des documents à un profil utilisateur. Notamment, en utilisant des techniques d'indexation et d'intelligence artificielle. L'utilisateur est indépendant des autres ce qui lui permet d'avoir des recommandations même s'il est le seul utilisateur du système [12].

Afin de recommander par exemple des films à un utilisateur, le système analyse les corrélations entre ces films et les films consultés antérieurement par cet utilisateur. Ces corrélations sont évaluées en considérant des attributs comme le titre et le genre. De ce fait, parmi ces films, ceux qui seront recommandés à l'utilisateur, sont les plus similaires (En terme d'attribut) aux films consultés par cet utilisateur [13]. Cependant, ce type de systèmes présente certaines limitations.

- L'effet "entonnoir" : les besoins de l'utilisateur sont de plus en plus spécifiques, ce qui l'empêche d'avoir une diversité de sujets. Même pire, un nouvel axe de recherche dans un domaine bien précis peut ne pas être pris en compte car il ne fait pas parti du profil explicite de l'utilisateur.
- Filtrage basé sur le critère thématique uniquement, absence d'autres facteurs comme la qualité scientifique, le public visé, l'intérêt porté par l'utilisateur, etc.
- Les difficultés à recommander des documents multimédia (images, vidéos, etc.) et ceci à cause de la difficulté à indexer ce type de documents, c'est en fait la même problématique dont souffrent les systèmes de recherche.
- Problème de démarrage à froid : Un nouvel utilisateur du système éprouve des difficultés à exprimer son profil en spécifiant des thèmes qui l'intéressent. Ceci malgré les techniques d'apprentissage ou l'utilisateur fournit des textes exemples.

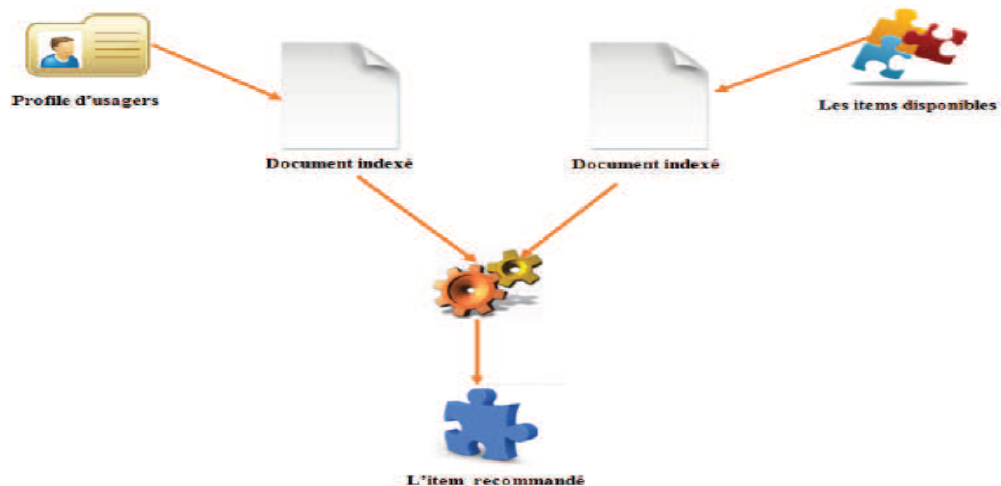


FIGURE 1.6 – Recommandation basé sur le contenu

On distingue deux types de recommandation basé sur le contenu : recommandation basé sur les mots clefs et recommandation basé sur la sémantique.

Recommandation basée sur les mots clefs

La technique de recommandation basée sur le contenu peut être appliquée à la recommandation de pages Web, de films, d'articles actualités, de restaurants, etc. Si nous prenons l'exemple d'un système de recommandation d'articles scientifiques basé sur le contenu, lorsqu'un utilisateur a tendance à consulter souvent des articles portant sur le domaine de la génétique, le système lui proposera des recommandations liées à la génétique. En effet, ces articles disposent de mots-clés communs tels que : "ADN", "gène" ou "protéine".

Il est à signaler que ces mots-clés sont généralement soit extraits sur la base d'une indexation automatique, soit attribués manuellement.

Pour ce qui est des systèmes de recommandation de films ou de restaurants, le contenu est plutôt structure et représente par des méta-données définies au préalable et valables pour tous les items [14].

La représentation par mots-clés à la fois pour les items et pour les profils peut donner des résultats précis. La plupart des systèmes basés sur le contenu sont conçus comme des classificateurs de textes construits à partir d'un ensemble de documents d'apprentissage qui sont soit des exemples positifs, soit des exemples négatifs des intérêts de l'utilisateur.

Mais le problème qui se pose avec cette approche est le « manque d'intelligence ».

Lorsque des caractéristiques plus complexes sont nécessaires, les approches à base de mots clefs montrent leurs limites. Si l'utilisateur, par exemple, aime « l'impressionnisme français », les approches à base de mots-clefs chercheront seulement des documents dans lesquels les mots « français » et « impressionnisme » apparaissent. Des documents concernant Claude Monet ou Renoir n'apparaîtront pas dans l'ensemble des recommandations, même s'ils sont susceptibles d'être pertinents pour l'utilisateur.

Recommandation basée sur la sémantique

La sémantique a été introduite par plusieurs méthodes dans le processus de recommandation.

Ces méthodes sont abordées en tenant compte de plusieurs critères :

- le type de source de connaissance impliquée (lexique, ontologie...).
- les techniques adoptées pour l'annotation ou la représentation d'items.
- le type de contenu inclus dans le profil utilisateur.
- la stratégie de correspondance entre items et profil.

1.5.5 Recommandation basé sur le filtrage collaboratif

Ce type de recommandation s'appuie sur les appréciations données par un ensemble d'utilisateurs sur un ensemble d'articles. Ces appréciations, traduites en valeurs numériques, peuvent être des notes, des comptes d'achats effectués, des nombres de visites, etc.

On distingue deux grandes approches de filtrage collaboratif :

- L'approche se référant aux utilisateurs.
- L'approche se référant aux articles.

L'approche se référant aux utilisateurs [15] consiste à comparer les utilisateurs entre eux et à retrouver ceux ayant des goûts en commun, les notes d'un utilisateur étant ensuite prédites selon son voisinage.

L'approche se référant aux articles [16] consiste à rapprocher les articles appréciés par les mêmes personnes et à prédire les notes des utilisateurs en fonction des articles les plus proches de ceux qu'ils ont déjà notés .

Dans un système du filtrage collaboratif, il faut que les utilisateurs fournissent des évaluations des items qu'ils ont déjà utilisés, sous forme des notes, pour constituer leurs profils. Il n'y a

aucune analyse du sujet ou du contenu des objets à recommander. Ce type de système est très efficace au cas où le contenu des objets sont complexes, il est compliqué ou impossible de l'analyser, l'utilisateur peut apercevoir divers domaines intéressants, car le principe du filtrage collaboratif ne se fonde absolument pas sur la dimension thématique des profils, et n'est pas soumis à l'effet « entonnoir ».

Parmi les avantages du filtrage collaboratif les jugements de valeur des utilisateurs intègrent non seulement la dimension thématique mais aussi d'autres facteurs relatifs à la qualité des items tels que la diversité, la nouveauté, l'adéquation du public visé, etc.

Un problème du système collaboratif est que sa performance dépend beaucoup de la distribution des évaluations (notes) données par utilisateurs. Dans le cas où il y a plusieurs items qui ont été utilisés et évalués par très peu d'utilisateurs, ces items seraient recommandés très rarement, même si ces utilisateurs ont donné des notes très hautes pour ces items. Ce problème est connu comme le problème de parcimonie (sparsity problem). De la même façon, si dans le système il existe des utilisateurs qui ont des goûts très différents en comparaison avec les autres, le système ne peut pas trouver des similarités entre utilisateurs et donc ne peut pas donner des bonnes recommandations.



FIGURE 1.7 – Recommandation basé sur le filtrage collaboratif

La figure 1.8 représente un tableau de place de touriste avec un axe des utilisateurs

d'un même Système et sur un autre les places de touriste. Chaque cellule de la matrice contient l'avis donné par un utilisateur pour une place de touriste, la cellule vide signifie qu'il n'a pas d'avis particulier sur cette place de touriste. Afin de prédire si Siham apprécierait la place "plage" et probablement lui recommander ce place, on compare les Votes de Siham à ceux des autres utilisateurs choisis. On peut alors voir que Siham et Mouna ont des Votes identiques, et que Mouna n'a pas aimé la place «plage», on pourrait alors prédire que Siham n'aimera pas aussi cette place et de ne lui pas faire cette suggestion.























	Hafidha	Mouna	Soraya	Siham
 plage				
 zoo				
 Casino				
 Musée d'art				
 plage				?

FIGURE 1.8 – Exemple de recommandation basé sur le filtrage collaboratif.

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

L'exploitation des données disponibles dans un système de filtrage peut se faire de plusieurs manières. Ces méthodes sont classées en deux familles principales :

les algorithmes basés mémoire est les algorithmes basés modèle.(voir la Figure 1.9)

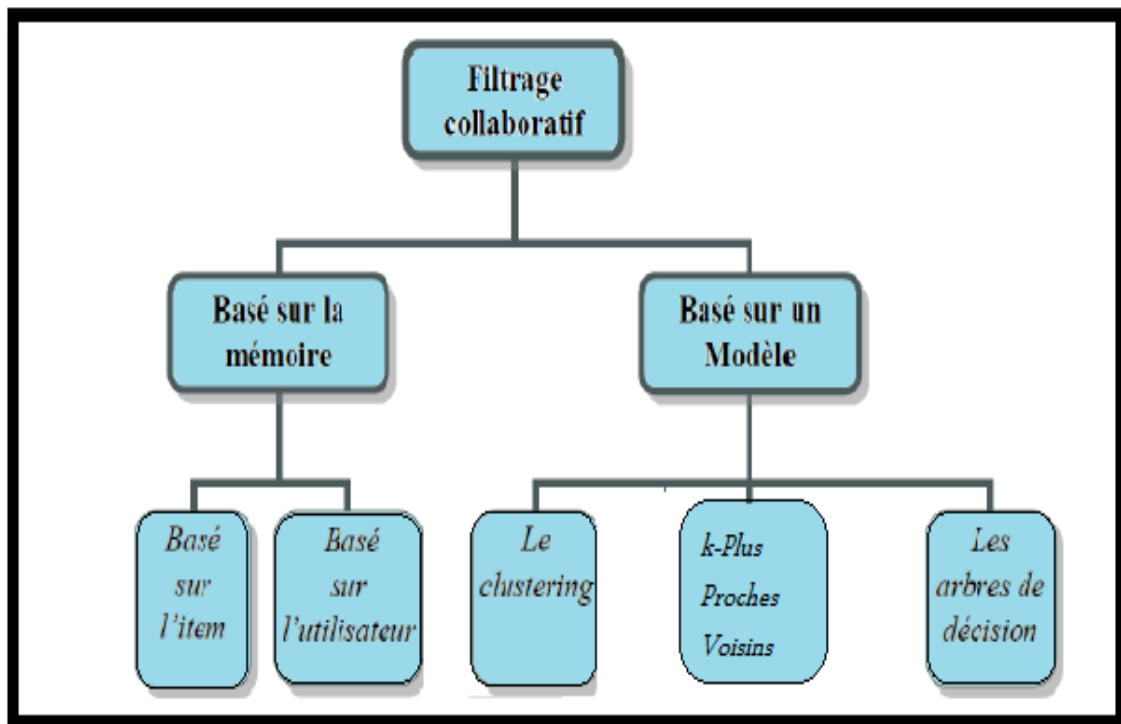


FIGURE 1.9 – Les méthodes de filtrage collaboratif

Filtrage collaboratif basé sur la mémoire

Le filtrage collaboratif basé sur la mémoire utilise une matrice des votes contenant des préférences des utilisateurs pour prédire des sujets additionnels ou des produits auxquels un nouvel utilisateur peut-être s'intéresse.

L'objectif d'un filtrage collaboratif basé sur la mémoire est de prédire l'utilité des ressources (items) pour un utilisateur particulier (l'utilisateur actif) basé sur la base des votes d'utilisateur.

Ainsi, dans le filtrage basé sur la mémoire, les notes des utilisateurs stockées par le système sont directement utilisées pour prédire les notes pour de nouveaux items. Cela peut être fait de deux manières connues sous le terme de recommandations basées sur les utilisateurs ou recommandations basées sur les items.

Filtrage collaboratif basé sur la mémoire (utilisateurs)

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

Filtrage collaboratif basé sur la mémoire (items)

Alors que les méthodes basées sur le voisinage utilisateur s'appuient sur l'avis d'utilisateurs partageant les mêmes idées pour prédire une note, les approches basées sur les items prédisent la note d'un utilisateur u pour un item i en se basant sur les notes de u pour des items similaires à i . Dans de telles approches, deux items sont similaires si plusieurs utilisateurs du système les ont notés d'une manière similaire. Les choix possibles pour calculer la similarité $Sim(i, j)$ entre les items i et j sont aussi la corrélation Pearson et la similarité vectorielle [17]. La similarité vectorielle se sert de l'estimation moyenne d'utilisateur de chaque paire évaluée, et fait face à la limitation de la similarité vectorielle. Empiriquement, la similarité entre deux items est calculée par la formule du Cosinus suivante :

$$Sim(i, j) = \sum_{A=1}^m \frac{(v_{A,i} - \bar{v}_A)}{\sqrt{\sum_{A=1}^m (v_{A,i} - \bar{v}_A)^2}} \times \frac{(v_{A,j} - \bar{v}_A)}{\sqrt{\sum_{A=1}^m (v_{A,j} - \bar{v}_A)^2}}$$

m : nombre d'utilisateurs qui ont votés pour les deux items.

$v_{A,i}$: vote de A pour l'item i .

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

\bar{v}_A : moyenne des votes de l'utilisateur A .

Une fois que la similarité parmi les items a été calculée, la prochaine étape est de

prévoir pour l'utilisateur cible A, une valeur pour l'item actif i. Une manière commune est de capturer comment l'utilisateur a évalué les items similaires[17]. La valeur prévue est basée sur la somme pondérée des estimations de l'utilisateur ainsi que les déviations des estimations moyennes et peut être calculée à l'aide de la formule suivante :

$$P_{A,i} = \bar{v}_i + \frac{\sum_{j=1}^m Sim(i,j) \times (v_{A,j} - \bar{v}_j)}{\sum_{j=1}^m |Sim(i,j)|}$$

m : nombre d'items présents dans le voisinage de item i, ayant déjà été voté par l'utilisateur A.

$v_{A,j}$: Vote de l'utilisateur A pour l'objet j.

\bar{v}_j : Moyenne des votes pour l'item j. —Sim(i,j)— : Similarité moyenne.

Ces prédictions seront comparées par la suite avec les valeurs réelles omises en utilisant deux mesures : MAE (Mean Absolute Error) et RMSE (Root Mean Squared Error) qui sont des mesures de qualité de la prédiction très utilisées dans ce domaine.

Filtrage collaboratif basé sur un modèle

Le deuxième type d'algorithmes, est comme le nom l'indique basés sur des modèles, ces méthodes ont été intégrées aux systèmes de recommandation pour améliorer et remédier aux problèmes des méthodes basées sur la mémoire. Les algorithmes basés sur le modèle se basent aussi sur les évaluations précédentes des utilisateurs, mais cette méthode ne calcule pas directement les prédictions, elle classe les utilisateurs suivant des groupes ou d'apprendre les modèles à partir de leurs données. Généralement, les méthodes basées sur le modèle utilisent les techniques d'apprentissage automatique pour la construction du modèle plusieurs.

Parmi ces modèles : Modèle de Clustering, K Plus proche voisin, l'arbre de décision[17].

Modèle de Clustering

Les méthodes de Clustering permettent de limiter le nombre d'individus considérés dans le calcul de la prédiction. Le temps de traitement sera donc plus court et les résultats

seront potentiellement plus pertinents puisque les observations porteront sur un groupe le plus proche de l'utilisateur actif. Autrement dit, au lieu de consulter l'ensemble de la population, nous estimons la préférence d'un groupe de personnes ayant les mêmes goûts que l'utilisateur.

K Plus Proche Voisin

C'est un des algorithmes d'apprentissage artificiel le plus simple, qui dispose une base de données d'apprentissage de m couple (entrée- sortie), son principe général est de normaliser les données et les récupérer (Growing).

Dans un contexte de classification d'une nouvelle observation x , l'idée fondatrice simple est de faire voter les plus proches voisins de cette observation. La classe de x est déterminée en fonction de la classe majoritaire parmi les k plus proches voisins de l'observation x .

Elle est donc une méthode à base de voisinage, non paramétrique ; où une nouvelle observation est classée dans la classe d'appartenance de l'observation de l'échantillon d'apprentissage qui lui est la plus proche, au regard des co-variables utilisées. La détermination de leur similarité est basée sur des mesures de distance.

Arbre de décision

Les arbres de décision sont des classificateurs en l'attribut concerné (ou classe) dans une forme d'un arbre. Les items à classer sont composés des attributs et leurs valeurs ciblent. Les noeuds d'un arbre peuvent être :

- Noeuds de décision : dans ces noeuds un attribut-valeur unique est testé pour déterminer à quelle branche du sous arbre applique.
- Noeuds de feuilles : ceux qui indiquent la valeur de l'attribut ciblent. Il y existe plusieurs algorithmes de l'arbre de décision parmi eux : Hunt's Algorithm, CART.

1.5.6 Filtrage hybride

Constatant les avantages et inconvénients de chacune des deux approches ci-dessus, on comprend que de nombreux systèmes reposent sur leur combinaison, ce qui en fait des systèmes de filtrage dits « hybrides ». En général, l'hybridation s'effectue en deux phases : (i) appliquer séparément le filtrage collaboratif et autres techniques de filtrage

pour générer des recommandations candidates, et (ii) combiner ces ensembles de recommandations préliminaires selon certaines méthodes telles que la pondération, la mixtion, la cascade, la commutation, etc., afin de produire les recommandations finales pour les utilisateurs [18].

Plus généralement, les systèmes hybrides gèrent des profils d'utilisateurs orientés contenu, et la comparaison entre ces profils donne lieu à la formation de communautés d'utilisateurs permettant le filtrage collaboratif. La meilleure description des méthodes hybrides a été faite par [Burke, 2002]. Alors, selon Burke on peut distinguer sept façons de combiner les méthodes traditionnelles : Pondération, Commutation, Technique mixte, Combinaison de caractéristiques, Cascade, Méta niveau.

Pondération (Weighted)

Une méthode hybride qui combine la sortie d'approches distinctes, utilisant, par exemple, une combinaison linéaire des scores de chaque technique de recommandation.

Commutation (Switching)

C'est une technique qui permet de faire le choix d'un modèle de recommandation parmi plusieurs, en se basant sur plusieurs critères. La détermination de la technique appropriée dépend de la situation. Le système se doit alors de définir les critères de commutation, ou les cas où l'utilisation d'une autre technique est recommandée. Ceci permet au système de connaître les points forts et les points faibles des techniques de recommandation qui le constituent.

Technique mixte (Mixed)

Dans cette approche, le recommandeur ne combine pas, mais augmente la description des ensembles de données, en prenant en considération les estimations des utilisateurs et la description des items. La nouvelle fonction de prédiction doit faire face aux deux types de descriptions et permet d'éviter les problèmes posés par le filtrage collaboratif, à savoir, le démarrage à froid.

Combinaison de caractéristiques (Features combination)

Dans un hybride basé sur la combinaison de caractéristiques, les données provenant de techniques collaboratives sont traitées comme une caractéristique, et une approche basée sur le contenu est utilisée sur ces données.

Cascade

La cascade implique un processus étape par étape. Dans ce cas, une technique de recommandation est appliquée en premier, produisant un ensemble de candidats potentiels. Puis, une deuxième technique raffine les résultats obtenus dans la première étape. Cette méthode a pour avantage que si la première technique génère peu de recommandations, ou si ces recommandations sont ordonnées afin de permettre une sélection rapide, la deuxième technique ne sera plus utilisée.

Augmentation de caractéristiques (Feature augmentation)

L'augmentation de caractéristiques est semblable à la cascade, mais dans ce cas-là les résultats obtenus (le classement ou la classification) de la première technique sont utilisés par le deuxième comme une caractéristique ajoutée.

Méta niveau (Meta-level)

Dans un hybride basé sur méta niveau, une première technique est utilisée, mais différemment que la précédente méthode (augmentation de caractéristiques), non pas pour produire de nouvelles caractéristiques, mais pour produire un modèle. Et dans la deuxième étape, c'est le modèle entier qui servira d'entrée pour la deuxième technique [19].

1.6 Les objectifs de la recommandation :

Généralement, les systèmes de recommandation possèdent plusieurs objectifs pour divers types de recommandation [20] telle que :

- Réduire l'effort de l'utilisateur dans la recherche d'items qui l'intéresseraient et lui proposer des items auxquels il n'aurait pas pensé ;
- Promouvoir les items impopulaires ;
- Assister l'utilisateur dans le choix d'items ;
- Recommander des items en temps réel.

1.7 Avantages et inconvénients des systèmes de recommandation

Le tableau 1.1 résume les forces et faiblesses des méthodes traditionnelles utilisées par les systèmes de recommandation, en l'occurrence le Filtrage Collaboratif (FC), le Filtrage Démographique (FD), le Filtrage à Base de Contenu (FBC), et le Filtrage à base de données communautaires.

Adaptabilité : Au fur et à mesure que la base de données des évaluations augmente, la recommandation devient plus précise.

Nouvel utilisateur : un nouvel utilisateur qui n'a pas encore accumulé suffisamment d'évaluations ne peut pas avoir de recommandations pertinentes.

Nouvel item : un item doit avoir suffisamment d'évaluations pour qu'il soit pris en considération dans le processus de recommandation.

Démarrage à froid : le démarrage à froid est un problème pour les nouveaux utilisateurs qui commencent à jouer avec le système, parce que le système ne dispose pas d'assez d'informations à leur sujet. Si le profil d'utilisateur est vide, il doit consacrer une somme d'efforts à l'aide du système avant d'obtenir une récompense (les recommandations utiles). D'autre part, quand un nouvel item est ajouté à la collection, le système doit avoir suffisamment d'informations pour être en mesure de recommander cet item aux utilisateurs.

Le tableau suivant présente les avantages et les inconvénients des systèmes de recommandation

Techniques	Avantages	Inconvénients
Filtrage démographique	N'exige aucun historique d'estimations..	<ul style="list-style-type: none"> • Problème de confidentialité • Utilisateur avec un gout unique • Nouvel Item
Filtrage a base de données communautaire	Adaptabilité : la qualité croit avec le nombre d'amis	<ul style="list-style-type: none"> • Nouvel utilisateur • Nouvel item
Filtrage a base du contenu	<ul style="list-style-type: none"> • Pas besoin d'une large communauté d'utilisateurs pour pouvoir effectuer des recommandations. • Une liste de recommandations peut être générée même s'il n'y a qu'un seul utilisateur. • La qualité croit avec le temps. • Pas besoin d'information sur les autres utilisateurs. • Prendre en considération les gouts uniques des utilisateurs. 	<ul style="list-style-type: none"> • L'analyse du contenu est nécessaire pour faire une recommandation. • Problème de recommandation des images et de vidéos en absence de Métadonnées. • Nécessité du profil d'utilisateur.
Filtrage collaboratif	<ul style="list-style-type: none"> • Ne demande aucune connaissance sur le contenu de l'item ni sa sémantique. • La qualité de la recommandation peut être évaluée. • Plus les nombre d'utilisateurs est grand plus la recommandation est meilleure. 	<ul style="list-style-type: none"> • Démarrage a froid. • Nouvel utilisateur. • Problème de confidentialité. • La complexité : dans les systèmes avec un grand nombre d'items et d'utilisateurs, le calcul croit linéairement. • Nouvel Item.

TABLE 1.1 – Les avantages et les inconvénients des techniques de recommandations.

1.8 Systèmes de recommandation sensibles au contexte

Systèmes de recommandation sensibles au contexte L'importance des informations contextuelles a été reconnue par les chercheurs dans plusieurs domaines, y compris la recherche d'information, l'informatique ubiquitaire, le marketing et management, etc. Cependant, les travaux de recherche sur les systèmes de recommandation n'ont pas exploité les informations contextuelles. Des informations telles que le temps, la localisation, la compagnie d'autres personnes peuvent pour tant améliorer le processus de recommandation dans certains domaines. Les systèmes de recommandation traditionnels traitent seulement de deux types d'entités, les utilisateurs et les items. Cependant, pour de nombreuses applications, par exemple les systèmes de recommandation dédiés au tourisme, il peut ne pas être suffisant de ne considérer que les utilisateurs et les items. Il est souvent important d'intégrer des informations sur le contexte. Par exemple, un système de recommandation des é jours de vacances doit tenir compte de la saison pour fournir une recommandation adaptée. De même, un système de recommandation pour le tourisme implémenté sur dispositif mobile peut privilégier la recommandation de lieux d'activités proches de la position de l'utilisateur[21].

1.9 L'apprentissage automatique :

L'apprentissage automatique (en anglais : Machine Learning) est une science qui consiste à développer des algorithmes d'apprentissage, qui apprennent à résoudre une tâche [22]. Plus précisément l'apprentissage automatique correspond au domaine se consacrant au développement d'algorithmes permettant à une machine d'apprendre à partir d'un ensemble des données, c-à-d., d'y extraire des concepts et patrons caractérisant ces données [23].

1.9.1 Les algorithmes d'apprentissage automatique :

Un algorithme est une suite d'opérations permettant d'obtenir un résultat ou résoudre des problèmes. Les algorithmes sont au coeur de nombreuses disciplines. Un domaine nous intéresse en particulier en raison de son développement significatif au cours de ces dernières années : l'apprentissage machine (machine learning) [24].

Les algorithmes d'apprentissage automatique permettant aux ordinateurs de s'entraîner

à partir d'exemples réels sur les entrées de données et utilisent l'analyse statistique pour produire des valeurs qui se situent dans une plage spécifique [23], ses algorithmes comme l'algorithme k-plus proche voisin, les réseaux de neurones, l'arbre de décision [24], l'algorithme de séparation, . . . , cherchent à découvrir la structure d'un ensemble de données à partir de valeurs observées. Ils sont utilisés dans de nombreux contextes comme la classification et la prévision et sont au coeur des algorithmes d'aide à la décision [23], nous allons les présenter dans les prochaines lignes.

Nous pouvons aussi définir un algorithme d'apprentissage comme suit ; « un algorithme d'apprentissage est un algorithme prenant en entrée un ensemble de données D et retournant une fonction f . Nous désignons alors D comme ensemble d'entraînement ou ensemble d'apprentissage, et la fonction f comme modèle. Suite à l'exécution d'un algorithme d'apprentissage, nous disons que le modèle f a été entraîné sur l'ensemble D . nous considérons que l'ensemble de données D contient sous forme de vecteurs l'information nécessaire pour résoudre un problème donné. Le modèle f doit ainsi pouvoir recevoir de tels vecteurs en argument et nous désignerons par sortie la valeur de l'application de f sur un vecteur. De plus, tous les éléments de D , que le nous nommons exemples d'apprentissage, sont indépendants et identiquement distribués »[23].

L'apprentissage automatique facilite l'utilisation des ordinateurs dans la construction de modèles à partir de données d'échantillonnage afin d'automatiser les processus de prise de décision en fonction des données saisies.

1.9.2 Types d'apprentissage automatique :

Les principaux types d'apprentissage sont distingués essentiellement par leur objectif, et peuvent être combinés dans un même système [23].

Apprentissage supervisé :

L'apprentissage supervisé correspond au cas où l'objectif de l'apprentissage est déterminé explicitement via la définition d'une cible à prédire. Dans ce cas, D correspond à un ensemble de n paires d'entrées

x_t et de cibles associées y_t :

$$D = [(x_t, y_t) | x_t \in X, y_t \in Y]$$

Typiquement, un tel ensemble est récolté en fournissant l'ensemble des entrées à un groupe de personnes et en leur demandant d'associer à chacune de ces entrées une cible appropriée dans le contexte du problème à résoudre. La tâche d'un algorithme d'apprentissage est alors d'entraîner un modèle qui puisse imiter ce processus d'étiquetage par un humain, c-à-d., qui puisse prédire pour une entrée x quelconque la valeur de la cible y qui aurait normalement été donnée par un humain. Cependant, les algorithmes d'apprentissage ne se limitent pas à la modélisation du comportement de l'humain, et peuvent être utilisés pour modéliser la relation liant des paires d'entrées et de cibles provenant d'un autre phénomène [23].

Quelques problèmes résolus par ce type d'apprentissage [25] sont :

- Détecteur de spam.
- Risque de crédit.
- Prédiction des pics d'ozone.
- Aide au diagnostic médical.
- Aide au pilotage.
- Apprentissage sur les graphes.

Ses problèmes d'apprentissage supervisé peuvent être regroupés en deux problèmes :

- **Classification** : Un problème de classification survient lorsque la variable de sortie est une catégorie, telle que «rouge», «bleu» ou «maladie» et «pas de maladie».
- **Régression** : Un problème de régression se pose lorsque la variable de sortie est une valeur réelle, telle que «dollars» ou «poids».

Apprentissage non-supervisé :

Un apprentissage sur des exemples non connus contrairement à l'apprentissage supervisé. Ne repose que sur des données sans étiquettes $(x_i)_{1, \dots, n} \in X_n$. La sortie n'est pas connue a priori [26]. « Les algorithmes d'apprentissage supervisé expérimentent un ensemble de données et apprennent des propriétés utiles de la structure de ces données. Ils modélisent fondamentalement la distribution de probabilité d'une propriété d'un ensemble de données. Dans l'apprentissage non supervisé, aucun libellé ou objectif n'est fourni, ce qui en fait une tâche beaucoup plus difficile que l'apprentissage supervisé. L'algorithme doit donner un sens aux données sans aucune supervision. Exemples d'algorithmes dans L'apprentissage non supervisé est l'analyse en composantes principales et la classification

en k-moyennes » [27].

Parmi les problèmes résolus par l'apprentissage non-supervisé les plus répandus nous avons :

- **Estimation de densité** : pour ce problème une fonction doit fournir une estimation de la fonction de densité ou de probabilité de la distribution ayant généré les éléments, cette estimation est faite avec les méthodes paramétriques par exemple, la méthode de l'estimation de vraisemblance ou bien la méthode de moindre carré, ..., ou avec des méthodes d'estimations non paramétriques pour les fonctions de densité inconnue [24].
- **Extraction de caractéristique** : apprendre une nouvelle représentation de l'entrée qui soit plus utile que la représentation vectorielle originale. L'utilité de cette représentation dépend de la tâche que le nous souhaitons réellement résoudre, et la qualité peut être mesurée de différentes façons, en fonction du contexte de l'application. Parmi les approches de l'extraction de caractéristiques, la technique la plus simple qui est la sélection de caractéristique [23].
- **Réduction de dimensionnalité** : c'est la solution optimale pour résoudre les problèmes d'apprentissage automatique par la sélection et extraction des attributs pertinents. Son principe est de réduire l'espace de stockage nécessaire et le temps d'apprentissage, afin d'améliorer la classification . Plus généralement c'est de passer d'un espace à D dimension vers un espace à K dimension où $K < D$ [23].

Les raisons pour réduire la dimensionnalité sont [23] :

- Malédiction de la dimensionnalité : l'ajout d'une dimension augmente exponentiellement l'espace mathématique.
- Simplicité de modèle diminuer les variances.
- Extraction de connaissances.
- Visualiser les données : analyse de résultats.
- **Regroupement (Clustering)** : un outil de visualisation d'un ensemble de données, peut aussi être utilisé afin de compresser un ensemble de données en le remplaçant par l'ensemble des prototypes. Il est même possible d'extraire des caractéristiques

à l'aide du regroupement. Le plus simple algorithme de regroupement est l'algorithme des K moyennes (K-means algorithm). Il s'agit d'un algorithme itératif, qui débute avec une initialisation aléatoire de la position des prototypes [Larochelle, 2008].

Voici quelques exemples populaires d'algorithmes d'apprentissage automatique supervisé et non-supervisé :(voir la Figure 1.10)

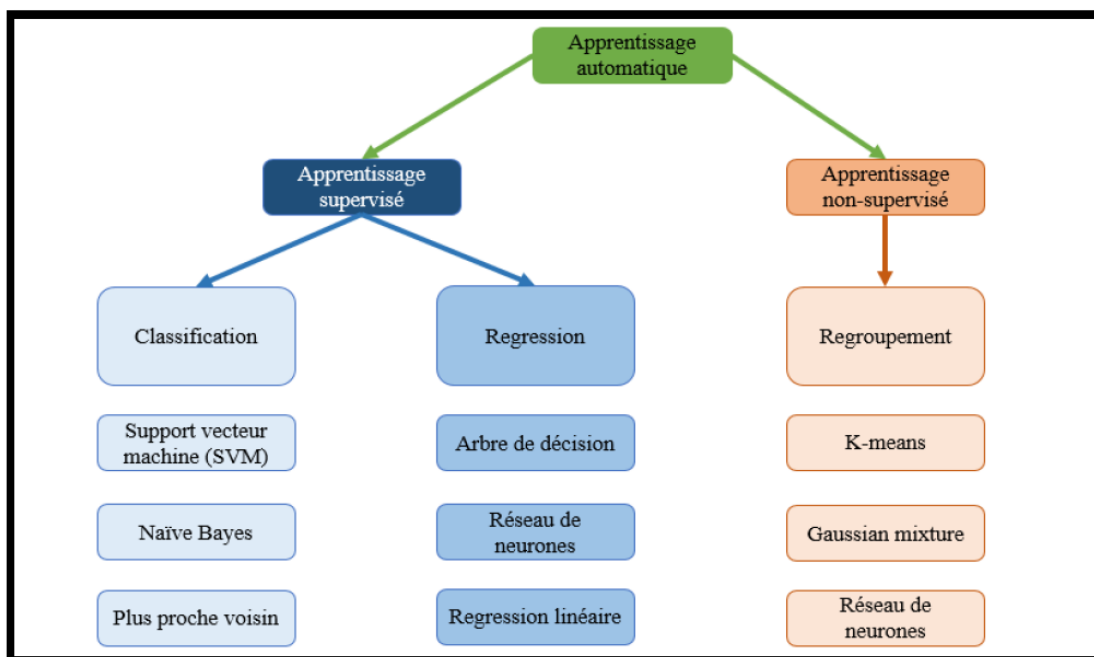


FIGURE 1.10 – Les algorithmes d'apprentissage supervisé et non-supervisé.

1.10 Conclusion

Dans ce chapitre, nous avons présenté les systèmes de recommandation qui sont devenus omniprésents ces dernières années dans de nombreux domaines. Ces systèmes sont conçus pour aider les utilisateurs à trouver des ressources qui les intéressent et qui sont adaptées à leurs préférences, parmi le nombre important des choix qui s'offrent à eux.

Nous allons détailler les approches les plus utilisées. Nous avons également passé en revue les différents algorithmes d'apprentissage automatique.

Dans Notre approche on s'intéresse au filtrage collaboratif en utilisant la classification supervisé.

Dans le prochain chapitre, nous présentons le contexte et les objets connecté.

Contexte, Internet des objets

2.1 Introduction

De nos jours, on entend de plus en plus parler sur le mot contexte. En 1997, seulement 5% des pages web contiennent le mot contexte. En 2006, il y en avait 15%. Bien que ce mot ne soit pas toujours bien employé. Aujourd’hui, le nombre de pages web contenant ce mot a une croissance exponentielle [28]. La notion de contexte est assez difficile à définir. En effet, tous les jours, les gens utilisent le contexte dans leurs prises de décision. La mise en évidence du contexte est particulièrement visible quand il y a plusieurs méthodes pour accomplir une tâche. Dans ce cas, chaque personne choisit sa méthode en fonction de ses connaissances et des informations contextuelles qu’il possède [29].

Dans ce 2^{ème} chapitre d’état de l’art, nous allons introduire par la notion de contexte, la Classification des informations de contexte et ses Catégories.

Nous allons finir ce chapitre en présentant l’internet des objets, ses domaines d’applications et on terminera en citant des exemples sur l’internet des objets.

2.2 Contexte

2.2.1 La notion de contexte

Le contexte n’est pas un concept nouveau en informatique. Cette notion est utilisée dès les premiers systèmes d’exploitation, où un contexte décrit l’ensemble des informations permettant de passer d’un processus d’exécution à un autre et de gérer les interruptions.

Le contexte sauvegardé doit au minimum inclure une portion notable de l'état du processeur (registres généraux, registres d'état, etc.) ainsi que les données nécessaires au système d'exploitation pour gérer un processus. La notion de contexte est aussi utilisée en linguistique et en traitement automatique des langues, où le contexte linguistique désigne tout trait linguistique présent autour d'un mot. En intelligence artificielle, la notion de contexte est principalement présente dans deux axes de recherches : la représentation de connaissances et la logique [30].

Il existe plusieurs définitions sur le mot contexte fourni par les dictionnaires, parmi ces définitions : Petit Robert "ensemble du texte qui entoure un élément de la langue (mot, phrase, fragment d'énoncé) et dont dépend son sens, sa valeur" [31].

Encyclopédie Larousse "ensemble de circonstances dans lesquelles se produit un évènement, se situe une action" [31].

Grand Dictionnaire : Office québécois de la langue française "le contexte est un ensemble d'informations concernant l'action du stylet, en rapport principalement avec sa localisation à l'écran, qui permet au système d'exploitation de l'ordinateur à stylet de différencier les commandes et l'entrée des données, et de fonctionner en conséquence"[31].

Avant de commencer la conception d'une application sensible au contexte, une séparation entre les données contextuelles et celles de l'application est très importante. Une donnée définie comme contextuelle dans un domaine peut être une donnée de l'application dans un autre domaine[32].

Dans le domaine informatique Le contexte selon schilit [33] est un ensemble d'informations de localisation, d'identité des personnes et des objets à proximité ainsi que les modifications pouvant intervenir sur ces objets. L'étude de contexte se fait par réponse aux questions : "On est où?", "Avec qui?", "quelles sont les ressources qu'on utilise?". Il est défini donc comme les changements de l'environnement physique de l'utilisateur et des ressources de calcul. Brown [34] restreint le contexte aux éléments de l'environnement de l'utilisateur, puis il introduit l'heure, la saison, la température, l'identité et la localisation de l'utilisateur. Ryan [35] assimile le contexte à l'environnement, l'identité et la localisation de l'utilisateur ainsi que le temps.

En 1998, Pascoe [36] introduit un élément important : l'intérêt. En effet, il définit le contexte comme un sous-ensemble d'états physiques et conceptuels qui ont un certain intérêt pour une entité donnée. Cette notion d'intérêt a été reprise par Dey et al [37] dans

leur définition : ” le contexte couvre toutes les informations pouvant être utilisées pour caractériser la situation d’une entité. Une entité est une personne, un lieu, ou un objet qui peut être pertinent pour l’interaction entre l’utilisateur et l’application, y compris l’utilisateur et l’application”.

Cette définition encapsule toutes les autres définitions précédentes puisqu’elle est d’ordre très générique. Dey [38] explique cette généralité du fait que les paramètres du contexte peuvent être fournis par l’utilisateur ou par des capteurs situés dans l’environnement de l’utilisateur et de l’application comme ils peuvent parvenir d’une interprétation plus ou moins complexe de ces paramètres, et a veillé à ce que sa définition englobe toute donnée implicite ou explicite qui peut être utile à l’application.

En 2000, Chen et Kots [39] citent que le contexte est divisé en trois catégories principales : contexte informatique, contexte physique et contexte utilisateur, le contexte informatique contient toutes les informations relatives aux ressources matérielles qui permettent l’exécution de l’application, le contexte utilisateur contient des informations relatives à l’identité de l’utilisateur, ses préférences, ses relations sociales, etc. Finalement le contexte physique caractérise l’environnement physique dans lequel se trouvent les personnes et les applications. Cette dernière catégorie regroupe également les données temporelles.

En 2001, Winograd [40] approuve la définition donnée par Dey et affirme qu’elle couvre tous les travaux existants sur le contexte, mais apporte plus de précision. Il propose ”le contexte est un ensemble d’informations qui est structuré et partagé et peut évoluer dans le temps”. Puis il donne une définition détaillée par rapport à la première ”la considération d’une information comme contexte est due à la manière dont elle est utilisée et non à ses propriétés inhérentes”. En 2002, Henricksen [12] définit le contexte comme ”la circonstance ou la situation dans laquelle une tâche informatique se déroule”.

Une analyse est faite en 2004 par Brezillon [41] et al concernant les définitions du terme contexte les a conduits à conclure que la plupart des définitions sont des réponses aux questions suivantes :

- Qui : identité de l’utilisateur courant et d’autres personnes présentes dans l’environnement.
- Quoi : percevoir et interpréter l’activité de l’usager.
- Où : localisation de l’usager, ou d’un évènement du système.

- Quand : repère temporel d'une activité, indexation temporelle d'un évènement, temps écoulé de la présence d'un sujet à un point donné.
- Pourquoi : il s'agit de comprendre la raison d'être de l'activité.
- Comment : la manière de déroulement de l'activité.

Les réponses aux questions citées ci-dessus peuvent engendrer un grand ensemble d'informations dont une grande partie est inutile. Cela requiert également un plus grand effort pour la gestion de ces informations.

En 2006 Chaari [42] définit le contexte comme : " l'ensemble des paramètres externes à l'application (localisation, temps, environnement, terminal utilisé, profil utilisateur...) pouvant influencer sur le comportement d'une application en définissant de nouvelles vues sur ses données et ses services. Ces paramètres ont un aspect dynamique qui leur permet d'évoluer durant le temps d'exécution. Une nouvelle instance de ces paramètres caractérise une nouvelle situation contextuelle qui ne modifie pas les données de l'application, mais qui peut mener à les traiter d'une façon différente.

En effet, toutes les définitions existantes sont soit très abstraites, ce qui rend la formalisation du contexte très difficile, soit très spécifiques à un domaine particulier. Par ailleurs, la définition la plus complète et la plus adoptée par les chercheurs et celle de Dey.

2.2.2 Classification des informations de contexte

L'information contextuelle peut être classifiée dans différentes classes. Plusieurs auteurs ont proposé des catégorisations selon différentes approches. voir Le tableau 2.1

2.2.3 Catégories du contexte

Etant donné la diversité des informations composant le contexte, il est utile d'essayer de les classer par catégorie pour faciliter leur utilisation. Dans cette section, nous présentons une classification qui synthétise les informations contextuelles utilisées dans les solutions existantes. Les entités principales concernées par la notion de contexte sont les lieux, les personnes ou les objets. Les lieux sont des régions d'espaces géographiques comme des chambres, des bureaux, des bâtiments, des rues ou des zones bien définies. Les personnes peuvent être des individus ou des groupes d'individus rassemblés ou répartis.

Auteur	Les classes	
Schilit et al [6] et Dey	le contexte primaire	primaire qui contient les informations sur la localisation, l'identité, le temps et l'activité (statut)
	le contexte secondaire	qui peut être déduit de ce dernier (exemple : de la localisation, on peut déduire les personnes à proximité)
Chen et Kotz [28]	le contexte actif	qui influence le comportement d'une application
	le contexte passif	qui est nécessaire, mais pas critique pour l'application
Petrelli et al [24]	le contexte matériel	localisation, machine, plateforme existante
	le contexte social	les aspects sociaux comme la relation les individus
Hofer et al [26]	le contexte physique	qui peut être mesuré par les capteurs physiques
	le contexte logique	qui contient les informations sur l'interaction (l'état émotionnel de l'utilisateur,

TABLE 2.1 – Classification des informations de contexte

Les objets peuvent être des entités physiques, des composants logiciels ou des artefacts (applications, fichiers, ressources...)[43].

les informations contextuelles utilisées dans la majorité des travaux existants sont classées en cinq catégories principales : entité, localisation, environnement (état), temps, relations (Figure 2.1).

Entité

L'un des éléments les plus importants du contexte. C'est un élément qui regroupe l'ensemble de tout ce qui peut être observé sur un être humain ou un objet physique ou virtuel. Une entité de contexte peut être soit individuelle ou un groupe qui partage un ensemble de caractéristiques et d'aspects de contexte communs. Une entité peut être

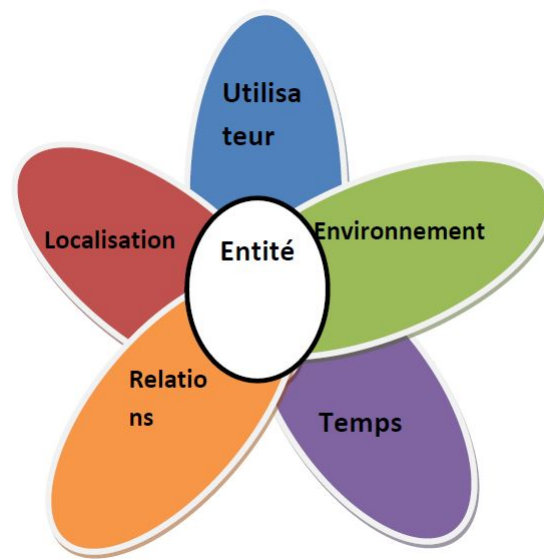


FIGURE 2.1 – Les différents éléments du contexte

divisée en quatre sous entités dans [43] :

- Naturelle : représente les caractéristiques des choses naturelles vivantes et non vivantes, et qui ne sont pas une fabrication de l'être humain. °Artificielle : représente les produits ou les phénomènes qui résultent des activités.
- Groupe d'entités : Est une collection d'entités, qui partagent certaines caractéristiques, ou ayant établi certaines relations entre elles.
- Humaine : regroupe les informations sur l'être humain qui généralement l'utilisateur du système. Il est généralement décrit par un ensemble d'éléments comme ses préférences, ses centres d'intérêt.

Utilisateur Un utilisateur est un élément central dans un système sensible au contexte, le représenter reste une tâche très difficile à cause de sa nature dynamique. Un utilisateur est représenté par un profil utilisateur.

Profil utilisateur

- *Définition1* : par profil utilisateur, on désigne habituellement l'ensemble des informations permettant de personnaliser le fonctionnement du système de façon à l'adapter à un utilisateur spécifique, soit pour fournir une meilleure qualité de services, soit pour améliorer la productivité de cet utilisateur. Pour ne pas obliger

l'utilisateur à effectuer un effort d'adaptation permanent, cette personnalisation doit être persistante et ses effets doivent se reproduire de façon quasi identique à chaque utilisation du système par un même utilisateur (même si une évolution lente est acceptable voire souhaitable). En conséquence, le profil est généralement construit autour de données ayant une durée de validité relativement longue [44].

- *Définition2* : le profil peut contenir des informations brutes, caractéristiques de l'utilisateur (nom, prénom, pseudonyme, adresse, historique,...). Généralement, ce sont des informations qui viennent directement de l'utilisateur, par un procédé de saisie. Outre les informations brutes, plusieurs techniques peuvent être adoptées pour la construction des profils des utilisateurs [45].

a. Modélisation du profil utilisateur La modélisation du profil repose sur des techniques et des outils permettant non seulement de représenter et construire le profil de l'utilisateur mais aussi de gérer son évolution de manière dynamique au cours du temps. Le modèle du profil consiste à spécifier sous quelle forme les données du profil doivent être représentées. Les modèles de représentations peuvent être simples basés sur des mots clés ou complexes par exemple basés sur des ontologies de domaines, des hiérarchies de concepts, etc. La construction du profil utilisateur consiste à collecter et exploiter les données pertinentes pour les représenter. La collecte d'information intègre la spécification du type des données pertinentes à collecter, et leur mode d'acquisition (explicite ou implicite). Le mode d'acquisition explicite des données est le plus facile à mettre en oeuvre et permet à l'utilisateur de saisir manuellement des informations utilisées dans la construction du profil. En revanche, le mode d'acquisition implicite repose sur des techniques d'extraction des informations basées sur des mesures de pertinence implicite (fréquence de clics, temps de lecture, etc.) appliqués sur l'historique d'interactions de l'utilisateur. Pour l'évolution du profil, son but fondamental est d'affiner la connaissance que le système a de l'utilisateur en mettant à profit l'expérience passée lors de ses activités de recherche [45].

- Domaine d'Intérêt : le domaine d'intérêt est la dimension centrale du profil utilisateur. Cette dimension regroupe tous les attributs qui concernent les objets de contenu (information ciblées). Elle peut définir aussi bien le domaine d'expertise et le niveau de qualification de l'utilisateur dans un domaine particulier que le contenu auquel s'intéresse

l'utilisateur[46].

- Données personnelles : pour que l'utilisateur d'un système personnalisable puisse en bénéficier, il est nécessaire qu'il soit identifié dans le système. Les données personnelles ont un double objectif, d'une part la gestion de l'identification de l'utilisateur (Nom, prénom,...) et d'autre part, elles permettent de catégoriser l'utilisateur en fonction des caractéristiques telles que les attributs d'authentification (Login, Mot de passe, etc.), les facteurs démographiques (Âge, Sexe, Première langue, Lieu de naissance, Particularités sociales et culturelles, etc.), les contacts personnels et professionnels et d'autres informations comme le groupe sanguin, le numéro du compte bancaire, etc. Ces données sont stables, rarement mises à jour[47].

- Préférences : aussi comme le domaine d'intérêts, les préférences est l'un des éléments les plus importants dans un profil utilisateur. Tous les utilisateurs ne sont pas intéressés par les mêmes informations, ou par la même présentation de l'information. Les préférences permettent la personnalisation des comportements des systèmes envers les utilisateurs [47]. Les préférences d'un utilisateur peuvent donc concerner :

- Le choix des méthodes de personnalisation et de services offerts.
- Le choix des composants graphiques à (ou ne pas) afficher.
- Les préférences concernant la présentation des informations.
- Les modalités d'exécution, décrivant le moment d'exécution d'une requête.
- Le niveau de détails souhaités.

- Expérience et compétences : selon [48], l'expérience de l'utilisateur représente son savoir-faire, la familiarité et l'aisance qu'il possède avec le type de système qui lui est présenté. Les compétences possédées par l'utilisateur correspondent aux connaissances qui ne relèvent ni du domaine, ni de l'expérience mais qui sont néanmoins considérées comme pertinentes dans le fonctionnement du système. La compétence est en relation avec d'autres concepts décrivant la capacité, l'habilité et le niveau d'expertise d'une personne.

b. Approches de représentation de profil utilisateur La représentation de l'utilisateur à travers un profil permet de cibler ses besoins spécifiques et à prendre en considérations ses préférences à travers l'amélioration du comportement des systèmes informatiques. Un modèle de représentation de profil permet d'organiser ces éléments afin de faciliter leur exploitation par les applications. On distingue quatre principales ap-

proches de représentation de profil utilisateur : Historique, Ensembliste, Connexionniste, et Multidimensionnelle.

- Représentation par l'historique : cette représentation est la plus répandue dans le domaine de la recherche d'informations. L'historique de recherche consiste en un ensemble de requêtes, de pages Web visitées ou cliquées ou des résumés textuels des résultats associés accumulés au cours des sessions de recherche de l'utilisateur. Dans cette catégorie, les travaux des auteurs [49] [50] représentent le profil utilisateur par son historique de recherche en se servant comme une base de données des requêtes utilisateur ainsi que leurs résultats associés. Cette base de données est utilisée pour sélectionner les requêtes les plus similaires à une requête en cours d'évaluation.

- Représentation ensembliste : la représentation ensembliste du profil utilisateur est basée sur un ensemble de mots clés (ou vecteurs de mots clés) pondérés représentés souvent selon le modèle vectoriel [51].

Ce type de représentation est le premier conçu pour modéliser le profil utilisateur. Les paquets de termes traduisent les centres d'intérêts de l'utilisateur. Dans [46], l'auteur divise la représentation ensembliste en trois sous-modèles de représentation :

- Un ensemble de termes pondérés où chaque terme représente un centre d'intérêt possible de l'utilisateur. °Un vecteur de termes pondérés représentant un centre d'intérêt [56].
- Un ensemble de vecteurs de termes pondérés.

- Représentation connexionniste : la représentation connexionniste du profil utilisateur consiste à représenter les centres d'intérêts de l'utilisateur par un réseau de nœuds pondérés dont chaque nœud représente un concept traduisant un centre d'intérêt. Cette représentation permet de résoudre les failles de la représentation ensembliste par la mise en place des relations de corrélation sémantiques entre les centres d'intérêts du profil. En effet, la richesse sémantique dans cette représentation permet de résoudre le problème de la polysémie des termes inhérents à la représentation ensembliste, l'incohérence possible entre les centres d'intérêts et l'identification d'un profil adéquat au sujet de la requête via les relations sémantiques [52][54].

Certains auteurs tels que [52] et [53], se sont intéressés à la représentation du profil utilisateur par un réseau de nœuds pondérés dans lequel chaque nœud représente un concept traduisant un centre d'intérêt utilisateur. Ce type de représentation offre le double avan-

tage de la structuration et de la représentation associative permettant de considérer l'ensemble des aspects représentatifs du profil. Les concepts composant le profil sont souvent représentés par des relations de paires de nœuds. L'auteur dans [55], suggère de créer les arcs reliant deux nœuds sur la base des occurrences entre ses termes.

• Représentation multidimensionnelle : les auteurs dans [44][47] ont adopté ce type de représentation pour le profil utilisateur. Il consiste en un ensemble de dimensions représentant chacune un aspect particulier (comme par exemple les données personnelles, le domaine d'intérêt). Son contenu est représenté par un modèle structuré de dimensions (ou catégories) prédéfinies [44] où par une structure générique [47] où chaque dimension est constituée d'un ensemble d'attributs éventuellement organisés en sous dimensions offrant ainsi, la possibilité d'élargir l'ensemble des données représentées en fonction de domaine d'application et des besoins utilisateur. Les attributs peuvent être simples ou composés (comme l'adresse). Une sous dimension regroupe un ensemble d'attributs simples qui sont liés sémantiquement (par exemple l'adresse est composée du numéro de la rue, du nom de la rue, du code postal etc.).

Localisation

La localisation peut concerner l'orientation, l'altitude et les relations spatiales entre les entités (comme les relations de proximité, de contenance). Par exemple, le fait qu'un objet A est orienté vers un objet B ou un autre objet C est une information qui peut être classée dans la catégorie localisation. La localisation peut aussi référencer des lieux [45].

Environnement (état)

Encapsule les caractéristiques des entités. Par exemple, pour un lieu donné, l'état peut caractériser la température, la luminosité, ou le niveau de bruit. Pour une personne, l'état peut se référer à des signes vitaux, sa fatigue ou son activité. Pour les composants logiciels, l'état peut être son temps de réponse, son taux d'utilisation [45].

Temps

Le temps peut aussi être une information contextuelle car il peut caractériser une entité. Il permet aussi d'établir un historique de valeurs permettant d'enrichir le contexte. En effet, l'enchaînement et l'ordonnancement d'actions ou d'évènements dans le temps peuvent aussi être importants pour la décision prise par l'application [42].

Relations

Finalement, cette catégorie de contexte capture les relations qu'une entité a établies avec d'autres entités, l'ensemble de toutes les relations construit une structure qui fait partie de contexte. Une relation exprime une dépendance sémantique entre deux entités qui se dégage de certaines circonstances. Chaque une joue un rôle spécifique dans une relation, cette dernière peut établir un certain nombre de relations différentes avec la même entité. En outre les relations ne sont pas nécessairement statiques et peuvent apparaître et disparaître dynamiquement, la catégorie de relation est subdivisé en social, fonctionnel et de composition relations [42].

- Relation sociale : décrit les aspects sociaux du contexte de l'entité courant, par exemple les relations entre les personnes sont des associations sociales.

- Relation fonctionnelle : c'est une relation fonctionnelle entre deux entités qui indique une entité utilise l'autre entité pour un certain but et avec un certain effet, ces relations présentent des propriétés physiques, des propriétés communicationnelles et interactionnelles.

- Relation de composition : est une relation entre un ensemble et ses parties. Ces derniers n'existeront plus si l'objet contenant est détruit [43].

2.2.4 Caractéristiques du contexte

Les différentes définitions existantes sur le contexte nous ont permis d'énumérer un ensemble de caractéristiques, qui sont :

- **Le contexte est lié à une entité** : le contexte est vu selon un certain point de vue très lié à une entité et n'est pas donné comme un concept absolu. Une entité peut être une personne, une machine ou un agent logiciel.

- **Le contexte n'est pas totalement contrôlable** : le contexte est déterminé par tout ce qui existe dans le monde et il n'est pas entièrement observable par une application qui est supposée interagir avec ses changements incontrôlables par la manière la plus adéquate possible.
- ● **Le contexte influence certains comportements** : le contexte a une influence sur les comportements d'une entité. Une entité qui évolue dans un certain contexte reflète les changements ayant lieu dans ce contexte. Les éléments n'ayant aucune influence sur le comportement d'une entité peuvent être exclus de son contexte.
- ● **Le contexte a une nature subjective** : puisque le contexte peut avoir une influence sur les décisions et actions d'une entité, la considération d'un élément comme étant une partie de contexte peut être faite selon les décisions qu'une entité est appelée à prendre.
- **Le contexte a une nature volatile** : les situations de contexte changent continuellement et évoluent dans le temps [42][43].

Sensible au contexte : Sensible à ce qui l'entoure ;

Autonome : Qui est construit ou qui fonctionne comme un tout indépendant ;

Everyware : Un terme introduit par Greenfield. La fusion du terme anglais everywhere signifiant " partout " et software pour " logiciel " ;

Internet des Objets : Une extension de l'Internet actuel à tous les objets pouvant communiquer de manière directe ou indirecte, avec des équipements électronique eux mêmes connectés à Internet.

2.3 Internet des objets

2.3.1 Définition de l' IoT

L'internet des objets (IoT) est une infrastructure dynamique d'un réseau global. Ce réseau global a des capacités d'auto configuration basée sur des standards et des protocoles

de communication interopérables. Dans ce réseau, les objets physiques et virtuels ont des identités, des attributs physiques, des personnalités virtuelles et des interfaces intelligentes et ils sont intégrés au réseau d'une façon transparente . En 2016 le nombre des objets connecté via internet à toucher 7.4 milliards d'objets, et les spécialiste du domaine estime 27 milliards d'objets connecté à l'échelle de 2025, la figure suivante représente un graphe sur le nombre existé et estimé des objets connecter [57].

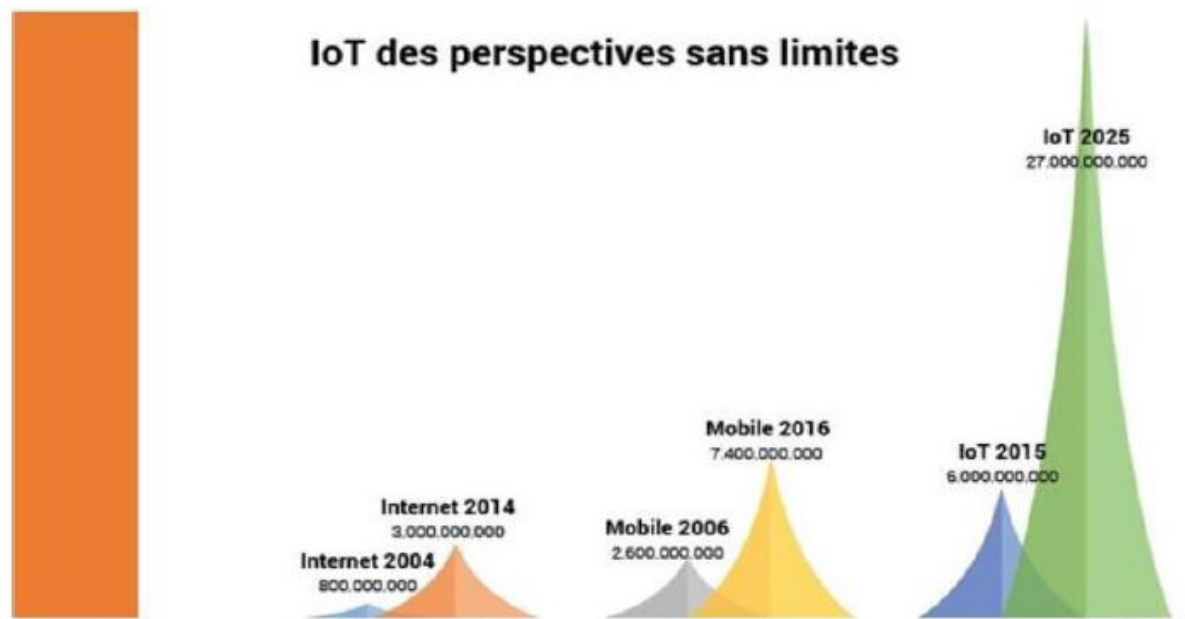


FIGURE 2.2 – Graphe du nombre existé et estimé des objets connecter.

2.3.2 Histoire de l' IoT

L'Internet des Objet (IoT) a été introduit pour la première fois par Kevin ASHTON [58] en 2009. Après 2009 cette technologie a connue un développement énorme, plusieurs projets et technique sont crée pour développer l'IDO, dans des plusieurs domaine de santé et transport et des villes intelligentes, nous expliquons plusieurs exemples dans les prochaines lignes. La figure suivante présente l'histoire de la technologie de la connectivité des objets depuis l'année 2000 .

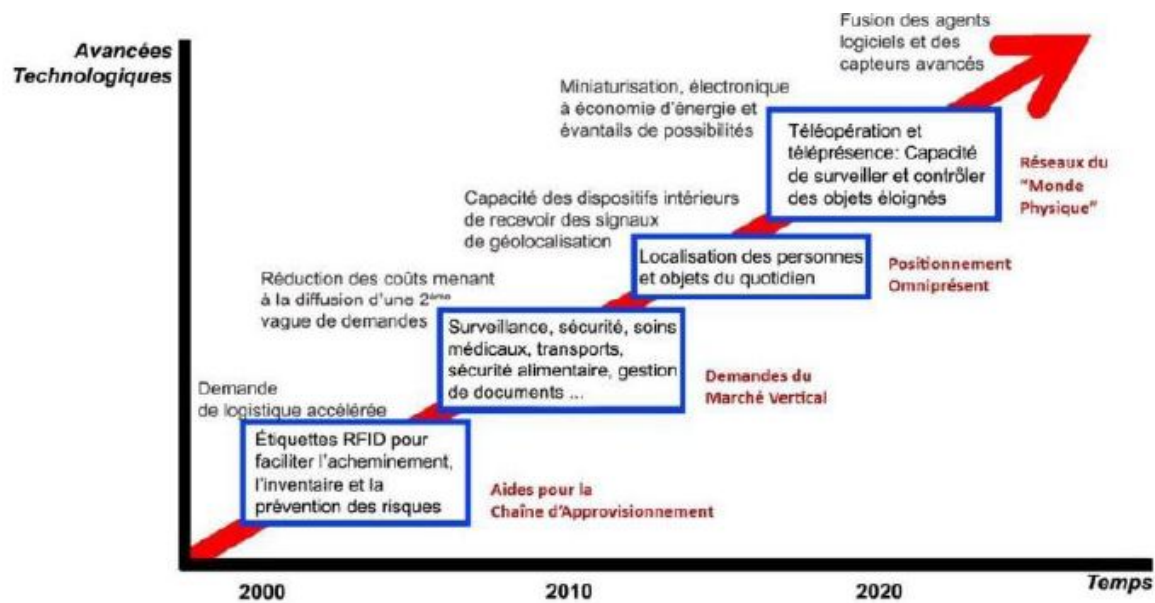


FIGURE 2.3 – Historique de la technologie de la connectivité des objets

2.3.3 Quelques domaines d' IoT

Internet des objets est représenté dans des plusieurs domaines, et après quelque année les chercheurs estimons que chaque objets de notre vie quotidienne va être connecté, et ce point est une cause depuis les cause de la création du protocole de IPV6, pour avoir un nombre suffisant des adresse IP pour toute les machines et les objets connecté dans le monde, voila les domaines les plus connue dans l' IoT [59] :

Transports : l'un des domaines les plus bénéficiaires de cette technologie, car ce domaine a connu un développement durable, dans le domaine ferroviaire, ou dans les voitures intelligentes, par la connaissance des points noir de la circulation, ou de trouver les plus courts chemins, toujours par l'utilisation des objets intelligentes et connecté via réseau.

smart city (bâtiments, villa, ville) : c'est un domaine vaste utilisé surtout dans les pays développé comme la Corée du sud et la suisse, les états unis et l'Allemagne, d'où nous trouvons des villa 100 pour 100 intelligentes ou bien des immeubles, des quartiers, Utilités et infrastructures ou même toute une ville connecté, et voila les top 5 villes les plus

intelligentes dans le monde d'après 1. Singapour 2. Barcelone 3. London 4. San Francisco 5. Oslo Et il existe plusieurs d'autres domaines comme l'agriculture, l'énergie, la sécurité alimentaire, les achats, ect.

2.3.4 Quelques exemples d'IoT

Dans les cinq dernières années, le domaine d'internet des objets a connu des bons projets qui ont vraiment changé et facilité la vie de l'être humain, nous présentons dans les prochaines lignes quelques exemples des projets de l'IoT :

Surveillance de la circulation : La surveillance de la circulation est l'un des meilleurs exemples dans la technique de l'IoT, par l'utilisation des différents types de capteurs, la circulation a connu un développement, surtout dans la détection des points d'encombrement, ou la détection d'un accident.

Trouver facilement une place de parking en ville : Par l'utilisation des capteurs installés dans chaque stationnement dans les parkings, les capteurs connaissent si cette place contient une voiture ou bien elle est libre, et aussi par l'utilisation d'une application web en temps réel, le conducteur peut recevoir la place vide la plus proche de lui, il peut la réserver par l'application pour qu'elle ne sera pas utilisée par un autre conducteur.

les véhicules sans chauffeur qui roule et stationne automatiquement : Il existe des voitures autonomes ont parcouru des milliers de kilomètres sans incidents majeurs, les voitures intelligentes basent sur plusieurs capteurs à l'intérieur et à l'extérieur de la voiture, par exemple elle a des capteurs qui détectent s'il y a un obstacle devant la voiture, si les capteurs ne détectent aucune obstacle, la voiture peut avancer, aussi la voiture se dépend sur le GPS pour connaître la destination, et elle a plusieurs capteurs à l'intérieur du véhicule pour connaître l'état de la voiture (l'existence d'essence, la vitesse, la position[59]).

2.4 Conclusion

Dans ce chapitre nous avons introduit par la notion du contexte qui désigne en général l'ensemble des informations qui entourent ou bien caractérisent la situation d'une entité. Plusieurs auteurs ont essayé de cerner et de couvrir le maximum d'informations sur ce concept au moyen de définitions et de classifications. Ainsi nous avons défini le profil utilisateur comme une source de connaissance qui contient des acquisitions sur tous les aspects de l'utilisateur qui peuvent être utiles pour modéliser l'utilisateur dont la modélisation se compose de trois phases, la première consiste la représentation de l'utilisateur, phase déconstruction ainsi que la phase d'évolution.

Nous avons présenté en dernier l'internet des objets qui est une infrastructure dynamique d'un réseau global, Qui permet d'interconnecter des objets (physiques ou virtuels) grâce aux technologies de l'information et de la communication interopérables.

Notre travail consiste à développer une application mobile sensible au contexte et au profil utilisateur et qui utilise les objets connectés. Nous détaillons plus dans les prochains chapitres.

Approche proposée

3.1 Introduction

Dans le premier chapitre, nous avons discuté plusieurs approches de recommandation existantes, chacune ayant ses avantages et ses limitations.

Dans notre cas, nous cherchons à recommander des endroits précis pour un utilisateur selon ses informations du contexte et du profil ainsi que ses préférences. Pour réaliser cela, nous avons choisi le filtrage collaboratif qui est le mieux adapté à notre situation.

Dans ce chapitre, nous allons commencer en proposant notre approche qui se base sur le filtrage collaboratif et l'apprentissage automatique. D'abord ; nous rappelons l'approche basée sur le filtrage collaboratif ainsi que l'apprentissage basé sur la classification supervisée. Ensuite, nous allons citer quelques projets de recommandation en relation avec l'apprentissage automatique. Enfin, nous allons appliquer l'algorithme KPP-v et l'algorithme CPP-v proposé qui signifie "Contexte, Profil, Préférence, score".

3.1.1 L'approche basées sur le filtrage collaboratif

Le filtrage collaboratif est une approche basée sur le partage d'opinions entre les utilisateurs [76]. L'hypothèse fondamentale de cette méthode est que les opinions des autres utilisateurs peuvent être utilisés pour fournir une prédiction raisonnable de la préférence de l'utilisateur sur un item qu'il n'a pas encore noté. Ces méthodes supposent que si des utilisateurs ont les mêmes préférences sur un ensemble d'items, alors ils auront probablement les mêmes préférences sur un autre ensemble d'items qu'il n'ont pas encore notés. Supposons par exemple que les voisins de Mouna trouvent que le nouveau restaurant qui

est ouvert dans son voisinage est intéressant, elle peut juger intéressant d'aller l'essayer. Si au contraire, la majorité de ses voisins estiment que ce n'est pas intéressant, alors il se peut qu'elle décide de s'abstenir d'y aller. Les techniques de filtrage collaboratif recommandent donc, à l'utilisateur courant, les items appréciés par les utilisateurs avec lesquels il partage les mêmes goûts. Dans ce cas, on parle d'utilisateurs similaires [60].

3.2 La classification automatique supervisée

A cause de la grande quantité de documents échangés et stockés sur les supports électroniques, la classification automatique supervisée est devenue plus que nécessaire pour faciliter l'utilisation et l'analyse des données. A la différence de la classification non supervisée, où l'ordinateur doit trouver automatiquement les classes, la classification supervisée se base principalement sur le fait qu'il existe déjà une classification de documents, c'est-à-dire qu'on dispose d'un ensemble de données déjà classées qu'on appelle « ensemble d'apprentissage » et qu'on utilise comme base, pour classer le reste des données. On essaie dans ce type de classification de trouver le maximum d'informations à partir des ensembles d'apprentissage, pour permettre un meilleur groupement des données restantes[61].

3.3 Les projets de recommandation en relation avec l'apprentissage automatique

Voici quelques systèmes de recommandation qui ont utilisé l'apprentissage automatique.

3.3.1 Recommandation pour le secteur de l'hôtellerie à la dernière minute

Rezza est une équipe commerciale spécialisée dans la réservation de chambres à la dernière minute sur smartphone. Cette entreprise essaie de donner les prix les plus intéressants du marché pour les clients et ceci en développant un système d'enchères à prix dégressif en temps réel.

Dans le but de maximiser les transactions, Rezza souhaite rendre l'utilisation des applications de réservation optimales, que ce soit pour les hôteliers en leur suggérant des offres de prix optimales en fonction de la demande future des clients et des objectifs de rentabilité de l'entreprise ; ou pour les clients finaux, en leur recommandant les offres les plus en adéquation à leurs goûts, leurs contextes ou aux autres critères de choix. Pour qu'un système de recommandation répond parfaitement au besoins du domaine de la réservation de dernière minute il doit combiner plus qu'un système similaire et ceci en prenant en compte : les stratégies de recommandation en fonction des seuils de données disponibles et des cas d'utilisation, les données d'évaluation déduites depuis les données collectées, la combinaison de données implicites et explicites et la pondération en fonction des objectifs de ce genre de système. L'architecture des prédictions et recommandations de l'équipe Rezza est conçu comme suite [62] :

- La prédiction "Price change trigger" : consiste à savoir la période et la valeur de changement de prix d'une chambre, afin d'assurer toujours d'être moins cher que la concurrence.
- La prédiction "Overbooking risk & room availability" : essaie de savoir la disponibilité des chambres, car souvent l'hôtelier ne prend pas le temps de la mettre à jour.
- Recommandations "Auto yield recommandation" et "Customer recommended offers" : Ce service de recommandation nécessite nombreuses données d'apprentissage qui peuvent aussi être quasi inexistantes pour pouvoir recommander sur la base de différents contextes (lieu, heure, concurrence locale).
- La prédiction "Customer booking probability" : tente de prédire en temps réel quand est-ce qu'un client prévoit de réserver : chez les concurrents, ou bien sur le service de Rezza.

Finalement, le rôle de ce projet est de fournir à la société Rezza des outils de prédiction et de recommandation, dédiés au secteur de la réservation d'hôtel à la dernière minute, et de découvrir de nouvelles opportunités pour l'entreprise en exploitant les techniques de machine Learning [62].

3.3.2 Moteur de recommandation des cadeaux

Le cadeau cherche à symboliser le bénéficiaire ainsi que le donateur, pour cela un réseau social a été mis afin de proposer à l'utilisateur de lui apporter une assistance dans son choix de cadeau, il s'agit de Pleazup, qui est un réseau social de partage et d'échange

d'idées de cadeau entre proches qui a vu le jour en 2014. L'idée de développer un moteur de recommandation de cadeaux pour ce réseau social est celle d'un étudiant à l'Université de la Réunion en 2017, en se basant comme objectif de répondre à la question : Comment recommander des idées cadeaux à une personne en se basant sur ses interactions précédentes sur le réseau social ?

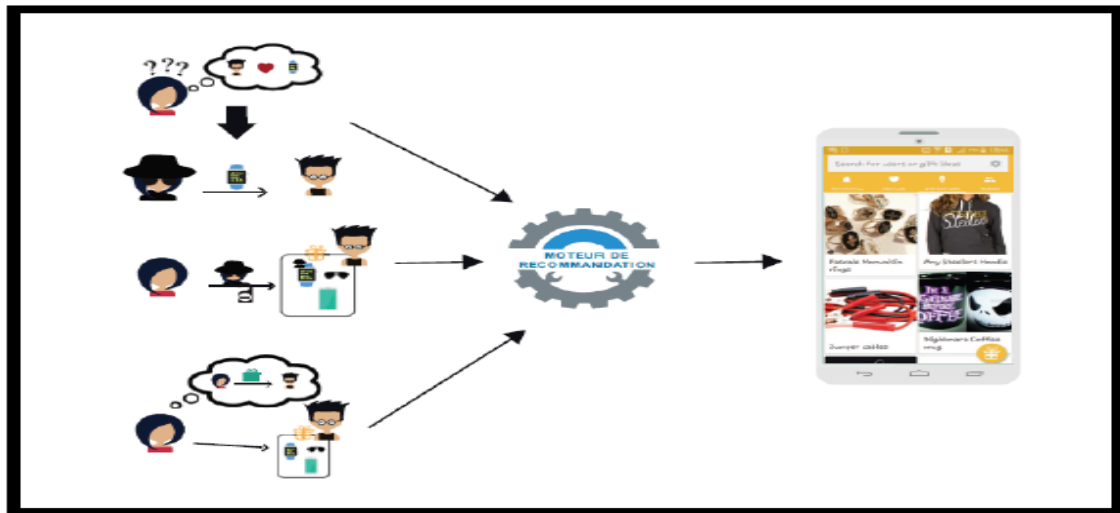


FIGURE 3.1 – Exemple de processus de la recommandation des cadeaux.

Dans un premier temps, l'étudiant a fixé comme objectif de développer une recommandation générale/statique basée sur les caractéristiques des idées cadeaux et exploitant les données sociales associées ; ensuite, développer une recommandation individuelle/dynamique basée sur les caractéristiques des utilisateurs ; pour améliorer l'estimation des centres d'intérêt des utilisateurs. Une API de classification automatique des images est aussi un objectif de plus. Afin de réaliser les objectifs cités, plusieurs règles métiers ont été mises en place, tel que : les idées de cadeau affiliées devaient être mises en avant après avoir calculé la recommandation, montrer dans les inspirations que les idées visibles et qui avaient des images associées. Les actions des utilisateurs ainsi que les mises à jour qui peuvent se faire sur les articles de cadeau ont été récoltées afin d'effectuer la tâche d'implémentation dans la plateforme de service mobile de Pleazup [63].

3.3.3 BlaBlaCar :

C'est une plateforme communautaire payante de covoiturage. La figure 3.2 montre un exemple d'une capture de l'application [64]

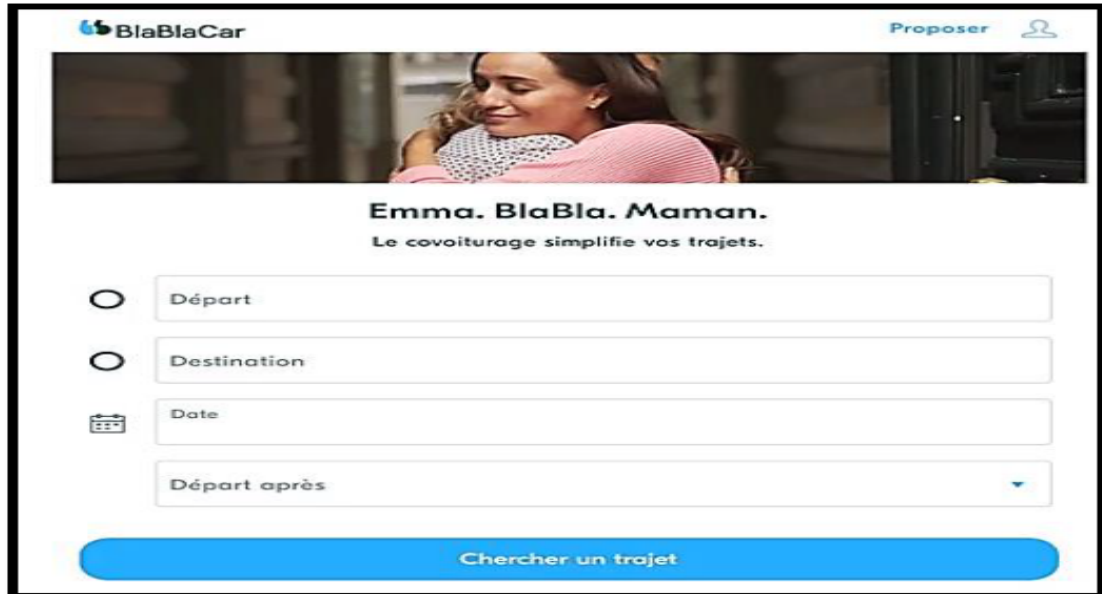


FIGURE 3.2 – Exemple de l'application BlaBlaCar

Cette application marche avec les systèmes de recommandation qui recommandent des meilleurs trajets aux voyageurs recherchant un co-voiturage sur la plateforme, à l'aide des algorithmes d'apprentissage automatique. Durant ces dernières années le Machine Learning a trouvé plusieurs axes d'application chez BlaBlaCar.

D'après Romain Serres, Data Scientist chez BlaBlaCar, quand on fait une recherche dans l'application par exemple (Paris-Toulouse), pleins de résultats seront retournés, donc c'est déjà beaucoup et ne nous pouvons pas trouver facilement le meilleur trajet sachant qu'il y'a beaucoup de critères à choisir : la distance, le cout, la duré d'arrivée, le profil de la personne, etc.

Donc le problème qui se pose est comment aider les passagers à trouver leur meilleur trajet plus facilement ? Pour cette raison les data scientist de BlaBlaCar ont développé des stratégies afin de recommander le trajet au passager selon son besoin, soit le trajet qui coûte moins cher ou bien le trajet le plus proche aussi sinon le trajet de meilleur profil.

Parmi ces stratégies il existe des algorithmes basés sur des règles en une dimension

(1D) par exemple le moins cher ou le plus proche dans le temps de la recherche. Une autre méthode un peu compliquée c'est les règles hiérarchique c'est une série des conditions comme le choix de trajet le plus proche de la distance de départ puis le plus proche de la distance d'arrivée, puis le moins cher. Mais toutes ces stratégies sont complexes et non justifiées même s'il faut sortir le même algorithme de recommandation de recherche sur de nouveaux pays où c'est un changement de produits assez significatif, il est nécessaire de réécrire toutes ces règles à la main, c'est pour ça ils ont utilisé l'apprentissage automatique.

La construction de recommandation basée sur l'apprentissage automatique passe par les grandes étapes d'apprentissages suivante :

- collecte des données de la recherche (date, trajet, l'heure, ...);
- Nettoyage des données (clean data) : c'est enlever les erreurs existantes, supprimer les incohérences, faire des hypothèses sur quelles données sont bonne et les quelles ne sont pas bonne;
- Entraînement de modèle (train model) : faire apprendre un algorithme avec les données. Le résultat est une équation qui vise à apprendre quelle est la probabilité que le passager réserve le trajet.

$$P[\text{passager}] = f(\text{recherche}, \text{trajet})$$

Le calcul de la probabilité se fait par une fonction qui prend la requête du passager et les caractéristiques du trajet, à la fin de calcul le trajet recommandé est celui qui a la probabilité la plus élevé.

- Validation de modèle : c'est l'estimation des performances est validation des paramètres.

3.4 Architecture générale de notre application

Nous commençons tout d'abord par illustrer notre architecture du modèle que nous avons créé[64].

Nous allons expliquer toutes ses étapes dans les prochaines lignes. Nous avons trois étapes à suivre : pré-traitement, apprentissage et recommandation.

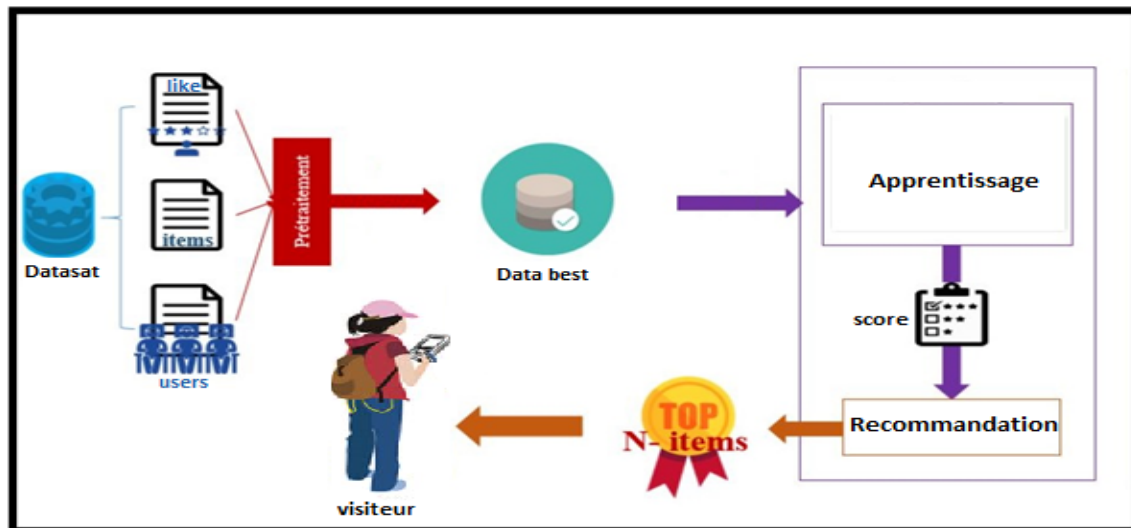


FIGURE 3.3 – Architecture de notre application

Etape 1 : Pré-traitement

L'objectif de cette partie est l'analyse et la normalisation des données. Cette étape consiste à nettoyer les données.

a- Premièrement, il s'agit de traiter les données pour préparer une collection de données.

b- Le deuxième pas est d'extraire les attributs essentiels (par exemple : nous avons éliminé nom, prénom) et éliminer les non essentiels, ou bien les doublants.

c- Finalement, nous allons faire la jointure des fichiers de données que nous avons sélectionné en un seul fichier, dans le but de faire une nouvelle méthode qui prend en considération tous les attributs et non pas un seul attribut comme dans les systèmes de recommandation classiques.

Etape 2 : Apprentissage

Une fois le fichier de données traité, nous allons utiliser un algorithme supervisé qui est le K plus proche voisins et notre algorithme de Score proposé.

Avant de passer à l'apprentissage du modèle nous allons faire :

- une normalisation des valeurs des attributs de notre best datasat.
- une création de notre modèle d'entraînement avec les algorithmes KPP-v et score.

Etape 3 : Recommandation

Arrivons à la dernière étape dans notre travail, qui est la recommandation où nous allons prédire des ressources à l'utilisateur.

avant de détailler les différentes étapes de l'algorithme KPP-v et notre algorithme proposé, nous présentons notre base de données

3.5 Base de données

Nous avons créé une base de données "Tourisme" qui contient 100 profils utilisateurs, le profil est composé de données personnelles et des préférences qui vont aider à la recommandation, Notre base de données contient aussi une liste de 136 endroits intéressants à visiter. Chaque endroit appartient a un type d'endroit, par exemple :tikjda est de type montagne.

Les tables utilisées sont les suivantes :

All_Place : cette table contient les informations des places tels que id_place,nom de la place, l'altitude et longitude, le type, le genre, la place exacte, les places validées, les places favorées

User : cette table contient les informations de l'utilisateur tel que le id,le nom, le prénom, les préférences, le genre, l'altitude et longitude.

3.6 K plus proches voisins

KPPV (k-Plus Proches Voisins) ou K-nn (K-nearest neighbor) est une méthode d'apprentissage supervisée qui raisonne avec le principe sous-jacent : "dis moi qui sont tes amis, je te dirais qui tu es". Elle diffère des méthodes d'apprentissages traditionnelles car aucun modèle n'est induit à partir d'exemple. A chaque fois que l'on veut classer un nouvel individu, on refait tourner l'algorithme et on cherche de nouveaux amis. KPP-v figure parmi les plus simples algorithmes d'apprentissage artificiel.

Dans un contexte de classification d'une nouvelle observation x , l'idée fondatrice simple est de faire voter les plus proches voisins de cette observation. La classe de x est déterminée en fonction de la classe majoritaire parmi les k plus proches voisins de l'observation x . La méthode KPP-v est donc une méthode à base de voisinage [66].

Dans les prochaines sections, nous expliquons l'architecture de l'algorithme KPP-v.

3.6.1 Architecture de l'algorithme KPP-v

La Figure 3.4 représente les trois étapes de l'algorithme KPP-v suivantes [65] :

1. Normaliser des attributs numériques.

2. Calculer la distance entre la nouvelle donnée et chaque donnée de la base d'apprentissage.
3. Trouver la classe majoritaire parmi les K données de la base d'apprentissage.

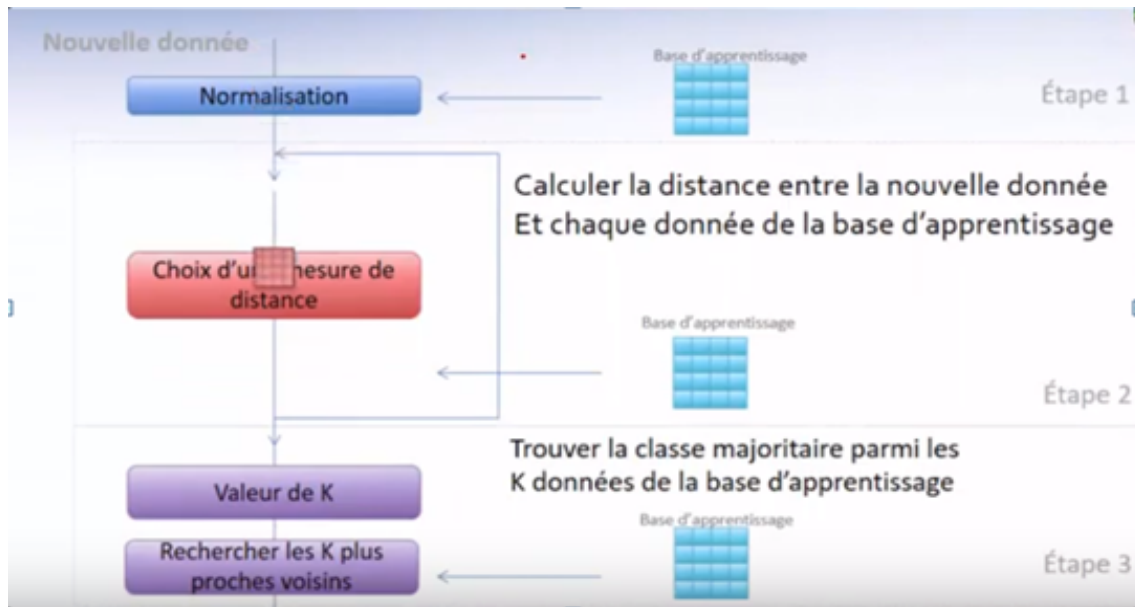


FIGURE 3.4 – Architecture de l'algorithme KPP-v

Calcul de distance dans l'algorithme KPP-v

Comme on vient de le voir dans notre écriture de l'architecture algorithmique, KPPv a besoin d'une fonction de calcul de distances entre deux observations. Plus deux points sont proches l'un de l'autre, plus ils sont similaires et vice versa.

Il existe plusieurs fonctions de calcul de distances, notamment, la distance euclidienne, la distance de Manhattan, la distance de Minkowski, etc. Nous allons utiliser la distance euclidienne [66] qui est la fonction la plus utilisée et qui donne de bons résultats [67].

3.6.2 Application de l'algorithme K Plus Proche voisin :

Après avoir défini l'algorithme kPP-v Nous allons l'appliquer sur notre base de données afin de faire des recommandations aux utilisateurs. Les recommandations concernent les anciens utilisateurs et les nouveaux utilisateurs :

- Dans la première partie nous allons recommander des places aux anciens utilisateurs.

- Dans la deuxième partie nous allons recommander des places a un nouveau utilisateur.

I. 1ère parties : La recommandation des places pour les anciens utilisateurs

Nous allons appliqué la méthode de KPP-v sur notre base de données.

La table 3.1 représente la base d'apprentissage constitué de huit places, et la table 3.2 représente la base de test pour deux utilisateurs a qui on va recommander des places .

Id-place	l'altitude	longitude	Genre	Ext-place	Type de place
20	36.449697	4.127364	Children	Tikjda	Montagne
18	36.380620	3.904818	Children	Parc De L'envi- -ronement	loisir
133	36.640488	2.694023	woman	jardin des pins	jardin
122	36.603594	5.491616	man	Parc national de Taza	Montagne
126	36.397906	3.900729	man	Uno Shopp -ing Center Bouira	Boutique
23	36.449697	4.127364	Children	Tikjda	Montagne
144	36.604145	5.660354	woman	Ain Mchaki	Montagne
118	36.821317	5.764172	man	Musée Kotama - Jijel	Musée

TABLE 3.1 – base d'apprentissage KPP (place)

Id-user	l'altitude	longitude	Genre	Nome	Type de place
85	36.396591	3.998636	Children	mayssa	Montagne
97	36,811123	5,763035	man	Moussa	Montagne

TABLE 3.2 – base de test KPP(user)

Après la sélection des tables, on va appliquer les étapes de l'algorithme KPP-v comme déjà expliqué (dans la Figure 3.4).

Etape 1 : Normaliser les données numériques

Nous allons normaliser les données numériques (latitude et longitude) par la formule suivante[65] :

$$V \text{ attribut}(\text{normalisé}) = \frac{V \text{ attribut} - V \text{ min}}{V \text{ max} - V \text{ min}}$$

Nous allons chercher le V min et le V max de chaque attribut.(voir la Table 3.3)

	l'altitude	longitude
V min	36.380620	2.694023
V max	36.821317	5.764172

TABLE 3.3 – V min et V max des attributs

Après le calcul des attributs normalisés, on aura les tables d'apprentissage (voir la Table 3.4) et de test (voir la Table 3.5) avec des valeurs normalisées.

Id-place	l'altitude	longitude	Genre	Ext-place	Type de place
20	0.156	0.466	Children	Tikjda	Montagne
18	0	0.394	Children	Parc De L'envi -ronement	loisir
133	0.589	0	woman	jardin des pins	jardin
122	0.505	0.911	man	Parc national de Taza	Montagne
126	0.037	0.393	man	Uno Shopp -ing Center Bouira	Boutique
23	0.156	0.466	Children	Tikjda	Montagne
144	0.507	0.966	woman	Ain Mchaki	Montagne
118	1	1	man	Musée Kotama - Jijel	Musée

TABLE 3.4 – Base d'apprentissage normalisée KPP-v(places)

Id-user	l'altitude	longitude	Genre	Nome	Type de place
85	0.036	0.424	Children	mayssa	Montagne
97	0.976	0.999	man	Moussa	Montagne

TABLE 3.5 – Base de test normalisé KPP(user)

Etape2 : Calculer la distance entre la nouvelle donnée et chaque donnée de la base d'apprentissage .

nous appliquons utilisé la mesure de distance euclidienne suivante [68] :

$$\sqrt{\sum_{i=1}^n (x_i - y_i)^2}$$

A- Exemple 1

Nous voulons recommander à l'utilisateur (id-user=85) les top-N places .

Pour cela, nous appliquons la formule suivante : distance(place/user) =

$$\sqrt{(altP - altU)^2 + (longP - longU)^2 + (genrP - genreU)^2 + (TypedeplaceP - PréférenceU)^2}$$

Exemple pratique :

$$\text{distance(place/user)} = \sqrt{0.0144 + 0.001754 + 0 + 0} = \mathbf{0.127}$$

Etape3 : Trouver la classe majoritaire parmi les k données de la base d'apprentissage

Après le calcul de la distance, on obtient les distances qui sont représentées dans la Table 3.6.

On a supposé que **k=2**

Après l calcul des distances comme illustré dans la table 3.6 entre les places et l'utilisateur (id-user=85), on aura :

$$\text{distance (place-20/user-85)} = 0.127$$

$$\text{distance (place-/user-85)} = 1.000$$

On obtient une classe Montagne et une classe Loisir, donc on recommande les places suivantes :

- Tikjda
- Parc de l'envirenement

B- Exemple 2

Nous voulons recommander à l'utilisateur (id-user=97) les top-N places.

Nous allons appliquer les mêmes étapes de l'algorithme KPP-v. La Table 3.7 représente les distances entre les places et l'utilisateur (id-user=97) :

Id-place	l'altitude	longitude	Genre	Ext-place	Type de place	distance
20	0.156	0.466	Children	Tikjda	Montagne	0.127
18	0	0.394	Children	Parc De L'envi- -ronement	loisir	1.000
133	0.589	0	woman	jardin des pins	jardin	1.56
122	0.505	0.911	man	Parc national de Taza	Montagne	1.38
126	0.037	0.393	man	Uno Shopp -ing Center Bouira	Boutique	1.141
23	0.156	0.466	Children	Tikjda	Montagne	1.008
144	0.507	0.966	woman	Ain Mchaki	Montagne	1.23
118	1	1	man	Musée Kotama - Jijel	Musée	1.58

TABLE 3.6 – base d'apprentissage normalisé + distance KPP(place)

On a supposé que $k=2$

Après le calcul de la distance entre les places et les positions de l'utilisateur (id-user=97) (voir la Table 3.7), on a trouvé une classe Montagne et une classe Musée, donc on lui recommande les places suivantes :

- Parc national de Taza
- Musée Kotama - Jijel

2 ème parties : La recommandation des places pour les nouveaux utilisateurs

Nous allons appliqué la méthode de KPP-v sur notre base de données.

La table 3.8 représente la base d'apprentissage constitué de sept utilisateurs , et la table

Id-place	l'altitude	longitude	Genre	Ext-place	Type de place	distance
20	0.156	0.466	Children	Tikjda	Montagne	1.398
18	0	0.394	Children	Parc De L'envi -ronement	loisir	1.144
133	0.589	0	woman	jardin des pins	jardin	1.78
122	0.505	0.911	man	Parc national de Taza	Montagne	0.47
126	0.037	0.393	man	Uno Shopp -ing Center Bouira	Boutique	1.49
23	0.156	0.466	Children	Tikjda	Montagne	1.39
144	0.507	0.966	woman	Ain Mchaki	Montagne	1.10
118	1	1	man	Musée Kotama - Jijel	Musée	1.000

TABLE 3.7 – base d'apprentissage normalisé kpp2(place)

3.9 représente base de test (Table 3.9) pour un utilisateur à qui on va lui recommander des places .

Id-user	l'altitude	longitude	Genre	P1	P2	P3
0	36.377899	3,876769	man	Musée	Boutique	Plage
77	36,746773	3,072751	Children	Jardin	Zoo	Loisir
26	36,376877	3,892184	Children	Musée	Zoo	Plage
63	36.823933	5,743015	Children	Montagne	Zoo	Plage
58	36,781242	3,012753	woman	Restaurant	Jardin	Loisir
70	36,725338	3,288645	Children	Loisir	Zoo	Plage
42	36.757683	2,913575	woman	Loisir	Plage	Autre

TABLE 3.8 – Base d'apprentissage KPP(user)

Id-user	l'altitude	longitude	Genre	P1	P2	P3
100	36,813034	2,971223	Children	Zoo	Loisir	Plage

TABLE 3.9 – Base de teste KPP (new user)

Etape 1 : Normaliser les données numériques

Nous allons normaliser les données numériques (latitude et longitude) par la formule suivant :

$$V \text{ attribut}(\text{normalisé}) = \frac{V \text{ attribut} - V \text{ min}}{V \text{ max} - V \text{ min}}$$

Nous allons chercher le V min et le V max de chaque attribut (voir la Table 3.10).

	l'altitude	longitude
V min	36,376877	2,913575
V max	36.823933	5.743015

TABLE 3.10 – V min et V max des attributs

Après le calcul des attributs normalisés, on aura la table d'apprentissage (voir la Table 3.11) et la table de test (voir la Table 3.12) avec des valeurs normalisées.

Id-user	l'altitude	longitude	Genre	P1	P2	P3
0	2,286	0,340	man	Musée	Boutique	Plage
77	0.827	0.056	Children	Jardin	Zoo	Loisir
26	0	0,345	Children	Musée	Zoo	Plage
63	1	1	Children	Montagne	Zoo	Plage
58	0,90	0,035	woman	Restaurant	Jardin	Loisir
70	0,779	0,1325	Children	Loisir	Zoo	Plage
42	0,8518	0	woman	Loisir	Plage	Autre

TABLE 3.11 – Base d'apprentissage normalisée KPP(user)

Id-user	l'altitude	longitude	Genre	P1	P2	P3	Classe
100	0,975	0,020	Children	Zoo	Loisir	Plage	!!!

TABLE 3.12 – Base de teste normalisée KPP(new user)

Etape2 :Calculer la distance entre la nouvelle donnée et chaque donnée de la base d'apprentissage .

On utilise la mesure de distance euclidienne. La Table 3.13 représente la distance entre les anciens utilisateurs et le nouvel utilisateur.

Id-user	l'altitude	longitude	Genre	P1	P2	P3	Distance
0	2,286	0,340	man	Musée	Boutique	Plage	2.41
77	0.827	0.056	Children	Jardin	Zoo	Loisir	1.73
26	0	0,345	Children	Musée	Zoo	Plage	1.743
63	1	1	Children	Montagne	Zoo	Plage	1.720
58	0,90	0,035	woman	Restaurant	Jardin	Loisir	2.0013
70	0,779	0,1325	Children	Loisir	Zoo	Plage	1.455
42	0,8518	0	woman	Loisir	Plage	Autre	2.004

TABLE 3.13 – Base d'apprentissage normalisée + distance KPP-v (user)

Etape3 : Trouver la classe majoritaire parmi les k données de la base d'apprentissage

Après le calcul de distance on obtient les distances qui sont représentées dans la Table 3.13.

On a supposé que $k=1$

Après le calcul des distances entre les utilisateurs et le nouvel utilisateur ($id-user=100$), on obtient la distance la plus minimale c'est 1.455, l'utilisateur le plus proche, qui est l'utilisateur ($id-user=70$). Donc nous recommandons au nouveau utilisateur les top-N places que nous avons déjà recommandé dans la 1ère partie.

3.7 Notre algorithme proposé "Algorithme CPP-s"

Nous proposons dans cette partie un algorithme basé sur le filtrage collaboratif, qui consiste à calculer les scores. Cet algorithme est sensible au contexte, il prend en considération le profil de l'utilisateur, ses préférences, et aussi sa localisation. Les recommandations changent par rapport à ces derniers.

Afin de réaliser la recommandation, nous devons calculer les scores et trouver les places qui conviennent à chaque utilisateur. Notre l'approche est divisé sur deux parties :

- La première partie consiste à recommander des places aux anciens utilisateurs.
- La deuxième partie consiste à recommander à un nouvel utilisateur des places selon son contexte.

Chaque partie est constituée de trois étapes :

- La première étape consiste à calculer le score.
- La deuxième étape consiste à trouver le score max.
- La troisième étape consiste à recommander les places selon les scores.

Cet algorithme permet une recommandation plus facile dans le cas où les items ne sont pas numériques.

1 ère parties : La recommandation des places pour les anciens utilisateur

Dans cette partie, nous allons recommander des places pour les utilisateurs existants suivant ces trois étapes :

- Le calcul des scores.
- Trouver le max scores.
- La recommandation des places selon les scores.

i. Le calcul des scores :

Le principe est de calculer le score de chaque utilisateur existant avec chaque place existante, par exemple : si la place A est destinée pour les **femmes (profil)** et que l'utilisateur U est une **femme** et la place A a un grand nombre de **likes** et **proche** de l'utilisateur U (**contexte**) et le type de la place A est parmi ses préférences (**préférences**) alors la place A sera recommandée à l'utilisateur U, Ce calcul de score s'appuie sur 4 points principaux :

- Sexe.
- Distance.
- Type de place.
- Likes.

Chaque point prend 25 % .(voir la Figure 3.5)

- Le sexe : prend 25% si l'utilisateur a le même sexe, 0% sinon.
- Le type de place : l'utilisateur doit choisir ses trois places préférées, ces dernières sont classer selon l'ordre suivant : p1 la plus préférée, p2 en 2 ème place, et p3 en dernier.
- la distance : elle est divisée en trois cas : si l'utilisateur est très proche de l'endroit proposé, Il prend 25%, s'il est moins proche il prend 15%, sinon il prend 5% (s'il est loin).
- Le like : l'utilisateur peut faire un like sur l'endroit qu'il a aimé, l'endroit le plus aimé (fort) prend 25%, le moins aimé prend 15%, et le dernier 15%.

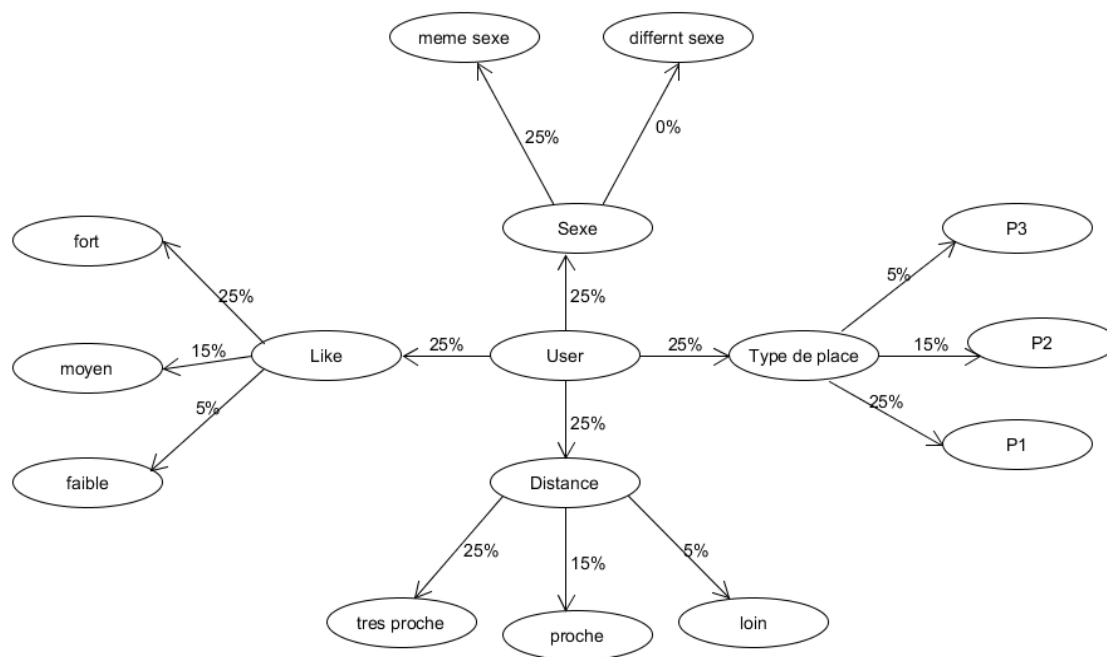


FIGURE 3.5 – Shéma représentant distribution des scores selon l’algorithme CPP-s

Le calcul de score pour le sexe

Ce nombre peut être soit 0% ou 25%. 0% indique que le genre est différent tandis que 25% indique que les utilisateurs ont le même genre.

Voici une matrice explicative avec les User-id en tant que lignes, genre en tant que colonnes (Figure 3.6) :

- M représente Man.
- W représente Woman.
- C représente Children.

$$\begin{matrix}
 id - 1 \\
 id - 2 \\
 id - 3 \\
 \dots \\
 id - 99
 \end{matrix}
 \begin{pmatrix}
 M & W & C \\
 1 & 0 & 0 \\
 0 & 1 & \\
 0 & 1 & 0 \\
 \dots & \dots & \dots \\
 0 & 1 & 0
 \end{pmatrix}$$

FIGURE 3.6 – Matrice obtenue pour le sexe

Le calcul de score pour la distance

La distance entre l'utilisateur et les endroits est calculé par la mesure de similarité "euclidienne" qui est déjà défini dans le chapitre 1.

Nous avons utilisée la formule suivante :

$$d = \sqrt{(x_1 - x_2)^2 + (y_1 - y_2)^2}$$

$$\sqrt{\sum_{i=1}^n (x_i - y_i)^2}$$

x_1 : exprime la longitude de l'utilisateur.

x_2 : exprime la longitude de chaque place.

y_1 : exprime l'altitude de l'utilisateur.

y_2 : exprime l'altitude de chaque place.

La distance peut prendre trois valeurs, si la distance est très proche elle prend 25%. Ce nombre peut prendre 5%, 15% ou 25% . 5% indique que la distance est loin tandis que 25% indique que la distance est proche de l'utilisateur.

La figure 3.7 illustre une matrice des user-id (lignes) et des places (colonnes).

P1, P2, ...Pi représentent les places.

$$\begin{array}{c} P_1 \quad P_2 \quad P_3 \quad \dots \quad \dots \quad P_{136} \\ \begin{array}{l} id - 1 \\ id - 2 \\ id - 3 \\ \dots \\ id - 99 \end{array} \end{array} \left(\begin{array}{cccccc} 25 & 10 & 5 & \dots & \dots & \dots \\ 5 & 10 & 15 & \dots & \dots & 25 \\ 5 & 15 & 25 & \dots & \dots & \dots \\ \dots & \dots & \dots & \dots & \dots & \dots \\ 15 & 25 & 15 & \dots & \dots & \dots \end{array} \right)$$

FIGURE 3.7 – Matrice obtenue pour la distance

Le calcul de score pour le type de places

Dans notre bases de données chaque utilisateur possède trois types de places qu'il préfère. Ces préférences sont classées par ordre : la première préférence est notée sur 25% la deuxième est notée sur 15% et la dernière sur 5%. Comme le montre la matrice suivante avec les user-id en tant que lignes, et les type de places en tant que colonnes (voir la figure 3.8).

P_1, P_2, \dots, P_i representent les places.

$$\begin{array}{c}
 P_1 \quad P_2 \quad P_3 \quad \dots \quad \dots \quad P_{136} \\
 id - 1 \quad \left(\begin{array}{cccccc}
 15 & 25 & 15 & \dots & \dots & \dots \\
 5 & 15 & 15 & \dots & \dots & 15 \\
 25 & 15 & 25 & \dots & \dots & \dots \\
 \dots & \dots & \dots & \dots & \dots & \dots \\
 25 & 15 & 5 & \dots & \dots & \dots
 \end{array} \right)
 \end{array}$$

FIGURE 3.8 – Matrice obtenue pour le type des places

Le calcul de score pour les likes

Chaque place contient un nombre de like obtenu par les utilisateurs.

La figure 3.9 illustre une matrice avec les likes en tant que lignes, et les places en tant que colonnes.

Tout d'abord on calcule la moyenne des likes

$$(\text{moyenne} = \sum \text{nombre} - \text{de} - \text{like} / \sum \text{nombre} - \text{de} - \text{places}) = 50.5).$$

En suite, on devise les likes en trois intervalles : le premier inférieur ou égal à 40 likes, le deuxième interval sera compris entre 40 et 60 likes, et le troisième supérieure à 60 likes. Les intervalles prennent les valeurs de 5%, 15% et 25% respectivement.

contenu... P_1, P_2, \dots, P_n représentent les places.

V_1, V_2, \dots, V_n representent les valeurs des likes

$$\begin{array}{c}
 P_1 \quad P_2 \quad P_3 \quad \dots \quad \dots \quad P_{136} \\
 v1 \left(\begin{array}{cccccc} 15 & 25 & 15 & \dots & \dots & \dots \end{array} \right) \\
 v2 \left(\begin{array}{cccccc} 5 & 15 & 15 & \dots & \dots & 15 \end{array} \right) \\
 v3 \left(\begin{array}{cccccc} 5 & 25 & 25 & \dots & \dots & \dots \end{array} \right) \\
 \dots \left(\begin{array}{cccccc} \dots & \dots & \dots & \dots & \dots & \dots \end{array} \right) \\
 vn \left(\begin{array}{cccccc} 25 & \dots & \dots & \dots & \dots & \dots \end{array} \right)
 \end{array}$$

FIGURE 3.9 – Matrice obtenue pour les likes

Après avoir calculer les scores des utilisateurs et les places, on obtient la matrice globale (user, places) avec 99 utilisateurs et 136 places.

Nous avons 13464 relations comme illustrée dans la Figure 3.10

$$\begin{array}{c}
 P_1 \quad P_2 \quad P_3 \quad \dots \quad \dots \quad P_{136} \\
 id - 1 \left(\begin{array}{cccccc} 60 & 5 & 10 & \dots & \dots & 45 \end{array} \right) \\
 id - 2 \left(\begin{array}{cccccc} 20 & 100 & 15 & \dots & \dots & \dots \end{array} \right) \\
 id - 3 \left(\begin{array}{cccccc} 5 & 15 & 25 & \dots & \dots & \dots \end{array} \right) \\
 \dots \left(\begin{array}{cccccc} \dots & \dots & \dots & \dots & \dots & \dots \end{array} \right) \\
 id - 99 \left(\begin{array}{cccccc} 80 & 15 & 60 & \dots & \dots & \dots \end{array} \right)
 \end{array}$$

FIGURE 3.10 – La matrice globale User/place

ii. Trouver le max score

Après le calcul des scores entre les places et les utilisateurs, nous trions les places par ordre décroissant, nous allons maintenant extrayons les scores maximales.

iii. La recommandation des places selon les scores

Nous allons recommander à chaque utilisateur les places a partir de l'étape précédente.

II. 2ème parties :La recommandation des places pour les nouveaux utilisateurs

Dans notre cas nous allons recommander à un nouvel utilisateur des places selon son **genre**, ses **préférences** et sa **wilaya**. Nous allons calculer le score du nouvel utilisateur

et le comparer avec tous les utilisateurs de notre base de données . Ce calcul s'appuie sur 5 points principaux : (Préférence1, Préférence2, Préférence3, Genre, Localisation) Chaque point prend 20%.

i. le calcul des scores

Le principe est de calculer le score du nouvel utilisateur U avec tous les utilisateurs existants, par exemple : si les deux utilisateurs sont des **hommes**(profil) et qu'ils ont les mêmes **préférences** et sont **proche** alors on recommande pour cet nouvel utilisateur les mêmes recommandations. **ii. Trouver le max score**

Après le calcul des scores entre le nouvel utilisateur et les anciens utilisateurs , nous allons maintenant extraire les Scores maximales entre eux.

iii. La recommandation des places selon les scores

Nous allons recommander au nouveau utilisateur des places de l'utilisateur qui a le meilleur score.

3.7.1 Application de l'algorithme CPP-s

Après avoir défini l'algorithme CPP-s Nous allons l'appliquer sur notre base de données (déjà présenté avant) afin de faire des recommandations aux utilisateurs. Les recommandations concernent les anciens utilisateurs et les nouveaux utilisateurs. Dans la première partie nous allons faire des recommandations aux anciens utilisateurs. Dans la deuxième partie nous allons recommander à un nouveau utilisateur des endroits selon son contexte.

I. 1ère parties : La recommandation des places pour les anciens utilisateurs

Nous allons appliquer l'algorithme de Score sur notre base de données.

La table 3.4 représente la table constituée de huit places, et la table 3.5 pour deux utilisateurs à qui on va recommander des places .

Id-place	l'altitude	longitude	likes	Genre	Ext-place	Type de place
20	36.449697	4.127364	65	Children	Tikjda	Montagne
18	36.380620	3.904818	66	Children	Parc De L'envi -ronement	Loisir
133	36.640488	2.694023	39	woman	jardin des pins	jardin
122	36.603594	5.491616	54	man	Parc national de Taza	Montagne
126	36.397906	3.900729	Bouira	man	Uno Shopp -ing Center Bouira	Boutique
23	36.449697	4.127364	72	Children	Tikjda	Montagne
144	36.604145	5.660354	66	woman	Ain Mchaki	Montagne
118	36.821317	5.764172	43	man	Musée Kotama - Jijel	Musée

TABLE 3.14 – La table des places

Id-user	l'altitude	longitude	Genre	Nome	Préf1	Préf2	Préf3
84	36.396591	3.998636	Children	mayssa	Montagne	Zoo	Plage
97	36,811123	5,763035	man	Moussa	Montagne	Jardin	Loisir

TABLE 3.15 – La table des utilisateurs

i. Le calcul de score Le calcul de Score pour chaque utilisateur avec les places.

Le calcul de score se base sur les 4 équations suivantes :

$$Score(Distance) \begin{cases} 25\% & \text{si } d(p,u) \leq 0.5 & (3.1) \\ 15\% & \text{si } d(p,u) \leq 0.5 & (3.2) \\ 5\% & \text{sinon} & (3.3) \end{cases}$$

$$Score(Likes) \begin{cases} 25\% & \text{si } \text{nbr-likes} \geq 60 & (3.4) \\ 15\% & \text{si } \text{nbr-likes} \in [40, 60] & (3.5) \\ 5\% & \text{sinon} & (3.6) \end{cases}$$

$$Score(Préférence) \begin{cases} 25\% & \text{si } P1 = \text{type de place} & (3.7) \\ 15\% & \text{si } P2 = \text{type de place} & (3.8) \\ 5\% & \text{sinon} & (3.9) \end{cases}$$

$$Score(Genre) \begin{cases} 25\% & \text{si même} & (3.10) \\ 0\% & \text{sinon} & (3.11) \end{cases}$$

A- Exemple 1

Nous allons recommander à l'utilisateur(id-user=85) les top-N places . Table 3.16 représente les places avec les score obtenus.

Id-place	l'altitude	longitude	likes	Genre	Ext-place	Type de place	score
20	36.449697	4.127364	65	Children	Tikjda	Montagne	100%
18	36.380620	3.904818	66	Children	Parc De L'envi- ronement	Loisir	57%
133	36.640488	2.694023	39	woman	jardin des pins	jardin	20%
122	36.603594	5.491616	54	man	Parc national de Taza	Montagne	65%
126	36.397906	3.900729	Bouira	man	Uno Shopp- ing Center Bouira	Boutique	50%
23	36.449697	4.127364	72	Children	Tikjda	Montagne	100%
144	36.604145	5.660354	66	woman	Ain Mchaki	Montagne	55%
118	36.821317	5.764172	43	man	Musée Kotama - Jijel	Musée	20%

TABLE 3.16 – La table des place + Score (CPP-s)

ii. Trouver le score max

Après le calcul des scores entre les places et les utilisateurs, nous allons maintenant extraire les scores maximales (voir la table 3.16).

iii. la recommandation des places selon les scores

On suppose qu'on recommande 2 places à l'utilisateur, d'après la table 3.16 on a :

$$\text{Score}(\text{place-20}/\text{user-85})=100\%$$

$$\text{Score}(\text{place-18}/\text{user-85})=57\%$$

Donc on recommande les places suivantes :

- Tikjda
- Parc De l'envirement

B- Exemple 2

Nous allons recommander à l'utilisateur(id-user=97) les top-N places . La table 3.17 représente les places et les scores obtenus :

Id-place	l'altitude	longitude	likes	Genre	Ext-place	Type de place	score
20	36.449697	4.127364	65	Children	Tikjda	Montagne	55%
18	36.380620	3.904818	66	Children	Parc De L'envi -ronement	Loisir	35%
133	36.640488	2.694023	39	woman	jardin des pins	jardin	25%
122	36.603594	5.491616	54	man	Parc national de Taza	Montagne	100%
126	36.397906	3.900729	Bouira	man	Uno Shopp -ing Center Bouira	Boutique	55%
23	36.449697	4.127364	72	Children	Tikjda	Montagne	55%
144	36.604145	5.660354	66	woman	Ain Mchaki	Montagne	75%
118	36.821317	5.764172	43	man	Musée Kotama - Jijel	Musée	90%

TABLE 3.17 – la base d'apprentissage normalisée CPP-s(place)

ii. Trouver le score max

Après le calcul des scores entre les places et les utilisateurs, nous allons maintenant extraire les scores maximales(voir la table 3.17).

iii. La recommandation des places selon les scores

On suppose qu'on recommande 2 places à l'utilisateur, donc d'après la Table 3.17 on a :

$$\text{Score}(\text{place-122}/\text{user-97})=100\%$$

$$\text{Score}(\text{place-118}/\text{user-97})=90\%$$

Donc on recommande les places suivantes :

- Parc national de Taza
- Musée Kotama

II. 2 ème parties : La recommandation des places pour les nouveaux utilisateurs

Nous allons appliqué l'algorithme CPP-s sur notre base de données.

La table 3.18 représente la table des utilisateurs constituée de sept utilisateurs , et la table 3.19 pour un utilisateur à qui on va recommander des places .

Id-User	Wilaya	Genre	P1	P2	P3
0	Bouira	man	Musée	Boutique	Plage
77	Alger	Children	Jardin	Zoo	Loisir
26	Bouira	Children	Musée	Zoo	Plage
63	Jijel	Children	Montagne	Zoo	Plage
58	Alger	woman	Restaurant	Jardin	Loisir
70	Alger	Children	Loisir	Zoo	Plage
42	Alger	Children	Loisir	Plage	Autre

TABLE 3.18 – la table des utilisateurs (user)

Id-User	Wilaya	Genre	P1	P2	P3	Class
100	Alger	Children	Zoo	Loisir	Plage	!

TABLE 3.19 – la table (new user)

i. Le calcul de score

Le calcul de score pour le nouvel utilisateur avec les anciens utilisateurs.

La calcule de score se basé sur les cinq équations suivante :

$$Score(Préférence1) \begin{cases} 20\% & \text{si meme Préférence 1} \\ 0\% & \text{sinon} \end{cases} \quad (3.12)$$

$$(3.13)$$

$$Score(Préférence2) \begin{cases} 20\% & \text{si meme Préférence 2} \\ 0\% & \text{sinon} \end{cases} \quad \begin{matrix} (3.14) \\ (3.15) \end{matrix}$$

$$Score(Préférence3) \begin{cases} 20\% & \text{si meme Préférence 3} \\ 0\% & \text{sinon} \end{cases} \quad \begin{matrix} (3.16) \\ (3.17) \end{matrix}$$

$$Score(Genre) \begin{cases} 20\% & \text{si meme Genre} \\ 0\% & \text{sinon} \end{cases} \quad \begin{matrix} (3.18) \\ (3.19) \end{matrix}$$

$$Score(Wilaya) \begin{cases} 20\% & \text{si meme Wilaya} \\ 0\% & \text{sinon} \end{cases} \quad \begin{matrix} (3.20) \\ (3.21) \end{matrix}$$

ii. Trouver le score max

Après le calcul des scores entre le nouvel utilisateur et les anciens utilisateurs, nous allons maintenant extraire les scores maximales(voir la table 3.20).

Id-User	Wilaya	Genre	P1	P2	P3	Similarité
0	Bouira	man	Musée	Boutique	Plage	20%
77	Alger	Children	Jardin	Zoo	Loisir	40%
26	Bouira	Children	Musée	Zoo	Plage	40%
63	Jijel	Children	Montagne	Zoo	Plage	40%
58	Alger	woman	Restaurant	Jardin	Loisir	40%
70	Alger	Children	Loisir	Zoo	Plage	60%
42	Alger	Children	Loisir	Plage	Autre	40%

TABLE 3.20 – la table des utilisateurs + Score CPP-s (user)

iii. La recommandation des places selon les scores

Nous allons recommander au nouveau utilisateur les places de l'utilisateur qui a le meilleur score. D'après l'étape précédente on a :

$Score_{MAX}(\text{id-user}=100, \text{id-user}=70) = 60\%$. Vu que l'utilisateur (id-user = 70) a le meilleur score, alors on recommande les places de l'utilisateur(id-user=70) a le nouvel utilisateur (id-user=100).

3.8 Conclusion

Dans cette partie, nous avons présenté deux approche se recommandation.

Nous avons bien détaillé les étapes de chacune des approches, nous avons crée un modèle à l'aide d'apprentissage automatique avec KPP-v, et un notre modèle que nous avons proposé. Nous avons trouvé que l'algorithme de Score est mieux adapté à notre situation et donne de meilleur résultat.

Dans le chapitre suivant nous allons prouver qui parmi les deux algorithmes est le meilleur, en calculant le rappel, la précision, et F-mesure.

Expérimentation et Test

4.1 Introduction

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

Ce chapitre est consacré à la partie expérimentation et réalisation de notre application, qui consiste à recommander des endroits aux utilisateurs en utilisant le systèmes de recommandation basé sur le filtrage collaboratif. Nous présentons, également l'architecture de notre système, l'environnement de travail choisi tout en présentant les langages et les outils utilisés ainsi que les captures d'écran de chaque étape.

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

4.2 Objectif de l'expérimentation

L'expérimentation d'un système de recommandation nécessite de détenir un ensemble de données permettant d'évaluer ses performances. Cependant, de tels ensembles de données ne sont pas disponibles pour tous les domaines. Nous avons expérimenté le système de recommandation défini dans le domaine du tourisme. Nous détaillons dans cette partie les différentes étapes de l'expérimentation et nous analysons les résultats obtenus, dans l'objectif de faire une nouvelle méthode de recommandation en lien avec les algorithmes de l'apprentissage automatique[64].

4.3 Environnement informatique utilisé

afin de réaliser notre objectif nous avons travaillé sur un matériel et des logiciels :

4.3.1 Environnement matériel

L'application de ces approches est faite avec un ordinateur "TOSHIBA" ayant 4 GO de mémoire RAM , 2.0 GHz vitesse de processeur, CPU i3 et carte graphique intel HD 5500.

4.3.2 Environnement logiciel

Chaque projet d'informatique requiert plusieurs ressources et outils car chaque ressource est exploitée dans une tâche bien déterminée, pour cela nous présentons les différents langages de programmation et les outils utilisés dans notre projet.

Python

Dans notre projet nous utilisons la version de python 3.5.0 qui est sortie le 13 Septembre 2015 et ce magnifique puissant langage est classé le 3ème meilleur langage de programmation au monde d'après le classement Tiobe qui se base sur les données de consultation issues des moteurs de recherche.

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

Ce langage fonctionne sur tous les principaux systèmes d'exploitation [59].

Bibliothèques de Python utilisées

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

Dans la Table 4.1 nous allons citer les différentes bibliothèques de Python utilisées dans notre projet :

L'installation se fait par l'utilisation de la boîte de commande CMD directement par l'accès au dossier où se trouve Python et taper la commande suivante "python -m pip install nom-de-bibliothèque".

La bibliothèque	Son but
Xlrd	lire un fichier Excel.
Xlwt	écrire dans un fichier Excel
matplotlib.pyplot	pour faire les différents diagrammes qui explique les résultats de notre travail.
Math	pour utiliser des fonctions de calcul prédéfini.
Pandas	utiliser pour les structures de données hautes performances et faciles à utiliser, ainsi que des outils d'analyse des données.

TABLE 4.1 – Les bibliothèques de Python utilisées dans notre projet

4.4 Analyse des résultats

Afin d'évaluer notre système de recommandation, nous avons construit un modèle avec le K plus proche voisins et l'algorithme de score dans le but de faire une comparaison entre eux et savoir qui est le plus performant.

4.4.1 Le résultat de l'algorithme KPP-v

Nous avons utilisé l'algorithme KPP-v pour construire un système de recommandation collaboratif qui permet de recommander des places aux utilisateurs.

Dans un premier temps nous avons recommandé dix places pour chaque utilisateur (K=10) de notre base selon son emplacement, son genre, et ses préférences.

Dans un deuxième temps, nous avons recommandé dix places pour un nouvel utilisateur, cet utilisateur doit saisir ses informations, Après la saisie le système extrait le profil le plus similaire et recommande les dix places préférées de l'ancien utilisateur au nouvel utilisateur.

La Figure(4.1) représente les 10 top places (nom de place, type de place et la distance trouver entre la place et l'utilisateur) recommandé pour les utilisateurs selon l'algorithme KPP-v.

```

***** L'algorithmme KPP-v *****

la recommandation des places pour les utilisateurs

donner le Nb de reccomondation
10
user = 0
0 -----> Musée National du Bardo ( Musée ) distance ---> 0.661
1 -----> Musée national des antiquités et des arts islamiques ( Musée ) distance ---> 0.705
2 -----> Place des Martyrs ( Musée ) distance ---> 0.708
3 -----> Dar Aziza ( Musée ) distance ---> 0.877
4 -----> Mosquée Ketchaoua ( Musée ) distance ---> 0.878
5 -----> Musée National des Arts et Traditions Populaires ( Musée ) distance ---> 0.879
6 -----> Basilique Notre-Dame d'Afrique Alger ( Musée ) distance ---> 0.906
7 -----> Galeries art ( Musée ) distance ---> 0.907
8 -----> Marka shoese ( Boutique ) distance ---> 0.946
9 -----> Magasin De Sport ( Boutique ) distance ---> 0.947
user = 1
0 -----> Musée National du Bardo ( Musée ) distance ---> 0.664
1 -----> Musée national des antiquités et des arts islamiques ( Musée ) distance ---> 0.708
2 -----> Place des Martyrs ( Musée ) distance ---> 0.711
3 -----> Dar Aziza ( Musée ) distance ---> 0.88
4 -----> Mosquée Ketchaoua ( Musée ) distance ---> 0.881
5 -----> Musée National des Arts et Traditions Populaires ( Musée ) distance ---> 0.882
6 -----> Basilique Notre-Dame d'Afrique Alger ( Musée ) distance ---> 0.909
7 -----> Galeries art ( Musée ) distance ---> 0.91
8 -----> Marka shoese ( Boutique ) distance ---> 0.949
9 -----> Magasin De Sport ( Boutique ) distance ---> 0.95
user = 2
0 -----> Musée national des antiquités et des arts islamiques ( Musée ) distance ---> 0.009
1 -----> Dar Aziza ( Musée ) distance ---> 0.009
2 -----> Musée National des Arts et Traditions Populaires ( Musée ) distance ---> 0.01
3 -----> Musée Kotama - Jijel ( Musée ) distance ---> 0.026
4 -----> Alambra Alger ( Restaurant ) distance ---> 0.027
5 -----> Sanctuaire des Martyrs ( Musée ) distance ---> 0.069
6 -----> Musée du martyr ( Musée ) distance ---> 0.069
7 -----> Musée des Beaux Arts ( Musée ) distance ---> 0.07

```

FIGURE 4.1 – Les top 10 places recommandées pour les utilisateurs selon le KPP-v

La Figure(4.2) représente les 10 top places (nom de place, type de place) recommandé pour le nouvel utilisateur selon l’algorithmme KPP-v.

```

***** L'algorithme KPP-v *****

la recommandation des places pour le nouvel utilisateur

donner les information du nouvel USER
donner LAT:36.255421
donner LOT :3.025478
donner PRE 1 :Musée
donner PRE 2 :Zoo
donner PRE 3 :Jardin
donner GENRE :man
donner le Nb de reccomondation
10
l'utilisateur le plus proche est : 22
1 ----> Musée national des antiquités et des arts islamiques(Musée)
2 ----> Dar Aziza(Musée)
3 ----> Musée National des Arts et Traditions Populaires(Musée)
4 ----> Musée Kotama - Jijel(Musée)
5 ----> UNO De Bab Ezzouar(Boutique)
6 ----> Lion CrossFit(Loisir )
7 ----> Ardis Mall(Boutique)
8 ----> Opirate aquapark Ardis(Boutique)
9 ----> Parc Zoologique de Ben Aknoun(Zoo)
10 ----> Stade caroubier (la base)(Loisir )
>>>

```

FIGURE 4.2 – Les top 10 places recommandées pour le nouvel utilisateur selon le KPP-v

4.4.2 résultat de l'algorithme CPP-s

De même pour l'algorithme de Score on a d'abord recommandé dix places a chaque utilisateur.

La Figure(4.1) représente les 10 top places (nom de place, type de place et score trouver entre les utilisateurs et le nouvel utilisateur) recommandé pour les utilisateurs selon l'algorithme CPP-s.


```

***** l'algorithme de Score *****

la recommandation des places pour les utilisateurs

voila les top 10 places pour l'utilisateurs 0
1 ---> Mosquée Ketchaoua( Musée ) score ---> 0.9
2 ---> Ruines Romaines De Tipaza( Musée ) score ---> 0.9
3 ---> Uno Shopping Center Bouira( Boutique ) score ---> 0.9
4 ---> Musée National des Arts et Traditions Populaires( Musée ) score ---> 0.8
5 ---> Place des Martyrs( Musée ) score ---> 0.8
6 ---> Tikjda( Montagne ) score ---> 0.75
7 ---> Ardis Mall( Boutique ) score ---> 0.7
8 ---> UNO De Bab Ezzouar( Boutique ) score ---> 0.7
9 ---> Magasin De Sport ( Boutique ) score ---> 0.7
10 ---> Musée Kotama - Jijel( Musée ) score ---> 0.7
voila les top 10 places pour l'utilisateurs 1
1 ---> Mosquée Ketchaoua( Musée ) score ---> 0.9
2 ---> Ruines Romaines De Tipaza( Musée ) score ---> 0.9
3 ---> Place des Martyrs( Musée ) score ---> 0.8
4 ---> Musée National du Bardo( Musée ) score ---> 0.8
5 ---> Jardin Botanique du Hamma/jardin d'essai( Jardin ) score ---> 0.8
6 ---> Tikjda( Montagne ) score ---> 0.75
7 ---> Uno Shopping Center Bouira( Boutique ) score ---> 0.75
8 ---> Dar Aziza( Musée ) score ---> 0.7
9 ---> Musée Kotama - Jijel( Musée ) score ---> 0.7
10 ---> Musée du martyr( Musée ) score ---> 0.65
voila les top 10 places pour l'utilisateurs 2
1 ---> Mosquée Ketchaoua( Musée ) score ---> 1.0
2 ---> Jardin Botanique du Hamma/jardin d'essai( Jardin ) score ---> 0.9
3 ---> Place des Martyrs( Musée ) score ---> 0.9
4 ---> Musée National du Bardo( Musée ) score ---> 0.9
5 ---> Ruines Romaines De Tipaza( Musée ) score ---> 0.9
6 ---> Maison d'art et déco( Boutique ) score ---> 0.8

```

FIGURE 4.3 – Les top 10 place recommandées pour les utilisateurs selon l'algorithme CPP-s

En suite, on a passé a recommandé dix places a le nouveau utilisateur a partir des recommandation des utilisateurs précédents.

La Figure(4.4) représente les 10 top places (nom de place, type de place et la distance trouver entre la place et l'utilisateur) recommandé pour le nouvel utilisateur selon l'algorithme CPP-s.

```

***** l'algorithme de Score *****

la recommandation des places pour le nouvel utilisateur

donner le genre :man
donner pref 1 :Musée
donner pref 2 :Zoo
donner pref 3 :Jardin
donner la wilaya :Alger
le user le plus proche 21

Voila les 10 places recommander pour le NEW USER
1 ----> Mosquée Ketchaoua( Musée )
2 ----> Sanctuaire des Martyrs( Musée )
3 ----> Musée des Beaux Arts( Musée )
4 ----> Parc Zoologique de Ben Aknoun( Zoo )
5 ----> Musée National des Arts et Traditions Populaires( Musée )
6 ----> Mausolée royal de Maurétanie (tombeau de la chrétienne)( Musée )
7 ----> Musée du martyr( Musée )
8 ----> Jardin Botanique du Hamma/jardin d'essai( Jardin )
9 ----> Basilique Notre-Dame d'Afrique Alger( Musée )
10 ----> Plage El Kettani( Plage )
>>>

```

FIGURE 4.4 – Les top 10 places recommandées pour le nouvel utilisateur selon l’algorithme CPP-s

4.5 Métrique dévaluation

Lorsqu’un système de recommandation est développé, il est important d’être en mesure d’évaluer son fonctionnement et sa capacité à répondre aux objectifs qui lui ont été fixés.

Le but de nos expérimentations est de tester la pertinence des préférences de l’utilisateur dans la recommandation. Pour cela nous avons utilisé deux mesures, **la précision** et **le rappel**.

Précision

La précision qui est la mesure la plus utilisée pour évaluer la pertinence des recommandations. La précision est calculée sur la base de l’équation (4.1) comme étant le rapport entre le nombre d’items pertinents sélectionnés N_{ps} et le nombre d’items sélectionnés par un utilisateur N_s .

$$P = \frac{N_{ps}}{N_s} \quad (4.1)$$

La précision générale du système de recommandation correspond ainsi à la moyenne des précisions calculées pour chaque utilisateur. Plus cette précision est élevée, plus le système de recommandation est performant.

Rappel

Le rappel mesure la probabilité qu'un item pertinent soit sélectionné par l'utilisateur. Il est calculé sur la base de l'équation (4.2) comme étant le ratio entre le nombre d'items pertinents sélectionnés par l'utilisateur N_{ps} et le nombre total d'items pertinents disponibles N_p .

$$R = \frac{N_{ps}}{N_p} \quad (4.2)$$

Comme pour la précision, le rappel relatif à la totalité du système est évalué comme étant la moyenne des rappels calculés individuellement.

Il existe une mesure combinant la précision et le rappel. Il s'agit de la mesure " F_1 ". Elle représente la moyenne harmonique entre la précision et le rappel, suivant l'équation (4.3). La valeur de F_1 varie de 0 à 1. Lorsque les scores de précision et de rappel sont équivalents, la qualité des recommandations est considérée comme parfaite[68].

$$F_1 = \frac{2P * R}{P + R} \quad (4.3)$$

4.6 Expérimentations et résultats

Nous avons comparé les résultats de la recommandation des places via les méthodes, KPP-v et notre méthode proposée. Pour évaluer la recommandation, nous utilisons les mesures classiques en recherche d'informations cités ci-dessus, précision, rappel et F_1 -mesure. Dans notre cas, l'ensemble de données permettant l'évaluation sont les préférences, l'utilisateur possède 3 types de places préférés et le système devra recommander dix places à partir de ces types. On considère que l'algorithme est performant si les recommandations appartiennent à ses préférences.

En appliquant l'algorithme KPP-v sur un ensemble d'utilisateurs, prenant en considération ses trois préférences.

4.6.1 Le calcul de Rappel, Précision et F-mesure de l'algorithme KPP-v

On obtient le tableau ci-dessous qui représente le résultat de recommandations des 10 utilisateurs (voir le Table 4.2).

KPP	Rec1	Rec2	Rec3	Autre
User1	8	0	0	1
User2	0	3	1	5
User3	2	2	2	3
User4	2	0	0	7
User5	0	1	1	7
User6	4	3	0	3
User7	2	0	0	8
User8	3	0	0	7
User9	2	3	0	5
User10	1	0	0	9
Moyenne	2.7	1.3	0.4	5.6

TABLE 4.2 – La moyenne de recommandation selon algorithme KPP-v

En moyenne, cet algorithme recommande 2.7 de placse de la première préférence (Rec1), 1.3 de la deuxième préférence (Rec2) et 0,4 de la troisième préférence(Rec3) et recommande 5.6 (Autre)de places qui ne figure pas dans ses préférences ce qui signifie que 56% de recommandations ne sont pas dans ses préférences (voir la Figure 4.5).

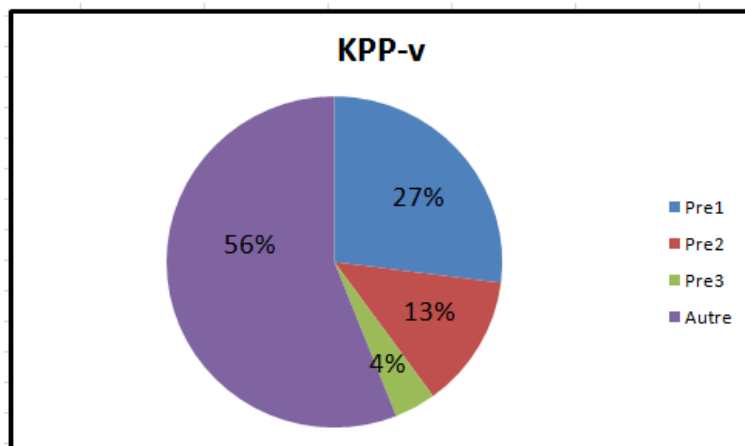


FIGURE 4.5 – Trigonométrie des pourcentages de recommandation de l’algorithme KPP-v

Maintenant nous allons passé a calculer le rappel, la précision et F-mesure de la moyenne des 10 utilisateurs on obtient le résultat dans le tableau suivant (voir le Table 4.3) :

Exemple de calcule le rappel de rec1 selon 10 utilisateurs :

Dans cet exemple nous allons montrée comment calculer le rappel de rec1(recommandation 1) selon 10 utilisateurs. Ce calcule se base sur la moyenne des rappels de 10 utilisateurs. Nous allons d’abord montrer un exemple de calculer le rappel d’un seul utilisateur. Nous supposons que l’utilisateur 1 contient comme préférences, Boutique, Jardin et Montagne, sa recommandation est 8 Boutique, 1 Musée et 1 Restaurant et sa lère recommandation est Boutique : alors $\text{rappel}(\text{user1}, \text{rec1}) = \frac{1}{8} = 0.13$

Après avoir calculé le rappel de chaque utilisateur, nous calculons la moyenne des rappels de tous les utilisateurs.

$$\text{moyenne}(10 \text{ user}, \text{rec1}) = \text{rappel}(\text{user1}, \text{rec1}) + \text{rappel}(\text{user2}, \text{rec1}) + \text{rappel}(\text{user}, \text{rec1}) + \dots + \text{rappel}(\text{user10}, \text{rec1})$$

$$\text{rappel}(\text{user1}, \text{rec1}) = \frac{(0.13 + 0.25 + 0.25 + 0.13 + 0.33 + 0.11 + 0 + 0.33 + 0.2 + 0.1 + 0)}{10}$$

	Rappel	Précision	F-Mesure
Rec1	0.18	<u>0.80</u>	0.29
Rec2	0.35	0.80	0.47
Rec3	0.50	0.77	0.59
Rec4	0.54	0.65	0.57
Rec5	0.56	0.56	0.54
Rec6	0.62	0.52	0.54
Rec7	0.64	0.46	0.51
Rec8	0.76	0.47	0.55
Rec9	0.87	0.47	0.58
Rec10	<u>0.99</u>	0.29	<u>0.60</u>

TABLE 4.3 – Tableau de rappel, précision et f-mesure de l’algorithme KPP-V

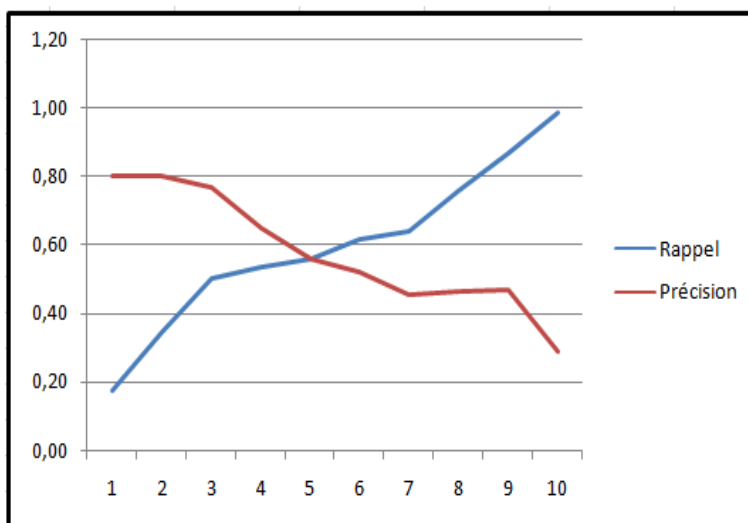


FIGURE 4.6 – Graphe de Rappel-Précision de l’algorithme KPP-v

La figure ci-dessus représente une courbe de la précision qui est entre 0.80 de valeur maximale et 0.29 valeur minimale, et une autre courbe de rappel qui est entre 0.18 et 0.99 (voir la Figure 4.7).

4.6.2 Le calcul de Rappel, Précision et F-mesure de l'algorithme CPP-s

En appliquant l'algorithme CPP-s sur un ensemble d'utilisateurs, prenant en considération ses trois préférences, on obtient le tableau ci-dessous qui représente le résultat de recommandations des 10 utilisateurs (voir la Table 4.4).

Score	Rec1	Rec2	Rec3	Autre
User1	7	1	1	1
User2	5	2	2	1
User3	3	2	2	3
User4	9	1	0	0
User5	6	2	2	0
User6	8	0	1	1
User7	4	0	1	5
User8	3	1	3	3
User9	1	5	0	4
User10	6	0	1	3
Moyenne	5.2	1.4	1.3	2.1

TABLE 4.4 – Nb de recommandation pour chaque utilisateur selon l'algorithme CPP-s

On se voit qu'il recommande en moyenne 5.2 de places de la première préférence ce qui signifie que 52% de recommandation dépend de la première préférence (Rec1), et que 1.4 de recommandation par la deuxième préférence (Rec2) et 1.3 de la troisième préférence (Rec3) tandis que 2.1 de recommandation n'appartient pas aux préférences (Autre) (voir la Figure 4.7).

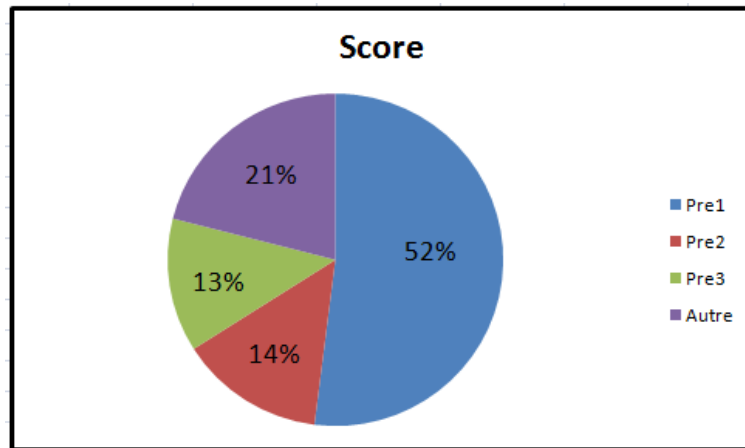


FIGURE 4.7 – Trigonométrie des pourcentages de recommandation de l'algorithme CPP-s

Après avoir calculé le rappel et la précision de chaque utilisateur, on a calculé la moyenne de ces derniers.

Voici le résultat de rappel et de précision ainsi que F-mesure de l'algorithme de Scores (voir la Table 4.5).

	Rappel	Précision	F-Mesure
Rec1	0.13	<u>1.00</u>	0.23
Rec2	0.22	0.9	0.35
Rec3	0.34	0.90	0.49
Rec4	0.43	0.85	0.57
Rec5	0.52	0.84	0.64
Rec6	0.62	0.83	0.71
Rec7	0.74	0.84	0.78
Rec8	0.81	0.83	0.81
Rec9	0.90	0.82	0.90
Rec10	<u>0.97</u>	0.79	<u>0.86</u>

TABLE 4.5 – Tableau de rappel, précision et f-mesure de l'algorithme CPP-s

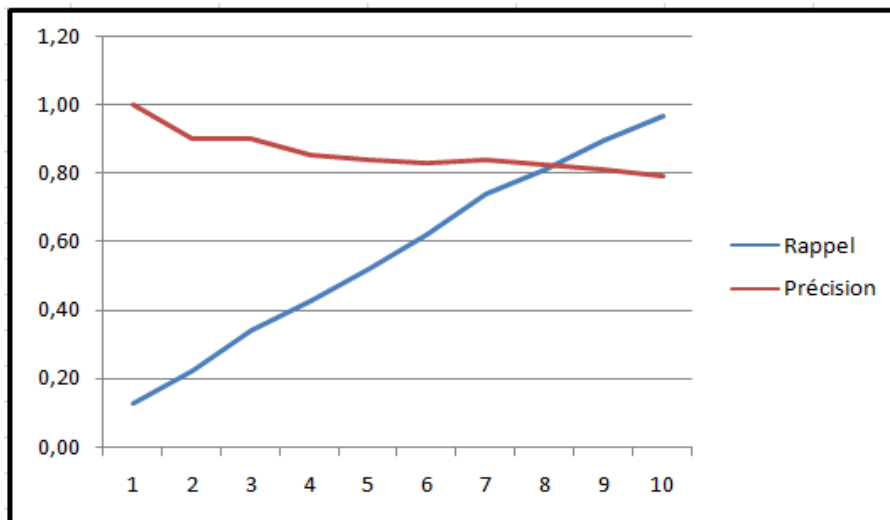


FIGURE 4.8 – Graphe de les résultats de l’algorithme Score

La figure ci-dessus représente une courbe de la précision qui est entre 1 de valeur maximale et 0.79 valeur minimale et une autre courbe de rappel qui est entre 0.13 et 0.97 (voir la Figure 4.8).

4.7 Comparaison des résultats obtenus

Nous allons passé maintenant a comparer les résultats obtenus. Nous remarquons que les résultats des deux algorithmes sont différents comme le représente le graphe suivant (voir la Figure 4.10) :

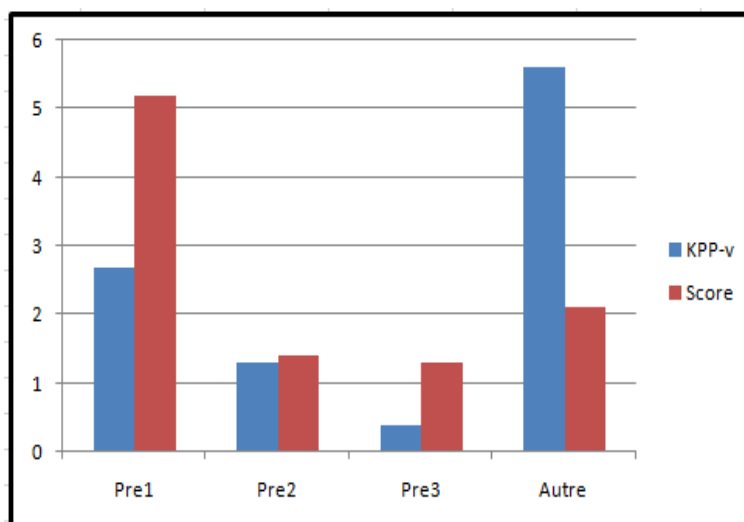


FIGURE 4.9 – Histogramme comparatif des deux algorithmes

Ce graphe montre que l'algorithme CPP-s a donné de meilleurs résultats par rapport a KPP-v, tel qu'il nous a donnée 5.2/10 de la première préférence, 1.4/10 de la deuxième préférence et 1.3/10 de la troisième préférence ce qui signifie que 7.9/10 places recommandées sont de leur préférences 79%. Contrairement a KPP-v qui nous a donnée 4.4/10 seulement des places qui appartiennent aux préférences 44%. et 5.6/10 n'appartiennent pas Se qui implique que l'algorithme de Score répond mieux a nos besoins, il fournit une liste de recommandations personnalisée a chaque utilisateur.

4.8 Partie application mobile

Une application mobile est un type de logiciel ou programme conçu pour s'exécuter sur un appareil mobile, tel qu'un smartphone ou une tablette. Les applications mobiles servent à fournir aux utilisateurs des services similaires à ceux du PC. Le développement de notre application ce fait avec un enveniment de développement "Android Studio" et des différentes langages[69].

4.8.1 Environnement de développement

Nous allons utilisé comme un environnement de développement, android studio.

Android Studio :

Android Studio est un nouvel environnement pour le développement et programmation entièrement intégré qui a été récemment lance par Google pour les systèmes Android, il a été conçu pour fournir un environnement de développement et une alternative à Eclipse qui est l'IDE le plus utilisé.

Android Studio permet de voir chacun des changements visuels que vous effectuez sur votre application et en temps réel, vous pourrez voir aussi son effet sur différents appareils Android. [69].

4.9 Langages de développement

Nous allons utiliser comme un langage de développement, JAVA, PHP, SQL, XML, JSON et quatre APIS.

4.9.1 JAVA

C'est un langage de programmation orienté objet. Il permet de créer des logiciels compatibles avec de nombreux systèmes d'exploitation (Windows, Linux, Macintosh, Solaris). Java donne aussi la possibilité de développer des programmes pour téléphones portables et assistants personnels. Enfin, ce langage peut être utilisé sur internet pour des petites applications intégrées à la page web (applet) ou encore comme langage serveur (JSP)[71].

4.9.2 PHP

PHP (Hypertext Preprocessor) est un langage de scripts généraliste et Open Source, spécialement conçu pour le développement d'applications web. Il peut être intégré facilement au HTML[72].

4.9.3 SQL

Le SQL (Structured Query Language) est un langage permettant de communiquer avec une base de données. Ce langage informatique est notamment très utilisé par les développeurs Web pour communiquer avec les données d'un site web. SQL.sh recense des cours de SQL et des explications sur les principales commandes pour lire, insérer, modifier et supprimer des données dans une base[74].

4.9.4 XML

Le XML (eXtensible Markup Language) est un langage informatique qui sert à enregistrer des données textuelles. Ce langage, grosso-modo similaire à l'HTML de par son système de balisage, permet de faciliter l'échange d'information sur l'internet. Contrairement à l'HTML qui présente un nombre fini de balises, le XML donne la possibilité de créer de nouvelles balises à volonté [75].

4.9.5 JSON

JSON (JavaScript Object Notation – Notation Objet issue de JavaScript) est un format léger d'échange de données. Il est facile à lire ou à écrire pour des humains. Il est aisément analysable ou générable par des machines. Il est basé sur un sous-ensemble du langage de programmation JavaScript [76].

4.9.6 les APIs utilisé

Dans ce qui suit, nous présentons les différentes API que nous avons utilisé :

Google Maps Android API

Google Maps Android API nous permet d'ajouter des cartes utilisant des données Google Maps à notre application. Cette API gère automatiquement l'accès aux serveurs Google Maps, le téléchargement de données, l'affichage de la carte et la réponse aux gestes effectués sur la carte [77].

API OpenWeatherMap

Le "Guide de l'API OpenWeatherMap" aide à trouver des informations utiles, des liens et des documents pour commencer à utiliser les services d'API météo sans à-coups.

OpenWeatherMap est l'un des principaux fournisseurs d'informations météorologiques numériques [78].

http request (api volley)

la volée Volley est une bibliothèque HTTP qui facilite la mise en réseau d'applications Android et, plus important encore, plus rapidement. Volley est disponible sur GitHub [?].

GSON

Gson est une bibliothèque open source développée par Google pour convertir un objet Java dans sa représentation JSON et vice versa.

Pour utiliser Gson dans une application, il faut ajouter le fichier gson-version.jar au classpath.(site <https://www.jmdoudoux.fr/java/dej/chap-gson.htm>)[76].

4.10 L'application "Tour-App"

Lors de développement de l'application nous avons ajouté des tables dans notre base de données pour avoir plus de fonctionnalités dans l'application.

Voici les tables utilisées dans notre base de données :

- **All_places** :cette table représente toutes les places disponible sur la bdd elle contient les information des places tel que le nom la géolocalisation,ect.
- **Commentaire** : cette table stocke tous les commentaires effectué par l'utilisateur, elle contient toutes les infomations du commentaire tel que le nom de l'utilisateur qui a commenté, l'article, ainsi que la date et l'heure du commentaire
- **login_admin** :c'est la table de l'administrateur.
- **prefered_place** : cette table repésente les places préféré de chaque utilisateur, elle contient les informations des places préféré.
- **token_user** : contient la liste de toutes les personnes qui ont installé cette application.
- **user_sing_up** : cette table représente les utilisateurs inscrit dans l'application.

L'objectif est de créer une application mobile dédiée au domaine du tourisme en implémentant l'approche de recommandation que nous avons proposée.

Étant donné les préférences de l'utilisateur et son contexte,l'application exécute notre algorithme de recommandation pour offrir à l'utilisateur une recommandation sous forme de liste.

Quand l'utilisateur lance l'application il est en mesure d'effectuer les actions suivantes :

- Consulter l'application en créons un compte ou sans crée compte.
- Voir toutes les wilayas disponible sa description et ses places intéressant.
- Voir la météo des wilayas et de son emplacement.
- Voir la distance entre son emplacement et une autre place choisis.
- Voir les endroits les plus aimé par les utilisateurs.
- Faire un like ou un commentaire sur les endroits.
- Ajouter une place.

4.11 les interfaces de l'application "Tour-App"

Ces interfaces sont dédiées pour les utilisateurs de l'application qui ont un compte et qui n'ont pas. après le lancement de l'application une page s'affiche, dans cette page l'utilisateur doit sélectionner le sexe, pour aider le système à choisir les endroits (voir la figure 4.10).



FIGURE 4.10 – Interface d'accueil

En cliquant sur le bouton let's start, une liste de fonctionnalités s'affiche (voir la figure 4.11).

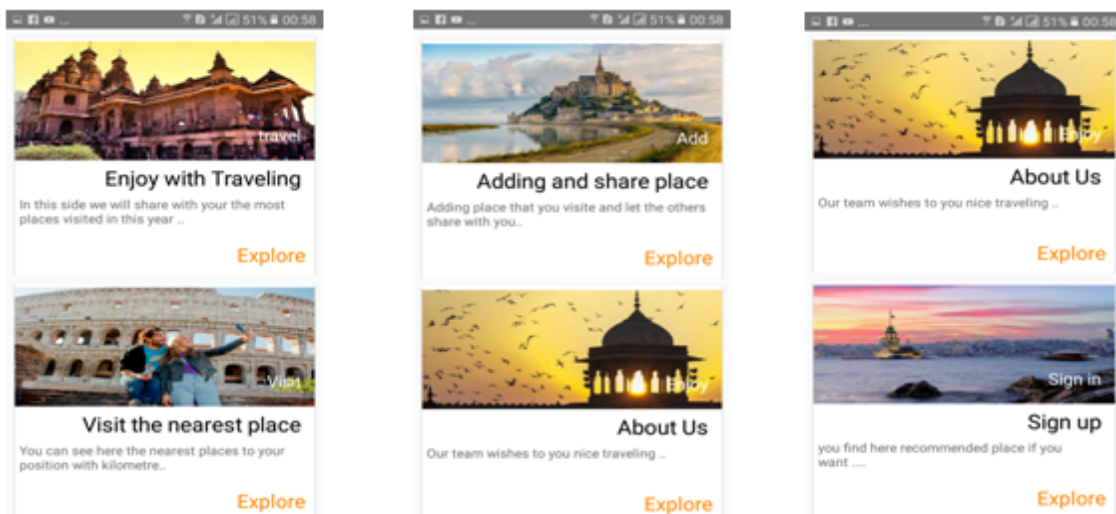


FIGURE 4.11 – Liste de fonctionnalités

4.11.1 Enjoy with traveling

cette page contient des descriptions pour chaque wilaya ainsi que sa localisation et sa météo.

4.11.2 Visit the nearest place

en cliquant dessus, une liste des wilayas apparait, ces wilayas sont trié du plus proche au plus loin. Après la sélection d'une wilaya l'utilisateur peut voir toute les places disponible et ses localisations.

le système peut recommandé aussi aux utilisateurs une liste de trois recommandations trié par le nombre de like.

4.11.3 adding and share place

l'utilisateur peut ajouté une place qui a visité et quila trouve intéressante a visité, cette place ne peut pas etre ajouter qu'après la validation de l'administrateur.

4.11.4 sing in

cette partie est dédié aux utilisateurs qui veulent s'inscrire, ils doivent remplir les champs, si il est déjà inscrit il passe a la page "Sing in" :(voir la Figure 4.12)

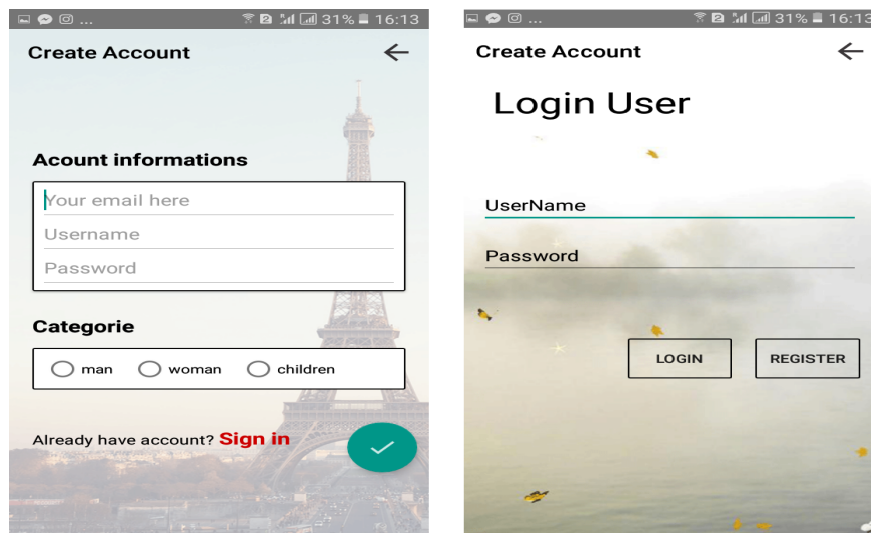


FIGURE 4.12 – Liste de fonctionnalités

Après l'authentification la page suivante apparait (voir la figure 4.13).

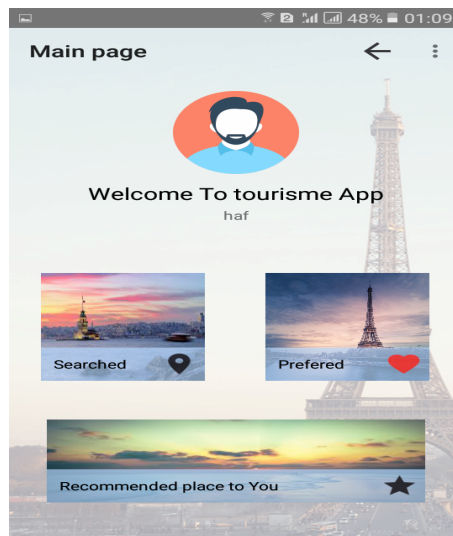


FIGURE 4.13 – Page Main

L'utilisateur a la possibilité de chercher un endroit précis(voir la Figure 4.14).

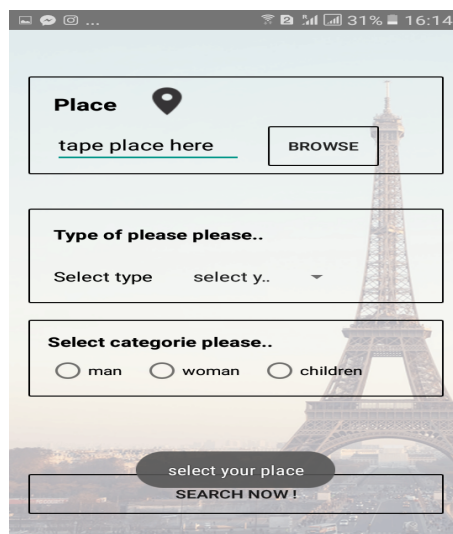


FIGURE 4.14 – Chercher endroit

L'utilisateur a la possibilité de voir ses préférences(voir la Figure 4.15).

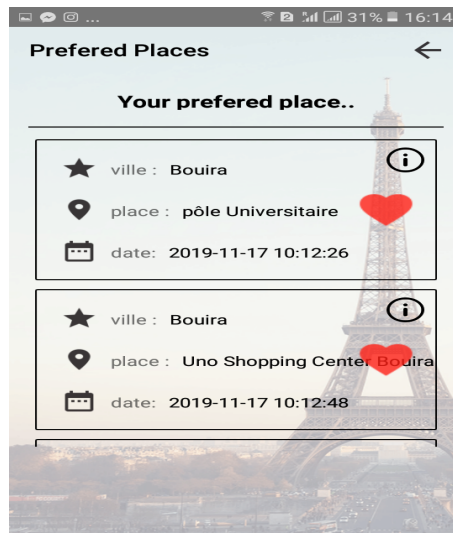


FIGURE 4.15 – Liste des préférences

En fin, il possède une liste de recommandation (recommander par les places favori et la distance)(voir la Figure 4.16).

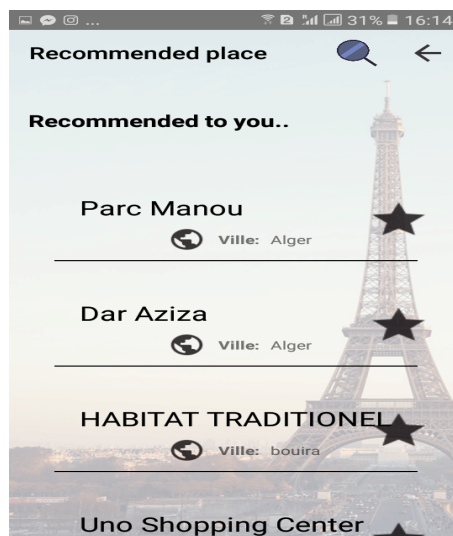


FIGURE 4.16 – Liste de recommandations

4.12 Conclusion

Nous avons présenté dans ce chapitre les technologies que nous avons utilisés pour la réalisation de l'application, ensuite nous avons testé et comparé notre approche (l'algorithme CPP-s) avec le KPP-v en terme de précision, rappel et de compromis entre précision et rappel, nous avons montré que notre approche est compétitive. Finalement nous avons terminé par illustrer les fonctionnalités de l'application et ces interfaces

Conclusion générale et Perspective

Dans notre travail, nous nous sommes intéressés aux systèmes d'aide à la visite touristiques. Notre but était de concevoir un système de recommandation sensible au contexte, implémenté sur dispositif mobiles, afin d'améliorer l'expérience du visiteur. Nous nous sommes focalisé sur les systèmes de recommandation dans le but de personnaliser les visites, afin d'offrir une visite sur mesure pour chaque visiteur. Nous avons proposé une approche de recommandation collaborative et sensible au contexte. Nous avons formellement défini le problème et nous avons travaillé sur l'algorithme KPP-v et notre algorithme proposé qu'on a appelé " CPP-s " pour construire une liste de recommandation.

Nous avons étalé en détail le processus de recommandation et ses approches, pareille pour l'apprentissage automatique, nous avons présenté une illustration des modèles que nous avons créés, et nous avons effectué une évaluation pour tester les modèles.

Cette évaluation a démontré la qualité de notre approche et sa capacité à améliorer à la fois la précision et la diversité des recommandations.

Perspective

- Nous pouvons améliorer notre système de recommandation en utilisant une base de données plus grande qui contient plus de places et plus d'utilisateurs.
- Nous désirons recommander un parcours au lieu de recommander une liste des places.
- Travailler plus sur l'IoT, par exemple mettre en considération la route et la météo par conséquent, la recommandation change.
- Intégrer dans notre application les services clientèle (par exemple Radio taxi, Yassir, ect.)

Bibliographie

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

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

- [3] Resnick, P., Iacovou, N., Suchak, M., Bergstrom, P., and Riedl, J. (1994). Grouplens : an open architecture for collaborative filtering of net news. In *Proceedings of the 1994 ACM conference on Computer supported cooperative work*, pages 175–186. ACM.

- [4] Shardanand, U. and Maes, P. (1995a). Social information filtering : algorithms for automating? word of mouth? In *Proceedings of the SIGCHI conference on Human factors in computing systems*, pages 210–217. ACM Press/Addison-Wesley Publishing Co.

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

- [6] [Burke, R. (2002). Hybrid recommender systems : Survey and experiments. *User Modeling and User-Adapted Interaction*, 12(4) :331–370.

- [7] A. T. NGUYEN. COCoFil2 : Un nouveau système de filtrage collaboratif basé sur le modèle des espaces de communautés. Thèse. Université Joseph Fourier Grenoble I. Novembre 2006.
- [8] Margaritis, K. and Vozalis, E. (2003). Analysis of recommender systems' algorithms. In 6th Hellenic European Conference on Computer Mathematics its Applications (HERCMA), Athens, Greece.
- [9] Adomavicius, G. and Tuzhilin, A. (2005). Toward the next generation of recommender systems : A survey of the state-of-the-art and possible extensions. *IEEE Transactions on Knowledge and Data Engineering*, 17(6) :734–749.
- [10] Bouchindhomme and Rochlitz (1992). Dans le flou artistique. *Éléments d'une théorie de la "rationalité esthétique"*. page 203–238.
- [11] Piamrat, K., Viho, C., Bonnin, J.-M., and Ksentini, A. (2009). Quality of experience measurements for video streaming over wireless networks. In *Sixth International Conference on Information Technology New Generations, 2009.ITNG '09*, page 1184 –1189.
- [12] Belloui, A. (2008). L'usage des concepts du web sémantique dans le filtrage d'information collaboratif. Master's thesis, Institut National d'Informatique d'Alger.
- [13] Adomavicius, G. and Tuzhilin, A. (2005). Toward the next generation of recommender systems : A survey of the state-of-the-art and possible extensions. *IEEE Transactions on Knowledge and Data Engineering*, 17(6) :734–749.
- [14] Pazzani, M. and Billsus, D. (2007). The Adaptive Web, chapter Content-Based Recommendation Systems, page 325–341. *Lecture Notes in Computer Science*.

- [15] Resnick, P. and Varian, H. (1997). Recommender systems. In *Communications of the ACM*, volume 40, pages 56–58.
- [16] Sarwar, B., Karypis, G., Konstan, J., and Riedl, J. (2001). Item-based collaborative filtering recommendation algorithms. In *WWW10*, pages 285–295, Hong Kong. ACM.
- [17] Arnautu, O. R. (2012). Mures : Un système de recommandation de musique. Master’s thesis, La Faculté des arts et des sciences Université de Montréal.
- [18] Nguyen, A. T. (2006). COCoFil2 : Un nouveau système de filtrage collaboratif basé sur le modèle des espaces de communautés. PhD thesis, université Joseph Fourier-Grenoble I.
- [19] Arnautu, O. R. (2012). Mures : Un système de recommandation de musique. Master’s thesis, La Faculté des arts et des sciences Université de Montréal.
- [20] Lémdani, Roza. (2016). Thèse : Système Hybride d’Adaptation dans les Systèmes de Recommandation, Université Paris-Saclay préparée à centrale Supélec, école doctorale N580, France.
- [21] Idir Benouaret.(2017) Un système de recommandation contextuel et composite pour la visite personnalisée de sites culturels. Thèse de doctorat Université de Technologie de Compiègne,France.
- [22] Roy, Jean-Francis. (2018). Thèse : Apprentissage automatique avec garanties de généralisation à l’aide de méthodes d’ensemble maximisant le désaccord, Université Laval, Quebec, Canada.
- [23] Larochelle, Hugo. (2008). Thèse : Étude de techniques d’apprentissage non-supervisé pour l’amélioration de l’entraînement supervisé de modèles connexionnistes, Faculté

des études supérieures et postdoctorales, Université de Montréal, Canada.

- [24] Bertail, Patrice. Bounie, David. Cléménçon, Stephan, et Patrick Waelbroeck (Février 2019).
Algorithmes : biais, discrimination et équité, Télécom ParisTech, la fondation ABEONA, Université Paris Nanterre, Paris, France.
- [25] Garivier, Aurélien. (12 et 13 juin 2013). Support de cour : Apprentissage supervisé.
- [26] Barlow, H. (1989). Finding Minimum Entropy Codes, Physiological Laboratory, Downing Street, Cambridge, CB2 3EG, England.
- [27] Bankman, Daniel. Et Moons, Bert. Verhelst, Marian. (2019). Embedded deep learning Algorithms, Architectures and Circuits for Always-on Neural Network Processing, Springer Nature Switzerland AG 2019, Library of Congress Control Number : 2018956280.
- [28] I.Wanner.(Novembre 2019). Notion du Contexte.
<http://contexte.lip6.fr/CxG/index2.php?Contexte>.
- [29] A.BOURAMOUL.(Septembre 2011). Recherche d'information contextuelle et sémantique Sur le web. Thèse de doctorat, Université Mentouri Constantine.
- [30] J.Pierson.(Décembre 2009). Une infrastructure de gestion de l'information de contexte pour l'intelligence ambiante. Thèse de doctorat, Université Joseph Fourier - Grenoble 1, France.
- [31] G.Rey. (aout 2010).Modélisation du contexte et Adaptation dans CONTINUUM. Journal of documentation.

- [32] M.Miraoui.(juillet 2009). Architecture logicielle pour l'informatique diffuse : Modélisation du contexte et adaptation dynamique des services. thèse de doctorat, Ecole de Technologie Supérieure, Université du Québec.
- [33] B.Schilit. N.Adams et al. (1994). Context-aware computing applications. In Proceedings of the IEEE Workshop on Mobile Computing Systems and Applications, pages 85-90,Santa Cruz.
- [34] P.Brown. (1996) .The Stick-e Document : a Framework for Creating Context-Aware Applications.In : Proceedings of Electronic Publishing. Page 259-272.
- [35] N.Ryan. J.Pascoe et al.(1997). Enhanced Reality Fieldwork : The Context-aware Archaeological Assistant. In Computer Applications in Archaeology. Oxford, UK.
- [36] J.Pascoe.(October 1998). Adding generic contextual capabilities to wearable computers. In Proceedings of International Symposium on Wearable Computers, page. 92-99.
- [37] K.Dey. D.Salber et al. An Architecture To Support Context-Aware Applications. GVU Technical Report GIT-GVU-99-23. Submitted to the 12th Annual.
- [38] T.Winograd.(2001).Architectures for Context. Human-Computer Interaction Journal, page. 401-419.
- [39] K.Henricksen. J.Indulska et al.(2002). Modeling context information in pervasive computing systems. In Proceedings of the 1st International Conference on Pervasive Computing, pages 167-180, London, UK.
- [40] P.Brezillon. B.Marcos et al.(July 2004.) .Context-awareness in group work : Three case studies. In Proceedings of Decision Support Systems, pages 115-124, Prato,

Italie.

- [41] T.CHAARI.(Septembre 2007) .Adaptation d'applications pervasives dans des environnements multicontextes. Thèse de doctorat, Institut National des Sciences Appliquées de Lyon, France.

- [42] R.Ali.(mars 2010). Modeling and Reasoning about Contextual Requirements : Goal-based framework. International Doctorate School in Information and Communication Technologies University of Trento.

- [43] Y.SKLAB .(2013). Spécification d'un modèle formel pour le contexte utilisateur. Mémoire de Magister, Université de Bejaia.

- [44] U.Straccia. G.Amato.(1999) .User profile modelling and applications to digital libraries. In Proceedings of the 3rd European Conference on Research and advanced technology for digital libraries, London, UK, pages 184-187.

- [45] D.Petrelli. E.Not et al.(2000). Modeling Context is Like Taking Pictures. In CHI'2000 Workshop on Context Awareness. The Hague, Netherlands.

- [46] M.Daoud. (Décembre 2009).Accès personnalisé à l'information : approche basée sur l'utilisation d'un profil utilisateur sémantique dérivé d'une ontologie de domaines à travers l'historique des sessions de recherche. Thèse de Doctorat en informatique présentée à l'Université Paul Sabatier, France.

- [47] D.Kostadinov. (Septembre 2008).Personnalisation de l'information : une approche de gestion de profils et de reformulation de requêtes. Thèse de Doctorat en informatique présentée à l'Université Versailles Saint-Quentin-En-Yvelines, France.

- [48] K.Abbas. (Décembre 2008).Système d'accès personnalisés à l'information : application au domaine médical. Thèse de Doctorat en informatique présenté à l'Institut

National des Sciences Appliquées de Lyon, France.

- [49] H.Sever. V.Raghavan. (1995).On the reuse of past optimal queries. In Proceedings of the 18th annual international ACM SIGIR conference on Research and development in information retrieval, NY, USA, pages 344-350.

- [50] B.Tan. X.Shen et al.(August 2006) .Mining long-term search history to improve search accuracy. In Proceedings of the 12th international conference on Knowledge Discovery and Data Mining, University of Illinois at Urbana-Champaign, pages 718-723.

- [51] C.Yang. G.Salton.(1973). On the specification of term values in automatic indexing. Journal of documentation.

- [52] H.Lieberman.(April 1997). Autonomous interface agents. In Proceedings of the SIGCHI conference on Human Factors in Computing Systems, New York USA, pages : 67-74.

- [53] C.Strapparava. A.Stefani.(June 1998). Personalizing access to web sites. The site if project.In Proceedings of the 2nd Workshop on Adaptive Hypertext and Hypermedia, pages : 69-75, Pittsburgh, USA.

- [54] G.Gentili. A.Micarelli et al.(2003) .An adaptive information tering system for the cultural heritage domain. Applied Articial Intelligence, pages :715-744.

- [55] N.Zemirli.(Juin 2008). Modèle d'accès personnalisé à l'information basé sur les diagrammes influence intégrant un profil utilisateur évolutif. Thèse de Doctorat en informatique présentée à l'Université Paul Sabatier de Toulouse III.

- [56] N.Lasierra. A.Alesanco et al.(2010). An Ontology approach to manage individual patient profiles in home-based telemonitoring scenarios. Journal of documentation.

- [57] CHALLAL Y.(Juin 2012). “Sécurité de l’Internet des Objets : vers une approche cognitive et systémique”, HDR, UTC.
- [58] Ashton, K. (2009).[http ://www.rfidjournal.com/articles/view?4986](http://www.rfidjournal.com/articles/view?4986)
. That “Internet of Things” Thing : In the Real World Things Matter More than Ideas. RFID Journal.
- [59] Hayi.M .(31 mai 2019).”Vers une gestion intelligente des services sensibles au contexte dans l’Internet des Objets : Approche centrée utilisateur”, rapport de thèse .
- [60] An Te NGUYEN.(23 Novembre 2006).”Un nouveau système de filtrage collaboratif basé sur le modèle des espaces de communautés”. THÈSE de DOCTEUR DE L’UNIVERSITÉ JOSEPH FOURIER- GRENOBLE I.
- [61] Marref Nadia.(18 décembre 2013) .UApprentissage Incrémental et Machines à Vecteurs Supports. Mémoire Présenté en vue de l’obtention du diplôme de Magister en Informatique,Université HADJ LAKHDAR BATNA.
- [62] Daull, Xavier. (2015). Recherche et développement d’une plateforme de prédiction et de recommandation pour le secteur de l’hôtellerie last-minute, CONSERVATOIRE NATIONAL DES ARTS ET METIERS, PROVENCE-ALPES-COTES D’AZUR.
- [63] Rabenarivo, Andrianambinina Marius. (2017). Rapport de stage de Master M2 Informatique : Développement d’un moteur de recommandation, UFR Sciences et Technologies, Université de la Réunion, France.
- [64] Y.Amraoui , D.Aggoun.(2018).”Exploitation du système de recommandation en utilisant l’apprentissage automatique”. Projet de fin d’études Présenté pour l’obtention du Diplôme de Master en Informatique.UNIVERSITE DR. YAHIA

FARES DE MEDEA.

- [65] Hadj Ahmed Bouarara .(07 Novembre 2019).”Data Mining (Fouille de données)”. Editions universitaires europeennes.
- [66] R. O. Duda and P. E. Hart.(1973). “Pattern Classification and Scene Analysis”. New York : John Wiley et Sons.
- [67] E Mathieu-Dupas .(2010).”Algorithme des k plus proches voisins pondérés et application en diagnostic”.
- [68] W.Ahmed Yousfi , M Y Hayi .(2018).”La création d’une collection de test Arabe pour évaluer des algorithmes et des techniques des moteurs de recherche”. Projet de fin d’études Présenté pour l’obtention du Diplôme de Master en Informatique. UNIVERSITE YAHIA FARES DE MEDEA.
- [69] “[http ://www.techno-science.net/?onglet=glossairedefinition=517.](http://www.techno-science.net/?onglet=glossairedefinition=517)” consulté le 10 octobre 2019.
- [70] “[http ://glossaire.infowebmaster.fr/xml/.](http://glossaire.infowebmaster.fr/xml/)” consulté le 10 octobre 2019.
- [71] “[http ://www.futura-sciences.com/tech/definitions/internet-java-485/.](http://www.futura-sciences.com/tech/definitions/internet-java-485/)” consulté le 10 octobre 2019.
- [72] “[http ://php.net/manual/fr/intro-what-is.php.](http://php.net/manual/fr/intro-what-is.php)” consulté le 10 octobre 2019.
- [73] “[http ://www.json.org/json-fr.html.](http://www.json.org/json-fr.html)” consulté le 10 octobre 2019.
- [74] “[http ://slideplayer.fr/slide/173472/.](http://slideplayer.fr/slide/173472/)” consulté le 10 octobre 2019.

[75] “[https ://developers.google.com/maps/documentation/android-api/intro](https://developers.google.com/maps/documentation/android-api/intro).”
consulté le 10 octobre 2019.

[76] “[https ://openweathermap.org/guide](https://openweathermap.org/guide).” consulté le 10 octobre 2019.

[77] Gautier Mechling .(2013). ”Android – Appels réseau avec Volley ”
.”[https :https ://blog.xebia.fr/2013/07/08/android-appels-reseau-avec-volley/](https://blog.xebia.fr/2013/07/08/android-appels-reseau-avec-volley/).”
consulté le 11 Novembre 2019 10 octobre 2019.

[78] ” [http ://code.google.com/p/google-gson](http://code.google.com/p/google-gson).” consulté le 11 Novembre 2019.

Titre Annexe 1

Titre Annexe 2

Résumé

Notre travail concerne les systèmes d'aide à la visite personnalisée. L'objectif est de concevoir un système de recommandation implémenté sur un dispositif mobile pour améliorer l'expérience de visiteur, en lui recommandant les items le plus pertinents. Ces items concernent les points d'intérêt qui peuvent être de différents types (Musée, Loisir, Montagne, ...). Chaque type contient plusieurs places et notre système doit recommander aux visiteurs des places sensibles à leur contexte, profil et préférences.

Mots clés : Système de recommandation , sensibilité au contexte, profil, préférence, mesure de similarité, application mobile.

Abstract

Our work concerns custom visitation systems. The objective is to design a recommendation system on a mobile device, to improve the visitor experience by recommending the most advanced items. These items concern the points of interest to recommend can be of different types (Museum, Leisure, Mountain, ...) each type contains several places and the system must recommend to the visitors places sensitive to their context, profile and preference of the user.

Keywords : Recommendation System, Context Sensitivity, Profile, Preference, Similarity measure, mobile application.