

République Algérienne Démocratique et Populaire  
Ministère de l'Enseignement Supérieur et de la Recherche Scientifique  
Université de 8 Mai 1945-Guelma-  
Faculté des Mathématiques, d'Informatique et des Sciences de la matière  
Département d'Informatique



## *Mémoire de fin d'études Master*

Filière : Informatique

Option :

Système informatique

## Thème

---

Une approche de recommandation pour le tourisme dans  
un environnement Smart City

---

Encadré par :

*P<sup>r</sup> KOUAHLA MOHAMED NADJIB*

Présenté par :

*M<sup>r</sup> ZIAYA ILYAS*

Année Universitaire 2021/2022

# Remerciements

*On dit souvent que le trajet est aussi important que la destination. Les années qu'on a passées à l'université nous ont permis de bien comprendre la signification de cette phrase toute simple. Ce parcours, en effet, ne s'est pas réalisé sans défis et sans soulever de nombreuses questions pour lesquelles les réponses nécessitent de longues heures de travail.*

*Avant tout, je voudrais remercier Dieu de m'avoir donné la force, la connaissance, la capacité et l'opportunité d'entreprendre ce travail, de persévérer et de la terminer de manière satisfaisante.*

*Je tiens à remercier sincèrement mon encadreur Professeur **KOUAHLA MOHAMED NADJIB** pour m'avoir dirigé, conseillé et ses encouragements tout au long de la réalisation de ce travail et surtout pour sa patience et sa compréhension afin de donner le fruit d'un travail long et difficile. Merci de m'avoir fait confiance et pour tout ce que vous avez fait pour m'amener jusqu'ici.*

*Je remercie également mes parents pour tous les sacrifices qu'ils ont consentis pour me permettre de poursuivre mes études dans les meilleures conditions possibles et de ne jamais cesser de m'encourager tout au long de ces années.*

*Enfin, je remercie tous les enseignants du département de l'informatique de l'université du 8 mai 1945 de Guelma pour la qualité de leur formation. Un grand merci aux équipes du laboratoire LabSTIC qui ont contribué, chacune à leur manière.*

# Dédicace

*Je dédie ce travail*

*À mes très chers parents*

*À ma mère, qui a œuvré pour ma réussite, de par son amour, son soutien, tous les sacrifices consentis et ses précieux conseils, pour toute son assistance et sa présence dans ma vie, reçois à travers ce travail aussi modeste soit-il, l'expression de mes sentiments et de mon éternelle gratitude.*

*À mon père, qui peut être fier et trouver ici le résultat de longues années de sacrifices et de privations pour m'aider à avancer dans la vie. Puisse Dieu faire en sorte que ce travail porte son fruit. Merci pour les valeurs nobles, l'éducation et le soutien permanents.*

*À ma soeur **IMANE***

*Les mots ne suffisent guère pour exprimer l'attachement, l'amour et l'affection que je te porte, mon ange gardien et mon fidèle compagnon dans les moments les plus délicats de cette vie mystérieuse. Je te dédie ce travail avec tous mes vœux de bonheur, de santé et de réussite.*

*À ma soeur **FAIZA***

*Elle est présente dans tous les moments de mes examens avec son soutien moral et ses belles douces surprises. A travers ce travail, je vous exprime les sentiments de fraternité et d'amour.*

*Sans oublier mes très chers amis **SOUFIANE, AMINE, ADEL** et tous ceux qui m'ont soutenu et encouragé durant toute la période de mon travail. . .*

**ILYAS**

# Résumé

Au cours de la dernière décennie, les systèmes intelligents fournissent des services publics qui ont conduit un changement significatif du cadre de vie de la ville vers le concept de la ville intelligente. Pour cela, nous avons concentré nos recherches sur le domaine de tourisme et mobilité afin de créer une stratégie intelligente visant à proposer un système de recommandation avec un filtrage hybride sensible au contexte, aux localisations et aux préférences des utilisateurs.

Pour construire notre système de recommandation, nous devons d'abord établir un modèle d'utilisateur qui nous permet de spécifier les caractéristiques qui seront incluses dans le système. Ces caractéristiques sont offertes par trois profils : le profil démographique, le profil de préférence et le profil de localisation. À la suite de cette modélisation, nous avons élaboré notre solution sur la base de trois facteurs qui sont : un facteur clé qui nous permet d'effectuer une analyse de sentiment de l'avis de l'utilisateur qui se présente sous forme de commentaire textuel et faire une prédiction de ce commentaire a une note, un facteur de préférences de l'utilisateur qui permet de connaître les restaurants les plus similaires et appropriés au regard de ces préférences, et enfin un facteur de localisation qui facilite une mobilité intelligente dans la ville afin de trouver les restaurants les plus proches en matière de distance. Les résultats sont ensuite fusionnés dans notre système pour améliorer les suggestions en fonction de ces trois facteurs.

Lors de la phase d'expérimentation, le sous-système d'analyse des sentiments a été développé dans un environnement Colab avec deux ensembles de données, permettant la création d'un modèle de prédiction sophistiqué avec un taux de précision de 60%, et le système de recommandation, qui est développé dans l'environnement Jupyter sur la base des trois facteurs précédents que nous avons pu découvrir un modèle utilisable pour créer de meilleures suggestions des restaurants pour l'utilisateur.

**Mot clés :** *ville intelligente, tourisme, mobilité, contexte, localisations, préférences, analyse de sentiment.*

# Abstract

Over the last decade, intelligent systems that provide public services have been successfully applied in all areas of the city, leading to a significant shift in the city's living environment towards the concept of the smart city. For this purpose, we focused our research on the tourism and mobility domain to create an intelligent strategy to propose a recommendation system with hybrid filtering sensitive to context, location and user preferences.

To build our recommender system, we must first establish a user model that allows us to specify the features that will be included in the system. These features are offered by three profiles : the demographic profile, the preference profile and the location profile. Following this modeling, we developed our solution based on three factors : a key factor that allows us to perform a sentiment analysis of the user's opinion in the form of a textual comment and make a prediction of this comment in the form of a score, a user preference factor that allows us to know the most similar and appropriate restaurants with respect to these preferences, and finally a location factor that facilitates intelligent mobility in the city in order to find the restaurants closest in distance. The results are then merged in our system to improve the suggestions based on these three factors.

In the experimentation phase, the sentiment analysis subsystem was developed in a Colab environment with two datasets, allowing the creation of a sophisticated prediction model with an accuracy rate of 60%, and the recommendation system, which is developed in the Jupyter environment based on the previous three factors we were able to discover a usable model to create better suggestions for the user.

**Key words** : *smart city, tourism, mobility, context, locations, preferences, sentiment analysis.*

# Table des matières

Table des figures	viii
Liste des tableaux	ix
<b>Introduction générale</b>	<b>1</b>
Contexte de recherche . . . . .	1
<b>I État de l’art</b>	<b>3</b>
<b>1 Smart cities</b>	<b>4</b>
1 Introduction . . . . .	4
2 Smart city . . . . .	4
2.1 Définitions . . . . .	4
2.2 Technologies TICs . . . . .	5
2.3 Caractéristiques . . . . .	6
2.4 Besoins . . . . .	7
2.5 Dimensions . . . . .	8
2.6 Architecture . . . . .	9
2.7 Défis . . . . .	11
3 Smart city : Domaines d’application . . . . .	11
3.1 Tourisme intelligent . . . . .	12
3.2 Destination intelligente - Cas d’étude : Les restaurants . . . . .	13
4 Conclusion . . . . .	14
<b>2 Systèmes de recommandation</b>	<b>15</b>
1 Introduction . . . . .	15
2 Système de recommandation . . . . .	15
2.1 Définition . . . . .	15
2.2 Type de filtrage . . . . .	16
2.2.1 Filtrage collaboratif . . . . .	16
2.2.2 Filtrage basé sur le contenu . . . . .	16
2.2.3 Filtrage hybride . . . . .	17
3 Système de recommandation : Moyens et méthodes . . . . .	18
3.1 Phase collecte d’information - Feedback . . . . .	18

3.1.1	Explicite . . . . .	18
3.1.2	Implicite . . . . .	18
3.2	Méthodes de similarité . . . . .	19
3.2.1	Distance euclidienne . . . . .	19
3.2.2	Coefficient de corrélation de Pearson . . . . .	19
3.2.3	Similarité cosinus . . . . .	19
3.3	Classification . . . . .	20
3.3.1	Naive Bayes (NB) . . . . .	20
3.3.2	Support Vector Machines (SVM) . . . . .	20
3.3.3	Régression Logistique (RL) . . . . .	20
3.3.4	K Nearest Neighbors (KNN) . . . . .	21
3.4	Évaluation . . . . .	21
3.4.1	Mesures de précision statistique . . . . .	21
3.4.2	Mesures de précision de l'aide à la décision . . . . .	21
3.5	Défis . . . . .	22
3.5.1	Démarrage à froid . . . . .	22
3.5.2	Rareté des données . . . . .	22
3.5.3	Évolutivité . . . . .	22
4	Travaux connexes . . . . .	22
4.1	Challenges des travaux connexes . . . . .	30
4.1.1	Avis, données démographiques et météo . . . . .	30
5	Conclusion . . . . .	31

## **II Conception et Implémentation 32**

### **3 Conception et modélisation de l'approche proposée 33**

1	Introduction . . . . .	33
2	Fonctionnement et objectif . . . . .	33
3	Architecture générale du système . . . . .	34
4	Modélisation de l'utilisateur . . . . .	36
4.1	Acquisition des données . . . . .	36
4.2	Construction des profils . . . . .	37
4.2.1	Profil démographique . . . . .	38
4.2.2	Profil de localisation . . . . .	38
4.2.3	Profil des préférences . . . . .	39
5	Modélisation du restaurant . . . . .	39
5.1	Caractéristiques . . . . .	40
6	Module d'analyse des sentiments . . . . .	41
6.1	Prétraitement et extraction des données . . . . .	42
6.2	Sélection des caractéristiques . . . . .	44
6.3	Technique de classification . . . . .	45
6.4	Résultat d'analyse des sentiments . . . . .	45

7	Module de recommandation - Filtrage . . . . .	45
7.1	Représentation de restaurant et utilisateur . . . . .	46
7.1.1	Restaurant . . . . .	46
7.1.2	Utilisateur . . . . .	46
7.2	Facteurs de recommandation . . . . .	48
7.2.1	Note prévue à partir de commentaire . . . . .	48
7.2.2	Filtrage basé sur le contenu - Calcule de similarité . . . . .	49
7.2.3	Distance entre utilisateur et restaurant . . . . .	49
7.3	Algorithme de recommandation . . . . .	50
7.3.1	Score finale . . . . .	50
7.3.2	Filtrage collaboratif . . . . .	51
7.3.3	Résultat de recommandation . . . . .	53
7.3.4	Pseudo-code de l'algorithme de recommandation . . . . .	53
8	Conclusion . . . . .	55
<b>4</b>	<b>Mise en oeuvre et résultat de l'approche proposée</b>	<b>56</b>
1	Introduction . . . . .	56
2	Présentation des outils de développement . . . . .	56
2.1	Matériels . . . . .	56
2.2	Environnement de travail . . . . .	56
2.3	Langage de programmation . . . . .	57
2.4	Bibliothèques . . . . .	57
3	Expérimentation du module d'analyse des sentiments . . . . .	58
3.1	Collecte des données . . . . .	58
3.2	Prétraitement des données . . . . .	59
3.3	Sélection des caractéristiques . . . . .	62
3.4	Évaluation et discussion des résultats de classification . . . . .	63
3.5	Scénario de test . . . . .	67
4	Expérimentation du système de recommandation . . . . .	68
4.1	Collecte des données . . . . .	68
4.2	Prétraitement des données . . . . .	70
4.3	Codage des données . . . . .	74
4.4	Résultat du score final . . . . .	75
4.5	Évaluation et discussion des résultats finaux . . . . .	77
5	Conclusion . . . . .	80
	<b>Conclusion générale</b>	<b>81</b>
	Contribution . . . . .	81
	Perspective . . . . .	81
	Acquis . . . . .	82

# Table des figures

1.1	Caractéristiques d'une ville intelligente [2]	6
1.2	Les six dimensions d'une ville intelligente	8
1.3	L'architecture en couches de la ville intelligente	10
1.4	Composants et couches de tourisme intelligent [29]	13
2.1	Schéma explicatif de SR basé sur le Filtrage collaboratif (FC) [40]	16
2.2	Schéma explicatif de SR basé sur le contenu [40]	17
2.3	Système de recommandation hybride [40]	17
3.1	L'architecture générale du système proposé	35
3.2	Les dimensions d'un profil utilisateur	38
3.3	Le profil démographique	38
3.4	Le profil de localisation	39
3.5	Le profil des préférences	39
3.6	Les caractéristiques de restaurant	40
3.7	Le processus d'analyse des sentiments	41
3.8	L'opération de stemming	43
3.9	L'opération de lemmatization	43
3.10	Exemple explicatif de Bag of Word	44
3.11	La matrice des caractéristiques des restaurants	46
3.12	Vecteur des préférences de l'utilisateur	47
3.13	Mise à jour de vecteur des préférences de l'utilisateur	47
3.14	La prédiction du note à partir de commentaire	48
3.15	Les positions de l'utilisateur et les restaurants	51
4.1	Le texte avant la suppression des stop word	59
4.2	Le texte après la suppression des stop word	59
4.3	Le texte après l'application de stemming	60
4.4	Le texte après l'application de lemmatization	60
4.5	Le groupement des mots à partir de stemming	61
4.6	Le groupement des mots à partir de lemmatization	61
4.7	Le nombre des mots avant et après le prétraitement	62
4.8	Vecteur numérique du premier texte avec Tfidf	62
4.9	Vecteur numérique du premier texte avec BoW	63
4.10	Résultats graphiques pour le classificateur SVM	64

4.11 Résultats graphiques pour le classificateur RL . . . . .	65
4.12 Résultats graphiques pour les quatre expérimentations . . . . .	66
4.13 Résultats de prédiction du commentaire 1 . . . . .	67
4.14 Résultats de prédiction du commentaire 2 . . . . .	67
4.15 Résultats de prédiction du commentaire 3 . . . . .	68
4.16 La distribution des caractéristiques des restaurants . . . . .	69
4.17 La distribution des caractéristiques des utilisateurs . . . . .	69
4.18 Les données des restaurants après le premier traitement . . . . .	70
4.19 Les données des utilisateurs après le premier traitement . . . . .	71
4.20 Les données des restaurants après la suppressions des valeurs manquants . . . . .	72
4.21 Les données des utilisateurs après la suppressions des valeurs manquants . . . . .	72
4.22 Les données des restaurants après le filtrage des caractéristiques . . . . .	73
4.23 Les données des utilisateurs après le filtrage des caractéristiques . . . . .	73
4.24 Les données des restaurants après le codage binaire . . . . .	74
4.25 Les données des utilisateurs après le codage binaire . . . . .	74
4.26 Le résultat de similarité entre l'utilisateur et le restaurant . . . . .	75
4.27 Le résultat de distance entre l'utilisateur et le restaurant . . . . .	75
4.28 L'ensemble de données des notes - Rating . . . . .	76
4.29 L'ensemble de données total . . . . .	76
4.30 L'ensemble de données pour le score final . . . . .	77
4.31 Le graphe des résultats de benchmark pour 11 algorithmes . . . . .	78
4.32 Le graphe des résultats de différentes combinaisons . . . . .	79

# Liste des tableaux

1.1	Les domaines d'application dans la ville intelligente . . . . .	12
2.1	Résumé des systèmes de recommandation des restaurants . . . . .	30
3.1	Exemple explicatif de TfidfVectorizer . . . . .	44
3.2	Matrice de score utilisateur-restaurant . . . . .	52
3.3	Exemple de recommandation des restaurants . . . . .	53
4.1	Résultats pour le classificateur SVM . . . . .	63
4.2	SVM et Tfidf . . . . .	64
4.3	SVM et BoW . . . . .	64
4.4	Résultats pour le classificateur RL . . . . .	64
4.5	RL et Tfidf . . . . .	65
4.6	RL et BoW . . . . .	65
4.7	Les résultats des quatre expérimentations . . . . .	66
4.8	SVM et Tfidf . . . . .	66
4.9	SVM et BoW . . . . .	66
4.10	RL et Tfidf . . . . .	67
4.11	RL et BoW . . . . .	67
4.12	Les résultats de benchmark pour 11 algorithmes . . . . .	77
4.13	Les résultats d'évaluation du système de recommandation . . . . .	79

# Abréviations et Acronymes

<BMCS> <Baidu Map Cloud Service>

<BOW> <Bag Of Word>

<FBC> <Filtrage Basé sur le Contenu>

<FC> <Filtrage Collaboratif>

<KNN> <k-Nearest Neighbor>

<MAE> <Mean Absolute Error>

<NLP> <Natural Language Processing>

<NLTK> <Natural Language Tool Kit>

<NMF> <Non-negative Matrix Factorization>

<NUMPY> <NUMerical PYthon>

<PMF> <Probabilistic Matrix Factorization>

<POI> <Points Of Interest>

<RL> <Régression Logistique>

<RMSE> <Root Mean Squared Error>

<SR> <Système de Recommandation>

<SVD> <Singular Values Decomposition>

<SVM> <Support Vector Machine>

<TFIDF> <Term Frequency Inverse Document Frequency>

<TIC> <Technologies de l'information et de la Communication>

# Introduction générale

Cette introduction présente le contexte global du travail, la problématique et l'objectif qui nous avons traité ainsi que la structure générale du mémoire.

## Contexte du travail

Les technologies de l'information et de la communication (TIC) et les systèmes intelligents ont créé de nouvelles formes de connectivité et de partage d'information dans la ville, l'intelligence est devenue une nouvelle expression et nouvel objectif pour définir ces progrès. Ce mot a été inventé pour les villes afin de caractériser les tentatives de promouvoir une gouvernance efficace, une durabilité et une qualité de vie par l'application de stratégies innovantes. L'objectif de ce projet est de proposer une stratégie de développement des villes intelligentes, notamment dans le domaine du tourisme intelligent, qui repose sur l'échange d'informations et de services publics entre les personnes et les infrastructures intelligentes de la ville.

Dans le domaine du tourisme intelligent, les visiteurs de la ville rencontrent plusieurs problèmes dans la recherche d'informations, pour cela nous avons dédié les systèmes de recommandation pour résoudre ce problème. Ces systèmes deviennent de plus en plus populaires et sont aujourd'hui une composante majeure de nombreuses applications dans le tourisme. Cela permet de recommandation adaptée aux préférences et aux besoins des utilisateurs exprimés implicitement ou explicitement, et s'adapte également en matière de localisation, afin de fournir des recommandations plus précises à l'utilisateur.

Notre approche proposée concerne un système de recommandation sensible au contexte et localisation basé sur le filtrage des données avec deux techniques différents de filtrage et contient un sous-système d'analyse des sentiments pour un environnement de tourisme intelligent.

Ce travail est réparti sur quatre chapitres différents dont l'organisation est comme suit :

**Une introduction générale** situant le contexte et explicitant la problématique et les objectifs à atteindre ainsi que la structure du mémoire.

**La première partie** est un état de l'art de notre travail elle est composée de deux chapitres suivants.

**Chapitre 1** : Ce chapitre est dédié au terme de smart city, ou nous présentons les concepts de base pour les smart cities tels que les caractéristiques, les besoins, les défis et l'architecture. Enfin nous soulignons le domaine d'étude pour notre travail et prenons le cas des restaurants

comme un cas d'étude.

**Chapitre 2 :** Nous avons consacré ce chapitre aux systèmes de recommandation, nous avons commencé par les concepts de base de ces systèmes tels que la recommandation contextuelle et les techniques de filtrage, puis identifiaient les moyens et les méthodes qui permettent le bon fonctionnement de ces systèmes tels que les modèles de similarité, les techniques de classification et les techniques d'évaluation. Enfin, nous citons des travaux pertinents pour ces systèmes dans le contexte des restaurants afin de bénéficier et exploiter les avantages de leurs approches.

**La deuxième partie** du mémoire contient notre contribution. Elle est composée de deux chapitres suivants.

**Chapitre 3 :** Représente notre conception, qui est décrite par la modélisation de l'utilisateur et du restaurant, suivie par la modélisation du processus d'analyse des sentiments pour la construction d'une stratégie de recommandation du tourisme pour les villes intelligentes.

**Chapitre 4 :** Ce chapitre est consacré à la mise en œuvre et l'expérimentation de notre approche. Nous présentons d'abord les outils de développement et les environnements de travail, puis l'expérimentation et les résultats du module d'analyse des sentiments sur deux ensembles de données distinctes. Enfin, nous illustrons les processus expérimentaux du système de recommandation ainsi que les évaluations des résultats réels.

**Une conclusion générale** qui souligne l'approche proposée et identifie les directions de recherche pour l'avenir.

Première partie

État de l'art

# Chapitre 1

## Smart cities

### 1 Introduction

L'augmentation rapide de la population urbaine du monde a créé des défis complexes pour les villes et une concurrence croissante partout dans le monde, dans ce cas le concept de ville intelligente s'est imposé dans la première fois par l'utilisation des technologies de l'information et de la communication (TIC) qui coordonnent toutes les activités et tous les services, ce qui se traduit par des citoyens mieux connectés, informés et engagés.

Afin de bien définir les villes intelligentes, nous évoquerons dans ce chapitre les différents concepts et les domaines liés à ce terme.

### 2 Smart city

Dans cette section nous abordons les nombreuses idées et principes de base associés aux villes intelligentes, ainsi que leurs caractéristiques, besoins et architecture.

#### 2.1 Définitions

Au début des années 1990, le terme smart city est apparu pour la première fois car l'accent était mis sur l'importance des nouvelles technologies de l'information et de la communication pour l'infrastructure d'une ville moderne [1].

L'adjectif "intelligent" dépend du sens qu'on lui attribue. Dans différents articles, différentes appellations ont été utilisées pour désigner ce concept : smart city, knowledge city, digital city, etc. C'est pourquoi la distinction entre une « ville intelligente » et une « ville numérique » est si importante. Ces deux termes sont souvent confondus ou utilisés comme synonymes, alors qu'ils ne sont pas vraiment les mêmes [2].

- **Ville numérique** : la ville numérique est basée sur les TICs. Cette notion devient intéressante à partir du moment où le numérique est mis au service de la ville intelligente et de la population [3].

- **Ville intelligente** : la ville intelligente est basée sur les outils numériques pour améliorer la qualité de vie des populations. En effet, la technologie est utilisée pour le développement intelligent de l'espace urbain tant en matière de mobilité que d'environnement, de participation citoyenne, etc. Il peut donc sembler évident que la ville intelligente découle souvent de la ville numérique pour une meilleure gestion urbaine [4].

Passons maintenant en revue quelques définitions, pour comprendre les différentes définitions couvertes par le concept de smart city.

**Définition 1.1.** *Une ville prospectivement performante en termes d'environnement et de vie, de mobilité, d'économie, de population et de gouvernance, conçue sur la base d'une combinaison intelligente de dotations et d'activités [5].*

**Définition 1.2.** *La ville intelligente est une ville qui relie les éléments de l'infrastructure informatique, de l'infrastructure sociale, de l'infrastructure physique et de l'infrastructure commerciale pour atteindre une intelligence collective de la ville [6].*

**Définition 1.3.** *La ville intelligente est une ville interconnectée, instrumentée et intelligente. Elle comprend des dispositifs et des outils de capture de données tels que les compteurs intelligents, les smartphones, internet, les réseaux sociaux et une multitude d'autres systèmes de collecte de données [7].*

Ce qui caractérise une ville intelligente est l'utilisation combinée des logiciels, des matériels et de systèmes de réseau (appelés technologies informatiques) selon une approche basée sur la technologie pour améliorer les services dans sept domaines principaux tels que la gestion de la ville, l'éducation, la santé, la sécurité publique, l'immobilier, le transport et services publics [8].

## 2.2 Technologies TICs

Le déploiement d'une ville intelligente est une tâche multidisciplinaire, qui implique diverses parties prenantes issues de différents domaines thématiques, tels que la politique, la finance, la gestion et l'organisation de la ville, ainsi que les technologies de l'information et de la communication (TIC).

Du point de vue des TIC, le nombre des dispositifs, technologies et des normes impliquées sont énormes donc nous résumons les principaux acteurs de la technologie comme suit :

- **Capteurs** : grâce aux progrès de la technologie, des nombreux capteurs différents sont désormais disponibles, qui sont un élément essentiel de tout système de contrôle intelligent et permettre des applications qui étaient auparavant inimaginables. Du point de vue des exigences des villes intelligentes, l'étendue de ces technologies se traduit par un grand nombre de possibilités pour des nombreuses opportunités en matière de détection, par exemple des compteurs pour identifier le gaz, l'électricité, l'eau, etc [9].

- **Smartphones** : représentent un autre scénario de détection intéressant. En effet, le nombre toujours croissant et la présence d'un mélange des capteurs tels que GPS, accéléromètres et boussoles, permettent une variété d'applications augmentées par l'IoT (internet of things). Par exemple, comme les utilisateurs mettent régulièrement à jour leur situation géographique sur des réseaux sociaux tels que Twitter et Facebook, il est possible, sur la base de ces informations de localisation, d'agréger ces données, ce qui permet d'envoyer des tâches à des personnes se trouvant dans des endroits spécifiques [10].

En ce qui concerne les communications à courte distance, deux technologies intéressantes dans le contexte des villes intelligentes sont l'identification par radio fréquence (RFID) et la communication en champ proche (NFC) [9].

## 2.3 Caractéristiques

La ville intelligente comprend des thèmes définissent les caractéristiques et les piliers de la ville intelligente. La plupart des propositions des villes intelligentes reposent sur quatre attributs importants : durabilité, commodité, qualité de vie, urbanisation et intelligence (voir figure 1.1) [11].

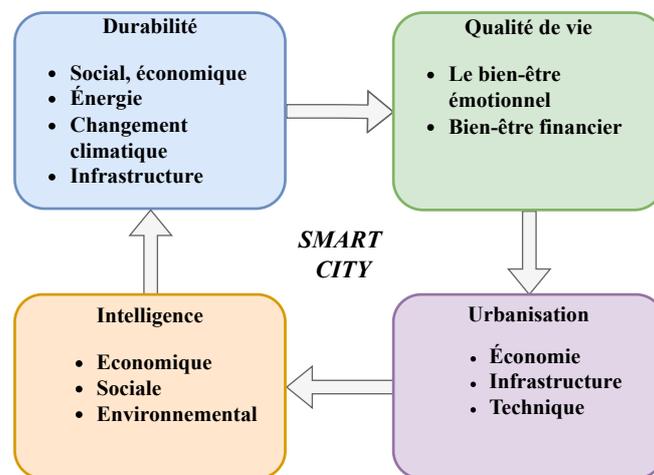


FIGURE 1.1 – Caractéristiques d'une ville intelligente [2]

La durabilité concerne la pollution, l'énergie, le changement climatique et les écosystèmes, l'attribut qualité de vie quotidienne des citoyens vise à améliorer le bien-être des citoyens, l'urbanisation et l'intelligence sont liées aux domaines de la technologie, des infrastructures et de la gouvernance pour transformer l'environnement rural en environnement urbain [12].

Une ville intelligente se compose aussi des propriétés et des infrastructures fournissent les plates-formes d'exploitation de la ville intelligente et se composent d'un ensemble des fonctionnalités [13].

- L'utilisation d'une infrastructure en réseau pour améliorer l'efficacité économique et politique et permettre le développement social, culturel et urbain ;
- Un accent sous-jacent sur le développement urbain dirigé par les entreprises ;
- L'accent mis sur l'objectif d'inclusion sociale des différents résidents urbains dans les services publics ;
- Un accent sur le rôle crucial des industries de haute technologie et de la création dans la croissance urbaine à long terme ;
- Une attention profonde au rôle du capital social et relationnel dans le développement urbain.

## 2.4 Besoins

Dans un domaine aussi complexe et multidisciplinaire que les "Smart cities", il y aura des nombreux besoins futurs pour compléter le démarche des villes intelligentes nous citons les différents types des besoins dans les sections suivantes.

- **Besoins technologiques** : les villes intelligentes fournissent des systèmes et des solutions complexes au moyen d'un large éventail des technologies homogènes capables de fonctionner ensemble et ont besoin de ces technologies puissantes et flexibles [14], pour les aider à :
  - Développer des modèles cohérents pour les villes intelligentes et ainsi améliorer l'interopérabilité des systèmes grâce à une compréhension partagée des concepts clés qui les sous-tendent ;
  - Échange facile et rapide de données entre différents types de réseaux et utilisation de différents types de transport et de communication ;
  - La possibilité d'utiliser et de collecter des données par des systèmes et services tiers ;
  - Permettre de présenter les données sous différentes formes, selon la personne ou le système technique qui en a besoin, en facilitant leur affichage, leur accès et leur interaction pour les rendre plus utiles ;
  - Le travail des systèmes hétérogènes des technologies de l'information et de la communication collectivement ;
  - Automatisation fréquente pour exécuter les fonctions de la ville de manière fiable et efficace, réduisant le besoin d'intervention humaine directe là et quand cela est approprié.
- **Besoins du marché** : l'innovation peut être bénéfique et appuyer les villes intelligentes à l'avenir, afin que les villes intelligentes disposent d'un environnement économique dans lequel l'innovation technologique peut prospérer [14], cet environnement nécessite ce qui suit :
  - Une main-d'œuvre correctement formée, disponible pour des nouvelles opportunités d'emploi et capable de travailler de manière flexible selon les besoins ;

- Des espaces d'apprentissage adaptatifs, ainsi que des outils d'apprentissage à distance, pour permettre le développement des compétences personnalisées, partout où les étudiants se rassemblent dans la ville ;
  - Une place de marché qui permet la découverte automatique des services et des ressources, la mise en correspondance des besoins et des fournisseurs des solutions possibles, ainsi que des transactions à faible friction ;
  - Une infrastructure stable et adaptée à la création d'entreprise et à l'investissement (physique et numérique) ;
  - Une infrastructure plus intelligente qui peut satisfaire les besoins des entreprises et du secteur public.
- **Besoins sociaux** : les villes incluent des environnements physiques et virtuels capables d'améliorer la communication complexe entre les citoyens, les entreprises et les services (publics ou privés) et qui avec les décideurs politiques doivent répondre aux politiques économiques, sociales et émergentes [14], ces défis sociaux nécessitent les besoins suivants :
    - L'élaboration des scénarios et des modèles sociaux qui tiennent compte des tendances démographiques et des besoins en constante évolution de la population ;
    - Une industrie plus moderne "écologique" et plus respectueuse de l'environnement ;
    - Promouvoir une meilleure gestion de la ville grâce à des citoyens qui jouent un rôle actif dans la prise de décision sur la vie et les orientations futures de la ville ;
    - Améliorer la qualité de vie et la sécurité des citoyens et fournir une variété des services ;
    - Un réseau d'espaces collaboratifs, pour favoriser la création des communautés dynamiques qui stimulent l'innovation, la croissance et améliorent le bien-être des citoyens ;
    - Environnement durable (qualité de l'air, gestion des déchets, résilience au changement et aux menaces climatiques, etc).

## 2.5 Dimensions

Cette partie se concentre sur un aperçu rapide des aspects de la ville intelligente, qui ont été divisés en six catégories : économie intelligente, mobilité intelligente, environnement intelligent, personnes intelligentes, vie intelligente et gouvernance intelligente [15].



FIGURE 1.2 – Les six dimensions d'une ville intelligente [16]

- **Économie intelligente** : déterminée par la flexibilité de la main-d'œuvre, l'innovation, l'esprit d'entreprise, l'image économique et la capacité d'une ville à se transformer en ville verte [17], cela inclut la forte compétitivité économique qui fait croître l'économie de la ville [11].
- **Mobilité intelligente** : comprends la disponibilité des systèmes de transport sûrs, d'installations modernes et d'infrastructures vertes qui permettent un accès facile [11, 18]. Il fait référence à l'accessibilité locale, modernes et durables. En outre, la mobilité intelligente consiste à fournir aux citoyens un accès à des technologies innovantes, pour faciliter la routine urbaine dans les villes [17].
- **Environnement intelligent** : l'environnement intelligent est lié à l'attractivité des conditions naturelles et à la gestion responsable des ressources en constante diminution et des demandes en constante augmentation [11]. Ainsi, l'écologie intelligente fait référence à l'utilisation des technologies TIC pour promouvoir les initiatives culturelles de sécurité et la numérisation des biens traditionnels [19].
- **Personnes intelligente** : la dimension "smart people" comprend le capital social et humain, la tolérance, la créativité et la participation aux événements publics. Dans une ville intelligente les résidents sont intelligents en terme de compétence et de niveau d'éducation, ainsi que de la valeur de la collaboration sociale dans la ville [15].
- **Vie intelligente** : comprend toutes les composantes d'une vie heureuse et bénéfique (comme la santé, l'éducation, le tourisme, la sécurité, la culture, etc.) [20], ce qui conduit à une vie plus harmonieuse et agréable pour la population [19].
- **Gouvernance intelligente** : la gouvernance intelligente vise à aider les décideurs à prendre des décisions et à planifier le déploiement des TIC [15]. Cela comprend l'amélioration des procédures d'autogestion et la modification de la répartition des services communautaires [21].

## 2.6 Architecture

L'architecture de la ville intelligente se compose de quatre couches, à savoir la couche de détection, la couche de transmission, la couche de gestion des données et la couche d'application (voir figure 1.3). Où chaque couche comprend des modules de sécurité [12].

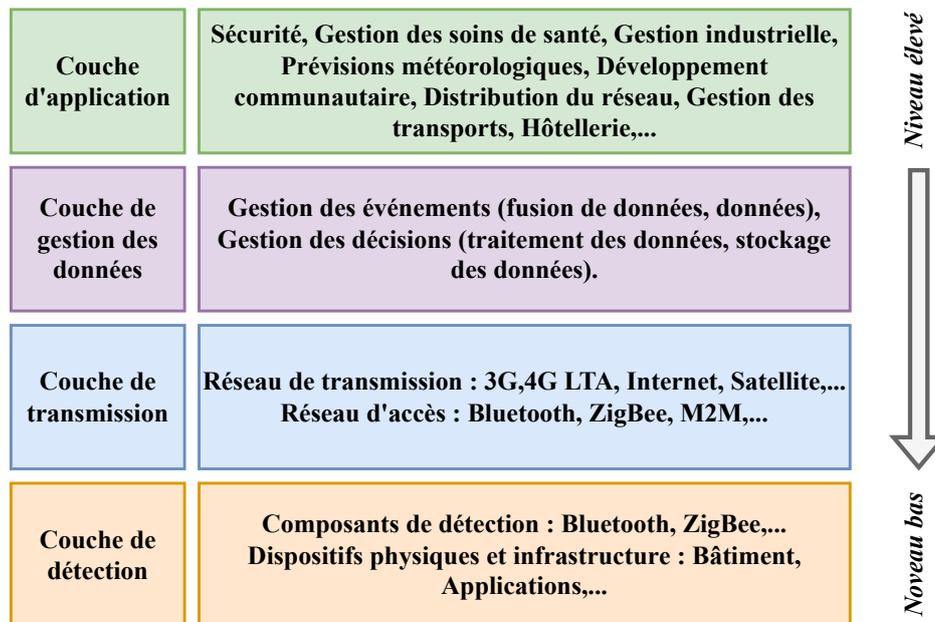


FIGURE 1.3 – L'architecture en couches de la ville intelligente [12]

Donc pour la suite nous allons voir la description de chaque couche [12] :

- **Couche de détection** : située au bas de l'architecture et responsable de la collecte des données.
- **Couche de transmission** : la couche de transmission est la confluence des différents réseaux de communication. Il comprend différents types des technologies filaires, sans fil et par satellite. Il est divisé en deux sous-couches, à savoir la transmission d'accès et la transmission de réseau. Cependant, Bluetooth, RFID qui sont des technologies de réseau d'accès avec une couverture relativement courte. De même, les technologies de couverture plus large, telles que LTE, 3G, 4G, 5G.
- **Couche de gestion** : la couche de gestion des données responsables du traitement et du stockage des données de charge utile par la couche d'application, et cette couche est la plus élevée de l'architecture de la maison intelligente.
- **Couche d'application** : les applications de ville intelligente reposent sur différents types de données et de calculs en raison de leur importance dans la prise de décision. D'autre part, la collecte de données est considérée comme le rôle le plus important car elle contrôle le reste des opérations de la ville intelligente. D'autre part, la collecte de données est la tâche la plus difficile en raison de la grande variance des données. Une ville intelligente se compose des différentes données provenant des différentes opérations de la ville : équilibrage de charge dans le réseau intelligent, contrôle des appareils dans la maison intelligente, surveillance de la santé personnelle, gestion des épidémies, des catastrophes, etc.

## 2.7 Défis

Nous décrivons certains des principaux défis importants auxquels sont confrontées les villes intelligentes et qui contribuent à leur acceptation et à leur croissance [22], nous avons cité trois défis importants [23].

- **Consommation d'énergie élevée** : l'absence des ressources naturelles dans l'estimation de la consommation d'énergie joue un rôle négatif dans les investissements dans les villes intelligentes. L'avenir du coût de l'énergie et de l'accès à l'énergie est incertain, principalement en raison de leur dépendance à l'égard des scénarios, socio-économiques et démographiques projetés.
- **Citoyens intelligents** : le caractère « intelligent » de la ville dépend en grande partie de la participation des citoyens aux projets de ville intelligente, à travers des multiples outils de communication (ex : réseaux sociaux et applications pour smartphone). Les villes intelligentes ont besoin que les citoyens soient constamment connectés dans les lieux publics, dans les transports publics et à la maison afin de partager leurs connaissances et leurs expériences et pour l'objectif de la gestion efficace des ressources naturelles et l'amélioration de la qualité de vie des citoyens.
- **Vie privée** : le respect de la vie privée jouera un rôle central dans toutes les stratégies de ville intelligente, les citoyens interagissent avec les services des villes intelligentes par leurs smartphones et leurs ordinateurs connectés par des réseaux et des systèmes hétérogènes. Il est donc impératif que les villes intelligentes fondées sur l'utilisation des TIC, sachent gérer les questions importantes relatives au respect de la vie privée (telles que les écoutes et la confidentialité) [23].

## 3 Smart city : Domaines d'application

Avec l'abondance et l'augmentation des applications des villes intelligentes, les définitions de champ deviennent plus larges et plus spécifiques qui guidant la planification globale d'une ville intelligente, en particulier pour des systèmes ou des applications spécifiques il est nécessaire de faire comprendre aux concepteurs les domaines dans lesquels les projets s'inscrivent, et les aspects à considérer et à couvrir [24].

Les domaines d'application des villes intelligentes sont répartis en plusieurs domaines et sous-domaines [25] représentés dans le tableau ci-dessous 1.1.

Domaine	Sous-domaine	Description
Gouvernement	Administration électronique Gouvernement transparent Service publique Sécurité publique Surveillance des villes Urgence	Améliorer l'efficacité interne du gouvernement. L'accès simple aux documents pour les citoyens et les autres organisations. Assurer l'efficacité des services et gestion de sécurité. Répondre rapidement dans les situations d'urgences.
Citoyen	Transport public Trafic intelligent Tourisme Divertissement Soins de santé Éducation Cohésion sociale	Voyager et se déplacer plus efficacement. Accéder à des informations contextualisées, précises et en temps réel dans la vie quotidienne. Services publics essentiels de haute qualité tels que l'éducation, les soins de santé et le sport.
Affaires	Niveau élevé Gestion d'entreprise Logistique Chaîne d'approvisionnement Transaction Publicité Innovation Esprit d'entreprise Agriculture	Utiliser des plateformes et des méthodes de logistique et des chaînes d'approvisionnement plus efficaces. Publicité plus large et plus précise. Faciliter l'entreprise et l'investissement. Améliorer l'activité commerciale d'une ville et encourager l'innovation.
Environnement	Réseau intelligent Énergie renouvelable Gestion de l'eau Gestion des déchets Contrôle de la pollution Bâtiment Logement Communauté Espace public	Réduisant et prévenir la pollution dans la ville. Offrir la mobilité intelligent. La télécommunication, l'information et toutes les autres facilités dans les différents espaces urbains.

TABLE 1.1 – Les domaines d'application dans la ville intelligente

[25]

### 3.1 Tourisme intelligent

Le tourisme intelligent est l'un des domaines d'application les plus importants dans une smart city. Voici les différentes définitions apportées pour le tourisme intelligent :

**Définition 1.4.** *L'appel à la mobilité numérique pour établir des liens plus efficaces et plus durables entre les visiteurs et leur destination [26]. Grâce à la quête d'informations, de communication, des réseaux sociaux et des capacités liées à la mobilité pour aider les passagers partout où ils vont, la technologie mobile semble avoir accéléré cette tendance [27].*

**Définition 1.5.** *Le tourisme intelligent est défini comme une destination touristique innovante construite sur une infrastructure technologique de pointe qui assure le développement à long terme des zones touristiques, est accessible à tous et facilite l'interaction et l'intégration du*

visiteur avec son environnement, améliorant ainsi la qualité de l'expérience du visiteur et la qualité de vie de la population [28].

**Définition 1.6.** *Le tourisme est soutenu par des efforts intégrés dans une destination pour collecter et agréger/exploiter les données dérivées de l'infrastructure physique, des connexions sociales, des sources gouvernementales/réglementaires et des corps/esprits humains, ainsi que par l'utilisation des technologies avancées pour transformer ces données en expériences de proposition des valeurs sur place et commerciales, en mettant clairement l'accent sur l'efficacité, la durabilité et l'enrichissement de l'expérience [29].*

Le tourisme intelligent comprend plusieurs composantes et couches de « l'intelligence », y compris dans la figure 1.4.

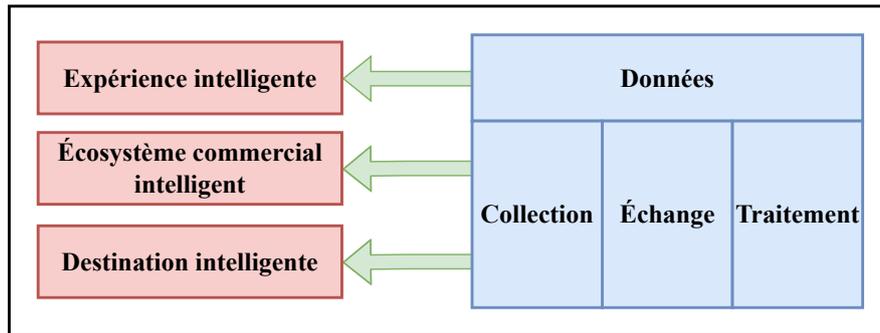


FIGURE 1.4 – Composants et couches de tourisme intelligent [29]

Chaque composante du tourisme intelligent a été définie comme suit [29] :

- **Expérience intelligente** : concentre spécifiquement sur l'expérience touristique médiatisée par la technologie et son partage grâce à la personnalisation, la sensibilisation contextuelle et le suivi en temps réel.
- **Écosystème commercial intelligent** : référence à l'écosystème commercial complexe qui crée et soutient l'échange des ressources touristiques et la co-crédation des expériences touristiques [28].
- **Destination intelligente** : sont des cas particuliers des villes intelligentes qui intègrent les TIC dans l'infrastructure physique. Les destinations intelligentes sont des sous-ensembles des villes intelligentes : elles appliquent les idées de la ville intelligente à des sites urbains ou ruraux, en tenant compte non seulement des habitants mais aussi des visiteurs dans leurs tentatives de promouvoir la mobilité, la disponibilité et l'allocation des ressources, la durabilité et la qualité de vie des visiteurs. Les groupes touristiques gouvernementaux se soucient souvent des destinations et investissent dans leur marketing.

### 3.2 Destination intelligente - Cas d'étude : Les restaurants

Les restaurants dans les villes intelligentes définies comme un lieu utilisant les outils et techniques technologiques disponibles pour permettre à la demande et à l'offre de créer une

valeur de plaisir et des expériences pour le touriste et de la richesse, des bénéfices et des avantages pour les organisations et la bonne destination [30].

L'étude de smart destination pour les restaurants comprend plusieurs recherches sur la qualité des restaurants, la nourriture et la compétitivité. Les touristes et les résidents qui aiment manger choisissent les restaurants en fonction des diverses caractéristiques, notamment l'emplacement, le climat, la qualité et le prix. Cependant, la diversité des restaurants, des services, des attractions touristiques, des activités et des événements spéciaux sont autant des facteurs importants à prendre en compte lors du choix d'un lieu [31].

La ville touristique intelligente doit améliorer l'expérience touristique en proposant des restaurants plus personnalisés et qui répondent aux intérêts uniques de chaque citoyen au bon moment en fonction des préférences des clients. Pour cela dans notre travail de recherche, nous examinons le tourisme intelligent pour les restaurants en tant qu'étude de cas, qui est influencée par un certain nombre d'aspects, dont le plus important est la mobilité intelligente et nous avons également besoin d'autres éléments qui sont utilisés dans notre recherche, comme les systèmes de recommandation, que nous étudions dans le chapitre suivant.

## 4 Conclusion

Dans ce premier chapitre, nous avons présenté les concepts fondamentaux de la ville intelligente avec un accent particulier sur le domaine du tourisme et les restaurants comme destination intelligente, qui aide les citoyens et les touristes à interagir de manière intelligente dans une ville touristique sur la base d'informations. Pour cela nous soulignons la présence des systèmes de recommandation, qui sont organisés en fonction des activités et des objectifs de la ville intelligente.

Le chapitre suivant est consacré aux systèmes de recommandation avec des détails sur les concepts de base, les moyens et les méthodes.

# Chapitre 2

## Systemes de recommandation

### 1 Introduction

Les systèmes de recommandation sont un élément important qui contribue de manière significative à l'intelligence de la ville et facilite la vie des touristes et des résidents de tous horizons. Par conséquent, ce chapitre concentrera sur ces systèmes et examinera plus en profondeur leurs concepts de base, leurs moyens et méthodes, et enfin nous exposons un certain nombre de travaux dans les domaines du tourisme intelligent et des destinations pour les restaurants basés sur ces systèmes.

### 2 Système de recommandation

Une multitude de données sont créées dans les villes intelligentes, fournissant une grande quantité d'informations. Il est donc nécessaire de disposer d'une méthode capable de filtrer les données afin de faciliter la recherche d'informations [32, 33]. Par conséquent, un système de recommandation peut être utilisé pour aider les utilisateurs à obtenir les informations contextuelles nécessaires pour faire des bons choix [34, 35].

Dans cette section nous abordons les systèmes de recommandation, ainsi que leurs techniques de filtrage, les moyens et les méthodes inclus dans notre travail, afin d'introduire de ces systèmes.

#### 2.1 Définition

Les systèmes de recommandation sont des systèmes de filtrage qui gèrent la surcharge d'informations en extrayant des éléments clés d'un énorme volume de données produites dynamiquement, en fonction des préférences, des intérêts ou du comportement observé de l'utilisateur [36, 37]. Les systèmes de recommandation profitent à la fois aux fournisseurs des services et aux clients. Ils réduisent les coûts de transaction liés à la recherche et au choix des produits dans un environnement d'achat en ligne [38], on a également constaté que les systèmes de recommandation améliorent le processus et la qualité de la prise de décision [39].

## 2.2 Type de filtrage

Lorsqu'il s'agit de la façon dont les recommandations sont créées, il y a plusieurs choix à considérer lors de l'adoption d'un SR. Le filtrage dans les systèmes de recommandation peut être divisé en plusieurs catégories ce que nous allons voir dans les sous-sections suivantes.

### 2.2.1 Filtrage collaboratif

Le filtrage collaboratif est une méthode de prédiction indépendante du domaine qui améliore les informations difficiles à définir à l'aide de métadonnées.

La méthode de filtrage collaboratif fonctionne en stockant les préférences des utilisateurs pour des éléments dans une base de données (matrice utilisateur/élément). Elle facilite ensuite les correspondances entre les utilisateurs ayant des intérêts et des préférences comparables en évaluant les similitudes entre leurs profils afin de proposer des suggestions, ainsi qu'en obtenant des évaluations implicites des utilisateurs (par exemple, le nombre de fois qu'une chanson est entendue, les informations consultées et l'accès à une ressource) [36].

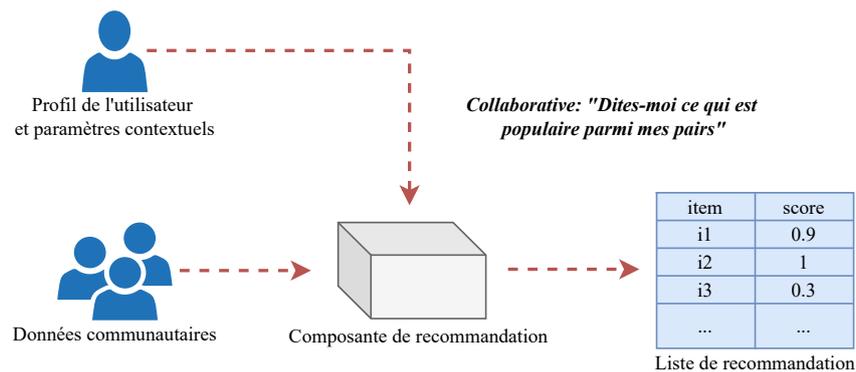


FIGURE 2.1 – Schéma explicatif de SR basé sur le Filtrage collaboratif (FC) [40]

### 2.2.2 Filtrage basé sur le contenu

L'approche basée sur le contenu est un algorithme spécifique au domaine qui se concentre sur l'évaluation des qualités des articles pour fournir des prédictions. La stratégie de filtrage basée sur le contenu est la plus efficace lorsqu'il s'agit de recommander des documents tels que des pages Web, des publications et des articles d'actualité [41].

La suggestion dans l'approche de filtrage basée sur le contenu est basée sur les profils des utilisateurs et les caractéristiques extraites du contenu des choses que l'utilisateur a précédemment évaluées [42].

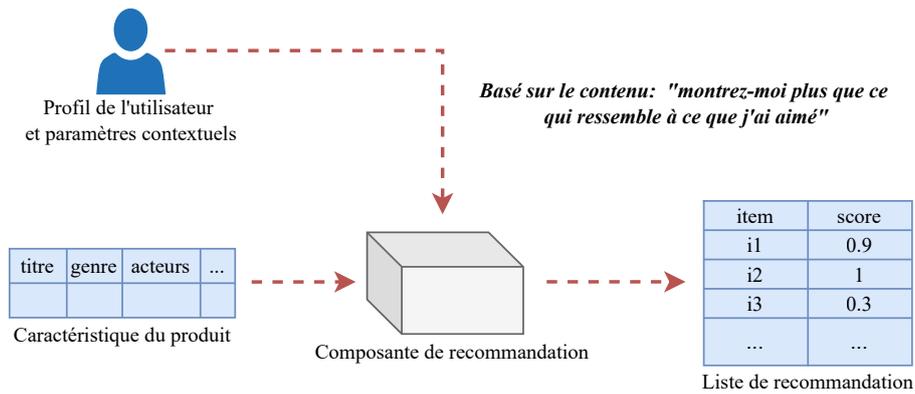


FIGURE 2.2 – Schéma explicatif de SR basé sur le contenu [40]

### 2.2.3 Filtrage hybride

Le filtrage hybride comprend plusieurs méthodes de recommandation mixtes pour améliorer l'optimisation du système et éviter certaines des limitations et difficultés associées aux systèmes de recommandation pure [43], l'idée derrière les méthodes mixtes est qu'un ensemble d'algorithmes fournira des recommandations plus précises et efficaces qu'un seul algorithme, puisque les défauts de l'un des algorithmes peuvent être surmontés [44].

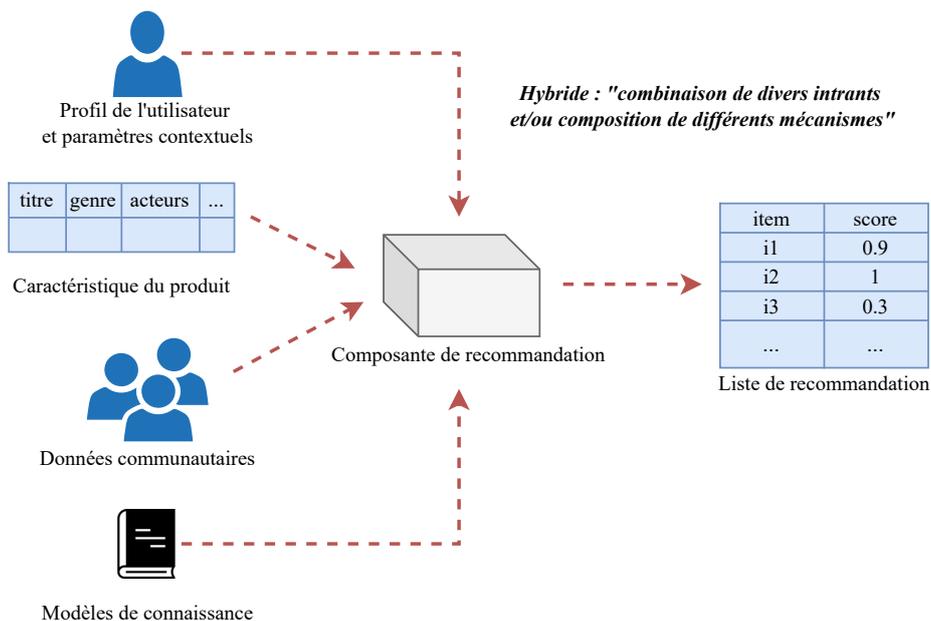


FIGURE 2.3 – Système de recommandation hybride [40]

La combinaison des approches peut être réalisée de l'une des manières suivantes [45].

- **Pondéré** : les résultats des techniques choisies (sous la forme de scores ou de votes) sont combinés avec différents degrés d'importance pour offrir une recommandation finale.
- **Commutation** : le type de situation affecte la technique utilisée par le système.
- **Mixte** : les recommandations de plusieurs techniques sont présentées simultanément à l'utilisateur.

- **Combinaison d'éléments** : les éléments provenant des différentes sources de recommandation sont combinés en entrée d'une seule technique.
- **Augmentation des caractéristiques** : la sortie d'une technique est utilisée comme caractéristique d'entrée pour une autre technique.
- **Cascade** : un système de recommandation affine les recommandations données par un autre.
- **Méta-niveau** : le modèle appris par un système de recommandation est utilisé comme entrée pour un autre [45].

### 3 Système de recommandation : Moyens et méthodes

Les systèmes de recommandation sont constitués d'un ensemble d'outils et des techniques qui leur permettent de travailler et de s'adapter à la situation en cours afin de trouver une solution rapide et efficace.

#### 3.1 Phase collecte d'information - Feedback

Pour effectuer des tâches de prédiction des modèles, des informations pertinentes sur l'utilisateur sont recueillies, telles que ses caractéristiques, ses actions ou le contenu des ressources auxquelles il a accédé [46].

Il existe deux types de commentaires d'utilisateurs qui peuvent être enregistrés : explicites et implicites.

##### 3.1.1 Explicite

Le système demande généralement à l'utilisateur d'évaluer les aspects à travers l'interface du système afin de créer et d'améliorer son modèle. La quantité d'évaluations soumises par l'utilisateur détermine la justesse de la suggestion [47].

Il existe principalement trois types différents de collecte d'information d'une manière explicite [48] :

- **Binaire** : le bouton "J'aime" ou "J'aime pas" de facebook est un exemple bien connu.
- **Classement** : l'utilisateur peut par exemple, classer les objets sur une échelle de un à dix, lancer un dé ou leur attribuer un certain nombre d'"étoiles".
- **Commentaires textuels** : les sites de commerce électronique utilisent souvent des commentaires textuels. Ils permettent aux utilisateurs de commenter les articles, ce qui facilite le processus de décision pour les utilisateurs.

##### 3.1.2 Implicite

La technologie extrait automatiquement les préférences de l'utilisateur en suivant ses nombreuses actions, notamment l'historique des achats, l'historique de navigation, le temps passé sur certains sites Web et les liens suivis. Entre autres, en déduisant les préférences de l'utilisateur

à partir de son interaction avec le système, le feedback implicite allège la charge de l'utilisateur. Cette approche est moins précise et ne nécessite aucun effort de la part de l'utilisateur [47].

## 3.2 Méthodes de similarité

La phase la plus cruciale des algorithmes de filtrage collaboratif consiste à trouver des objets et des utilisateurs comparables. Après cette phase, il est simple de considérer les similarités entre ces personnes et ces objets et de choisir un ensemble d'utilisateurs et d'objets qui ressemblent le plus à l'utilisateur cible [49].

Voici quelques mesures de similarité populaires utilisées dans le filtrage qui sont également utilisées dans cette étude pour le contrôle.

### 3.2.1 Distance euclidienne

Elle est principalement basée sur la distance, puisque plus la distance est faible, plus la similarité entre les utilisateurs est élevée, comparable à la mesure de similarité cosinus, qui ne donne jamais de valeurs de similarité négatives [50].

Pour déterminer la similarité, il faut d'abord calculer la distance entre les utilisateurs à l'aide de l'équation (2.2), puis la similarité à l'aide de l'équation (2.1) comme suit :

$$\sum(Ua, Ub) = \frac{1}{1 + DIS(Ua, Ub)} \quad (2.1)$$

$$Dist(Ua, Ub) = \sum_{i=1}^j |Ru(a, i) - Ru(b, i)|^2 \quad (2.2)$$

### 3.2.2 Coefficient de corrélation de Pearson

La technique de Pearson, qui révèle une relation linéaire entre les utilisateurs et les objets, est l'une des mesures de similarité classiques les plus populaires et les plus utiles [51].

Les éléments qui ont été évalués conjointement sont inclus dans la procédure, qui est représentée par le rapport entre la variance conjointe de deux utilisateurs et l'écart type entre eux. La similarité est estimée à l'aide de la formule (2.3) suivante :

$$Sim(Ua, Ub) = \frac{\sum_i^j (Ru(a, i) - Ru(a)) (Ru(b, i) - Ru(b))}{\sqrt{\sum_{i=1}^j (Ru(a, i) - Ru(a))^2} \times \sqrt{\sum_{i=1}^j (Ru(b, i) - Ru(b))^2}} \quad (2.3)$$

### 3.2.3 Similarité cosinus

Pour ce faire, on mesure les angles en cosinus [51], qui sont créés par les deux vecteurs de notation des utilisateurs. Une valeur d'angle plus faible représente un plus grand degré de ressemblance, et vice versa. Cette métrique modifie les valeurs des préférences plutôt que de se concentrer sur les données [52].

La similarité en cosinus est calculée [50] par la formule (2.4) :

$$Sim(Ua, Ub) = \frac{\sum_{i=1}^j (Ru(a, i))(Ru(b, i))}{\sqrt{\sum_{i=1}^n (Ru(a, i))} \times \sqrt{\sum_{i=1}^n (Ru(b, i))}} \quad (2.4)$$

Les résultats de cette méthode sont très similaires, malgré la grande différence des évaluations et ceci est considéré comme une limite pour cette similarité [53].

### 3.3 Classification

L'une des approches d'apprentissage automatique les plus importantes pour prédire la classe d'échantillons frais est la classification, qui utilise un modèle appris à partir des données de formation. En général, la classification est un processus d'apprentissage qui fait correspondre ou classe les instances de données dans les étiquettes de classe spécifiées de l'ensemble de données [54].

La section suivante énumère certains des algorithmes de catégorisations les plus souvent utilisées :

#### 3.3.1 Naive Bayes (NB)

Il s'agit d'une méthode de classification basée sur le théorème de Bayes et sur l'hypothèse de l'indépendance des prédicteurs. Un classificateur Naive Bayes, en termes simples, postule que l'existence d'une caractéristique dans une classe n'est pas liée à la présence de toute autre caractéristique. L'application de Naive Bayes se fait principalement dans le domaine de la classification de texte. Il est surtout utilisé pour le regroupement et la classification des probabilités conditionnelles d'occurrence [55].

#### 3.3.2 Support Vector Machines (SVM)

Une autre technologie d'apprentissage automatique supervisé avec des algorithmes d'apprentissage connexes est la machine vectorielle de support, qui analyse les données pour la classification et l'analyse de régression.

Le SVM peuvent effectuer une classification non linéaire rapidement en utilisant une approche à noyau, qui comprend la création des marges entre les couches et la traduction implicite de leurs entrées en espaces de caractéristiques à haute dimension. Les marges sont conçues pour maximiser la distance entre la marge et les catégories, ce qui réduit l'erreur de classification [55].

#### 3.3.3 Régression Logistique (RL)

L'approche d'apprentissage supervisé de la régression logistique peut être utilisée pour un certain nombre de tâches, notamment la classification de texte. Il s'agit d'un modèle de régression qui applique la régression logistique à des questions comportant plus de deux valeurs de sortie possibles.

### 3.3.4 K Nearest Neighbors (KNN)

La technique KNN (k-Nearest Neighbor) est une approche simple d'apprentissage automatique supervisé pour résoudre les problèmes de classification et de régression, elle est simple à mettre en place et à comprendre, mais elle a le défaut de devenir sensiblement plus lente à mesure que la quantité de données consommées augmente [55].

## 3.4 Évaluation

Différentes sortes de métriques peuvent être utilisées pour évaluer la qualité d'un algorithme de recommandation, et ces métriques changent en fonction de la stratégie de filtrage utilisée. La pertinence de chaque statistique est déterminée par les propriétés de l'ensemble de données et les tâches que le système de recommandation doit accomplir. La précision statistique et les métriques d'aide à la décision sont deux types de métriques utilisées pour évaluer la précision des systèmes de recommandation [39].

### 3.4.1 Mesures de précision statistique

Il évalue la précision de la technologie de filtrage en comparant directement les scores prédits et réels de l'utilisateur [39]. Les mesures de l'exactitude statistique comprennent l'erreur absolue moyenne (MAE), l'erreur quadratique moyenne (RMSE) et la corrélation [56].

MAE est calculer par la formule (2.5) [57] :

$$MAE = \frac{1}{N} \sum_{u,i} |Pu, i - Ru, i| \quad (2.5)$$

RMSE est calculer par la formule (2.6) [58] :

$$RMSE = \sqrt{\frac{1}{N} \sum_{u,i} (Pu, i - Ru, i)^2} \quad (2.6)$$

### 3.4.2 Mesures de précision de l'aide à la décision

Une façon courante d'évaluer les résultats des expériences d'apprentissage automatique est d'utiliser le rappel, la précision et la F-mesure. Ces mesures sont nommées en raison de leur origine dans le domaine de la recherche d'information et aident l'utilisateur à sélectionner des éléments de haute qualité parmi un grand nombre de possibilités [59].

$$Precision = \frac{\textit{Correctly recommended items}}{\textit{Total recommended items}} \quad (2.7)$$

$$Recall = \frac{\textit{Correctly recommended items}}{\textit{Total relevant items}} \quad (2.8)$$

$$F - \textit{measure} = \frac{2PR}{P + R} \quad (2.9)$$

## 3.5 Défis

En raison de l'évolution des besoins des entreprises qui utilisent et déploient des systèmes de recommandation, il devient de plus en plus important d'évaluer les performances de ces systèmes en fonction de leur capacité à résoudre une série des problèmes auxquels ils sont confrontés. L'un des plus grands défis est le démarrage à froid, la précision, la rareté des données et l'évolutivité, dont nous parlerons ci-dessous.

### 3.5.1 Démarrage à froid

Le problème du démarrage à froid. C'est le cas lorsqu'un système de recommandation n'a pas suffisamment des connaissances sur une personne ou un article pour faire des prédictions significatives [60].

L'un des principaux problèmes qui dégradent les performances des systèmes de recommandation est le suivant. Le profil de ce nouvel utilisateur ou de ce nouvel élément sera vide puisqu'il n'a rien évalué et que le système ne sait pas ce qu'il aime [39].

### 3.5.2 Rareté des données

Il s'agit d'un problème qui survient en raison d'un manque d'information, c'est-à-dire lorsque les utilisateurs n'évaluent que quelques éléments parmi le nombre total d'éléments accessibles dans une base de données [41, 61], cela se traduit invariablement par une matrice éparsée d'éléments utilisateur clairsemée, l'incapacité de trouver des voisins appropriés et par conséquent, des suggestions inadéquates. De plus, la rareté est toujours associée à des problèmes de couverture ou à la proportion des parties du système pour lesquelles des suggestions peuvent être données [62].

### 3.5.3 Évolutivité

L'évolutivité est un problème associé aux algorithmes de recommandation (le plus souvent dans les approches de filtrage collaboratif), puisque le calcul croît à un rythme linéaire par rapport aux utilisateurs et aux articles [61].

Lorsque le nombre total d'utilisateurs et d'éléments dans le système dépasse le niveau auquel les algorithmes traditionnels de filtrage collaboratif atteignent leurs limites de performance, des problèmes d'évolutivité apparaissent [63].

Le problème peut être résolu à l'aide de méthodes de réduction de dimension telles que le SVD, qui utilisent des nombreuses étapes de factorisation matricielle pour fournir des recommandations de haute qualité avec moins de ressources, ce qui permet une plus grande évolutivité et des résultats de meilleure qualité [63].

## 4 Travaux connexes

De nombreux systèmes de villes intelligentes fournissent des recommandations des restaurants aux utilisateurs. Nous avons choisi quelques-uns de ces systèmes afin d'en discuter,

d'apprendre comment ils fonctionnent et d'évaluer la qualité de leurs résultats. Ces systèmes sont regroupés en trois catégories comme suit.

- **Systèmes Basés sur le filtrage collaboratif :**

Afin d'extraire les préférences de l'utilisateur et de fournir la liste des suggestions, ce type de système calcule des recommandations basées sur les profils des utilisateurs et des restaurants, en prenant en compte les informations contextuelles ainsi que l'emplacement et la distance entre l'utilisateur et le restaurant. Cette stratégie a permis d'obtenir des recommandations significatives et efficaces. Nous allons passer en revue les travaux proposés dans ce domaine et découvrir les aspects qui sont généralement utilisés pour obtenir des bons résultats.

Un premier système proposé par wen-ying et Guo-Ming ont développé un système de recommandation qui réduit la quantité d'entrées censées initialisées pendant la phase de démarrage à froid. Les auteurs ont utilisé un algorithme basé sur des règles, qu'ils ont intégrées à des algorithmes de FC basés sur l'utilisateur et le contexte dans la phase d'analyse des données sur le comportement de l'utilisateur.

Les attentes des utilisateurs peuvent changer en fonction de leurs préférences à court terme, telles que le type de cuisine, l'heure et le lieu préféré, ainsi que de leurs préférences à long terme, telles que la gamme de prix ou les préoccupations environnementales.

Les informations sur le profil des utilisateurs, telles que le sexe, l'âge et les informations sur l'appareil mobile ; les informations contextuelles, telles que le lieu, la météo, l'heure et la saison ; les informations sur le restaurant, qui comprennent la catégorie, les caractéristiques et les informations de base ; et les enregistrements de l'utilisateur, qui comprennent les évaluations et les données d'interaction, ont également été collectés. Les suggestions ont été faites dans les mêmes paramètres que les évaluations et les plus proches voisins des évaluations. Cette méthode a permis d'obtenir des suggestions utiles et efficaces [64].

Hasegawa et Hayashi ont développé un système de localisation des environnements familiers et non familiers, en utilisant une application de localisation de restaurant. Les utilisateurs dont les régions de mouvement sont similaires et susceptibles de disposer d'informations ponctuelles qui peuvent être utiles à d'autres ; néanmoins, l'évaluation du caractère commun de ces régions de mouvement peut être inefficace si l'utilisateur quitte un site non familier.

Les auteurs ont examiné les historiques de visite des lieux d'autres utilisateurs appartenant à certaines zones, ainsi que l'emplacement actuel de l'utilisateur cible. Différents genres de lieux, où classifications des restaurants, ont été choisis pour les recommandations des restaurants. Les données de localisation ont été recueillies sur Tabelog.com, et des tests ont été effectués avec deux utilisateurs et 11 participants pour démontrer l'utilité de la méthode proposée [65].

Selon Sun et al, un système avec plusieurs variables, telles que : les lieux d'enregistrement, les suggestions d'amis, la popularité basée sur différentes zones, le comportement d'achat et les conditions de circulation autour du restaurant, ainsi que les comportements de mobilité dynamique des utilisateurs qui influencent le choix des restaurants.

Les auteurs ont proposé une approche RMSQ-MF, basée sur une analyse factorielle probabiliste, pour utiliser ces données multi sources. La popularité d'une zone a été déterminée à partir de la descente de taxi et combiner avec des informations sur le lieu. Les auteurs ont utilisé les données du quartier de Manhattan du site Yelp.com pour mener l'expérience, qui comprenait les profils des utilisateurs, les évaluations, les listes d'amis et les profils des restaurants.

En utilisant une approche gourmande, 1000 restaurants sur 7115 ont été choisis comme ayant la plus grande distance. En outre, les données des taxis de la ville de New York ont été acquises pour obtenir des informations sur le dépôt des taxis. En matière de distance en longitude et en latitude, l'emplacement du restaurant et le lieu de dépôt des clients des taxis ont été appariés, ce qui a donné un total de 21 684 273 données de dépôt des taxis pour les 1000 restaurants. La méthode proposée est appliquée sur 80% des données choisies d'une manière au hasard et comparer avec plusieurs approches telles que la factorisation matricielle non négative (NMF) [66], le modèle de décomposition en valeur singulière (SVD) basé sur SVD++ [67], BiasedMF [67], la factorisation matricielle probabiliste (PMF) [68] et SocialMF [69], et l'évaluation des performances a indiqué que la méthode suggérée était plus performante avec une MAE et une RMSE moyennes de 0,79 et 0,98, respectivement [70].

Dans [71], une technique de FC basée sur l'utilisateur a été proposée pour suggérer des restaurants en fonction des évaluations et des qualités des utilisateurs. Cette technique est composée de trois parties : le facteur de score moyen, le facteur de correction de similarité et le facteur de similarité des attributs de l'utilisateur, ainsi qu'une combinaison de ces similarités pour réaliser la technique suggérée. Les données utilisées pour l'expérience provenaient du site web dianping.com, qui couvrait la ville de Guilin, en Chine, qui comprenait 627 restaurants et 46718 évaluations de 30081 clients. En utilisant les mesures MAE et RMSE, la technique proposée a été comparée à la FC standard basée sur l'utilisateur.

Une autre approche nommée TripBuilder utilise le profil de l'utilisateur et la destination de vacances envisagées comme données d'entrée pour créer une visite personnalisée de nombreux points d'intérêt (PoI). Pour déterminer l'itinéraire, elle prend en compte à la fois le temps nécessaire pour profiter d'une attraction et le temps nécessaire pour se rendre à la suivante.

Wikipedia et Flickr sont deux sites collaboratifs accessibles au public. Flickr est utilisé pour collecter des photographies et des métadonnées de personnes du monde entier, tandis que Wikipedia est utilisé pour collecter des informations sur les points d'intérêt d'une ville particulière.

Toutes les entrées Wikipédia relatives à une entité située dans la région sont récupérées et considérées comme un PoI. Dans une base de données touristiques, la description, les coordonnées géographiques et l'ensemble des catégories pertinentes pour le PoI sont obtenues et enregistrées. Flickr est utilisé pour compiler une liste d'utilisateurs ainsi que des informations à partir des photos prises dans la ville pendant une certaine période. L'idée est que les albums photos des membres de Flickr décrivent des itinéraires touristiques à l'intérieur de la ville. Une photo est liée à un point d'intérêt si elle a été prise dans un rayon de 100 mètres de celui-ci, et le temps nécessaire pour visiter un point d'intérêt est supposé être le temps entre la première et la dernière photo de l'utilisateur du point d'intérêt.

Enfin, la popularité de chaque point d'intérêt est déterminée par le nombre des personnes uniques qui ont pris au moins une photo du point d'intérêt. Compte tenu des préférences des utilisateurs et des PoI, l'intérêt utilisateur-PoI est défini comme une combinaison de la similarité utilisateur-PoI et de la popularité des PoI [72].

- **Systèmes basés sur le filtrage hybride :**

Plusieurs techniques hybrides ont été présentées afin de fournir des suggestions plus précises [73]. Un algorithme hybride combine des nombreuses méthodologies, la structure d'une telle mise en œuvre varie en fonction de la technique d'approche hybride.

Une technique hybride, en particulier, combine des éléments d'évaluation provenant à la fois de l'approche basée sur le contenu et du filtrage collaboratif [74].

Katarya et Verma ont prédit des facteurs comme les contraintes de temps, le quartier, les spécificités culinaires, la gamme de prix, l'attitude et les objectifs d'une personne peuvent également être des facteurs déterminants dans le choix d'un restaurant. Pour la suggestion des restaurants basé sur le mobile, les auteurs ont également utilisé des variables démographiques, notamment l'âge, le sexe et l'emplacement actuel de l'utilisateur. Ils ont recueilli des informations à partir du site Web Foursquare.com ; l'élément psychographique est un autre type d'élément qui peut inclure des traits de personnalité et a été inclus dans les études, ce qui a permis d'obtenir un RMSE plus faible pour l'évaluation de la communauté et les détails des check-in [75].

Une application proposée par Utama et al, basée sur la logique floue et les techniques de tri à bulles pour recommander un restaurant dans une zone métropolitaine telle que Tangerang Selatan, en Indonésie. Les auteurs ont utilisé la logique floue pour décrire la découverte de l'intérêt du produit [76], la stratégie basée sur la FC, la commodité géographique [77], l'approche basée sur l'emplacement et la similarité de l'attention des utilisateurs [78], l'approche de FBC [79]. Le meilleur restaurant acceptable a été sélectionné à l'aide de l'algorithme de tri à bulles après que cette expérience ait été examinée pour 12 restaurants avec 77 répondants. Les utilisateurs peuvent également choisir un menu de plats et un emplacement de siège [80].

Pour tout service orienté vers l'utilisateur, la garantie du plaisir du client a toujours été une priorité absolue. Des notes plus élevées sur la nourriture, le service, le comportement du personnel, l'atmosphère et la satisfaction générale sont nécessaires pour qu'un SR recommande un certain restaurant à un nombre important d'utilisateurs. Une comparaison des techniques FC, FBC et hybrides pour prédire le taux de satisfaction des restaurants [81].

Cette expérience a été réalisée à l'aide d'une base de données publique [82] qui contenait des données sur les clients, telles que le profil et les caractéristiques de l'utilisateur avec 20 attributs, des données sur les restaurants, telles que diverses caractéristiques avec 21 attributs, et des données d'évaluation, telles que les valeurs de satisfaction des clients allant de 0 à 2, où 0 indique l'insatisfaction et 2 la satisfaction, pour les restaurants que les clients avaient visités. Les auteurs ont utilisé une somme d'évaluations, telles que la satisfaction, la nourriture et le service; les valeurs de notation globale variaient de 0 à 6, 6 indiquant que le restaurant était entièrement satisfait dans tous les domaines.

Pour tester les performances des techniques, les données ont été prétraitées et une validation croisée quadruple a été utilisée. L'approche hybride FC et FBC (la régression) proposée était une combinaison de ces deux techniques, avec une moyenne de 0,064 et un écart-type de 1,294, la MAE étant le plus faible, soit 1,032, parmi toutes les approches.

Dans [83], des similitudes entre les éléments et les utilisateurs ont été découvertes pour suggérer des restaurants lorsque les techniques FC et FBC ont été combinés. Les restaurants que l'utilisateur cible a visités récemment sont fournis en entrée, et des restaurants similaires sont découverts pour être suggérés. Les utilisateurs ayant des préférences similaires à l'utilisateur cible ont également été sélectionnés; Les restaurants récents fréquentés par les utilisateurs les plus similaires ont été fournis en tant qu'entrée pour trouver des similitudes entre les éléments, comme mentionné précédemment, et des restaurants classés ont été suggérés pour l'utilisateur cible. Les ensembles de données ont été produits à cette fin en utilisant un service d'annuaire des restaurants en ligne en Indonésie, qui comprend 67845 critiques et 83082 informations sur les restaurants [83]; les auteurs ont utilisé les données de 2014 à 2016 pour les trains et les données de 2017 pour les tests. La mesure F1 s'est avérée plus importante pour la similarité des restaurants dans la comparaison.

REJA est le SR qui combine FC et un système basé sur la connaissance. Sa principale caractéristique est l'utilisation de relations de préférences imparfaites dans le système à base de connaissances pour résoudre la difficulté de démarrage à froid de la composante collaborative. Le système collaboratif de REJA repose sur le moteur CoFE avec FC, qui est associé à une base de données de restaurant et d'utilisateur. Nous n'avons pu découvrir aucune information concernant le projet CoFE, et nous ne pouvons donc pas nous prononcer avec certitude sur le fonctionnement interne de l'algorithme FC.

Pour le traitement de démarrage à froid, certains auteurs sont convenus d'une méthode basée sur les connaissances qui ne nécessite qu'une petite quantité de données pour faire des suggestions appropriées. Cela fonctionne en utilisant la pensée basée sur des cas suivants :

1. L'utilisateur choisit un restaurant qui répond à ses besoins.
2. Le système présente au client trois restaurants connus et lui demande de les évaluer afin que le système puisse les comparer au restaurant choisi.
3. L'algorithme peut élaborer un profil d'utilisateur de base à partir de ces trois éléments d'information, qui seront utilisés pour fournir des suggestions.

REJA fournit également des informations géographiques aux utilisateurs par le biais de Google Maps, ce qui permet de minimiser la surcharge de données textuelles. L'utilisateur peut s'en servir pour localiser le restaurant sur une carte, identifier le trajet le plus rapide et localiser les principales attractions touristiques [84].

- **Systèmes basés sur le filtrage par contenu :**

Ce type de technologie faire des suggestions spécifiques à un domaine, il utilise des techniques de localisation pour créer des suggestions basées sur les caractéristiques personnelles et les préférences de l'utilisateur, cette méthode est simple et peut fournir des suggestions appropriées à partir d'une grande quantité de données.

Chung-Hua Chu et Se-Hsien Wu ont créé un système qui recueille des données contextuelles telles que l'emplacement, des informations sur les aliments saisonniers et les types des restaurants tels que les rafraîchissements et les restaurants gastronomiques en utilisant les services mobiles sensible au contexte. Les utilisateurs pouvaient sélectionner le moment préféré, leurs habitudes alimentaires et d'autres préférences, et les informations sur les restaurants, ainsi que les choix de chaque utilisateur sont transformés en vecteurs. Au total, 33 personnes ont participé à cette étude. Les auteurs ont calculé la moyenne et la variance de la satisfaction des utilisateurs et de la qualité du système, de l'information et du service ; la statistique de Cronbach a été déterminée comme étant supérieure à 0,7, ce qui suggère que le questionnaire employé dans cette technique était acceptable [85].

"My Eat!" un système de localisation des restaurants dans un environnement mobile a été proposé [86], les auteurs ont étudié la personnalisation et l'ubiquité [87], ainsi que les données de localisation afin d'améliorer les expériences alimentaires des utilisateurs. Foursquare.com est utilisé pour étudier les comportements d'enregistrement des utilisateurs et extraire les différentes informations [88, 89].

Les valeurs des suggestions des restaurants ont été calculées en omettant toutes les chaînes des restaurants souhaitées par l'utilisateur, c'est-à-dire en supprimant les différentes branches d'un même restaurant, et l'utilisateur a reçu une vue cartographique des restaurants proposés sur son téléphone mobile.

Un autre environnement mobile a été développé pour fournir de recommandation des restaurants basées sur les préférences de l'utilisateur, telles que les restaurants fréquentés avec les informations géographiques. Divers aspects du restaurant ont été évalués à l'aide d'un modèle de préférence de l'utilisateur pour la technique proposée, et la similarité

entre l'utilisateur et le restaurant a été calculée. La distance entre la position actuelle de l'utilisateur et le restaurant a donc été calculée à l'aide du GPS et du Baidu map cloud service (BMCS). Les N meilleurs restaurants ont été suggérés sur la base des scores estimés, six personnes ont participé à l'expérience et ont cliqué sur le restaurant de la liste suggérée s'ils l'avaient visité, sur la base de 11 caractéristiques extraites : hot pot, grill, buffet, restauration rapide, cuisine japonaise, fruits de mer, nouilles, soupe de canard, soupe de poulet, steak et boulettes, elles ont cliqué sur le restaurant de la liste proposée si elles l'avaient visité. Au total, 209 clics de ce type ont été enregistrés.

Les résultats ont montré que 89,5% des restaurants choisis figuraient parmi les cinq premiers, avec un classement moyen de 3,47 ; l'étude de cas a démontré l'efficacité de la stratégie recommandée [90].

R-cube est un système de recommandation et de réservation des restaurants. Avec des applications comme SIRI, Google Now et Watson, les auteurs ont choisi un système conversationnel pour améliorer les systèmes de reconnaissance vocale, une plus grande disponibilité des données pour permettre des applications et des technologies basées sur les données, et un accès aux informations et aux services via des plateformes mobiles.

Le système proposé, R-cube, est composé de trois sous-systèmes :

- Un sous-système de recommandation des restaurants qui collecte des informations sur les préférences de l'utilisateur afin de présélectionner des restaurants possibles ;
- Un sous-système de sélection des restaurants qui permet à l'utilisateur de poser des questions sur les restaurants présélectionnés afin de prendre une décision finale ;
- Un sous-système de réservation qui collecte les informations supplémentaires nécessaires pour compléter le processus de réservation.

Les choix de l'utilisateur sont pris en compte par le sous-système de suggestion des restaurants, qui se base sur quatre variables : le type de cuisine, la gamme de prix, la région de la ville et le nom de l'établissement, jusqu'à ce que la liste des restaurants probables soit réduite. l'algorithme continue à poser des questions au consommateur concernant ces qualités, puis nous passons à l'étape suivants, à savoir le sous-système de sélection des restaurants. Les clients peuvent maintenant en savoir plus sur les restaurants de la liste en consultant des critères tels que l'emplacement, la description de base et les critiques. Une fois que l'utilisateur a choisi son restaurant préféré, le système de réservation prend le relais.

De manière similaire à l'étape précédente, l'utilisateur est invité à saisir des informations sur le nombre de convives, le jour de la réservation et l'heure de la séance. Ensuite, la réservation est terminée et l'utilisateur reçoit un courriel de confirmation [91].

I'm feeling LoCo [92], est une application pour les smartphones décrite comme un système de recommandation contextuelle basée sur la localisation qui vise à fournir des recommandations plus approfondies en utilisant des données temporelles et de localisation. Au lieu de répondre à un sondage, le logiciel examine le profil Foursquare d'une personne et

transforme les données en préférences d'utilisateur. En outre, le programme reconnaît le mode de transport actuel de l'utilisateur pour estimer la distance qu'il est prêt à parcourir pour visiter une destination. La seule information explicite demandée à l'utilisateur est le type de lieu qu'il recherche.

L'application Foursquare utilise les données GPS de l'utilisateur pour générer une liste de lieux possibles et chaque élément de la liste est étiqueté avec un nom, une catégorie et des balises qui servent à définir le lieu. L'utilisateur n'a qu'à choisir son emplacement actuel. L'API de Foursquare peut ensuite être utilisée pour obtenir tous ces enregistrements de l'utilisateur, ainsi que les données qui les accompagnent. Ces informations sont utilisées pour créer le modèle d'utilisateur I'm Feeling LoCo lorsqu'un utilisateur visite un lieu, le modèle d'utilisateur est mis à jour avec le nom, la catégorie et le lieu.

Un système de recommandation est utilisé, incluant un filtrage basé sur le contenu. Voici un récapitulatif de la procédure de proposition :

1. L'utilisateur sélectionne une catégorie de recommandation. Parmi les catégories, on trouve "artsy", "nerdy" et "hungry" ;
2. L'emplacement actuel de l'utilisateur est utilisé pour obtenir tous les lieux situés dans un certain rayon. La distance est déterminée par le mode de transport de l'utilisateur ;
3. La liste des lieux est filtrée pour n'inclure que ceux qui entrent dans la catégorie donnée, et des étiquettes sont créées pour les lieux restants ;
4. Une collection des mots est créée pour chaque lieu en fonction de l'intersection entre les étiquettes de l'utilisateur et les étiquettes de ce site ;
5. Le poids de la fréquence logarithmique est calculé pour chaque phrase dans un ensemble de mots. Il s'agit d'un décompte du nombre de fois qu'une certaine phrase apparaît dans le profil d'un utilisateur, et ce pour chaque ensemble de mots ;
6. Une somme de toutes les pondérations est déterminée pour chaque site. Ce total représente le score d'un certain lieu ;
7. Enfin le client est informé des  $K$  sites ayant le meilleur score final.

Nous avons donné un aperçu des systèmes de recommandation des restaurants précédents ainsi que les caractéristiques, les techniques et les spécifications de données dans le tableau suivant 2.1.

Catégories	Réf	Caractéristiques	Base de données	Résultat
Systèmes avec filtrage collaboratif	[64]	Profil utilisateur Information sur les restaurants Contextes	Non spécifié	Recommandation efficace
	[65]	Historique des visites Localisation	2 utilisateurs 11 participants Données de Tabelog.com	Résultats améliorés par rapport à l'actuel SR
	[70]	Profil utilisateur Profil restaurant Évaluation Liste d'amis	1000 restaurants (yelp.com) 21684273 registres de dépôt de taxis	0.79 MAE, 0.98 RMSE
	[71]	Évaluation	627 restaurants 46718 évaluation 30081 utilisateurs de dianping.com	MAE, RMSE améliorés
	[72]	Profil utilisateur Informations sur les POIs	Base de donnée touristique Description POI et catégories Coordonnées géographique	Recommandation efficaces
Systèmes avec filtrage hybride	[75]	Mode de vie Intérêt Personnalité, âge et genre	Données de Foursquare.com	RMSE améliorés
	[80]	Intérêt Localisation Classement	12 restaurants, 77 répondants	Recommandation d'un restaurant approprié
	[81]	Satisfaction Nourriture Service	BDD publique avec contenu client, restaurant et évaluation	0.064 moyenne 1.294 SD 1.032 MAE
	[83]	Critiques	67845 critiques 83082 restaurants	F1-mesure supérieure
	[84]	Profil utilisateur Contextes Localisation	Non spécifique	Présentation de la carte et le trajet PoI touristique
Systèmes avec filtrage par contenu	[85]	Emplacement Informations sur les aliments Types de restaurants	33 utilisateurs	Statistique de Cronbach alpha supérieure à 0.7
	[86]	Enregistrement	Donnée de Foursquare.com	Présentation de la carte
	[90]	11 caractéristiques du restaurant	107 restaurants 6 participants 209 clicks	89.5 % des restaurants sélectionnés sont classés parmi les 5 premiers
	[91]	Profil utilisateur et restaurant Contextes	Non spécifique	Recommandation d'un restaurant approprié
	[92]	Données temporelles Localisation et mode de transport	Données de Foursquar.com	Recommandation plus précise

TABLE 2.1 – Résumé des systèmes de recommandation des restaurants

## 4.1 Challenges des travaux connexes

Selon les systèmes explorés dans la section 4 des travaux connexes nous couvrons certaines des études futures recommandées sur les systèmes de recommandation dans le domaine de la restauration, ce qui nous a conduits à des nouveaux aspects externes qui aident à la création de ces systèmes et à la façon dont ces aspects peuvent modifier le classement des résultats.

### 4.1.1 Avis, données démographiques et météo

La recherche montre que des bonnes critiques sur un restaurant particulier renforcent sa note élevée, et les données démographiques ont été liées à l'activité en ligne des gens. Dans [93] ont étudié si des aspects psychologiques tels que les commentaires en ligne, la démographie

et la météo influencent les comportements des utilisateurs. En analysant un grand nombre de restaurants, des critiques textuelles, des données démographiques et météorologiques des années précédentes, ils ont découvert que :

- Il existe une forte corrélation entre les notes, les avis et la qualité des restaurants ;
- Les restaurants qui s'annoncent comme prix faible ont moins d'avis et obtiennent de moins bonnes notes ;
- Les restaurants situés dans des zones où les niveaux d'éducation sont plus élevés sont plus susceptibles d'être fortement évalués, mais ne semblent pas affecter les notes.

En calculant et en corrigeant les biais qui sont couramment liés à les commentaires et les critiques des utilisateurs et à la météo, ces résultats ont des implications pour la conception des systèmes de recommandation.

## 5 Conclusion

Les systèmes de recommandation dans les villes intelligentes sont un domaine de recherche en pleine expansion. Grâce à diverses technologies, ils sont omniprésents dans notre vie quotidienne.

Dans ce chapitre, nous avons tenté d'analyser les travaux les plus intéressants et qui sont en relation direct avec notre approche proposée. Cette analyse elle permet de découvrir les types, les facteurs et les stratégies de ces systèmes sur lesquels nous nous concentrons entièrement pour établir notre propre système. L'étude des travaux de recherche ci-dessus nous a orienté vers une nouvelle approche de recommandation intelligente des restaurants basée sur des nombreux critères efficaces tels que les critiques des utilisateurs et les services de localisation qui contribuent de manière significative au développement de ces systèmes et les rendent plus performants.

Dans le chapitre suivant, nous expliquons en détail la conception et la formalisation de ce système proposé.

## Deuxième partie

# Conception et Implémentation

# Chapitre 3

## Conception et modélisation de l'approche proposée

### 1 Introduction

Ce chapitre est consacré à la modélisation de l'approche proposée, qui comporte plusieurs étapes importantes pour bien concevoir un système de recommandation des restaurants dans une ville intelligente. Nous verrons comment cette technique fonctionne pour obtenir des résultats efficaces, ainsi que les différents modules sur lesquels elle s'appuie pour atteindre notre objectif.

### 2 Fonctionnement et objectif

L'objectif de ce projet est de créer un système de recommandation intelligent des restaurants sensible au contexte et localisation, qui aide à fournir des services intelligents adaptés aux besoins des touristes et des citoyens de la ville. Ce système concentré sur la mobilité intelligente et la précision de recommandation selon les préférences. Cependant, cette technique est un moyen de personnalisation et un outil très puissant.

Notre approche décrit une situation dans laquelle une personne ayant besoin de nourriture reçoit des suggestions qui incluent un ensemble des restaurants qui répondent à ses besoins et qui sont utiles et efficaces. Le traitement avec le citoyen ou le visiteur commence par une demande de la liste des préférences suggérée et se termine par la liste des restaurants classés.

Les étapes de la modélisation de notre système peuvent être résumées comme suit :

- La présentation de l'architecture générale ;
- La modélisation de l'utilisateur et le restaurant ;
- Le module d'analyse des sentiments (la prédiction de la note à partir du commentaire) ;
- Enfin, Le module de recommandation.

### 3 Architecture générale du système

Le système de recommandation offre la possibilité de calculer et de filtrer les données pour obtenir des résultats de haute qualité pour l'utilisateur, le choix des lieux de restauration dans le cadre du tourisme intelligent pouvant être limité en fonction de précision et de situation spécifique.

Cette section présente le schéma général de notre système de recommandation des restaurants. En effet, la technique recommandée est axée sur la localisation et les préférences des personnes afin de remédier au faible choix des restaurants dans la ville.

Afin d'améliorer les services de restauration aux clients, nous avons rempli ce système avec plusieurs profils d'utilisateurs et des caractéristiques de restaurant, ainsi que de leurs coordonnées géographiques. Le schéma ci-dessous 3.1 représente l'architecture générale du système, ainsi que les modules qui le composent et son fonctionnement.

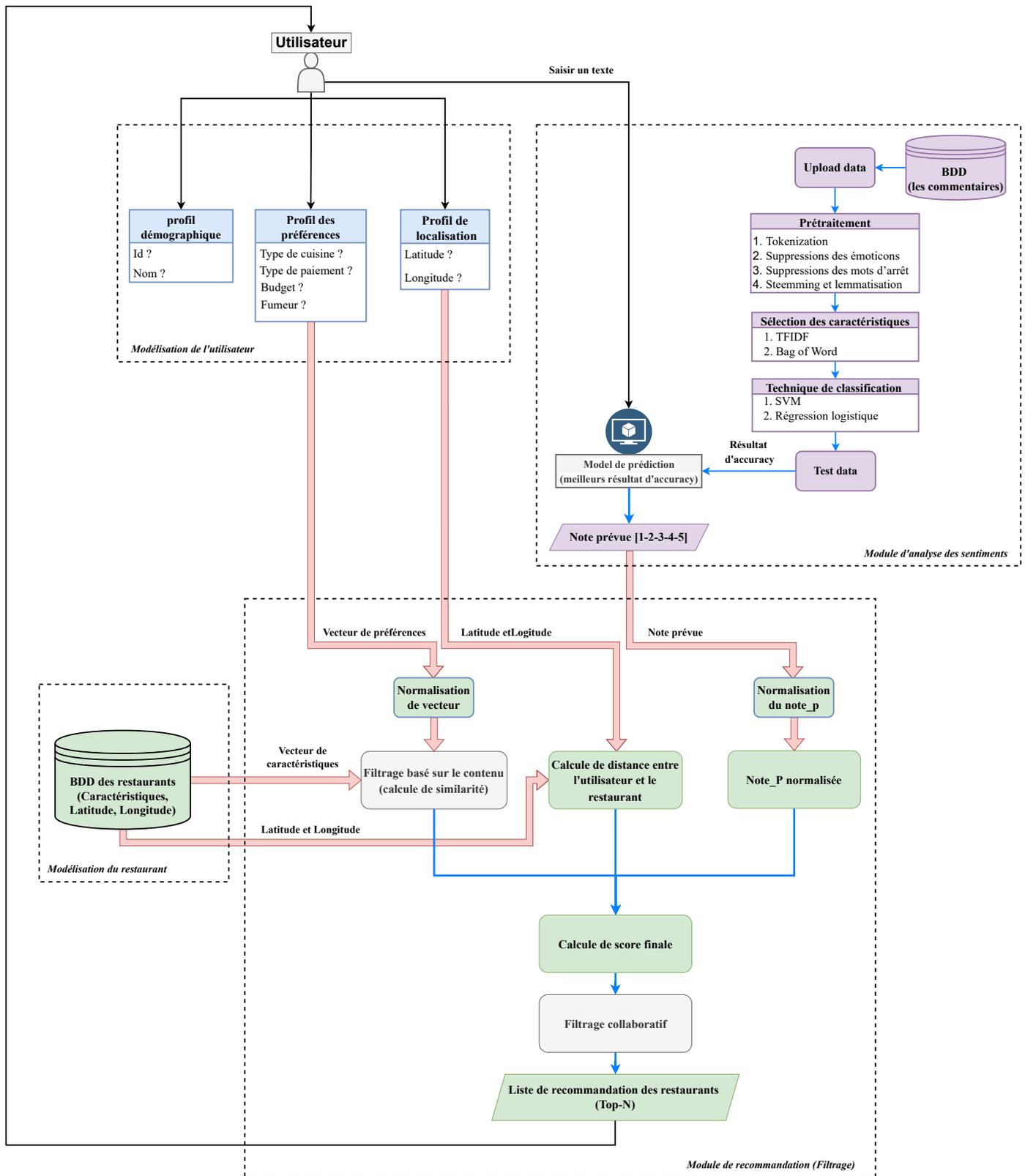


FIGURE 3.1 – L'architecture générale du système proposé

Notre approche est divisée en plusieurs composants, notamment :

- **Modélisation des utilisateurs :** ce module comprend toutes les données démographiques ou les données de localisation extraites explicitement pour créer un profil utilisateur. Étant donné que l'objectif est de fournir des suggestions individuelles, il s'agit d'un élément

important. Son but est de garder une trace de tous les profils d'utilisateurs et de les mettre à jour au fur et à mesure que des nouvelles informations deviennent disponibles.

- **Modélisation des restaurants** : déterminez quelles caractéristiques du restaurant sont saisies dans le système et comment filtrer et recommander en fonction de ses caractéristiques. En plus de ses données de localisation, ce qui vous permet de connaître son emplacement et aide à une mobilité intelligente dans la ville.
- **Module d'analyse du sentiment** : permet aux utilisateurs d'exprimer leurs opinions sur un restaurant particulier afin d'en faire une note de 1 à 5 sur lequel s'appuyer lors de l'étape de recommandation afin de fournir des suggestions cohérentes avec leurs opinions.
- **Module de recommandation** : ce module permet aux utilisateurs du système d'obtenir le résultat le plus efficace. Ce module c'est le cœur du système basé sur des informations pertinentes sur les personnes et les restaurants, il fournit une liste de recommandation à l'utilisateur actif. La section 7 examine plus en détail le fonctionnement interne de ce module.

Dans les sections suivantes, nous allons examiner de plus près ces unités et apprendre tout ce qu'il y a à savoir à leur sujet.

## 4 Modélisation de l'utilisateur

La mise en œuvre de la méthode de suggestion des restaurants repose en grande partie sur la modélisation des utilisateurs. Elle consiste à décrire les caractéristiques informationnelles des résidents et des visiteurs à l'aide du formulaire de profil. La modélisation des profils d'utilisateur est un processus en deux étapes [94].

1. La première étape implique des stratégies de collecte de données pour les utilisateurs.
2. La deuxième étape consiste à formuler et à développer le profil de l'utilisateur, ce qui peut être fait de différentes manières pour arranger les données acquises dans la structure représentative souhaitée.

### 4.1 Acquisition des données

Le processus de collecte d'informations cruciales en vue de créer un profil utilisateur est connu sous le nom acquisition des données, dans le processus de collecte des données, de nombreuses procédures de diagnostic ou d'évaluation sont utilisées. Ces techniques peuvent obtenir ces informations directement de l'utilisateur, c'est-à-dire explicitement, ou indirectement c'est-à-dire implicitement, par le biais du logiciel [94] :

- **Méthode explicite** : il s'agit d'une méthode rapide et facile pour en savoir plus sur l'utilisateur, l'utilisateur est invité à remplir des formulaires ou à être interrogé en personne pour obtenir des données personnelles et démographiques telles que son numéro d'identification, son nom, son âge, son état civil et sa profession, qui sont ensuite enregistrées pour établir un profil d'utilisateur.

- **Méthode implicite** : la méthode implicite consiste à recueillir des données auprès de l'utilisateur, à surveiller son comportement et analyser ses activités afin d'en déduire son profil. Cette technique ne nécessite pas que l'utilisateur participe directement à la recherche ; en fait, chaque interaction de l'utilisateur avec le système est considérée comme une estimation de ses intérêts.

Après avoir étudié la nature des données des citoyens, nous avons reconnu la nécessité des deux méthodes : un profil contient des informations personnelles suffisantes pour identifier un citoyen et ne peut être recueilli directement qu'au moyen de mécanismes tels que des formulaires. Lorsque le système est utilisé sans la participation directe du citoyen, d'autres données, telles que les données environnementales et comportementales, doivent être acquises implicitement [95].

## 4.2 Construction des profils

Une bonne gestion des meilleures recommandations permet d'ajuster les résultats en fonction des exigences du touriste et du citoyen en créant correctement leurs profils à partir de leurs données d'information. Pour cette raison, il existe trois types de base de représentation de profil [96] :

- **Représentation de l'assemblage** : le profil est généralement exprimé sous forme de vecteur des mots pondérés ou de classes de vecteurs.
- **Représentation sémantique** : la représentation du profil met l'accent sur les relations sémantiques entre les données qu'il contient. Les ontologies ou les réseaux sémantiques probabilistes sont largement utilisés dans cette représentation.
- **Représentation multidimensionnelle** : le profil est structuré autour de nombreuses dimensions qui sont affichées de différentes manières [96].

L'idée de profil est souvent liée aux préférences et au contexte, en plus des informations factuelles sur l'individu [97] :

- Un profil d'utilisateur est une collection de données sur un individu.
- Un contexte est une collection de données qui explique l'environnement d'interaction entre l'utilisateur et le système.
- Une préférence est un terme ou une expression qui nous permet de catégoriser l'importance des faits dans un profil ou un contexte.

En effet, les préférences d'un citoyen sont un élément important de sa description, et elles peuvent changer en fonction du scénario dans lequel il se trouve. L'objectif de la construction est d'offrir un modèle générique qui peut être utilisé pour expliquer un utilisateur, son environnement et ses préférences à l'aide d'un ensemble de caractéristiques ou de concepts qui peuvent être adaptés, améliorés et instanciés selon les besoins.

Nous avons analysé le profil en plusieurs dimensions qui reflètent les trois concepts tels que les données de localisation, le contexte et les préférences, qui sont les principaux aspects

qui composent le profil de l'utilisateur de notre système, et avons utilisé la représentation multidimensionnelle la plus appropriée pour décrire les informations sur les citoyens. mieux adapté à cet objectif.

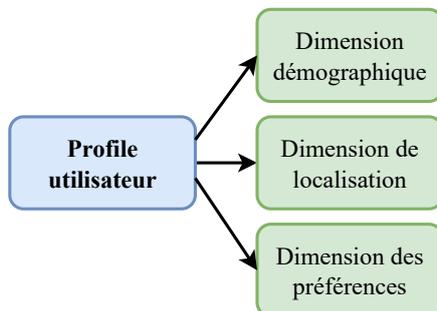


FIGURE 3.2 – Les dimensions d'un profil utilisateur

#### 4.2.1 Profil démographique

Cette dimension représente la partie du profil de l'utilisateur qui est fixé lors de son inscription initiale en remplissant un formulaire qui rassemble toutes les informations du citoyen et qui est formé en deux sortes de données : la première est l'identifiant de l'utilisateur, et la deuxième qui contient des informations telles que le nom de la personne, âge, sexe, etc. Ces informations relèvent de la catégorie des données personnelles.

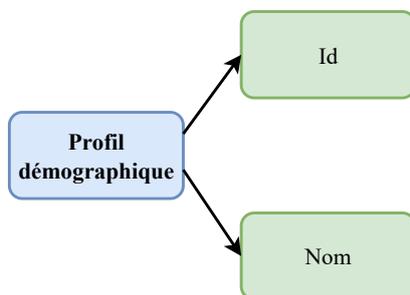


FIGURE 3.3 – Le profil démographique

#### 4.2.2 Profil de localisation

Cette dimension représente la partie dynamique du profil qui contient toutes les informations nécessaires pour définir la localisation actuelle du citoyen. Il s'agit principalement de l'élément spatial nécessaire pour décrire la position géographique actuelle du citoyen.

Les coordonnées de latitude et longitude sont deux caractéristiques de la géolocalisation qui indiquent l'emplacement de l'utilisateur. En raison de la sensibilité de ces informations et du fait qu'elles changent en temps réel, une mise à jour est effectuée à chaque demande pour éviter de mettre à jour des informations qui ne sont plus utiles et pour diminuer le nombre de mises à jour non sollicitées.

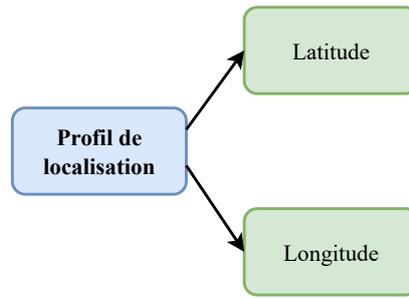


FIGURE 3.4 – Le profil de localisation

### 4.2.3 Profil des préférences

Le profil des préférences peut être établi à partir de critères relatifs aux restaurants qui sont examinés au moment de l'entrée dans le système. Les éléments les plus importants sont le type de cuisine, la gamme de prix, le parking et fumeur oui/non.

Pour créer un profil des préférences, l'utilisateur doit préciser correctement ses préférences. Chaque élément offre une variété d'options, telles que des gammes de prix basses, moyennes et élevées, ainsi que des nombreux styles de nourriture tels que l'italien, le chinois et l'espagnol etc.

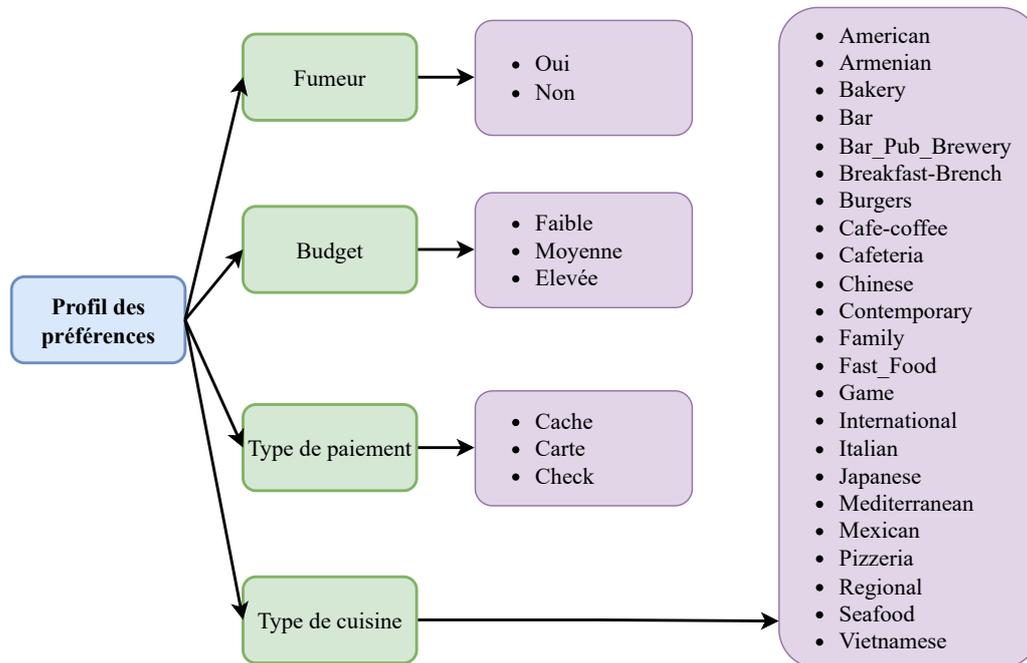


FIGURE 3.5 – Le profil des préférences

## 5 Modélisation du restaurant

L'entité restaurant représente la notion de base qui consiste à donner une liste d'idées, chaque restaurant étant représenté par un ensemble d'attributs que l'utilisateur peut person-

naliser en fonction de ses exigences et de ses goûts.

Nous avons examiné les conditions préalables ou les éléments de réussite d'un restaurant à service complet en raison du degré croissant de concurrence dans le secteur de la restauration. Les caractéristiques du restaurant qui influencent l'impression du client nous ont particulièrement intéressés. Nous avons donc dressé dans la figure 3.6 une liste des caractéristiques les plus répandues [98].

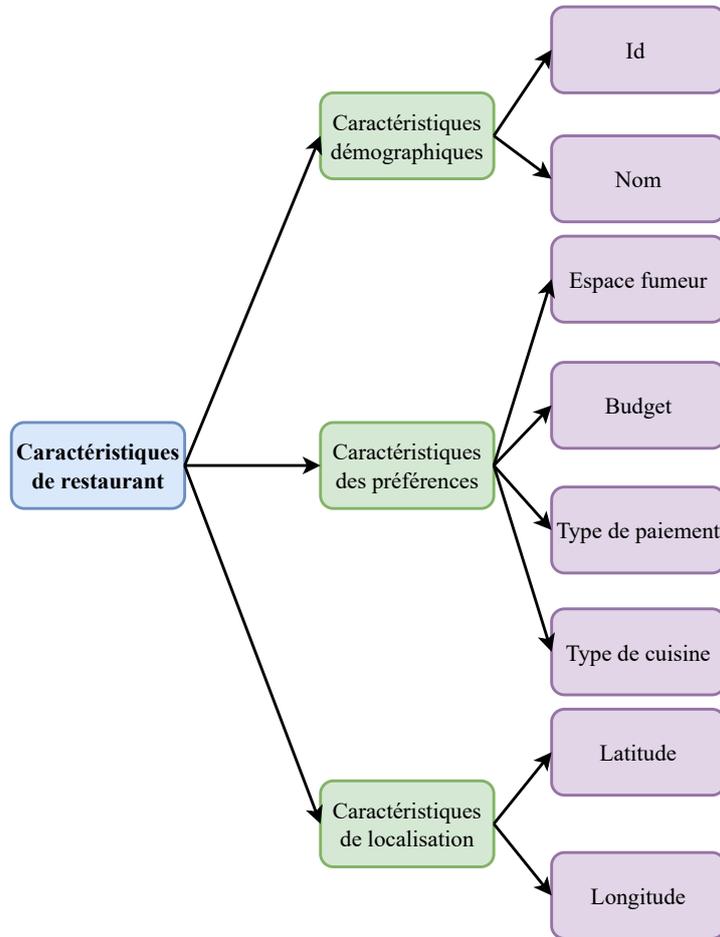


FIGURE 3.6 – Les caractéristiques de restaurant

## 5.1 Caractéristiques

Dans le monde réel, chaque restaurant possède plusieurs caractéristiques qui définissent son profil. Nous avons choisi quelque caractéristique telle que :

- Les caractéristiques contextuelles comme l'identifiant et le nom du restaurant définissent les informations relatives.
- La fonction qui aident l'utilisateur à localiser le restaurant ou à calculer la distance, les coordonnées de latitude et longitude de l'emplacement du restaurant sont incluses dans cette fonctionnalité.
- Les critères de préférence les plus importants sont : le mode de paiement, le budget du restaurant, la zone fumeur et le type de cuisine.

Si le restaurant A propose une cuisine "chinoise" et que le restaurant B sert à la fois une cuisine "chinoise" et une cuisine "italienne", la cuisine "chinoise" peut être considérée comme une caractéristique du restaurant A, tandis que la cuisine "chinoise" et la cuisine "italienne" peuvent être considérées comme des caractéristiques du restaurant B. Ces caractéristiques, ainsi que d'autres sont utilisées pour sélectionner le restaurant le plus approprié pour l'utilisateur.

## 6 Module d'analyse des sentiments

L'analyse des sentiments est une approche informatique basée sur le traitement du langage naturel (NLP) qui vise à apprendre aux ordinateurs à comprendre le langage humain de la même manière que les gens [99]. Afin de distinguer et de catégoriser une opinion textuelle exprimant l'attitude positive, négative ou neutre de l'auteur, ou de la convertir en une note de 1 à 5 pour un produit, un événement ou une personne [100].

Une fois nos données exportées, un processus d'analyse des sentiments en trois étapes sont effectués, à savoir le prétraitement, la sélection des fonctionnalités et la classification. La structure de ce processus est illustrée dans la figure 3.8, puis l'ensemble de données est formé à l'aide de deux classificateurs d'apprentissage automatique (SVM et RL), avec des taux de précision des tests unitaires utilisés pour comparer les méthodologies. Nous avons choisi l'algorithme le plus précis et l'avons appliqué aux commentaires pour une analyse des sentiments en temps réel. Nous étudions nos principales unités et sous modules en détail dans les sections suivantes [101].

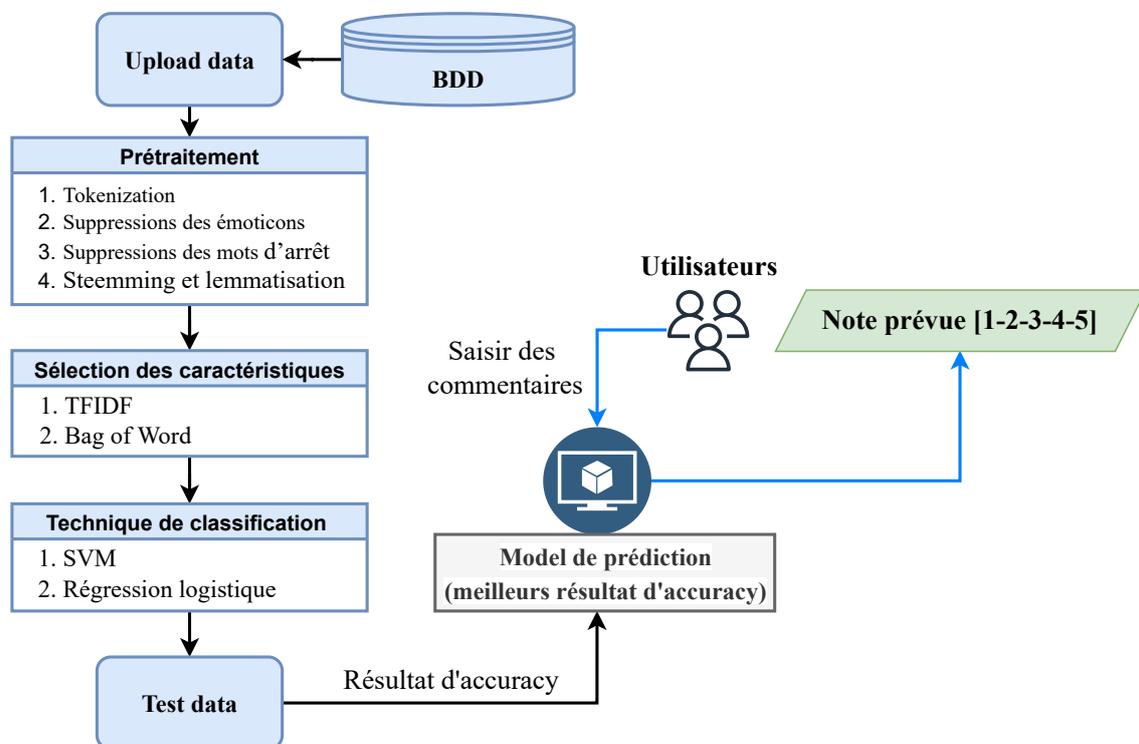


FIGURE 3.7 – Le processus d'analyse des sentiments [101]

## 6.1 Prétraitement et extraction des données

Tout au long de l'étape de préparation, nous devons nettoyer les données textuelles selon diverses méthodes. La pertinence de cette étape s'accroît lorsqu'il s'agit de données textuelles extraites du Web où des médias sociaux, car ces données sont souvent bruyantes et comprennent des nombreuses données relatives à la construction du Web, à des balises comme le HTML, aux espaces blancs ou à la publicité [102].

- **Tokenization** : tokenization est la représentation d'une seule notion par un seul mot ou une seule phrase. La tokénisation d'un texte s'effectue en le divisant en tokens séparés par des espaces, comme dans l'exemple "the quick fox jumped over the lazy dog", qui produit une liste de tokens [the, quick, fox, jumped, over, the, lazy, dog]. La séparation d'une chaîne de caractères en espaces n'est pas pertinente en termes techniques puisqu'un jeton peut inclure un certain nombre d'expressions. Des bibliothèques ou des plugins de tokénisation sont disponibles pour la plupart des langages de programmation actuels, notamment python, R et Java.
- **Suppression des émoticônes** : les utilisateurs utilisent fréquemment des émoticônes de joie, de tristesse et de colère pour exprimer leurs sentiments. Dans certaines circonstances, les émoticônes sont employées pour véhiculer le sarcasme, une tactique linguistique sophistiquée qui est populaire sur les médias sociaux. Considérez cette déclaration : "Je me suis éclaté à pelleter l'allée aujourd'hui! :)" Il est difficile de savoir si l'opinion d'un utilisateur sur un produit, une personnalité ou un événement est positive ou négative lorsque des émoticônes sont employées dans la communication. Pour cette étude, nous avons supprimé les émoticônes de notre ensemble de données d'entraînement.
- **Suppression des mots d'arrêt** : en anglais, les mots d'arrêt incluent the, is, at, which, on, et d'autres. Ils produisent du contenu lexical et peuvent conduire à l'oubli de résultats significatifs. Les mots d'arrêt ont été omis de notre base de données car ils sont trop larges pour être utiles. Ils peuvent même entraîner la confusion d'un système d'apprentissage automatique.
- **Stemming et lemmatization** : stemming est une approche de la recherche d'informations qui aborde le problème de l'inadéquation lexicale, qui survient lorsque les mots d'une requête ne correspondent pas au terme d'un document. Les différentes variantes d'un même mot sont mises sur un pied d'égalité par les stemmers, comme par exemple (so, sooo). En anglais et dans beaucoup d'autres langues, le stemming est en grande partie un processus d'abstraction de suffixes.

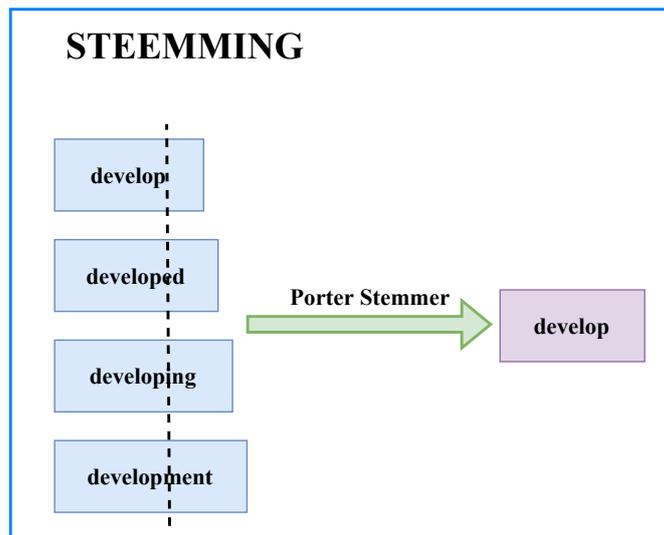


FIGURE 3.8 – L'opération de stemming [103]

La forme d'un mot que l'on peut trouver dans un dictionnaire est appelée lemmatization, donc on applique premièrement le processus de transformation d'un token est qui appelé la normalisation de l'orthographe (par exemple, remplacer "tonight" et "tonite" par "tonight") et l'extension des abréviations sont des exemples de lemmatization (par exemple, remplacer "DOE" par "dyspnea on exertion"). En linguistique, la forme fondamentale d'un mot est appelée lemmatization.

La plupart des langages de programmation modernes, y compris C, R, Python, Ruby et Java, fournissent des bibliothèques ou des modules d'extension qui peuvent effectuer l'extraction et lemmatization des mots anglais.

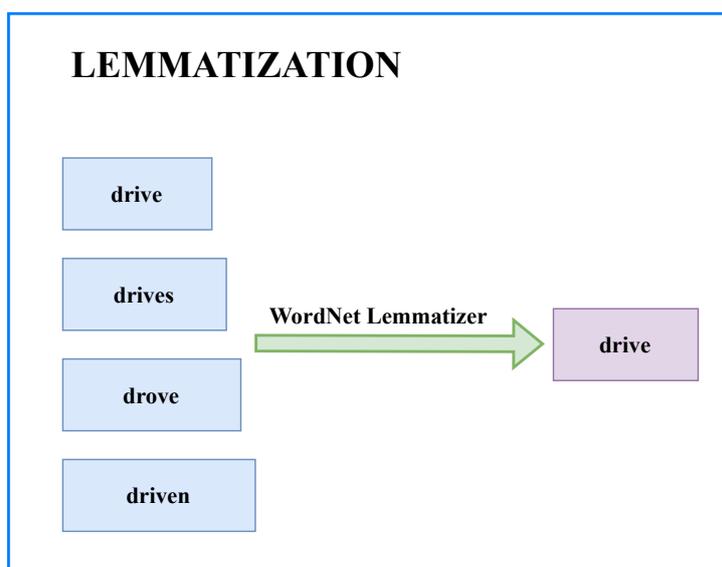


FIGURE 3.9 – L'opération de lemmatization [103]

## 6.2 Sélection des caractéristiques

Plusieurs chercheurs ont essayé d'extraire les meilleures caractéristiques qui pourraient servir de bon point de départ pour la catégorisation des émotions et la sélection autonome des caractéristiques [102]. Parmi les meilleures méthodes de sélection des caractéristiques, nous avons choisi les méthodes les plus connues pour les étudier.

- **TfidfVectorizer** : comme les algorithmes d'apprentissage automatique ne traitent que des données numériques, nous devons transformer les données en une représentation vectorielle numérique pour la catégorisation des textes. En général, il existe deux méthodes pour ce faire : l'utilisation du Count Vectorizer et l'utilisation du Tfidf Vectorizer. Le Count Vectorizer, une approche relativement simple est utilisé pour compter le nombre de mots dans un document. Le score Tfidf, quant à lui, sera utilisé comme donnée numérique pour le modèle vectoriel dans Tfidf Vectorizer [101]. Par exemple en prend les trois documents suivants ['problem of devil', 'devil queen', 'horizon problem'] et faire l'application de Tfidf-Vectorizer.

Document	devil	horizon	of	problem	queen
0	0.517	0.000	0.680	0.517	0.000
1	0.605	0.000	0.000	0.000	0.795
2	0.000	0.795	0.000	0.605	0.000

TABLE 3.1 – Exemple explicatif de TfidfVectorizer [101]

- **Bag of Word** : un modèle de comptage des mots peut être utilisé pour extraire des caractéristiques des données textuelles. Un algorithme, comme nous le savons tous, ne saisit pas le langage. Par conséquent, nous devons représenter numériquement les mots du corpus. Toute méthode peut ensuite utiliser cette représentation numérique pour un examen plus approfondi. Le concept de base du BoW consiste à compter le nombre des mots dans un texte. Il est important de noter que chaque mot dans l'idée de bag of word est traité séparément, et que l'ordre dans lequel les mots apparaissent n'a aucune importance [104].

	the	red	dog	cat	eats	food
1. the red dog	1	1	1	0	0	0
2. cat eats dog	0	0	1	1	1	0
3. dog eats food	0	0	1	0	1	1
4. red cat eats	0	1	0	1	1	0

FIGURE 3.10 – Exemple explicatif de Bag of Word [104]

### 6.3 Technique de classification

Plusieurs systèmes d'apprentissage automatique utilisent des techniques de classification telles que le SVM et les réseaux de neurones après avoir transformé l'ensemble de données dans TfidfVectorizer. Dans [105], Van et al utilisé ces classificateurs pour sélectionner des caractéristiques à partir de données biographiques relationnelles, en outre, dans [106], Kang et al a proposé d'utiliser le classificateur Naïve Bayes pour sélectionner les caractéristiques des critiques des restaurants.

Afin de former une classification efficace pour notre modèle de prédiction, nous utilisons parmi les méthodes de classification citées dans la sous-section 3.3 du chapitre 1 le SVM et la régression logistique, qui sont beaucoup plus utilisés sur les systèmes basés sur le contexte et employés pour de nombreuses applications de fouille de textes, y compris l'analyse des sentiments.

### 6.4 Résultat d'analyse des sentiments

En utilisant des approches d'apprentissage automatique, nous avons effectué une analyse des sentiments sur les résultats de notre système. Dans nos expériences, la méthodologie SVM a surpassé les approches de régression logistique en matière de précision.

Nous avons choisi la technique SVM pour l'analyse des sentiments en temps réel en raison de sa précision, et nous avons construit notre modèle d'analyse des sentiments pour prédire la note de l'utilisateur à partir de son texte de commentaire, que nous avons utilisé dans la phase de recommandation pour obtenir des résultats plus efficaces.

## 7 Module de recommandation - Filtrage

Le composant fondamental de notre système est l'unité de recommandation, où sont calculées les suggestions qui conduisent aux résultats. Nous montrons comment nous avons utilisé les procédures de filtrage collaboratif et de filtrage basé sur le contenu pour construire un modèle très précis basé sur les idées et les recherches décrites dans la section 4 du chapitre 1.

Afin de présenter une plus grande diversité d'idées, nous privilégions la méthode de filtrage mixte dans notre stratégie de filtrage. Ce système de recommandation hybride utilise à la fois des techniques de filtrage collaboratif et de filtrage basé sur le contenu. Le concept est simple comme mentionné précédemment dans la sous section 2.2.3 du chapitre 1, une approche hybride est employée pour intégrer les deux méthodologies.

Cependant, avant de commencer à travailler sur le moteur de recommandation, nous devons d'abord définir comment les utilisateurs et les restaurants seront représentés dans le système, ainsi que la manière dont ils interagiront.

## 7.1 Représentation de restaurant et utilisateur

Des vecteurs à  $N$  dimensions sont utilisés pour représenter les utilisateurs et les restaurants, chaque composante reflétant un attribut. Les caractéristiques booléennes sont simples et chacune est représentée par un seul élément du vecteur des caractéristiques.

### 7.1.1 Restaurant

Un restaurant est représenté par un vecteur des caractéristiques qui décrit la qualité d'un restaurant, chacun des attributs booléens ayant la valeur 1 ou 0.

Toutes les caractéristiques d'un restaurant peuvent être concaténées en un vecteur  $F$  avec 4 caractéristiques de taille  $n = 31$ .

$$F = (f_1, f_2, \dots, f_n) \quad (3.1)$$

Où  $f_i$  est la  $i$ ème caractéristique du restaurant. S'il y a  $f_i$  dans un restaurant, alors  $f_i$  égale à 1 sinon  $f_i$  égale 0.

Comme le montre le schéma 3.11, une matrice peut être construite en fusionnant les caractéristiques et les restaurants actuels. Les restaurants sont représentés par les lignes, tandis que les caractéristiques sont représentées par les colonnes de la matrice.  $(r_i, f_j) = 1$  si le  $i$ ème restaurant possède la  $j$ ème caractéristique. Il convient de noter que chaque restaurant peut fournir une ou plusieurs caractéristiques.

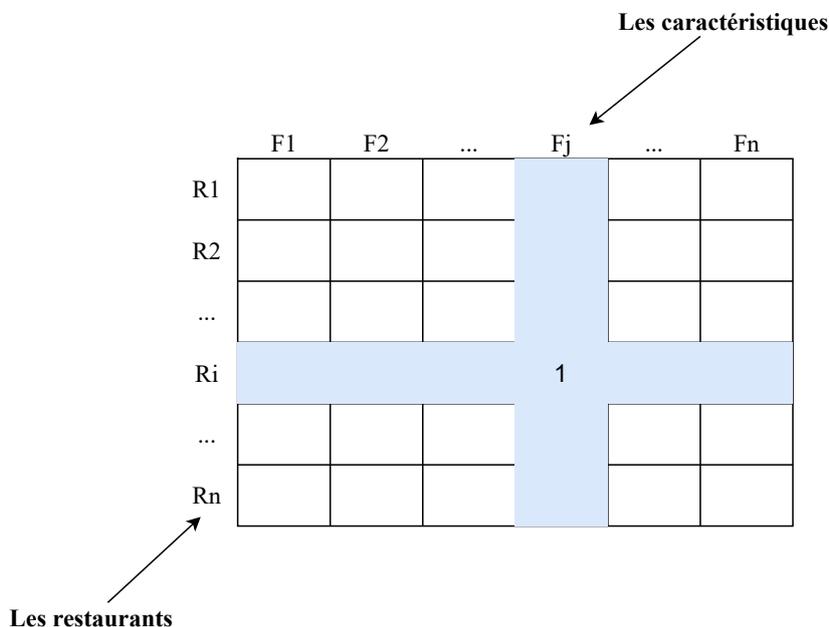


FIGURE 3.11 – La matrice des caractéristiques des restaurants

### 7.1.2 Utilisateur

Le vecteur d'un utilisateur contient des informations sur ses préférences ainsi qu'une paire de valeurs 0 et 1, qui montrent son affinité pour des caractéristiques spécifiques. Toutes les

préférences de l'utilisateur extraites de 5 caractéristiques peuvent être concaténées en un vecteur  $P$  de taille  $n = 31$ .

$$P = (P1, P2, \dots, Pn) \tag{3.2}$$

Les caractéristiques du restaurant ont été explorées dans la section précédente. Ces caractéristiques peuvent être utilisées pour représenter le vecteur des préférences d'un utilisateur puisqu'elles peuvent indiquer les choix de l'utilisateur. Lorsqu'un utilisateur utilise initialement le système proposé, il est invité à établir la valeur de départ de son vecteur des préférences en fonction des attributs uniques du restaurant ; une valeur de 0 indique une attitude négative à l'égard de l'attribut, tandis qu'une valeur de 1 indique une attitude positive (voir figure 3.12).

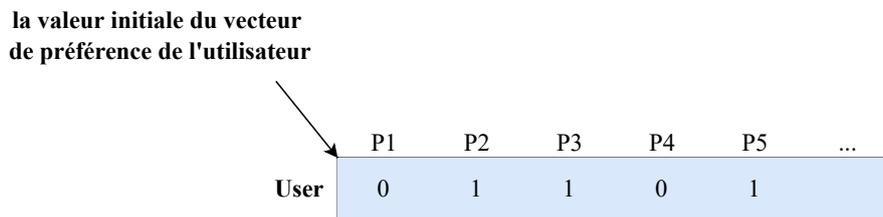


FIGURE 3.12 – Vecteur des préférences de l'utilisateur

Le modèle des préférences de l'utilisateur est illustré à la figure 3.13. L'utilisateur possède les préférences  $P2$ ,  $P3$  et  $P5$  lorsqu'on lui donne un restaurant. Si un utilisateur visite ce restaurant, les caractéristiques du restaurant seront ajoutées aux préférences de l'utilisateur.

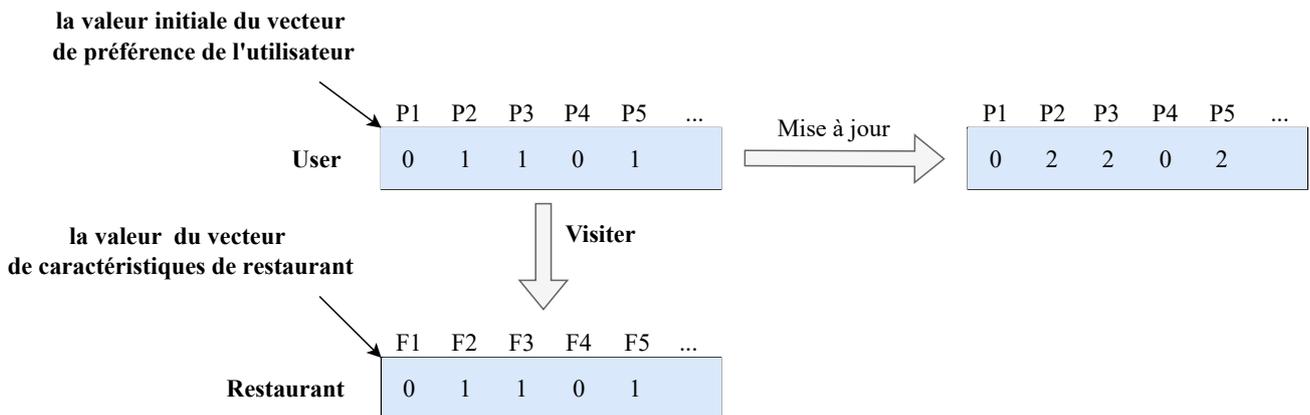


FIGURE 3.13 – Mise à jour de vecteur des préférences de l'utilisateur

Au fur et à mesure que le nombre des restaurants visités augmente, les caractéristiques préférées de l'utilisateur deviennent plus évidentes, donc nous utilisons la formule (3.3) pour normaliser la valeur des caractéristiques choisies par l'utilisateur :

$$Norm(P_i) = \frac{P_i}{\sum_{j=0}^n P_j} \tag{3.3}$$

De cette façon, la norme  $P_i$  sera limitée dans  $[0, 1]$ , et représentant les préférences d'un utilisateur, par exemple le vecteur  $P = (0, 2, 2, 1)$  sera après la normalisation et l'utilisation de la formule (3.3)  $P = (0, 0.5, 0.5, 0.25)$ . Par conséquent,  $P = (p_1, p_2, \dots, p_n)$  peut-être utilisé pour représenter le vecteur des préférences de l'utilisateur.

$L_u = (latitude, longitude)$  et  $L_r = (latitude, longitude)$  sont des vecteurs de position pour l'utilisateur et le restaurant, respectivement, qui intègrent leurs positions respectives sur la base de données des coordonnées géographiques pour estimer la distance entre eux.

## 7.2 Facteurs de recommandation

Le système de recommandation proposé utilise des méthodes statistiques et une analyse exploratoire de données, et prend en compte trois facteurs.

- La note anticipée du processus d'analyse des sentiments.
- Le filtrage basé sur le contenu qui détermine le degré de similarité entre l'utilisateur et le restaurant.
- La distance réelle entre le restaurant et l'utilisateur.

Cette section couvre l'ensemble de l'approche de recommandation du restaurant.

### 7.2.1 Note prévue à partir de commentaire

Chaque restaurant dispose plusieurs avis fournis par des visiteurs et des utilisateurs dans le monde réel, parce qu'ils permettent une interaction entre les utilisateurs et les restaurants. Par conséquent, dans les systèmes contextuels, dont l'objectif principal est de suggérer des restaurants de manière appropriée, les notes des restaurants sont essentielles.

Nous utilisons la méthode d'analyse des sentiments comme une boîte noire pour prévoir la note au fur et à mesure que nous progressons dans les phases de la procédure de prédiction, en permettant à l'utilisateur de donner son avis et son opinion sur le restaurant et en transformant cette entrée en une note réelle, comme le montre dans la figure 3.14.

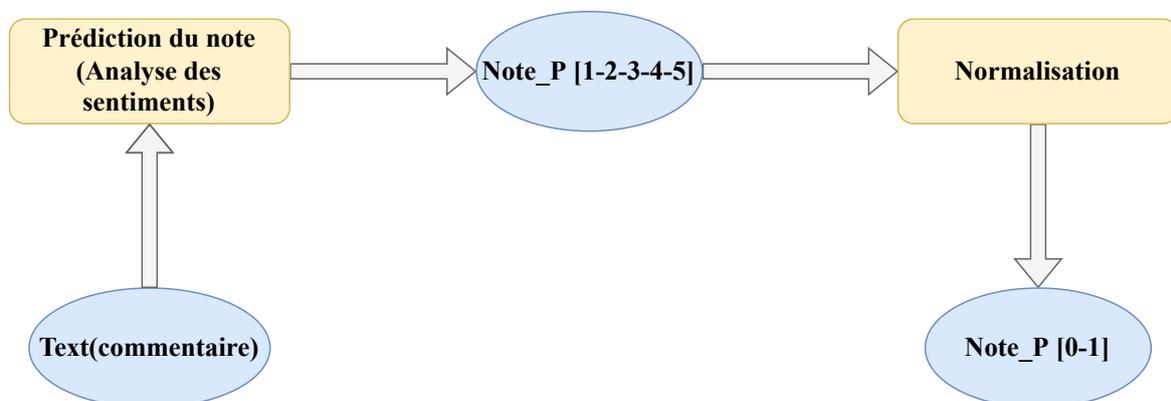


FIGURE 3.14 – La prédiction du note à partir de commentaire

La note de l'étape d'analyse des sentiments est essentiel pour la création de l'algorithme de recommandation et l'extraction d'un résultat de suggestion efficace.

Comme la valeur de la note du commentaire de l'utilisateur varie de 1 à 5, nous avons utilisé l'opération de normalisation de note prévue pour limiter ces valeurs à l'intervalle compris entre  $[0, 1]$ . La normalisation de note prévue est indiquée dans la formule (3.4).

$$Norm(Nota\_P(i)) = \frac{Nota\_P(i)}{\sum_{j=0}^n Nota\_P(j)} \quad (3.4)$$

### 7.2.2 Filtrage basé sur le contenu - Calcule de similarité

La recommandation par paires (élément/utilisateur) est basée sur le contenu qui détermine quels restaurants sont les plus similaires au profil de l'utilisateur. Par exemple, un utilisateur qui a donné des évaluations positives à plusieurs restaurants asiatiques aura un profil qui ressemble à un restaurant asiatique.

L'objectif est de développer une technique permettant de fusionner un utilisateur  $U$ , dont le vecteur des préférences  $Pi = (p1, p2, \dots, pn)$ , et un restaurant  $R$ , dont le vecteur des caractéristiques  $Fi = (f1, f2, \dots, fn)$ , pour obtenir des suggestions des restaurants de la part d'autres utilisateurs, nous devons calculer la similarité pour mesurer le niveau de ressemblance entre les deux vecteurs afin qu'ils puissent proposer des restaurants en fonction des préférences de l'utilisateur.

Pour savoir quel sont les restaurants que l'utilisateur préfère, nous devons trouver un moyen de calculer la similarité entre l'utilisateur et le restaurant. Dans la section 3.2 du chapitre 1, nous avons discuté de nombreux modèles de similarité qui pourraient être utilisés pour déterminer cette similarité et nous avons choisi d'utiliser une métrique de distance en cosinus pour ce système (3.5).

$$Sim(U, R) = cos(U, R) = \frac{\sum_{i=1}^n PiFi}{\sqrt{\sum_{i=1}^n Pi^2} \times \sqrt{\sum_{i=1}^n Fi^2}} \quad (3.5)$$

La fonction de similarité renvoie toujours la valeur entre  $[0, 1]$ , si la valeur est proche de 1 alors le restaurant et l'utilisateur sont très similaires sinon ne sont pas similaires.

### 7.2.3 Distance entre utilisateur et restaurant

Les utilisateurs sont plus susceptibles de choisir le restaurant le plus proche puisque le système de suggestion des restaurants recommandés est conçu pour les appareils mobiles. Par conséquent, la distance entre l'utilisateur et le restaurant est un facteur important qui peut influencer le résultat de la proposition pour localiser l'utilisateur et le restaurant, la méthode proposée exploite les données de localisation explicites telles que latitude et longitude. Par conséquent, elle calcule la distance entre les deux endroits.

Dans cette partie, nous allons suivre la procédure de détermination de la distance à partir de [90], étant donné deux coordonnées  $A1(x1, y1)$  et  $A2(x2, y2)$  on fixe  $a = \cos(x1 - x2)$ ,  $b = \cos(y1 - y2)$ ,  $c = \cos x1 * \cos x2$  et  $Rearth$  le rayon de la terre.

Pour calculer la distance entre  $A1$  et  $A2$ , on applique la formule (3.6).

$$Distance(A1, A2) = 2.Rearth.arcsin \left( \frac{\sqrt{2.(1 - a + c - c.b)}}{2} \right) \quad (3.6)$$

Cette fonction renvoie un nombre réel de distance qui n'est pas compris entre 0 et 1, nous utilisons donc la formule suivante (3.7) pour le transformer en une distance comprise entre 0 et 1, prêtons pour le prochain calcul de filtrage.

$$Dis(A1, A2) = 1 - \frac{Distance(A1, A2)}{Rearth} \quad (3.7)$$

### 7.3 Algorithme de recommandation

Dans cette section, nous définissons les concepts qui nous permettent d'effectuer le filtrage en utilisant deux algorithmes :

- La première méthode c'est la formule qui calcule les scores finals entre l'utilisateur et tous les restaurants contiennent un algorithme de filtrage basé sur le contenu pour calculer la similarité.
- Le second algorithme est l'algorithme de SlopeOne qui permet de faire le filtrage collaboratif entre les utilisateurs.

Nous avons utilisé ces deux algorithmes pour créer une liste finale de recommandation comprenant les restaurants qui sont similaires au profil de l'utilisateur.

#### 7.3.1 Score finale

Lorsqu'un utilisateur est connecté au système de recommandation, un score final est produit pour chaque utilisateur avec tous les restaurants voisins en utilisant quatre entrées :

- Le commentaire, qui reflète l'opinion de l'utilisateur sur un certain restaurant ;
- Le vecteur qui représente le modèle des préférences de l'utilisateur  $U$  ;
- Le vecteur qui représente les caractéristiques du restaurant  $R$  ;
- La distance réelle du restaurant par rapport à la position actuelle de l'utilisateur.

L'algorithme prend en compte la note prévue du restaurant par l'utilisateur telle que déterminé par son texte de commentaire à partir de processus d'analyse des sentiments, le taux de similarité et la distance entre l'utilisateur cible avec tous les restaurants pour calculer le score final.

Le score final de chaque restaurant sera calculé à l'aide de la formule (3.8), qui combine les trois facteurs pour obtenir le score final d'un utilisateur cible avec tous les restaurants.

$$Score\_f(U, Ri) = \alpha Note\_P(i) + \beta Sim(U, Ri) + \gamma Dis(U, Ri) \quad (3.8)$$

Dans l'équation (3.8), *alpha*, *bêta* et *gamma* sont des poids ; chacun commence par une valeur de 1/3, et la somme totale des poids est égale à 1.

Le système renvoie les résultats selon les facteurs précédents, selon la méthodologie, le facteur avec le score le plus élevé alors la recommandation des restaurants dépend souvent de ce facteur, par exemple si deux restaurants ont la même similarité, la même note et la distance attendue est différente, le restaurant le plus proche obtiendra un score plus élevé. Si deux restaurants sont à la même distance de l'utilisateur et ont la même note attendue et présentent de similarité différent, le restaurant le plus similaire obtiendra un score plus élevé. La différence dans les résultats est due aux poids similaires entre les facteurs.

Le score final de chaque restaurant sera utilisé sur l'étape suivante qui parler sur la technique de filtrage collaboratif et sa contribution sur l'augmentation de qualité du résultat de recommandation.

### 7.3.2 Filtrage collaboratif

Le filtrage collaboratif est un mécanisme de recommandation axé sur l'utilisateur, ce qui signifie que l'algorithme n'a pas besoin de savoir quoi que ce soit sur les restaurants, l'objectif est de promouvoir les restaurants populaires auprès des clients ayant des goûts similaires. En pratique, cela se fait en créant la matrice des scores finaux utilisateur-restaurant calculée à l'étape précédente.

Il est généralement considéré comme une méthode permettant de prévoir comment un utilisateur évaluera un certain élément en fonction des évaluations des autres utilisateurs. Nous avons inclus un exemple de matrice des notes dans la figure 3.15 pour aider à clarifier le concept de filtrage collaboratif.

		Les utilisateurs										
		U1	U2	U3	U4	U5	U6	U7	U8	U9	...	Un
les restaurants	R1	1	1		1		1	1				
	R2				1	1	-1	1		1		
	R3		1		1		1		1			
	R4	-1	-1			-1	-1	1				
	R5		1	1		1	-1	1	-1			
	...											
	Rn											

FIGURE 3.15 – Les positions de l'utilisateur et les restaurants

Lorsque deux utilisateurs n'ont aucune évaluation en commun, comme  $U1$  et  $U8$ , leur score de similarité est de 0. Les utilisateurs ayant des opinions exactement opposées sur les restaurants, comme  $U3$  et  $U6$ , auront un score de similarité proche de 0. Les utilisateurs qui classent les mêmes restaurants de manière similaire, par exemple  $U2$  et  $U4$  obtient une note de 1.

- **Algorithme de SlopeOne** : SlopeOne est une famille d'algorithmes utilisés pour le filtrage collaboratif. Il s'agit de la forme la plus simple de filtrage collaboratif non trivial basé sur les notes. La méthode de filtrage collaboratif "SlopeOne" est une technique traditionnelle de matrice de notation "utilisateur/élément". Par rapport aux techniques de filtrage collaboratif standard basé sur les utilisateurs et les éléments, l'algorithme est simple à concevoir, à maintenir et à utiliser, et il est capable de générer des avis d'utilisateurs sur des suggestions en l'absence de données [107]. Le concept principal de cette méthode est d'utiliser la moyenne plutôt que la différence des scores entre deux inconnus. La simplicité de cette méthode la rend particulièrement facile à mettre en pratique [108].

Les techniques de SlopeOne utilisent les données des autres utilisateurs qui ont évalué le même restaurant ainsi que les restaurants évalués par le même utilisateur. La méthode SlopeOne suppose un lien linéaire entre les deux éléments, avec  $f(x) = x + b$  représentant l'écart moyen et  $x$  étant la note de l'élément précédemment noté [109].

Les évaluations de l'utilisateur  $v$  aux éléments  $i$  et  $j$  sont notées  $r(v, i)$  et  $r(v, j)$  pour chaque utilisateur  $v (v \in S(i, j))$ , l'écart moyen des éléments  $i$  et  $j$  est alors :

$$dev(i, j) = \frac{1}{card(S(i, j))} \sum_{v \in S(i, j)} (r(v, i) - r(v, j)) \quad (3.9)$$

Où  $S(i, j)$  est l'ensemble des utilisateurs qui évaluent à la fois l'élément  $i$  et l'élément  $j$ . Après avoir obtenu l'écart moyen à l'aide de l'équation (3.9), le score de prédiction de l'utilisateur  $u$  pour l'élément  $i$  peut être calculé comme suit :

$$r(u, i) = \frac{1}{card(Ru)} \sum_{j \in Ru} (dev(i, j) + r(u, j)) \quad (3.10)$$

Où  $Ru$  est l'ensemble de tous les éléments pertinents,  $r(u, j)$  est l'évaluation de l'élément  $j$  par l'utilisateur  $u$ , et  $dev(i, j)$  est calculé à l'aide de l'équation (3.9).

Voici l'exemple suivant qui comprennent les avis de cinq restaurants soumis par les utilisateurs A, B, C et D, respectivement :

Utilisateurs \ Restaurants	Restaurants				
	1	2	3	4	5
A	3	4	2	-	3
B	2	2	3	2	2
C	4	-	3	5	4
D	-	3	4	-	-

TABLE 3.2 – Matrice de score utilisateur-restaurant

Si nous voulons déterminer la note de l'utilisateur D pour le cinquième restaurant, nous devons d'abord calculer la valeur moyenne de la différence entre les notes de l'utilisateur D pour le deuxième et le troisième restaurant. Le processus de calcul est le suivant :  $((3 - 4) + (2 - 2))/2 = -0.5$ ,  $((3 - 2) + (2 - 3) + (4 - 3))/3 = 0.3$

Le score de prédiction D de l'utilisateur pour le restaurant 5 peut donc être calculé en ajoutant les scores D de l'utilisateur pour les restaurants 2 et 3 à la moyenne arithmétique :  $((3 - 0.5) + (4 + 0.3))/2 = 3.4$

### 7.3.3 Résultat de recommandation

Le système de recommandation est un outil puissant pour fournir aux utilisateurs des informations personnalisées et utiles. Ce système de recommandation utilise un modèle des préférences de l'utilisateur basé sur les caractéristiques des restaurants visités, ainsi que des informations géographiques sur l'utilisateur et les restaurants, ainsi que des avis et des prédictions d'évaluation des utilisateurs et des restaurants pour fournir des résultats recommandés de manière dynamique.

Nous avons utilisé deux techniques de filtrage pour déterminer un résultat efficace pour l'utilisateur cible qui demandant des recommandations des restaurants : le filtrage basé sur le contenu, qui calcule la similarité comme facteur principal pour calculer le score final, et le filtrage collaboratif qui utilise par l'algorithme de SlopeOne pour donner un résultat de recommandation final plus précis.

Le tableau 4.1 donne un simple échantillon de ce qui sera montré à l'utilisateur cible ainsi que le résultat final d'une liste de recommandation *Top\_n* pour évaluer l'utilité du système de recommandation proposé dans cette section.

Restaurants	Note_P	Similarité cosinus	Distance	Score finale
R1	0.521	0.8137	0.352	0.562
R2	0.456	0.4321	0.382	0.283
R3	0.294	0.0600	0.160	0.351
R4	0.423	0.0921	0.172	0.232

TABLE 3.3 – Exemple de recommandation des restaurants

R1, R2, R3 et R4 sont la liste de recommandation final des restaurants, et elles sont similaires aux préférences de l'utilisateur cible et ont les meilleurs résultats finaux.

### 7.3.4 Pseudo-code de l'algorithme de recommandation

Cette section résume notre système de recommandation dans un pseudo-code qui étudie deux états : lorsqu'un utilisateur visite initialement le système et lorsqu'il est déjà existé dans le système. Ce code fournit toutes les méthodes nécessaires au calcul de la suggestion dont nous avons parlé précédemment et explique en détail les étapes de cette phase, où il reçoit en entrée la liste des restaurants *Lr* et la liste des utilisateurs *Lu* qui sont déjà dans le système afin de calculer le score final de chacun des restaurants et appliquent le filtrage collaboratif

avec l'algorithme SlopeOne sur ces scores finales, puis effectue un tri ascendant des restaurants pour obtenir la liste finale des recommandations  $Top\_n$  pour l'utilisateur.

---

**Algorithme 1** Algorithme général du système de recommandation

---

**Require:**  $Lu$  (liste des utilisateurs),  $Lr$  (liste des restaurants),  $Txt$  (commentaire)

**Ensure:**  $Top\_n$  (liste des restaurants recommandés)

$$\alpha = \beta = \gamma = 1/3$$

**foreach**  $Ri \in Lr$  **do**

**if**  $U \notin Lu$  **then**

▷ L'utilisateur entrer pour la première fois

$Note\_P(U, Ri) \leftarrow 1;$

$Sim(U, Ri);$

$Dis(U, Ri);$

$Score\_f(U, Ri) = \alpha * (1) + \beta * Sim(U, Ri) + \gamma * Dis(U, Ri);$

$Top\_n \leftarrow SlopeOne(Score\_f(U, Ri));$

▷ Filtrage collaboratif

**Return**  $Top\_n$

**if**  $U$  saisir un commentaire pour  $Ri$  **then**

$Note\_P(U, Ri) \leftarrow Pred(U, Ri, Txt);$

▷ Analyse de sentiment

$Score\_f(U, Ri) = \alpha * Note\_P(U, Ri) + \beta * Sim(U, Ri) + \gamma * Dis(U, Ri);$

$Top\_n = SlopeOne(Score\_f(U, Ri));$

▷ filtrage collaboratif

**Return**  $Top\_n$

▷ Nouvelle liste de recommandation

**end**

**end**

**else**

▷ l'utilisateur exister dans le système

$Note\_P(U, Ri);$

$Sim(U, Ri);$

$Dis(U, Ri);$

$Score\_f(U, Ri) = \alpha * Note\_P(U, Ri) + \beta * Sim(U, Ri) + \gamma * Dis(U, Ri);$

$Top\_n = SlopeOne(Score\_f(U, Ri));$

▷ Filtrage collaboratif

**Return**  $Top\_n$

**if**  $U$  saisir un commentaire pour  $Ri$  **then**

$Note\_P(U, Ri) \leftarrow Pred(U, Ri, Txt);$

▷ Analyse de sentiment

$Score\_f(U, Ri) = \alpha * Note\_P(U, Ri) + \beta * Sim(U, Ri) + \gamma * Dis(U, Ri);$

$Top\_n = SlopeOne(Score\_f(U, Ri));$

▷ Filtrage collaboratif

**Return**  $Top\_n$

▷ Nouvelle liste de recommandation

**end**

**end**

**end**

---

## 8 Conclusion

Dans ce chapitre, nous avons détaillé les étapes de développement d'un système de recommandation des restaurants dans le contexte du tourisme intelligent, qui repose sur de nombreuses étapes, notamment la modélisation des profils de l'utilisateur et de restaurant en fonction des informations contextuelles et de localisation, ainsi que la phase d'analyse des sentiments qui permet d'évaluer l'opinion de l'utilisateur et convertir à une note réelle.

Nous avons pu décrire la phase principale de la proposition et de la recommandation basée sur les deux techniques de filtrage et les performances globales du système, en utilisant les phases de modélisation des utilisateurs et des restaurants et la phase d'analyse des sentiments, et en ce décrivant leurs relations et leurs fonctionnements.

Après ce chapitre de conception et modélisation, nous passons au processus de mise en œuvre et d'exécution du système proposé, ainsi qu'aux résultats acquis.

# Chapitre 4

## Mise en oeuvre et résultat de l'approche proposée

### 1 Introduction

Après avoir terminé la phase de conception et formalisation de notre approche, nous passons à la phase de mise en œuvre, qui constitue la dernière partie de ce mémoire et qui tente de mettre en œuvre et de valider notre approche.

Dans ce chapitre nous allons décrire les outils de développement et l'environnement de travail qui ont été utilisés pour construire le système. Nous procédons au test du système de recommandation après avoir expérimenté l'analyse des sentiments afin de mettre en œuvre et d'évaluer les principes proposés au chapitre 3. Ensuite, à l'aide de quelques captures d'écran des résultats collectés, nous avons révélé les résultats de chaque étape pour ce système.

### 2 Présentation des outils de développement

Nous utilisons une collection d'outils pour nous aider à concevoir notre approche de manière appropriée pendant les étapes de développement et de mise en œuvre de notre système. Dans les sections ci-dessous, nous allons examiner ces outils en détail.

#### 2.1 Matériels

L'implémentation a été réalisée sur un ordinateur à processeur Core I5 avec 4 Go de RAM fonctionnant sous Windows 10, mais le système peut être exploité sur n'importe quelle machine utilisant des environnements de programmation interactive en ligne tels que jupyter et Google collaboratory.

#### 2.2 Environnement de travail

- **Jupyter** : Jupyter est une application open source basée sur un navigateur qui combine des langages interprétés, des bibliothèques et des outils de visualisation [110]. Un carnet de notes jupyter peut être utilisé localement ou à distance. Chaque document est composé de

plusieurs cellules, chacune d'entre elles contient du code en langage de script ou markdown, dont le résultat est intégré au texte. Les résultats les plus courants sont des textes, des tableaux, des graphiques et des images. Les tests et les résultats étant livrés dans un format autonome, cette technique facilite le partage et la reproduction des enquêtes scientifiques [111].

- **Google collaboratory** : Google Colab où Colaboratory est un service gratuit de Google basé sur Jupyter notebook et conçu pour la formation et la recherche en apprentissage automatique. Cette plateforme permet d'entraîner des modèles d'apprentissage automatique sur le cloud. Ainsi, à part un navigateur, nous n'avons pas besoin d'installer quoi que ce soit sur notre machine [112]. Les blocs-notes de Colaboratory et fonctionnent de manière similaire aux objets Google Docs en ce sens qu'ils peuvent être partagés et que les utilisateurs peuvent travailler ensemble sur le même bloc-notes. TensorFlow, Matplotlib et Keras figurent parmi les principales bibliothèques d'apprentissage automatique et d'intelligence artificielle incluses dans les moteurs d'exécution Python de Colaboratory [113].

## 2.3 Langage de programmation

- **Python** : Il a été conçu en 1990 par Guido van Rossum. Python est un langage de programmation interprété, interactif et orienté objet. Il fournit des structures de données de haut niveau telles que les listes et les tableaux associatifs (appelés dictionnaires), le typage dynamique et la liaison dynamique, les modules, les classes, les exceptions, la gestion automatique de la mémoire, etc. Il possède une syntaxe remarquablement simple et élégante, tout en étant un langage de programmation puissant et polyvalent [114].

La qualité des logiciels, la productivité des développeurs, la portabilité des programmes, les bibliothèques de support, l'intégration des composants et l'appréciation sont autant de facteurs qui influent sur l'utilisation de python.

## 2.4 Bibliothèques

- **Numpy** : Numerical Python est une bibliothèque python pour les structures de données, l'algèbre linéaire et la manipulation des matrices. Sa syntaxe est similaire à celle de matlab, tout comme la façon dont elle gère les structures de données et les matrices. Elle contient les structures de données, les méthodes et les bibliothèques nécessaires à la plupart des applications scientifiques basées sur python et nécessitant des données numériques [115, 116, 117].
- **Pandas** : Pandas est une bibliothèque python qui fournit des structures de données rapides, polyvalentes et expressives pour traiter des données "relationnelles" ou "étiquetées". Son objectif est de servir de base pour entreprendre des analyses de données réalistes et concrètes en python. En outre, il aspire à devenir l'outil d'analyse et de manipulation de données open source le plus puissant et le plus polyvalent accessible dans n'importe quel langage [118].

- **Matplotlib** : Matplotlib est une bibliothèque graphique python pour la visualisation de données, qui est étroitement intégrée à numpy et pandas, est devenue un composant important de la pile python pour la science des données. Le paquet pyplot ressemble beaucoup aux instructions graphiques de matlab. Par conséquent, les utilisateurs de matlab peuvent rapidement passer au traçage en python [119].
- **Scikit-learn** : Scikit-learn est une bibliothèque python pour le développement d’algorithmes d’apprentissage automatique. Il contient également des étapes de prétraitement des données, des stratégies de rééchantillonnage des données, des paramètres d’évaluation et des interfaces de recherche pour régler/optimiser les performances d’un algorithme, qui sont tous des éléments importants du pipeline d’apprentissage automatique [119].
- **Surprise** : Surprise est un module python qui vous permet de créer et de tester des systèmes de prédiction de taux. surprise comprend un ensemble d’estimateurs (ou algorithmes de prédiction) pour évaluer les prédictions. Les algorithmes classiques, tels que les algorithmes primaires basés sur la similarité, ainsi que les méthodes basées sur la factorisation matricielle, telles que SVD ou NMF, sont implémentés. Il comprend également des outils d’évaluation et de sélection des modèles [120].
- **NLTK** : Natural Language Tool Kit est un langage de programmation python populaire pour travailler avec les données du langage humain. Il comprend une suite des bibliothèques de traitement de texte pour la classification, tokenization, stemming, tagging, l’analyse syntaxique et le raisonnement sémantique, des wrappers pour des bibliothèques NLP industrielles, un forum de discussion actif, ainsi que des interfaces faciles à utiliser pour plus de 50 corpus et ressources lexicales comme WordNet [121].
- **Textblob** : Textblob est une bibliothèque pour le traitement du texte écrit en python. Il offre une API standardisée pour les activités impliquant le traitement du langage naturel (NLP) [122]. Textblob fonctionne de manière similaire à une chaîne de caractères python et contient plusieurs aspects tels que tokenization, l’extraction des phrases nominales, le marquage POS, l’analyse des sentiments, la traduction et l’identification de la langue, les n-grammes, la vérification orthographique et l’intégration de WordNet [122].

### 3 Expérimentation du module d’analyse des sentiments

Dans cette partie nous allons présenter la mise en œuvre et le développement du module d’analyse des sentiments. Nous commençons par détailler la collecte des données, puis l’expérimentation et les résultats de chaque étape, et enfin, une évaluation des résultats.

#### 3.1 Collecte des données

Dans ce travail nous avons besoin d’un ensemble de données contiennent les commentaires sous format de texte et les notes d’évaluation pour chaque texte afin de réaliser les travaux et les tests présentés dans cette étude.

Nous utilisons à cette fin un échantillon de 50 000 lignes et deux colonnes pour les commentaires (texte) et les notes d'évaluation (stars), cet échantillon est choisi au hasard de la collection originale de Yelp [123], qui comprend 6.9 millions des commentaires publics ainsi que 908 915 conseils par 1 987 897 utilisateurs Plus de 1,2 million d'attributs d'entreprises tels que les heures d'ouverture et le stationnement pour chacune des 131 930 entreprises, et qui est accessible sous forme de fichiers JSON.

## 3.2 Prétraitement des données

Les étapes essentielles de notre prétraitement sur l'échantillon de données précédentes seront présentées dans cette section; chaque étape nous permet d'effectuer un nettoyage sur notre ensemble de données qui nous permettant de construire un ensemble final propre et prêt pour la classification.

- **Suppression des stops word** : les mots vides et les symboles des émoticônes pour chaque commentaire sont supprimés en utilisant cette méthode et ne laissant que les termes cruciaux et clairs, l'exemple suivant montre le processus de suppression en action sur un commentaire particulier et voit son effet sur celui-ci.

Le capture suivant montrer le commentaire avant l'application de cette méthode :

```
[9] # le premier texte dans le dataframe
df_review['text'].values[0]
```

```
'If you decide to eat here, just be aware it is going to take about 2 hours from beginning to end. We have tried it multiple times, because I want to like it! I have been to it's other locations in NJ and never had a bad experience. \n\nThe food is good, but it takes a very long time to come out. The waitstaff is very young, but usually pleasant. We have just had too many experiences where we spent way too long waiting. We usually opt for another diner or restaurant on the weekends, in order to be done quicker.'
```

FIGURE 4.1 – Le texte avant la suppression des stop word

Après avoir appliqué cette méthode, nous verrons le changement qui s'est produit dans le commentaire :

```
[19] #le premier texte après l'élimination du stop word et émoticons
df_review['text'].values[0]
```

```
'decide eat here, aware going take 2 hours beginning end. tried multiple times, want like it! locations nj never bad experience. food good, takes long time come out. waitstaff young, usually pleasant. many experiences spent way long waiting. usually opt another diner restaurant weekends, order done quicker.'
```

FIGURE 4.2 – Le texte après la suppression des stop word

- **Stemming et lemmatization** : on divise chaque texte en mots dans la méthode de tokenization, puis on applique la méthode de stemming et lemmatization à chacun des mots qui composent le texte, ce qui permet l'unification d'un groupe de mots tournant

dans un même contexte, un exemple des deux méthodes est donné ci-dessous sur un commentaire spécifique.

L'application de la méthode de stemming sur le premier commentaire permet de donner le résultat suivant :

```
[106] #le stemming du premier texte
df_review['text_stemming'].values[0]

['decid',
 'eat',
 'here,',
 'awar',
 'go',
 'take',
 '2',
 'hour',
 'begin',
 'end.',
 'tri',
 'multipl',
 'times,',
 'want',
 'like',
 'it!',
 'locat',
 'nj',
 'never',
```

FIGURE 4.3 – Le texte après l'application de stemming

Voici le résultat de l'utilisation de l'étape de lemmatization sur le même commentaire :

```
[109] #lemmatization du premier texte
df_review['text_lemmatize'].values[0]

['decide',
 'eat',
 'here,',
 'aware',
 'going',
 'take',
 '2',
 'hour',
 'beginning',
 'end.',
 'tried',
 'multiple',
 'times,',
 'want',
 'like',
 'it!',
 'location',
 'nj',
 'never',
 .....
```

FIGURE 4.4 – Le texte après l'application de lemmatization

- **Collecter les mots de chaque commentaire** : les mots de chaque commentaire coupés par les méthodes de stemming et lemmatization sont maintenant regroupés en une

seule phrase pour reconstruire nos phrases en incluant uniquement les mots importants sur lesquels travailler dans la prochaine phase d'extraction des caractéristiques. Les captures d'écran suivantes montrent le groupe de mots précédent dans une phrase après le processus de regroupement.

Le résultat obtenu en regroupant les mots du premier commentaire en fonction du processus de stemming :

```
[33] #regrouper les mots de la première phrase(stemming)
df_review['text_stemming'].values[0]

'decid eat here, awar go take 2 hour begin end. tri multipl times, want like it! locat nj never bad experience.
iting. usual opt anoth diner restaur weekends, order done quicker. '
```

FIGURE 4.5 – Le groupement des mots à partir de stemming

Le résultat obtenu en regroupant les mots du même commentaire en fonction du processus de lemmatization :

```
[32] #regrouper les mots de la première phrase(lemmatize)
df_review['text_lemmatize'].values[0]

'decide eat here, aware going take 2 hour beginning end. tried multiple times, want like it! location nj never b
nce spent way long waiting. usually opt another diner restaurant weekends, order done quicker. '
```

FIGURE 4.6 – Le groupement des mots à partir de lemmatization

- **Résultat de prétraitement** : l'utilisation de processus de prétraitement des données permet de réduire et de nettoyer l'ensemble de données des textes afin de faciliter l'étape d'extraction des caractéristiques à laquelle il est appliqué, comme le montre la figure ci-jointe, qui illustre la transformation des données basées sur le nombre de mots avant et après ces étapes.





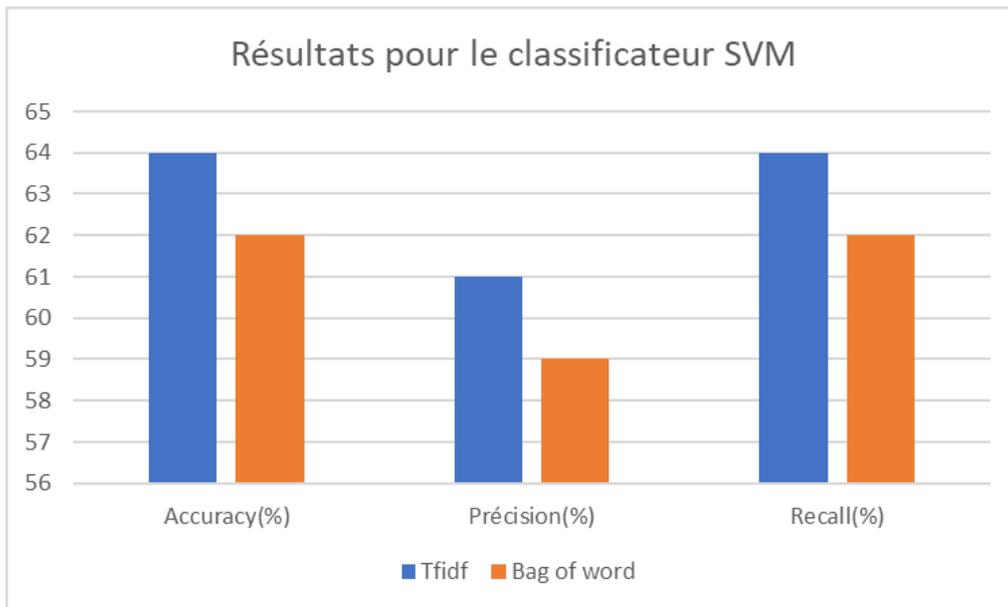


FIGURE 4.10 – Résultats graphiques pour le classificateur SVM

Notation	1	2	3	4	5
1	166	82	8	14	14
2	34	164	36	92	33
3	7	61	67	248	54
4	2	17	20	665	504
5	0	2	5	289	1515

TABLE 4.2 – SVM et Tf-idf

Notation	1	2	3	4	5
1	167	60	6	12	39
2	48	136	43	82	50
3	16	54	68	229	70
4	3	22	22	662	539
5	1	3	5	261	1541

TABLE 4.3 – SVM et BoW

- **Régression logistique** : il s'agit d'un modèle de régression qui est utilisé à des fins de classification, RL tente de trouver un hyperplan qui maximise l'écart de séparation entre les classes. Les valeurs des paramètres de classificateur RL sont réglées comme suit :  $C = 1.0$ ,  $iter\_max = 100$ . Le classificateur de régression a été utilisé pour classer le jeu de données obtenu après le prétraitement. Au total, deux tests ont été effectués en utilisant deux algorithmes de sélection des caractéristiques distinctes (TF-IDF et BoW).

Le tableau et la figure ci-jointe montrent les résultats produits par ce classificateur en utilisant les deux stratégies de sélection des caractéristiques :

Algorithmes	Mesures		
	Accuracy(%)	Precision(%)	Recall(%)
Tfidf	63	61	63
Bag of word	59	58	59

TABLE 4.4 – Résultats pour le classificateur RL

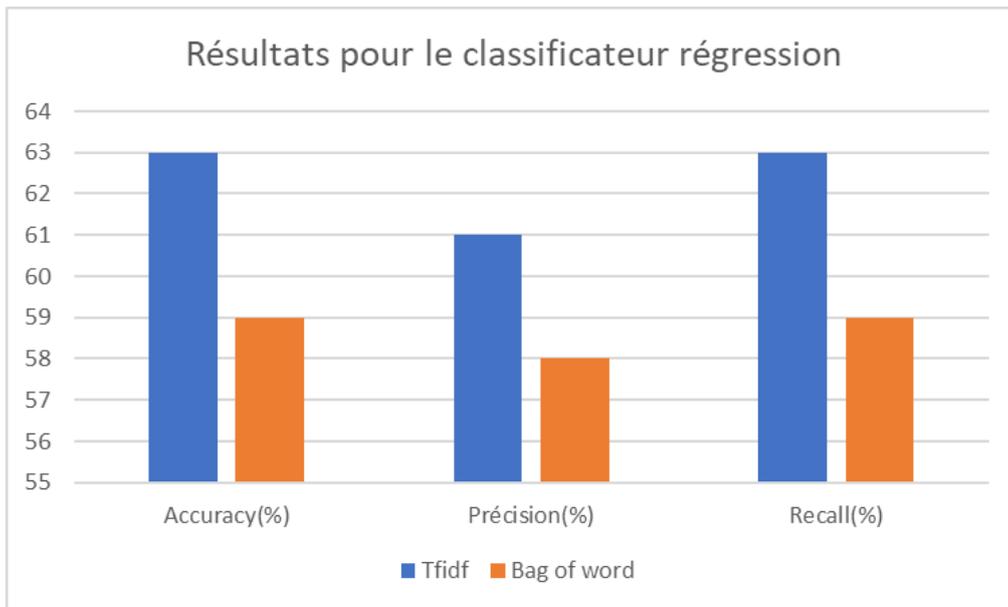


FIGURE 4.11 – Résultats graphiques pour le classificateur RL

Notation	1	2	3	4	5
1	797	133	40	38	68
2	218	226	195	100	62
3	83	110	364	406	172
4	22	32	149	1146	1195
5	28	5	38	589	3784

TABLE 4.5 – RL et Tfidf

Notation	1	2	3	4	5
1	797	133	40	38	68
2	218	226	195	100	62
3	83	110	364	406	172
4	22	32	149	1146	1195
5	28	5	38	589	3784

TABLE 4.6 – RL et BoW

- **Discussion :** nous pouvons constater d’après les résultats présentés dans les figures et les tableaux précédents, qui représentent les résultats de quatre cas que le taux d’accuracy des résultats d’apprentissage est similaire dans toutes les situations et se situe entre 50% et 64%.

Les meilleurs résultats ont été obtenus pour le modèle généré en utilisant le classificateur SVM et la méthode d’extraction de caractéristiques Tfidf avec un taux d’accuracy = 64%, une précision = 61%, un rappel = 64%, et le résultat de matrice de confusion de 5 classes de notation pour ce modèle est illustré dans le tableau 4.2 , qui montrent de nombreuses erreurs sur la prédiction ponctuelle de la notation comme dans l’observation actuelle pour les valeurs 3 et 5, il les prédit principalement avec la valeur 4 avec des nombres d’erreurs 248 et 289 respectivement, et aussi pour le reste des classes de notation.

En outre, nous concluons que la qualité des résultats du modèle d’apprentissage est liée à de nombreux éléments de contrôle, tels que les paramètres et la méthode de classification, ainsi que le degré de prétraitement du jeu de données, et nous avons construit un modèle d’apprentissage plus robuste sur la base de ces aspects.

Pour démontrer la validité de notre système de prédiction nous appliquons ce système sur un nouveau jeu de données des hôtels provenant de TripAdvisor [124], qui

comprend 20 491 lignes et deux colonnes d'avis et des catégories d'évaluation allant de 1 à 5. Ce modèle nous permet d'effectuer les phases préalables du processus d'analyse des sentiments, telles que le prétraitement et la sélection des caractéristiques, ainsi que la classification, comme le montrent les figures et les tableaux suivants, qui illustrent les résultats de nombreux essais sur cette nouvelle base de données :

Algorithmme	Mesures		
	Accuracy(%)	Precision(%)	Recall(%)
SVM et Tfidf	63	62	63
SVM et BoW	62	60	62
RL et Tfidf	62	61	62
RL et BoW	59	58	59

TABLE 4.7 – Les résultats des quatre expérimentations

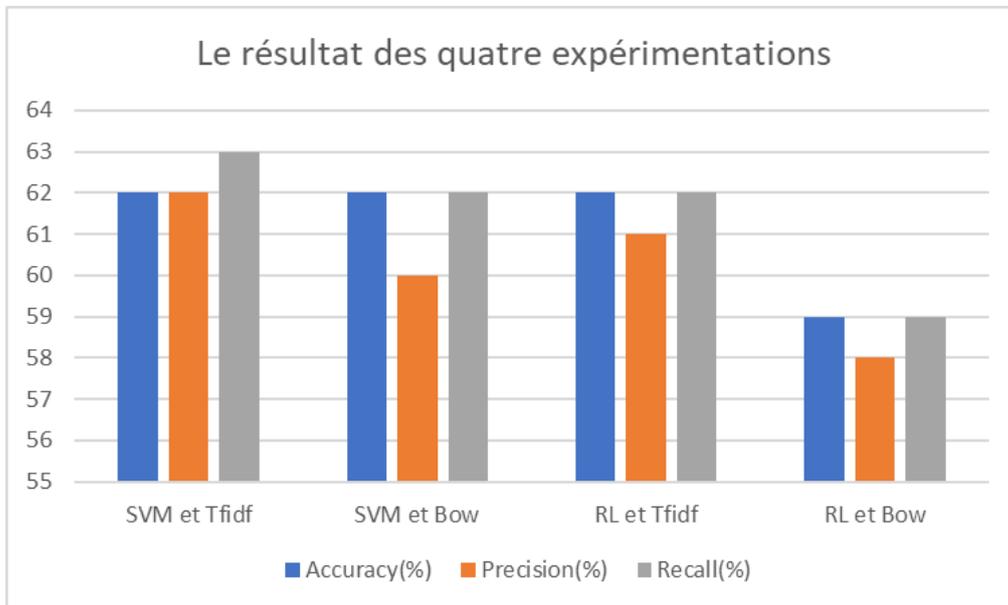


FIGURE 4.12 – Résultats graphiques pour les quatre expérimentations

Notation	1	2	3	4	5
1	166	82	8	14	14
2	34	164	36	92	33
3	7	61	67	248	54
4	2	17	20	665	504
5	0	2	5	289	1515

TABLE 4.8 – SVM et Tfidf

Notation	1	2	3	4	5
1	167	60	6	12	39
2	48	136	43	82	50
3	16	54	68	229	70
4	3	22	22	622	539
5	1	3	5	261	1541

TABLE 4.9 – SVM et BoW

Notation	1	2	3	4	5
1	176	72	10	10	16
2	44	145	61	73	36
3	11	46	102	215	63
4	6	14	38	626	524
5	0	1	6	297	1507

TABLE 4.10 – RL et Tfidf

Notation	1	2	3	4	5
1	178	73	16	9	8
2	71	138	85	48	17
3	13	63	156	154	51
4	3	41	120	560	484
5	4	3	27	411	11366

TABLE 4.11 – RL et BoW

### 3.5 Scénario de test

Nous avons appliqué des cas de tests au système de prédiction que nous avons conçu. Plusieurs commentaires suggérés et différents seront entrés dans cette section pour afficher les résultats obtenus à partir de ce système et voir leur validité. Voici les trois exemples et les résultats obtenu à partir de ce système de prédiction.

#### Exemple 1 :

Commentaire 1 = "Food is superb as always.??? One of the best places for value for money Thai food. Staff is very warm n helpful. We are repeat customers there."

```
[164] #test du text positif
test = input("Please enter a Review:\n")
text_pred(test)

Please enter a Review:
Food is superb as always. 😊 One of the best places for value for money Thai food. Staff is very warm n help-
5
```

FIGURE 4.13 – Résultats de prédiction du commentaire 1

#### Exemple 2 :

Commentaire 2 = "slow service, food nice and menu is good"

```
[166] #test du text neutre
test = input("Please enter a Review:\n")
text_pred(test)

Please enter a Review:
slow service, food nice zand menu is good
3
```

FIGURE 4.14 – Résultats de prédiction du commentaire 2

#### Exemple 3 :

Commentaire 3 = "Possibly the worst Thai food I have ever had. Jungle curry consisted of inedible beef prices covered with a tasteless watery liquid (?), lemon grass stir fry (chicken) possibly chicken, more like old bush Turkey. Too tired to take it back but if it was served up in the restaurant would have walked out."

```
[163] #test du text négative
test = input("Please enter a Review:\n")
text_pred(text)

Please enter a Review:
Possibly the worst Thai food I have ever had. Jungle curry consisted of inedible beef prices covered with a t
1
```

FIGURE 4.15 – Résultats de prédiction du commentaire 3

## 4 Expérimentation du système de recommandation

Dans cette partie, nous allons passer en revue les étapes de la mise en œuvre et du développement du module de recommandation, ainsi que les résultats de chaque étape. Nous commençons par voir comment recueillir les données et comment les traiter pour qu’elles puissent être utilisées dans les étapes suivantes, puis nous passons aux résultats de chacune des procédures qui mènent à la conclusion finale suggérée, et enfin nous verrons comment analyser ces résultats.

### 4.1 Collecte des données

Notre ensemble de données comprend les données nécessaires sur les restaurants et les utilisateurs pour notre système de recommandation qui est les recueillir d’après le site de UCI Machine Learning Repository [125]. Cet ensemble de données comprend 9 fichiers incluant les caractéristiques et les attributs des restaurants et des utilisateurs.

Pour les restaurants, on contient plusieurs caractéristiques où chaque caractéristique a une capacité spécifique, cet ensemble des restaurants est représenté dans les quatre fichiers suivants :

- **Fichier 1** : pour la caractéristique de type de paiement avec une capacité de 1314 lignes et 2 colonnes.
- **Fichier 2** : pour la caractéristique de type de cuisine avec une capacité de 916 lignes et 2 colonnes.
- **Fichier 3** : pour la caractéristique de parking avec une capacité de 702 lignes et 2 colonnes.
- **Fichier 4** : pour les autres caractéristiques comme latitude, longitude et la gamme de prix, etc, avec une capacité de 130 lignes et 21 colonnes.

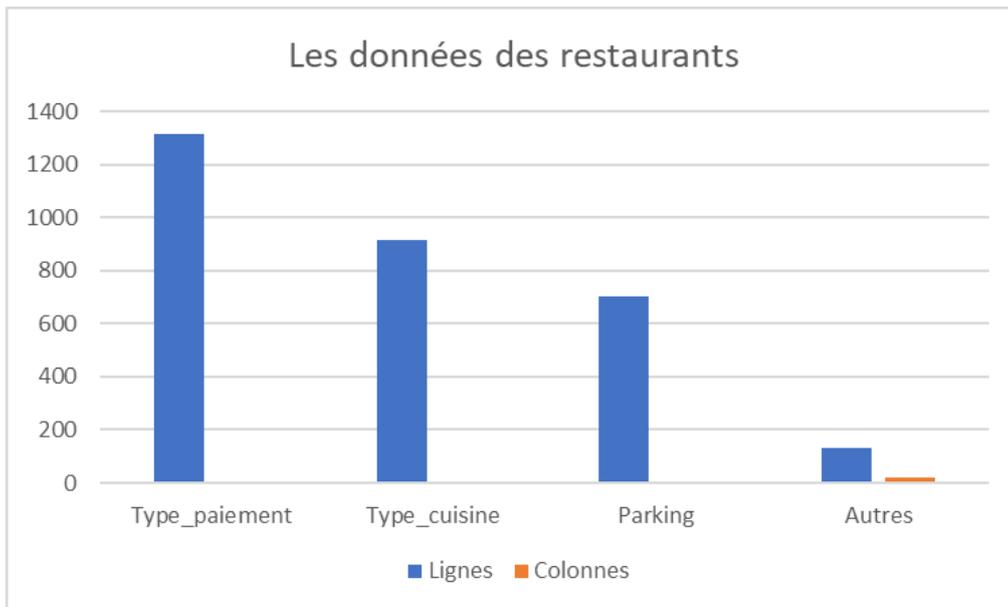


FIGURE 4.16 – La distribution des caractéristiques des restaurants

Pour les utilisateurs nous avons de nombreuses caractéristiques, qui se reflètent dans les trois fichiers suivants :

- **Fichier 1** : pour la caractéristique de type de paiement avec une capacité de 177 lignes et 2 colonnes.
- **Fichier 2** : pour la caractéristique de type de cuisine avec une capacité de 330 lignes et 2 colonnes.
- **Fichier 3** : pour les autres caractéristiques comme latitude, longitude et la gamme de budget, etc. avec une capacité de 138 lignes et 19 colonnes.

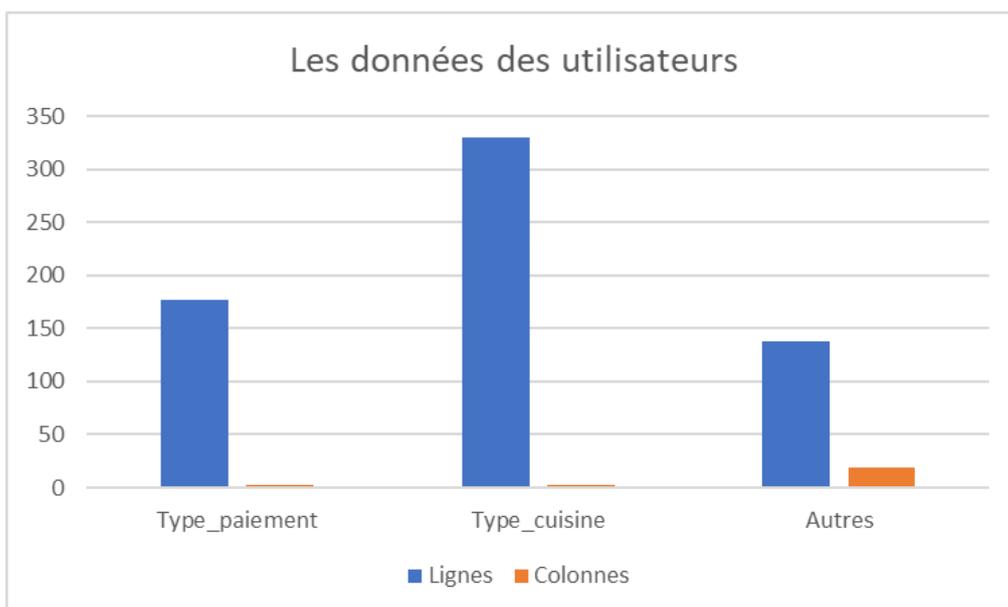


FIGURE 4.17 – La distribution des caractéristiques des utilisateurs

## 4.2 Prétraitement des données

Après la validation et la consultation des fichiers de notre ensemble de données, nous décrivons le processus de prétraitement de cet ensemble de données, ainsi que les résultats acquis à cette étape.

La liste des fonctions de prétraitement essentielles comprend (1) la gestion des données répliquées, (2) la gestion des valeurs manquantes, (3) le filtrage des caractéristiques et attributs.

- **Gestion des données répliquées** : la gestion des répliques comprend une première étape pour grouper les données selon le numéro d'identification et supprimer les répliques incohérentes et une seconde pour faire une combinaison des 4 ensemble de données du restaurant et 3 ensembles de données de l'utilisateur qui expliquent les deux à base d'identificateur. Dans les captures suivantes, nous verrons les résultats obtenus à partir de cette étape. Après ce traitement initial, le nouvel ensemble de données pour les restaurants, qui comprend 146 lignes et 16 colonnes, ressemble à ceci :

Entrée [16]: df\_restaurants

Out[16]:

	placelD	latitude	longitude	name	address	city	state	country	alcohol	smoking_area	price	area	other_services
0	134999	18.915421	-99.184871	Kiku Cuernavaca	Revolucion	Cuernavaca	Morelos	Mexico	No_Alcohol_Served	none	medium	closed	none
1	132825	22.147392	-100.983092	puesto de tacos	esquina santos degollado y leon guzman	s.l.p.	s.l.p.	mexico	No_Alcohol_Served	none	low	open	none
2	135106	22.149709	-100.976093	El Rincón de San Francisco	Universidad 169	San Luis Potosi	San Luis Potosi	Mexico	Wine-Beer	only at bar	medium	open	none
3	132667	23.752697	-99.163359	little pizza Emilio Portes Gil	calle emilio portes gil	victoria	tamaulipas	?	No_Alcohol_Served	none	low	closed	none
4	132613	23.752903	-99.165076	carnitas_mata	lic. Emilio portes gil	victoria	Tamaulipas	Mexico	No_Alcohol_Served	permitted	medium	closed	none
...	...	...	...	...	...	...	...	...	...	...	...	...	...
141	132866	22.141220	-100.931311	Chaires	Ricardo B. Anaya	San Luis Potosi	San Luis Potosi	Mexico	No_Alcohol_Served	not permitted	medium	closed	none
142	135072	22.149192	-101.002936	Sushi Itto	Venustiano Carranza 1809 C Polanco	San Luis Potosi	SLP	Mexico	No_Alcohol_Served	none	medium	closed	none
143	135109	18.921785	-99.235350	Paniroles	?	?	?	?	Wine-Beer	not permitted	medium	closed	Internet
144	135019	18.875011	-99.159422	Restaurant Bar Coty y Pablo	Paseo de Las Fuentes 24 Pedregal de Las Fuentes	Jiutepec	Morelos	Mexico	No_Alcohol_Served	none	low	closed	none
145	132877	22.135364	-100.934948	sirloin stockade	?	?	?	?	No_Alcohol_Served	none	low	closed	none

146 rows x 16 columns

FIGURE 4.18 – Les données des restaurants après le premier traitement

Après le traitement initial, la capture d'écran suivante affiche le nouvel ensemble de données pour les utilisateurs, qui comprend 138 lignes et 20 colonnes :

Entrée [41]: df\_users

Out[41]:

	userID	latitude	longitude	smoker	drink_level	dress_preference	ambience	transport	marital_status	birth_year	interest	personality	religion
0	U1001	22.139997	-100.978803	false	abstemious	informal	family	on foot	single	1989	variety	thrifty-protector	none
1	U1002	22.150087	-100.983325	false	abstemious	informal	family	public	single	1990	technology	hunter-ostentatious	Catholic
2	U1003	22.119847	-100.946527	false	social drinker	formal	family	public	single	1989	none	hard-worker	Catholic
3	U1004	18.867000	-99.183000	false	abstemious	informal	family	public	single	1940	variety	hard-worker	none prof
4	U1005	22.183477	-100.959891	false	abstemious	no preference	family	public	single	1992	none	thrifty-protector	Catholic
...	...	...	...	...	...	...	...	...	...	...	...	...	...
133	U1134	22.149654	-100.998610	false	casual drinker	no preference	family	public	single	1991	variety	hard-worker	Catholic
134	U1135	22.170396	-100.949936	false	casual drinker	informal	family	on foot	single	1988	variety	hunter-ostentatious	Catholic
135	U1136	22.149607	-100.997235	true	social drinker	no preference	friends	car owner	single	1990	retro	thrifty-protector	Catholic
136	U1137	22.144803	-100.944623	false	social drinker	formal	family	public	single	1989	eco-friendly	hard-worker	Catholic
137	U1138	22.152884	-100.939663	false	social drinker	formal	friends	public	single	1991	variety	thrifty-protector	none

138 rows x 20 columns

FIGURE 4.19 – Les données des utilisateurs après le premier traitement

- **Gestion des valeurs manquants** : cette étape concerne les valeurs manquantes ; les cases comportant trop de valeurs inconnues peuvent être supprimées car elles posent un problème pour les techniques de calcul qui utilisent une matrice d'ensemble de données. Après avoir supprimé les valeurs manquantes, le jeu de données restaurants comprend désormais 103 lignes et 16 colonnes, la capture suivante présente le nouvel ensemble de données :

```
Entrée [38]: #df après la suppression des Nan
df_restaurants
```

```
Out[38]:
```

	placeID	latitude	longitude	name	address	city	state	country	alcohol	smoking_area	budget	area	other_service
1	132825	22.147392	-100.983092	puesto de tacos	esquina santos degollado y leon guzman	s.l.p.	s.l.p.	mexico	No_Alcohol_Served	none	0	open	nor
2	135106	22.149709	-100.976093	El Rincón de San Francisco	Universidad 169	San Luis Potosi	San Luis Potosi	Mexico	Wine-Beer	only at bar	1	open	nor
3	132667	23.752697	-99.163359	little pizza Emilio Portes Gil	calle emilio portes gil	victoria	tamaulipas	?	No_Alcohol_Served	none	0	closed	nor
4	132613	23.752903	-99.165076	camitas_mata	lic. Emilio portes gil	victoria	Tamaulipas	Mexico	No_Alcohol_Served	permitted	1	closed	nor
6	132732	23.754357	-99.171288	Taqueria EL amigo	Calle Mezquite Fracc Framboyanes	Cd Victoria	Tamaulipas	Mexico	No_Alcohol_Served	none	0	open	nor
...	...	...	...	...	...	...	...	...	...	...	...	...	...
139	132583	18.922290	-99.234332	McDonalds Centro	Rayon sn col. Centro	Cuernavaca	Morelos	Mexico	No_Alcohol_Served	not permitted	0	closed	nor
140	132866	22.141220	-100.931311	Chaires	Ricardo B. Anaya	San Luis Potosi	San Luis Potosi	Mexico	No_Alcohol_Served	not permitted	1	closed	nor
141	132866	22.141220	-100.931311	Chaires	Ricardo B. Anaya	San Luis Potosi	San Luis Potosi	Mexico	No_Alcohol_Served	not permitted	1	closed	nor
142	135072	22.149192	-101.002936	Sushi Itto	Venustiano Carranza 1809 C Polanco	San Luis Potosi	SLP	Mexico	No_Alcohol_Served	none	1	closed	nor
143	135109	18.921785	-99.235350	Paniroles	?	?	?	?	Wine-Beer	not permitted	1	closed	Intern

103 rows x 16 columns

FIGURE 4.20 – Les données des restaurants après la suppressions des valeurs manquants

Pour les données des utilisateurs, après avoir supprimé les valeurs manquantes, elles comportent 133 lignes et 20 colonnes, comme illustré dans la capture ci-dessous :

```
Entrée [36]: #df après la suppression des Nan
df_users
```

```
Out[36]:
```

	userID	latitude	longitude	smoker	drink_level	dress_preference	ambience	transport	marital_status	birth_year	interest	personality	religion
0	U1001	22.139997	-100.978803	0	abstemious	informal	family	on foot	single	1989	variety	thrifty-protector	none
1	U1002	22.150087	-100.983325	0	abstemious	informal	family	public	single	1990	technology	hunter-ostentatious	Catholic
2	U1003	22.119847	-100.946527	0	social drinker	formal	family	public	single	1989	none	hard-worker	Catholic
3	U1004	18.867000	-99.183000	0	abstemious	informal	family	public	single	1940	variety	hard-worker	none prof
4	U1005	22.183477	-100.959891	0	abstemious	no preference	family	public	single	1992	none	thrifty-protector	Catholic
...	...	...	...	...	...	...	...	...	...	...	...	...	...
133	U1134	22.149654	-100.998610	0	casual drinker	no preference	family	public	single	1991	variety	hard-worker	Catholic
134	U1135	22.170396	-100.949936	0	casual drinker	informal	family	on foot	single	1988	variety	hunter-ostentatious	Catholic
135	U1136	22.149607	-100.997235	1	social drinker	no preference	friends	car owner	single	1990	retro	thrifty-protector	Catholic
136	U1137	22.144803	-100.944623	0	social drinker	formal	family	public	single	1989	eco-friendly	hard-worker	Catholic
137	U1138	22.152884	-100.939663	0	social drinker	formal	friends	public	single	1991	variety	thrifty-protector	none

133 rows x 20 columns

FIGURE 4.21 – Les données des utilisateurs après la suppressions des valeurs manquants

- **Filtrage des caractéristiques** : nous supprimons les colonnes superflues et donnons à chaque colonne un nom commun dans les bases de données des utilisateurs et des restaurants au cours de cette étape, qui consiste à gérer les caractéristiques et les attributs des ensembles de données. Donc l'objectif de cette étape consiste à produire des ensembles de données finales qui peuvent être traités ultérieurement.

L'ensemble de données final des restaurants, qui ne comprend que les données importantes pour notre travail, est illustré par la capture suivante :

Entrée [93]: `#df finale`  
`df_restaurants`

Out[93]:

	placeID	latitude	longitude	name	smoking_area	budget	parking_lot	payment	cuisine
1	132825	22.147392	-100.983092	puesto de tacos	none	0	[none]	[cash]	Mexican
2	135106	22.149709	-100.976093	El Rincón de San Francisco	only at bar	1	[none]	[cash, VISA, MasterCard-Eurocard]	Mexican
3	132667	23.752697	-99.163359	little pizza Emilio Portes Gil	none	0	[none]	[cash]	Armenian
4	132613	23.752903	-99.165076	carnitas_mata	permitted	1	[yes]	[cash]	Mexican
6	132732	23.754357	-99.171288	Taqueria EL amigo	none	0	[none]	[cash]	Mexican
...	...	...	...	...	...	...	...	...	...
139	132583	18.922290	-99.234332	McDonalds Centro	not permitted	0	[none]	[cash, VISA, MasterCard-Eurocard, bank_debit_c...	American
140	132866	22.141220	-100.931311	Chaires	not permitted	1	[yes]	[cash]	Bakery
141	132866	22.141220	-100.931311	Chaires	not permitted	1	[yes]	[cash]	Cafeteria
142	135072	22.149192	-101.002936	Sushi Itto	none	1	[none]	[cash, VISA, MasterCard-Eurocard, American_Exp...	Japanese
143	135109	18.921785	-99.235350	Paniroles	not permitted	1	[none]	[cash]	Italian

103 rows × 9 columns

FIGURE 4.22 – Les données des restaurants après le filtrage des caractéristiques

La capture d'écran suivante montre l'ensemble de données pour les utilisateurs, qui contient uniquement des données pertinentes pour notre travail :

Entrée [90]: `#df finale`  
`df_users`

Out[90]:

	userID	latitude	longitude	budget	payment	cuisine
0	U1001	22.139997	-100.978803	medium	[cash]	[American]
1	U1002	22.150087	-100.983325	low	[cash]	[Mexican]
2	U1003	22.119847	-100.946527	low	[cash]	[Mexican]
3	U1004	18.867000	-99.183000	medium	[cash, bank_debit_cards]	[Bakery, Breakfast-Brunch, Japanese, Contempor...
4	U1005	22.183477	-100.959891	medium	[cash]	[American]
...	...	...	...	...	...	...
133	U1134	22.149654	-100.998610	medium	[cash]	[Mexican]
134	U1135	22.170396	-100.949936	low	[cash]	[Organic-Healthy, Steaks, Middle_Eastern, Medi...
135	U1136	22.149607	-100.997235	low	[cash]	[Mexican]
136	U1137	22.144803	-100.944623	low	[cash]	[Mexican]
137	U1138	22.152884	-100.939663	medium	[cash]	[Pizzeria]

133 rows × 6 columns

FIGURE 4.23 – Les données des utilisateurs après le filtrage des caractéristiques

### 4.3 Codage des données

Le codage des données est une phase importante de notre travail, dans cette étape nous avons converti en binaire les données de chaque caractéristique dans les bases de données des utilisateurs et des restaurants, comme les types de paiement, les gammes de budget, smoking et les types de cuisine.

La conversion pour toutes les caractéristiques de la base de données des restaurants et des utilisateurs permet de produire un ensemble de données comprenant les données sous forme binaire, comme l'illustrent les captures suivantes :

Entrée [79]: #Df finale  
df\_rs\_preference

Out[79]:

	placeID	name	latitude	longitude	no_smoking	smoking	low_budget	medium_budget	high_budget	cash	...	Game	International	Italian	Ja
0	132825	puesto de tacos	22.147392	-100.983092	1	0	1	0	0	1	...	0	0	0	
1	135106	El Rincón de San Francisco	22.149709	-100.976093	0	1	0	1	0	1	...	0	0	0	
2	132667	little pizza Emilio Portes Gil	23.752697	-99.163359	1	0	1	0	0	1	...	0	0	0	
3	132613	carnitas_mata	23.752903	-99.165076	0	1	0	1	0	1	...	0	0	0	
4	132732	Taqueria EL amigo	23.754357	-99.171288	1	0	1	0	0	1	...	0	0	0	
...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...
98	132583	McDonalds Centro	18.922290	-99.234332	1	0	1	0	0	1	...	0	0	0	
99	132866	Chaires	22.141220	-100.931311	1	0	0	1	0	1	...	0	0	0	
100	132866	Chaires	22.141220	-100.931311	1	0	0	1	0	1	...	0	0	0	
101	135072	Sushi Itto	22.149192	-101.002936	1	0	0	1	0	1	...	0	0	0	
102	135109	Paniroles	18.921785	-99.235350	1	0	0	1	0	1	...	0	0	0	1

103 rows x 35 columns

FIGURE 4.24 – Les données des restaurants après le codage binaire

Entrée [78]: #Df finale  
df\_u\_preference

Out[78]:

	userID	name	latitude	longitude	no_smoking	smoking	low_budget	medium_budget	high_budget	cash	...	Game	International	Italian	Japanese
0	U1001		22.139997	-100.978803	1	0	0	1	0	1	...	0	0	0	0
1	U1002		22.150087	-100.983325	1	0	1	0	0	1	...	0	0	0	0
2	U1003		22.119847	-100.946527	1	0	1	0	0	1	...	0	0	0	0
3	U1004		18.867000	-99.183000	1	0	0	1	0	1	...	0	0	0	1
4	U1005		22.183477	-100.959891	1	0	0	1	0	1	...	0	0	0	0
...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...
128	U1134		22.149654	-100.998610	1	0	0	1	0	1	...	0	0	0	0
129	U1135		22.170396	-100.949936	1	0	1	0	0	1	...	1	1	1	1
130	U1136		22.149607	-100.997235	0	1	1	0	0	1	...	0	0	0	0
131	U1137		22.144803	-100.944623	1	0	1	0	0	1	...	0	0	0	0
132	U1138		22.152884	-100.939663	1	0	0	1	0	1	...	0	0	0	0

133 rows x 35 columns

FIGURE 4.25 – Les données des utilisateurs après le codage binaire



```
Entrée [84]: #dataframe de rating (Note_P)
rating_df

Out[84]:
```

	userID	placelID	rating	food_rating	service_rating
0	U1077	135085	2	2	2
1	U1077	135038	2	2	1
2	U1077	132825	2	2	2
3	U1077	135060	1	2	2
4	U1068	135104	1	1	2
...	...	...	...	...	...
1156	U1043	132630	1	1	1
1157	U1011	132715	1	1	0
1158	U1068	132733	1	1	0
1159	U1068	132594	1	1	1
1160	U1068	132660	0	0	0

1161 rows × 5 columns

FIGURE 4.28 – L'ensemble de données des notes - Rating

Nous pouvons voir dans cet ensemble de données qu'un utilisateur peut évaluer plusieurs restaurants et nous avons donc utilisé le prétraitement précédent sur cet ensemble et l'avons combiné avec les ensembles de données finales des utilisateurs et des restaurants présentés dans les deux figures 4.24 4.25 sur la base des identifiants, ce qui donne un ensemble de données total avec 936 couples (utilisateurs/restaurants) et toutes les données nécessaires pour calculer le score final, cet ensemble de données est présenté dans la figure suivante.

```
Entrée [85]: #DF des données totales
df

Out[85]:
```

	userID	placelID	rating	food_rating	service_rating	name_x	latitude_x	longitude_x	no_smoking_x	smoking_x	...	International_y	Italian_y	Japanese
0	U1077	135085	2	2	2	Tortas Locas Hipocampo	22.150802	-100.982680	1	0	...	0	0	
1	U1077	132825	2	2	2	puesto de tacos	22.147392	-100.983092	1	0	...	0	0	
2	U1077	135060	1	2	2	Restaurante Marisco Sam	22.156883	-100.978485	1	0	...	0	0	
3	U1077	135027	0	1	1	Restaurant Orizatlan	22.147145	-100.974494	1	0	...	0	0	
4	U1108	135085	1	2	1	Tortas Locas Hipocampo	22.150802	-100.982680	1	0	...	0	0	
...	...	...	...	...	...	...	...	...	...	...	...	...	...	
931	U1038	132862	2	2	2	La Posada del Virrey	22.150643	-100.987015	1	0	...	0	0	
932	U1047	132875	0	0	0	shi ro ie	22.149901	-100.993779	0	1	...	0	0	
933	U1138	132921	2	2	1	crudalia	22.150305	-100.989134	0	1	...	0	0	
934	U1138	132922	2	2	2	cafe punta del cielo	22.151135	-100.982311	0	1	...	0	0	
935	U1138	132925	1	2	1	el pueblito	22.153500	-100.976243	1	0	...	0	0	

936 rows × 74 columns

FIGURE 4.29 – L'ensemble de données total

```
Entrée [86]: #Df des scores finales
df_final

Out[86]:
```

	userID	placelD	score
0	U1077	135085	0.890237
1	U1077	132825	0.890219
2	U1077	135060	0.723569
3	U1077	135027	0.666586
4	U1108	135085	0.666617
...	...	...	...
931	U1038	132862	0.740928
932	U1047	132875	0.407625
933	U1138	132921	0.749733
934	U1138	132922	0.833103
935	U1138	132925	0.749803

936 rows × 3 columns

FIGURE 4.30 – L’ensemble de données pour le score final

## 4.5 Évaluation et discussion des résultats finaux

Dans cette partie, nous fournissons les résultats des tests de comparaison et d’analyse des erreurs de prédiction en utilisant les deux mesures RMSE (Root Mean Squared Error) et MAE (Mean Absolute Error ) pour l’évaluation des résultats finale.

Les résultats du système de recommandation sont basés sur un cadre d’évaluation comparative qui appelée benchmark utilisé le filtrage collaboratif et nous permet de relever les défis de la comparaison de 11 algorithmes de recommandation sur notre ensemble de données du score final en utilisant plusieurs mesures statistiques et de qualité et enfin pour conclure le meilleur algorithme qui déduit un bon résultat pour notre système. Le tableau et le graphique suivants illustrent le résultat de benchmark de notre système :

Algorithmes \ Mesures	Test_RMSE	Test_MAE	Fit_time	Test_time
SlopeOne	0.128145	0.099624	0.001999	0.003001
BaselineOnly	0.130385	0.110465	0.001659	0.001331
KNNWithZScore	0.130722	0.098512	0.008660	0.008672
SVDpp	0.132249	0.106452	0.130997	0.007003
KNNWithMeans	0.132471	0.101867	0.003664	0.006669
KNNBaseline	0.133002	0.109239	0.003993	0.011328
NMF	0.142340	0.111865	0.057662	0.002004
KNNBasic	0.155882	0.128166	0.001668	0.005997
SVD	0.158728	0.126956	0.059665	0.002667
NormalPredictor	0.209052	0.170258	0.000669	0.001997
CoClustering	0.714653	0.695006	0.035666	0.001334

TABLE 4.12 – Les résultats de benchmark pour 11 algorithmes

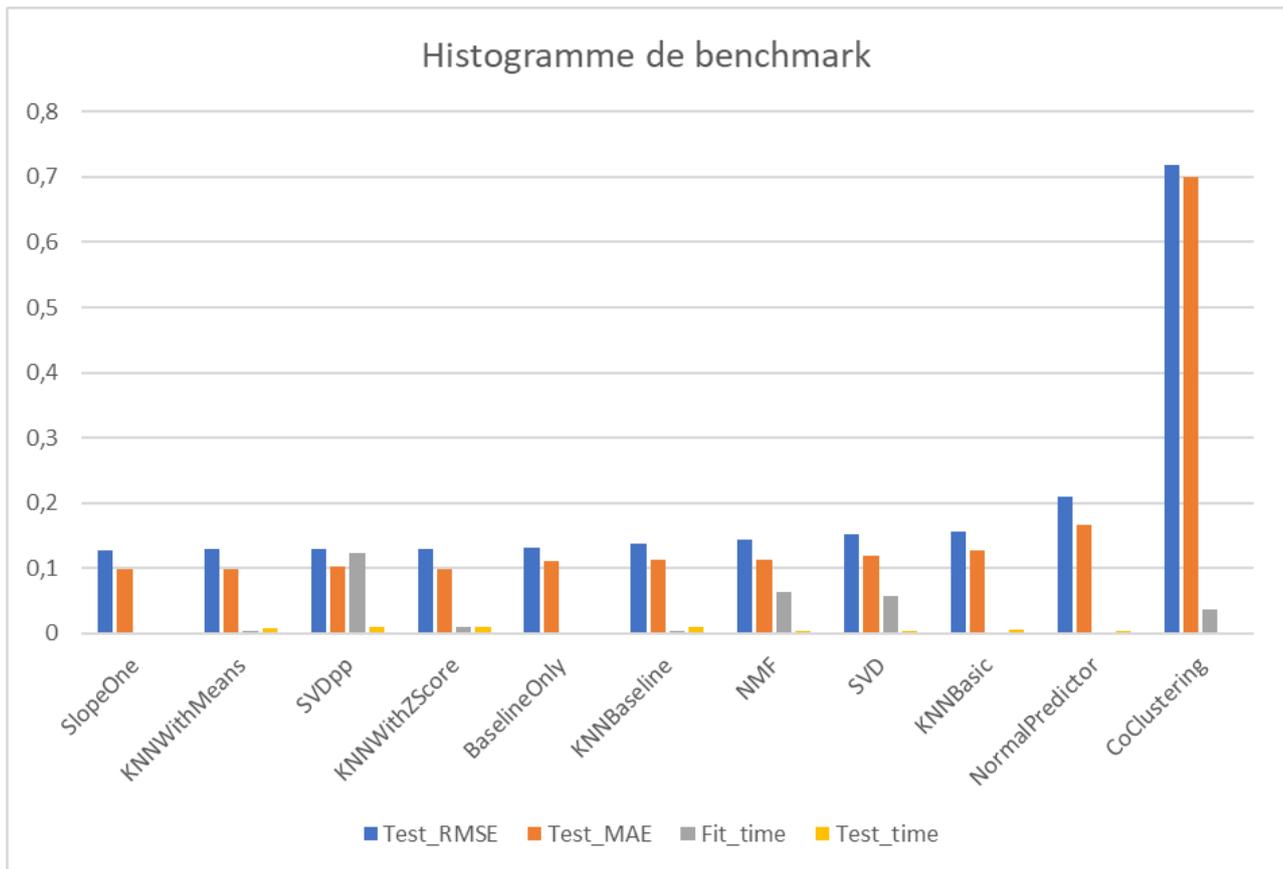


FIGURE 4.31 – Le graphe des résultats de benchmark pour 11 algorithmes

le framework de benchmark affiche les résultats de la performance moyenne de la recommandation. Les résultats sont organisés dans ce tableau par deux techniques d'évaluation, RMSE et MAE, ainsi que par le temps de formation et de test pour 11 algorithmes différents, par ordre décroissant, ce qui permet de voir facilement les différences absolues et relatives entre les algorithmes.

Les résultats produits par les 11 algorithmes pour les quatre paramètres sont similaires et relativement proches, comme le montre le graphique ci-dessus. Par rapport aux autres méthodes, l'algorithme de SlopeOne donne les meilleurs résultats, avec  $RMSE = 0,127419$ ,  $MAE = 0,097967$ , temps d'ajustement =  $0,002330$  et temps de test =  $0,002333$ .

À cette fin, nous allons créer un modèle d'apprentissage basé sur cette méthode qui va construire et fournir des valeurs RMSE et MAE pour le système de recommandation en utilisant le même ensemble de données. Nous avons utilisé l'algorithme SlopeOne basée sur le filtrage collaboratif comme algorithme de base pour notre modèle d'apprentissage, et notre jeu de données a été réparti entre 75 % de formation (749 utilisateurs) et 25 % de test (250 utilisateurs). les résultats des mesures d'évaluation pour l'algorithme de SlopeOne qui permet de définir la qualité de prédiction de nos expériences sont présentées dans le tableau ci-dessous.

Algorithme	Mesures	
	RMSE	MAE
SlopeOne	0.1119	0.0867

TABLE 4.13 – Les résultats d'évaluation du système de recommandation

Les résultats pour MAE et RMSE avec une valeur de 0,0867 et 0,1119 semblent bons car la plage de la valeur de MAE et RMSE est de 0 à 5 de sorte que si la valeur est proche de 0, les résultats sont bons. Sinon, si la valeur de l'échelle n'est pas proche de 0, les résultats obtenus ne sont pas bons.

Le volume et le type de jeu de données, les attributs choisis de l'utilisateur, l'algorithme d'apprentissage et d'autres aspects influencent tous la qualité des résultats d'évaluation des systèmes de recommandation. En plus de la composante de notation, notre système de recommandation est basé sur trois critères significatifs : la similarité des préférences, la distance entre l'utilisateur et le restaurant, et le facteur de notation, qui sont tous considérés comme des éléments améliorant la qualité des résultats d'évaluation.

Pour déduire l'importance des trois facteurs précédents dans notre système de recommandations, nous avons construit l'évaluation des trois cas où chaque cas dépend de la sommation de chacun des deux facteurs et de l'isolement de l'autre. Cette méthode nous permet de préciser l'effet de chacun des trois facteurs précédents sur le résultat de notre évaluation. La figure suivante illustre l'évaluation de ces cas.

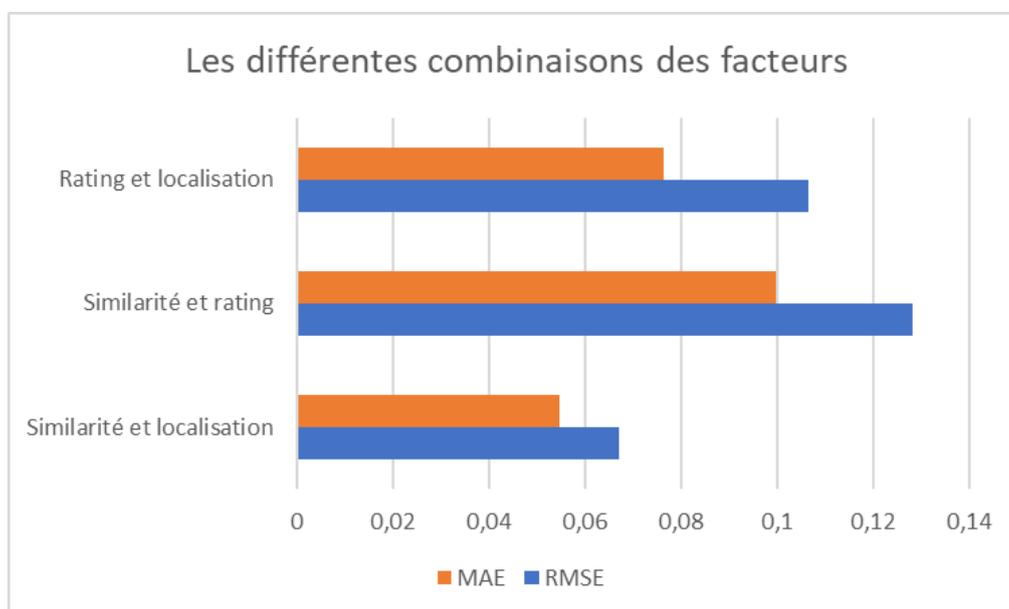


FIGURE 4.32 – Le graphe des résultats de différentes combinaisons

La qualité des évaluations pour les trois cas peut être observée dans la figure précédente, où les résultats révèlent que le troisième cas, qui est représenté par la combinaison des variables de similarité et de localisation, présente RMSE et MAE les plus faibles.

Nous concluons que la similarité des préférences et le calcul de la distance entre l'utilisateur

et le restaurant sont parmi les contributeurs les plus importants à l'augmentation de la qualité des notes du système de recommandation. Étant donné que les caractéristiques spécifiques des préférences de nos utilisateurs et de nos restaurants jouent un rôle important dans le résultat final tout en minimisant le taux d'erreur et qui contribuent de manière significative à la qualité du système, le but de ses caractéristiques est de construire une juste valeur de similarité entre l'utilisateur et le restaurant pour déterminer une bonne recommandation.

## 5 Conclusion

Dans ce chapitre, nous avons décrit les étapes du développement de notre système de recommandation des restaurants basé sur l'analyse des sentiments et la mobilité intelligente dans une smart city, les outils utilisés et les différents résultats obtenus lors des tests des deux modules d'analyse des sentiments et recommandation. Les résultats sont excellents et satisfaisants dans une certaine mesure, et nous pouvons affirmer qu'ils répondent aux critères fondamentaux d'un système de recommandation pour le tourisme dans une ville intelligente.

# Conclusion générale

Dans ce projet de fin d'études, nous nous sommes intéressés au domaine de la ville intelligente et avons proposé une approche pour recommander des restaurants orientés vers le tourisme intelligent.

Les systèmes de recommandation sont un bon moyen de donner aux gens des informations individuelles et précieuses afin qu'ils puissent prendre des meilleures décisions dans leur vie quotidienne. Par conséquent, nous avons introduit notre nouvel algorithme de recommandation dans ce mémoire pour améliorer ces décisions, qui fournissent des suggestions des restaurants contextuelles. Notre système de recommandation est un système hybride et contextuel s'adapte aux préférences de type de l'utilisateur et utilisent les caractéristiques des restaurants visités, et produisent également des résultats de suggestion dynamiques basés sur les méthodes précédentes.

Les résultats de l'étude de cas ont révélé que le système de recommandation des restaurants proposé peut exploiter avec succès les préférences des utilisateurs, les informations géographiques et d'autres critères pertinents tels que l'analyse des sentiments pour proposer des restaurants personnalisés et appropriés aux différents utilisateurs.

## Contribution

Les contributions de ce travail peuvent être résumées comme suit :

1. Traitement et modélisation des données pour les utilisateurs et les restaurants et construire les deux vecteurs des caractéristiques et des préférences qui sont inclus dans l'étape de calcul de similarité et distance qui définit dans quelle mesure un utilisateur et un restaurant sont similaires et proches.
2. Modéliser et mettre en œuvre la technique d'analyse des sentiments sur l'environnement Colab, y compris toutes les phases de traitement du texte, et mener des expériences et des évaluations sur deux ensembles de données distinctes.
3. Implémentation et expérimentation de l'approche globale proposée sous l'environnement Jupyter avec l'évaluation de ces résultats finale.

## Perspective

Une des limites du travail que nous avons présentées et dans une perspective de recherches futures, nous prévoyons ce qui suit :

1. Des contextes multiples, tels que la météo, le temps, les conditions de circulation et d'autres aspects pertinents, peuvent être ajoutées à l'algorithme de recommandation contextuelle existant pour produire un système de recommandation plus sophistiqué et plus robuste avec une meilleure précision de prédiction.
2. Utiliser la technologie IoT pour évaluer les demandes nutritionnelles des clients dans le contexte de la santé et augmenter la qualité des résultats des suggestions des restaurants en fusionnant différents types de capteurs et senseurs.
3. L'utilisation de données de médias sociaux dans le système, telles que Facebook et Twitter, pour obtenir des informations sur les visites et les interactions précédentes des utilisateurs sur le contexte des restaurants.
4. Ajouter une nouvelle caractéristique (proximité des amis) si les amis de l'utilisateur se trouvent à proximité, leur position est indiquée sur une carte, et des recommandations pour les mêmes restaurants sont fournies.
5. Une autre perspective intrigante dans le domaine du tourisme et de visite des restaurants serait la possibilité de fournir des suggestions pour un groupe d'utilisateurs plutôt que pour un utilisateur individuel. Par conséquent, nous aimerions élargir notre approche afin de pouvoir faire d'excellentes suggestions à un groupe d'utilisateurs tout en essayant de tenir compte du plus grand nombre possible de leurs goûts spécifiques.

## Acquis

- **Sur le théorique** : Maîtriser le domaine des systèmes de recommandation et comprendre leur importance dans le développement de smart city, notamment dans le domaine du tourisme, en maîtrisant les techniques de travail de toutes les différentes type de filtrage qui intervient dans leur fonctionnement, ainsi que tous les outils de base et facteurs nécessaires à leur fonctionnement.
- **Sur la pratique** : Nous nous sommes familiarisés avec les deux environnements de traitement du langage naturel, Jupyter et Colab, en plus d'approfondir notre connaissance du langage python. La mise en œuvre de notre approche est une concrétisation des plusieurs connaissances acquises durant nos années de formation en informatique, telles que la modélisation formelle, le traitement des données, l'apprentissage automatique et l'intelligence artificielle.
- **Sur la méthodologique** : Nous avons appris le processus de recherche scientifique, ses méthodes, ses phases, et nous les avons mis en pratique en travaillant sur notre projet, qui comprenait : la compréhension du sujet, la localisation du problème, la sélection d'une technique de recherche, la proposition d'une solution et son exécution.

# Bibliographie

- [1] Suha Alawadhi, Armando Aldama-Nalda, Hafedh Chourabi, J Ramon Gil-Garcia, Sofia Leung, Sehl Mellouli, Taewoo Nam, Theresa A Pardo, Hans J Scholl, and Shawn Walker. Building understanding of smart city initiatives. In *International conference on electronic government*, pages 40–53. Springer, 2012.
- [2] Edward O’Dwyer, Indranil Pan, Salvador Acha, and Nilay Shah. Smart energy systems for sustainable smart cities : Current developments, trends and future directions. *Applied energy*, 237 :581–597, 2019.
- [3] Helen Couclelis. The construction of the digital city. *Environment and Planning B : Planning and design*, 31(1) :5–19, 2004.
- [4] K Keane and V Nisi. Experience prototyping, 224–237, 2013.
- [5] Vito Albino, Umberto Berardi, and Rosa Maria Dangelico. Smart cities : Definitions, dimensions, performance, and initiatives. *Journal of urban technology*, 22(1) :3–21, 2015.
- [6] David M Gann, Mark Dodgson, and Dheeraj Bhardwaj. Physical–digital integration in city infrastructure. *IBM Journal of Research and Development*, 55(1.2) :8–1, 2011.
- [7] Eugenio Leanza and Gianni Carbonaro. Attaining sustainable, smart investment : The smart city as a place-based capital allocation instrument. In *E-Planning and Collaboration : Concepts, Methodologies, Tools, and Applications*, pages 179–204. IGI Global, 2018.
- [8] Doug Washburn, Usman Sindhu, Stephanie Balaouras, Rachel A Dines, Nicholas Hayes, and Lauren E Nelson. Helping cios understand “smart city” initiatives. *Growth*, 17(2) :1–17, 2009.
- [9] Riccardo Petrolo, Valeria Loscri, and Nathalie Mitton. Towards a smart city based on cloud of things, a survey on the smart city vision and paradigms. *Transactions on emerging telecommunications technologies*, 28(1) :e2931, 2017.
- [10] Gerhard P Hancke, Bruno de Carvalho e Silva, and Gerhard P Hancke Jr. The role of advanced sensing in smart cities. *Sensors*, 13(1) :393–425, 2012.
- [11] Saraju P Mohanty, Uma Choppali, and Elias Kougiianos. Everything you wanted to know about smart cities : The internet of things is the backbone. *IEEE Consumer Electronics Magazine*, 5(3) :60–70, 2016.
- [12] Djamel Saba, Youcef Sahli, Brahim Berbaoui, and Rachid Maouedj. Towards smart cities : challenges, components, and architectures. In *Toward Social Internet of Things (SIoT) : enabling technologies, architectures and applications*, pages 249–286. Springer, 2020.

- [13] Retno Susanti, Sugiono Soetomo, Imam Buchori, and PM Brotosunaryo. Smart growth, smart city and density : In search of the appropriate indicator for residential density in indonesia. *Procedia-Social and Behavioral Sciences*, 227 :194–201, 2016.
- [14] ISO/IEC JTC 1. Smart cities, 2014.
- [15] Rudolf Giffinger and Haindlmaier Gudrun. Smart cities ranking : an effective instrument for the positioning of the cities? *ACE : architecture, city and environment*, 4(12) :7–26, 2010.
- [16] Martina Kmecová. of thesis : Transformation of the city into smart city and its impact on the. *Journal of urban technology*, 18(2) :65–82.
- [17] Rudolf Giffinger, Christian Fertner, Hans Kramar, Evert Meijers, et al. City-ranking of european medium-sized cities. *Cent. Reg. Sci. Vienna UT*, 9 :1–12, 2007.
- [18] Anthony J Bokolo, Mazlina Abdul Majid, and Awanis Romli. A trivial approach for achieving smart city : A way forward towards a sustainable society. In *2018 21st Saudi Computer Society National Computer Conference (NCC)*, pages 1–6. IEEE, 2018.
- [19] Anthony Junior Bokolo and Sobah Abbas Petersen. A smart city adoption model to improve sustainable living. Norsk konferanse for organisasjoners bruk av informasjonsteknologi, 2019.
- [20] Zurinah Tahir and Jalaluddin Abdul Malek. Main criteria in the development of smart cities determined using analytical method. *Planning Malaysia*, 14, 2016.
- [21] Bokolo Anthony Jr, Mazlina Abdul Majid, and Awanis Romli. Green information technology system practice for sustainable collaborative enterprise : a structural literature review. *International Journal of Sustainable Society*, 9(3) :242–272, 2017.
- [22] Zheng Xiang and Daniel R Fesenmaier. *Analytics in smart tourism design : concepts and methods*. Springer, 2017.
- [23] Rida Khatoun and Sherali Zeadally. Smart cities : concepts, architectures, research opportunities. *Communications of the ACM*, 59(8) :46–57, 2016.
- [24] Mariacristina Roscia, Michela Longo, and George Cristian Lazaroiu. Smart city by multi-agent systems. In *2013 International Conference on Renewable Energy Research and Applications (ICREERA)*, pages 371–376. IEEE, 2013.
- [25] ChuanTao Yin, Zhang Xiong, Hui Chen, JingYuan Wang, Daven Cooper, and Bertrand David. A literature survey on smart cities. *Science China Information Sciences*, 58(10) :1–18, 2015.
- [26] Yunpeng Li, Clark Hu, Chao Huang, and Liqiong Duan. The concept of smart tourism in the context of tourism information services. *Tourism management*, 58 :293–300, 2017.
- [27] Ulrike Gretzel, Chulmo Koo, Marianna Sigala, and Zheng Xiang. Special issue on smart tourism : convergence of information technologies, experiences, and theories. *Electronic Markets*, 25(3) :175–177, 2015.
- [28] Rua-Huan Tsaih and Chih Chun Hsu. Artificial intelligence in smart tourism : A conceptual framework. *Artificial Intelligence*, 2, 2018.

- [29] Ulrike Gretzel, Marianna Sigala, Zheng Xiang, and Chulmo Koo. Smart tourism : foundations and developments. *Electronic markets*, 25(3) :179–188, 2015.
- [30] Kim Boes, Dimitrios Buhalis, and Alessandro Inversini. Conceptualising smart tourism destination dimensions. In *Information and communication technologies in tourism 2015*, pages 391–403. Springer, 2015.
- [31] Taehyee Um and Namho Chung. Does smart tourism technology matter? lessons from three smart tourism cities in south korea. *Asia Pacific Journal of Tourism Research*, 26(4) :396–414, 2021.
- [32] Fabiana Lorenzi and Francesco Ricci. Case-based recommender systems : A unifying view. In *IJCAI Workshop on Intelligent Techniques for Web Personalization*, pages 89–113. Springer, 2003.
- [33] Fatemeh Alyari and Nima Jafari Navimipour. Recommender systems : a systematic review of the state of the art literature and suggestions for future research. *Kybernetes*, 2018.
- [34] Maayan Zhitomirsky-Geffet and Avital Zadok. Risk analysis and prediction in welfare institutions using a recommender system. *AI & SOCIETY*, 33(4) :511–525, 2018.
- [35] Abolfazl Zare, Mohammad Reza Motadel, and Aliakbar Jalali. Presenting a hybrid model in social networks recommendation system architecture development. *AI & SOCIETY*, 35(2) :469–483, 2020.
- [36] Joseph A Konstan and John Riedl. Recommender systems : from algorithms to user experience. *User modeling and user-adapted interaction*, 22(1) :101–123, 2012.
- [37] Chenguang Pan and Wenxin Li. Research paper recommendation with topic analysis. In *2010 International Conference On Computer Design and Applications*, volume 4, pages V4–264. IEEE, 2010.
- [38] Gediminas Adomavicius and Alexander Tuzhilin. Context-aware recommender systems. In *Recommender systems handbook*, pages 217–253. Springer, 2011.
- [39] Folasade Olubusola Isinkaye, Yetunde O Folajimi, and Bolande Adefowoke Ojokoh. Recommendation systems : Principles, methods and evaluation. *Egyptian informatics journal*, 16(3) :261–273, 2015.
- [40] Dietmar Jannach, Markus Zanker, Alexander Felfernig, and Gerhard Friedrich. *Recommender systems : an introduction*. Cambridge University Press, 2010.
- [41] Robin Burke. Hybrid recommender systems : Survey and experiments. *User modeling and user-adapted interaction*, 12(4) :331–370, 2002.
- [42] Jesús Bobadilla, Fernando Ortega, Antonio Hernando, and Abraham Gutiérrez. Recommender systems survey. *Knowledge-based systems*, 46 :109–132, 2013.
- [43] David H Stern, Ralf Herbrich, and Thore Graepel. Matchbox : large scale online bayesian recommendations. In *Proceedings of the 18th international conference on World wide web*, pages 111–120, 2009.
- [44] Mhamed HADJ HENNI, Nassim DENNOUNI, and Zohra SLAMA. Sig et la recommandation contextuelle des poi.

- [45] Robin Burke. Survey and experiments. *User Model. User-Adap. Inter*, 12(4) :331–370, 2002.
- [46] Douglas W Oard, Jinmook Kim, et al. Implicit feedback for recommender systems. In *Proceedings of the AAAI workshop on recommender systems*, volume 83, pages 81–83. AAAI, 1998.
- [47] Jürgen Buder and Christina Schwind. Learning with personalized recommender systems : A psychological view. *Computers in Human Behavior*, 28(1) :207–216, 2012.
- [48] Adrian Kristoffer Tvette and Matias Pettersen. A hybrid recommender system for context-aware recommendations of restaurants. Master’s thesis, NTNU, 2016.
- [49] Karamollah Bagheri Fard, Mehrbakhsh Nilashi, Mohsen Rahmani, and Othman Ibrahim. Recommender system based on semantic similarity. *International Journal of Electrical & Computer Engineering (2088-8708)*, 3(6), 2013.
- [50] Gourav Jain, Tripti Mahara, and Kuldeep Narayan Tripathi. A survey of similarity measures for collaborative filtering-based recommender system. In *Soft computing : theories and applications*, pages 343–352. Springer, 2020.
- [51] Yong Wang, Jiangzhou Deng, Jerry Gao, and Pu Zhang. A hybrid user similarity model for collaborative filtering. *Information Sciences*, 418 :102–118, 2017.
- [52] Taner Arsan, Efecan Köksal, and Zeki Bozkuş. Comparison of collaborative filtering.
- [53] Zhenhua Tan and Liangliang He. An efficient similarity measure for user-based collaborative filtering recommender systems inspired by the physical resonance principle. *IEEE Access*, 5 :27211–27228, 2017.
- [54] Jiawei Han, Jian Pei, and Micheline Kamber. *Data mining : concepts and techniques*. Elsevier, 2011.
- [55] Batta Mahesh. Machine learning algorithms-a review. *International Journal of Science and Research (IJSR).[Internet]*, 9 :381–386, 2020.
- [56] Ken Goldberg, Theresa Roeder, Dhruv Gupta, and Chris Perkins. Eigentaste : A constant time collaborative filtering algorithm. *information retrieval*, 4(2) :133–151, 2001.
- [57] Mark Claypool, Anuja Gokhale, Tim Miranda, Pavel Murnikov, Dmitry Netes, and Matthew Sartin. Combining content-based and collaborative filters in an online newspaper. In *Proceedings of ACM SIGIR workshop on recommender systems*, volume 60, pages 1853–1870. Citeseer, 1999.
- [58] Paul Cotter and Barry Smyth. Ptv : Intelligent personalised tv guides. In *AAAI/IAAI*, pages 957–964, 2000.
- [59] Marina Drosou and Evaggelia Pitoura. Search result diversification. *ACM SIGMOD Record*, 39(1) :41–47, 2010.
- [60] Robin Burke. The adaptive web : Methods and strategies of web personalization. Incs, vol. 4321, 2007.
- [61] Deuk Hee Park, Hyea Kyeong Kim, Il Young Choi, and Jae Kyeong Kim. A literature review and classification of recommender systems research. *Expert systems with applications*, 39(11) :10059–10072, 2012.

- [62] Xiaoyuan Su and Taghi M Khoshgoftaar. A survey of collaborative filtering techniques. *Advances in artificial intelligence*, 2009, 2009.
- [63] C Sardianos, N Tsirakis, and I Varlamis. A survey on the scalability of recommender systems for social networks. In *Social Networks Science : Design, Implementation, Security, and Challenges*, pages 89–110. Springer, 2018.
- [64] Zhang Wen-ying and Qian Guo-ming. A new framework of a personalized location-based restaurant recommendation system in mobile application. In *2013 International Conference on Management Science and Engineering 20th Annual Conference Proceedings*, pages 166–172. IEEE, 2013.
- [65] Takafumi Hasegawa and Takahiro Hayashi. Collaborative filtering based spot recommendation seamlessly available in home and away areas. In *2013 IEEE/ACIS 12th International Conference on Computer and Information Science (ICIS)*, pages 547–548. IEEE, 2013.
- [66] Daniel Lee and H Sebastian Seung. Algorithms for non-negative matrix factorization. *Advances in neural information processing systems*, 13, 2000.
- [67] Yehuda Koren. Factorization meets the neighborhood : a multifaceted collaborative filtering model. In *Proceedings of the 14th ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 426–434, 2008.
- [68] Andriy Mnih and Russ R Salakhutdinov. Probabilistic matrix factorization. *Advances in neural information processing systems*, 20, 2007.
- [69] Mohsen Jamali and Martin Ester. A matrix factorization technique with trust propagation for recommendation in social networks. In *Proceedings of the fourth ACM conference on Recommender systems*, pages 135–142, 2010.
- [70] Jing Sun, Yun Xiong, Yangyong Zhu, Junming Liu, Chu Guan, and Hui Xiong. Multi-source information fusion for personalized restaurant recommendation. In *Proceedings of the 38th International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 983–986, 2015.
- [71] Ling Li, Ya Zhou, Han Xiong, Cailin Hu, and Xiafei Wei. Collaborative filtering based on user attributes and user ratings for restaurant recommendation. In *2017 IEEE 2nd Advanced Information Technology, Electronic and Automation Control Conference (IAEAC)*, pages 2592–2597. IEEE, 2017.
- [72] Igo Brilhante, Jose Antonio Macedo, Franco Maria Nardini, Raffaele Perego, and Chiara Renso. Where shall we go today ? planning touristic tours with tripbuilder. In *Proceedings of the 22nd ACM international conference on Information & Knowledge Management*, pages 757–762, 2013.
- [73] Alexander Felfernig, Ludovico Boratto, Martin Stettinger, and Marko Tkalčič. *Group recommender systems : An introduction*. Springer, 2018.
- [74] Gediminas Adomavicius and Alexander Tuzhilin. 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, 2005.

- [75] Rahul Katarya and Om Prakash Verma. Restaurant recommender system based on psychographic and demographic factors in mobile environment. In *2015 International Conference on Green Computing and Internet of Things (ICGCIoT)*, pages 907–912. IEEE, 2015.
- [76] Edjalma Queiroz Da Silva, Celso G Camilo-Junior, Luiz Mario L Pascoal, and Thierson C Rosa. An evolutionary approach for combining results of recommender systems techniques based on collaborative filtering. *Expert Systems with Applications*, 53 :204–218, 2016.
- [77] Yung-Ming Li, Chia-Ling Chou, and Lien-Fa Lin. A social recommender mechanism for location-based group commerce. *Information Sciences*, 274 :125–142, 2014.
- [78] Ludovico Boratto, Salvatore Carta, Gianni Fenu, and Roberto Saia. Semantics-aware content-based recommender systems : Design and architecture guidelines. *Neurocomputing*, 254 :79–85, 2017.
- [79] Peter Brusilovski, Alfred Kobsa, and Wolfgang Nejdl. *The adaptive web : methods and strategies of web personalization*, volume 4321. Springer Science & Business Media, 2007.
- [80] Ditdit Nugeraha Utama, Luqman Isyraqi Lazuardi, Hersy Ayu Qadrya, Bella Marisela Caroline, Tris Renanda, and Atthiya Prima Sari. Worth eat : An intelligent application for restaurant recommendation based on customer preference (case study : Five types of restaurant in tangerang selatan region, indonesia). In *2017 5th international conference on information and communication technology (ICoICT7)*, pages 1–4. IEEE, 2017.
- [81] Nanthaphat Koetphrom, Panachai Charusangvittaya, and Daricha Sutivong. Comparing filtering techniques in restaurant recommendation system. In *2018 2nd International Conference on Engineering Innovation (ICEI)*, pages 46–51. IEEE, 2018.
- [82] Dheeru Dua, Casey Graff, et al. Uci machine learning repository. 2017.
- [83] Aji Achmad Mustafa and Indra Budi. Recommendation system based on item and user similarity on restaurants directory online. In *2018 6th International Conference on Information and Communication Technology (ICoICT)*, pages 70–74. IEEE, 2018.
- [84] Luis Martinez, Rosa M Rodriguez, and Macarena Espinilla. Reja : a georeferenced hybrid recommender system for restaurants. In *2009 IEEE/WIC/ACM International Joint Conference on Web Intelligence and Intelligent Agent Technology*, volume 3, pages 187–190. IEEE, 2009.
- [85] Chung-Hua Chu and Se-Hsien Wu. A chinese restaurant recommendation system based on mobile context-aware services. In *2013 IEEE 14th International Conference on Mobile Data Management*, volume 2, pages 116–118. IEEE, 2013.
- [86] Anant Gupta and Kuldeep Singh. Location based personalized restaurant recommendation system for mobile environments. In *2013 International Conference on Advances in Computing, Communications and Informatics (ICACCI)*, pages 507–511. IEEE, 2013.
- [87] Daniele Quercia, Neal Lathia, Francesco Calabrese, Giusy Di Lorenzo, and Jon Crowcroft. Recommending social events from mobile phone location data. In *2010 IEEE international conference on data mining*, pages 971–976. IEEE, 2010.

- [88] Moon-Hee Park, Jin-Hyuk Hong, and Sung-Bae Cho. Location-based recommendation system using bayesian user’s preference model in mobile devices. In *International conference on ubiquitous intelligence and computing*, pages 1130–1139. Springer, 2007.
- [89] Souvik Debnath, Niloy Ganguly, and Pabitra Mitra. Feature weighting in content based recommendation system using social network analysis. In *Proceedings of the 17th international conference on World Wide Web*, pages 1041–1042, 2008.
- [90] Jun Zeng, Feng Li, Haiyang Liu, Junhao Wen, and Sachio Hirokawa. A restaurant recommender system based on user preference and location in mobile environment. In *2016 5th IIAI International Congress on Advanced Applied Informatics (IIAI-AAI)*, pages 55–60. IEEE, 2016.
- [91] Seokhwan Kim and Rafael E Banchs. R-cube : a dialogue agent for restaurant recommendation and reservation. In *Signal and Information Processing Association Annual Summit and Conference (APSIPA), 2014 Asia-Pacific*, pages 1–6. IEEE, 2014.
- [92] Norma Saiph Savage, Maciej Baranski, Norma Elva Chavez, and Tobias Höllerer. I’m feeling loco : A location based context aware recommendation system. In *Advances in Location-Based Services*, pages 37–54. Springer, 2012.
- [93] Saeideh Bakhshi, Partha Kanuparth, and Eric Gilbert. Demographics, weather and online reviews : A study of restaurant recommendations. In *Proceedings of the 23rd international conference on World wide web*, pages 443–454, 2014.
- [94] Nesrine Zemirli. *Modèle d’accès personnalisé à l’information basé sur les Diagrammes d’Influence intégrant un profil utilisateur évolutif*. PhD thesis, Université Paul Sabatier-Toulouse III, 2008.
- [95] Abir Hoadjli. *Une approche Cloud Computing pour le développement de ‘Smart City’ à base de recommandations*. PhD thesis, Université Mohamed Khider–BISKRA, 2020.
- [96] Lynda Tamine and Wahiba Bahsoun. Définition d’un profil multidimensionnel de l’utilisateur : vers une technique basée sur l’interaction entre dimensions. In *Actes de la Conférence francophone en Recherche d’Information et Applications (CORIA 2006)*, pages 225–236, 2006.
- [97] Dimitre Kostadinov. *Personnalisation de l’information : une approche de gestion de profils et de reformulation de requêtes*. PhD thesis, Université de Versailles-Saint Quentin en Yvelines, 2007.
- [98] Alex M Susskind and Edwin K Chan. How restaurant features affect check averages : a study of the toronto restaurant market. *The Cornell Hotel and Restaurant Administration Quarterly*, 41(6) :56–63, 2000.
- [99] Ahlem Bouguettaya. *Une Approche Basée Deep Learning Pour La Classification des Sentiments*. PhD thesis, Université laarbi tebessi tebessa, 2019.
- [100] Saqib Alam and Nianmin Yao. The impact of preprocessing steps on the accuracy of machine learning algorithms in sentiment analysis. *Computational and Mathematical Organization Theory*, 25(3) :319–335, 2019.

- [101] Rai B Shamantha, Sweekriti M Shetty, and Prakhyath Rai. Sentiment analysis using machine learning classifiers : evaluation of performance. In *2019 IEEE 4th International Conference on Computer and Communication Systems (ICCCS)*, pages 21–25. IEEE, 2019.
- [102] Samira Zad, Maryam Heidari, James H Jones, and Ozlem Uzuner. A survey on concept-level sentiment analysis techniques of textual data. In *2021 IEEE World AI IoT Congress (AIIoT)*, pages 0285–0291. IEEE, 2021.
- [103] Stemming vs lemmatization nlp : Data science and machine learning.
- [104] Aditi Mukerjee. Spam filtering using bag-of-words, Jan 2021.
- [105] Matje Van de Camp and Antal Van den Bosch. The socialist network. *Decision Support Systems*, 53(4) :761–769, 2012.
- [106] Hanhoon Kang, Seong Joon Yoo, and Dongil Han. Senti-lexicon and improved naïve bayes algorithms for sentiment analysis of restaurant reviews. *Expert Systems with Applications*, 39(5) :6000–6010, 2012.
- [107] Daniel Lemire and Anna Maclachlan. Slope one predictors for online rating-based collaborative filtering. In *Proceedings of the 2005 SIAM International Conference on Data Mining*, pages 471–475. SIAM, 2005.
- [108] Tong Qiang Jiang and Wei Lu. Improved slope one algorithm based on time weight. In *Applied Mechanics and Materials*, volume 347, pages 2365–2368. Trans Tech Publ, 2013.
- [109] Ruisheng Zhang, Qidong Liu, Rongjing Hu, Huiyi Ma, and Yongna Yuan. Collaborative filtering : User similarity in slope one algorithm. In *JCIS*, volume 10, pages 10413–10422, 2014.
- [110] Fernando Pérez and Brian E Granger. Ipython : a system for interactive scientific computing. *Computing in science & engineering*, 9(3) :21–29, 2007.
- [111] Bernadette M Randles, Irene V Pasquetto, Milena S Golshan, and Christine L Borgman. Using the jupyter notebook as a tool for open science : An empirical study. In *2017 ACM/IEEE Joint Conference on Digital Libraries (JCDL)*, pages 1–2. IEEE, 2017.
- [112] Henri Michel and Olivier. Google colab : Le guide ultime, Nov 2019.
- [113] Tiago Carneiro, Raul Victor Medeiros Da Nóbrega, Thiago Nepomuceno, Gui-Bin Bian, Victor Hugo C De Albuquerque, and Pedro Pedrosa Reboucas Filho. Performance analysis of google colaboratory as a tool for accelerating deep learning applications. *IEEE Access*, 6 :61677–61685, 2018.
- [114] Michel F Sanner et al. Python : a programming language for software integration and development. *J Mol Graph Model*, 17(1) :57–61, 1999.
- [115] Wes McKinney. *Python for data analysis : Data wrangling with Pandas, NumPy, and IPython*. " O'Reilly Media, Inc.", 2012.
- [116] Mark Lutz. *Learning python : Powerful object-oriented programming*. " O'Reilly Media, Inc.", 2013.

- [117] Andreas Holzinger. Machine learning for health informatics. In *Machine learning for health informatics*, pages 1–24. Springer, 2016.
- [118] Wes McKinney and PD Team. Pandas-powerful python data analysis toolkit. *Pandas—Powerful Python Data Analysis Toolkit*, 1625, 2015.
- [119] Ekaba Bisong. Introduction to scikit-learn. In *Building machine learning and deep learning models on Google cloud platform*, pages 215–229. Springer, 2019.
- [120] Nicolas Hug. Surprise : A python library for recommender systems. *Journal of Open Source Software*, 5(52) :2174, 2020.
- [121] Steven Bird, Ewan Klein, and Edward Loper. Nltk documentation. *Online : accessed April*, 2008.
- [122] Venkateswarlu Bonta and Nandhini Kumaresh2and N Janardhan. A comprehensive study on lexicon based approaches for sentiment analysis. *Asian Journal of Computer Science and Technology*, 8(S2) :1–6, 2019.
- [123] Yelp open dataset.
- [124] Larxel. Trip advisor hotel reviews, Sep 2020.
- [125] Dheeru Dua and Casey Graff. UCI machine learning repository, 2017.