ÉCOLE NATIONALE SUPÉRIEURE D'INFORMATIQUE ET D'ANALYSE DES SYSTÈMES

FILIÈRE : E-MANAGEMENT BUSINESS INTELLIGENCE

Rapport de projet de fin de deuxième année :

Recommandation de menus de restaurants basée sur l'apprentissage automatique

---

Juin 2022

Réalisé par :

ELARGAB oulaya
L'HASSNAOUI Kaoutar

Encadré par : Mr. ALAMI Yasser

Jury : Lamia BENHIBA
M. A. JANATI IDRISSI

Année académique 2021/2022

Résumé

L'agriculture représente une part importante de l'économie et la plupart des sociétés en dépendent pour leur subsistance. Cela fait de l'eau une ressource importante qui doit être préservée à l'aide des dernières technologies disponibles.

Outre son rôle fondamental dans l'industrie 4.0, l'IdO étend également ses capacités à l'agriculture intelligente. Le travail proposé ici vise à développer un système intelligent à faible coût pour l'irrigation intelligente.

Il utilise l'IdO pour que les dispositifs utilisés dans le système parlent et se connectent d'eux-mêmes, avec des capacités telles que : le mode administrateur pour l'interaction avec l'utilisateur, la configuration unique pour l'estimation du calendrier d'irrigation, la prise de décision basée sur les neurones pour un soutien intelligent et la surveillance des données à distance. Un banc d'essai de culture a été choisi pour présenter les résultats du système proposé, notamment le programme d'irrigation, la prise de décision par réseau neuronal et la visualisation des données à distance.

Le réseau neuronal fournit l'intelligence nécessaire au dispositif qui prend en compte l'entrée du capteur actuel et masque le programme d'irrigation pour une irrigation efficace. Le système utilise MQTT et HTTP pour tenir l'utilisateur informé de la situation actuelle des cultures, même à distance.

Le système proposé s'avère bénéfique grâce à son intelligence, son faible coût et sa portabilité, ce qui le rend approprié pour les serres, les fermes, etc.

Mot clés : **IdO , Evapotranspiration, MQTT , Réseau neuronal , Irrigation .**

Abstract

Agriculture is an important part of the economy and most societies depend on it for their livelihood. Water is therefore an important resource that must be preserved using the latest available technologies.

In addition to its fundamental role in Industry 4.0, IoT is extending its capabilities to smart agriculture. The work proposed here aims to develop a low-cost smart system for smart irrigation.

It uses IoT to make the devices used in the system talk and connect on their own, with capabilities such as : administrator mode for user interaction, one-time configuration for irrigation scheduling estimation, neural network decision making for intelligent assistance, and remote data monitoring. A crop test bed was selected to showcase the results of the proposed system, including irrigation scheduling, neural network decision making, and remote data visualization.

The neural network provides the necessary intelligence to the device that takes into account the current sensor input and masks the irrigation schedule for efficient irrigation. The system uses MQTT and HTTP to keep the user informed of the current crop situation even from a distance.

Keywords : **IoT , Evapotranspiration, MQTT , Neural network , Irrigation**

Table des matières

Table des figures

Introduction générale

De nombreux travaux ont été et sont menés par différents groupes de recherche dans le domaine de l'agriculture intelligente. L'eau s'avère l'une des ressources les plus importantes pour l'agriculture. Cela a conduit à un travail intensif lié à l'utilisation efficace de cette ressource limitée. Une grande quantité d'eau, environ 100x plus que l'utilisation personnelle, est consommée par l'alimentation et l'agriculture et près de 70% des eaux fluviales et souterraines sont utilisées pour l'irrigation, ce qui en fait les plus grands consommateurs de ressources en eau Siebert et al. (2010). Actuellement, sur les 3600 km3 d'eau douce, près de la moitié est perdue en raison de l'évaporation, de la transpiration des cultures, etc. tandis que la moitié restante s'ajoute au niveau des eaux souterraines. La plupart de l'eau utilisée dans l'agriculture est fournie directement par les précipitations, mais les endroits où sévit la sécheresse n'ont que l'irrigation comme option pour l'agriculture. Un système d'irrigation typique se compose d'un plan : méthode d'irrigation, source : plan d'eau à proximité, transporteurs : canalisations vers la ferme, actionneurs : mécanisme marche/arrêt pour contrôler l'eau, vannes : contrôle du débit d'eau.**prédire la note d'un restaurant avec précision est la clé**. Le seul fait de regarder les notes moyennes données par les utilisateurs n'est pas suffisant pour prédire les étoiles de notation du restaurant, notre hypothèse est qu'en intégrant l'analyse des sentiments sur les évaluations des utilisateurs pour le restaurant, nous pouvons générer un meilleur score d'évaluation pour le restaurant. Dans notre projet, nous nous intéressons à l'amélioration du système de recommandation avec l'analyse des sentiments, pour prédire l'évaluation des restaurants à partir des avis des utilisateurs et pour recommander des des restaurants pour différents utilisateurs en fonction d'autres restaurants similaires.

Nous avons divisé notre projet en trois parties. Notre rapport sera donc structuré comme suit :

• Le chapitre 1 intitulé **" Présentation du projet et méthodologie "** nous permettra de présenter notre objectif et de souligner notre problématique suivant une méthodologie adaptée au déroulement de notre réalisation de système intelligent.

• Le chapitre 2 intitulé **" Contexte général du travail"** est consacré à l'introduction générale de l'IdO, son importance et les différentes approches possibles qu'il nous presente.

• Le chapitre 3 nommé **"Approches et méthode proposées"** expose l'idée du système proposé, l'architecture de celui-ci et sa conception.

• Le chapitre 4 nommé **"Expérimentation et résultats"** présente les résultats aboutis et leurs analyses.

• Le chapitre 5 nommé **"Résultats et évaluations"** présente les résultats aboutis et les propables améliorations possibles.

• Dans la **conclusion** on va mettre le point sur l'apport de ce système intelligent, ses limites et les perspectives qu'il peut engendrer.

Chapitre 1

Présentation du projet et méthodologie

1.1 Introduction

Ce premier chapitre a pour but de contextualiser le projet. Nous allons, tout d'abord, présenter le sujet et leur principaux volets. Ensuite on va présenter les objectifs de ce dernier, sa problématique et l'ordonnancement pris pour le réaliser.

1.2 Présentation du sujet

Notre travail concerne les systèmes d'aide à la visite de restaurants et l'essai de nouveaux restaurants en se basant sur les préferences du client .
L'objectif est de concevoir un système de recommandation qui par la suite pourra être implémenté sur dispositifs mobiles, pour améliorer l'expérience du visiteur, en lui recommandant les items les plus pertinents et en l'aidant à personnaliser son parcours.

1.3 Problématique

Dans ce sens le défi relevé est celui-ci ; à quel point peut-on fournir des recommandations intelligentes qui attirent de plus en plus nos clients ?

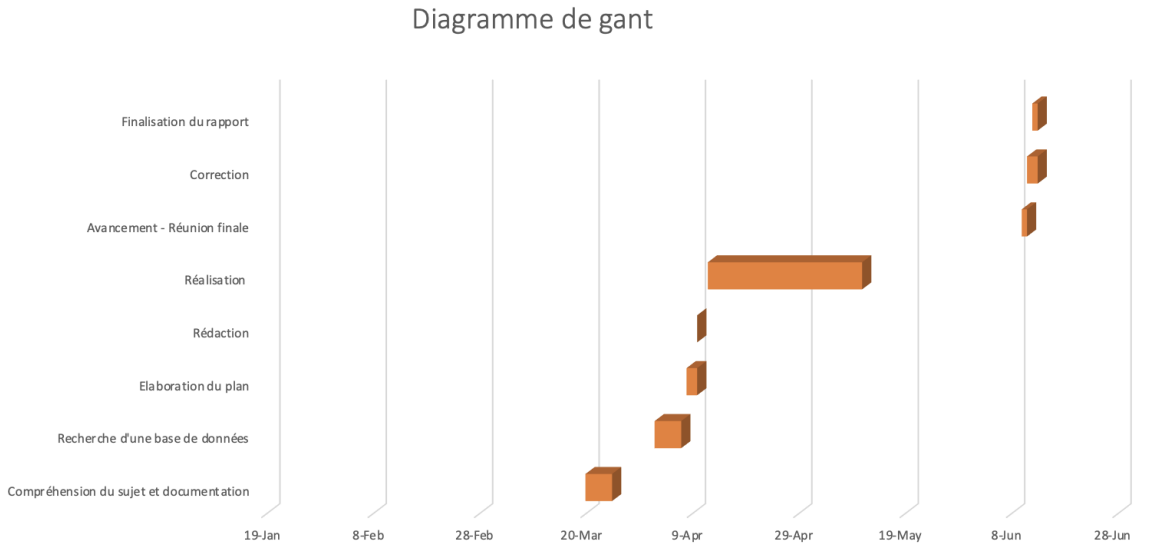
1.4 Objectifs du projet

L'objectif principal de ce projet est de concevoir et réaliser un système adéquat de recommandations de restaurants basé sur le collaborative filtering et la classification à l'aide de l'analyse de sentiments qui pourraient intéresser les clients d'après leurs préférences.
Le défi s'étend aussi à la bonne minimisation de l'erreur en nous usant de toutes les données disponibles dans notre DataSet pour affiner le ciblage et par suite les recommandations.

1.5 Planification du projet Et méthodologie de travail

1.5.1 Diagramme de Gant :

Afin de réaliser le projet dans les délais établis, nous avons utilisé le diagramme de GANTT puisqu'il est considéré comme l'un des outils les plus efficaces pour représenter l'état d'avancement des différentes tâches . Comme le montre la figure ci-dessous, nous avons adopté le modèle en cascade pour réaliser notre projet. Ce dernier représente une organisation des activités sous forme de phases linéaires et séquentielles, où chaque phase correspond à une spécialisation des tâches et dépend des résultats de la phase précédente.



	Compréhension du sujet et documentation	Recherche d'une base de données	Elaboration du plan	Rédaction	Réalisation	Avancement - Réunion finale	Correction	Finalisation du rapport
Début	20-Mar	2/4/2022	8/4/2022	10/4/2022	12/4/2022	10/6/2022	11/6/2022	12/6/2022
Jours passés	5	5	2	0	29	1	2	1
Jours restants	4	0	3	1	10	0	0	0

FIGURE 1.1 – Diagramme de GANTT

1.5.2 Méthodologie : CRISP-DM :

CRISP-DM, qui signifie Cross-Industry Standard Process for Data Mining, est la méthode mise à l'épreuve nous permettant d'orienter notre travail d'exploration de données et de réalisation de projet.

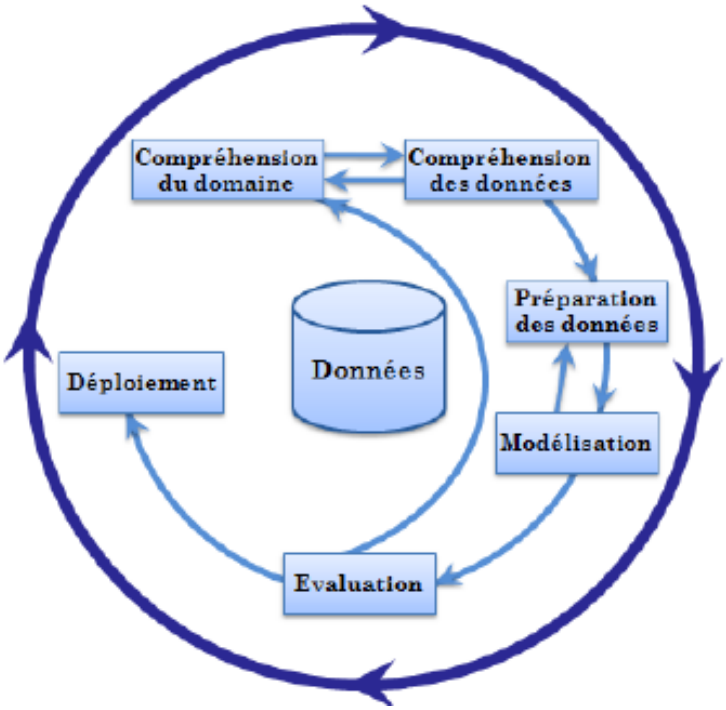


FIGURE 1.2 – Crisp-DM

1.6 Conclusion

Dans ce chapitre nous avons abordé différentes parties. Premièrement, on a présenté le sujet de notre projet. Ensuite, on a expliqué la problématique et précisé l'objectif du projet et la planification suivie.

Le prochain chapitre sera donc dédié à la définition des systèmes de recommandation et leur intérêt pour ensuite passer à l'exploration de nos données.

Chapitre 2

Contexte général

2.1 Introduction

Dans ce chapitre on va définir ce qu'est un système de recommandation en premier lieu. Puis, on entame la phase de conception en presentant les différents algorithmes et modèles existants. Ensuite, on détaillera la démarche suivie pour la réalisation du projet.

2.2 Compréhension du domaine

Avec la facilité d'accès à Internet, nous sommes de plus en plus exposés à une multitude d'informations. Avec tous les blogues, les journaux, les magasins et bien d'autres, cela apporte beaucoup de diversités aux utilisateurs.
Toutefois, avec cette multitude de sources d'informations, la surcharge d'information peut devenir problématique. Afin de remédier à ce problème, divers outils ont été développé pour filtrer les informations avant de les transmettre aux utilisateurs. Les systèmes de recommandations sont des outils permettant un tel filtrage. Le but principal de ces systèmes est de faciliter la prise de décisions pour les utilisateurs en leur offrant des informations selon leurs préférences. Il existe déjà des systèmes de recommandations pour divers produits tels que des films (Netflix), de la musique (last.fm)et des livres (Amazon).

2.3 Catégories d'un système de recommandation

Il y a principalement deux grandes catégories de systèmes de recommandations. L'une se fait avec un filtrage basé sur le contenu et l'autre se fait avec un filtrage collaboratif.

2.3.1 Classification basée sur l'analyse des sentiments

Cette approche basée sur le contenu est mise en oeuvre en prenant en compte les caractéristiques des restaurants recommandés et en créant des groupes de produits

en utilisant une mesure de similitude sur leur contenu. Par contenu, on sous-entend une relation entre un mot et un restaurant. L'un des principaux inconvénients de cette approche est que les caractéristiques sont généralement acquises à l'aide des informations externes des clients qui ne sont pas toujours disponibles ni fiables.
Cette approche est basée sur la mémoire. C'est-à-dire que le système va émettre ses recommandations en se basant sur un voisinage contenu en mémoire.

2.3.2 Filtrage collaboratif

Dans un autre ordre d'idées, le filtrage collaboratif utilise généralement un voisinage d'utilisateurs similaires et les produits recommandés en fonction de l'historique des autres utilisateurs au sein du même voisinage. Lorsque les systèmes émettent leurs recommandations, ils peuvent le faire globalement ou localement. «Globalement» signifie que tous les utilisateurs recevront les mêmes recommandations. Alors que «localement» signifie que les éléments recommandés ne seront pas les mêmes pour tous les utilisateurs.
Cette seconde approche est plutôt basée sur un modèle. Ainsi, il faut d'abord créer des modèles qui ressemblent aux comportements des utilisateurs et ensuite, utiliser ces modèles afin d'émettre les recommandations.

Chapitre 3

Exploration de données

3.1 Description des données :

Les données que nous avons utilisées sont business.json, reviews.json, checkin.json, user.json et tips.json de les ensembles de données ouvertes du Yelp Challenge et que nous avions convertit en des fichiers .csv pour pouvoir travailler avec des DataFrames.

Il y a environ 5,1 millions d'avis dans Review.json, 210 000 Entreprises dans Business.json et 1,97 millions d'utilisateurs dans Users.json. Tout d'abord, nous voulons jeter un œil aux statistiques des données Yelp via la réalisation de certaines visualisations de données.

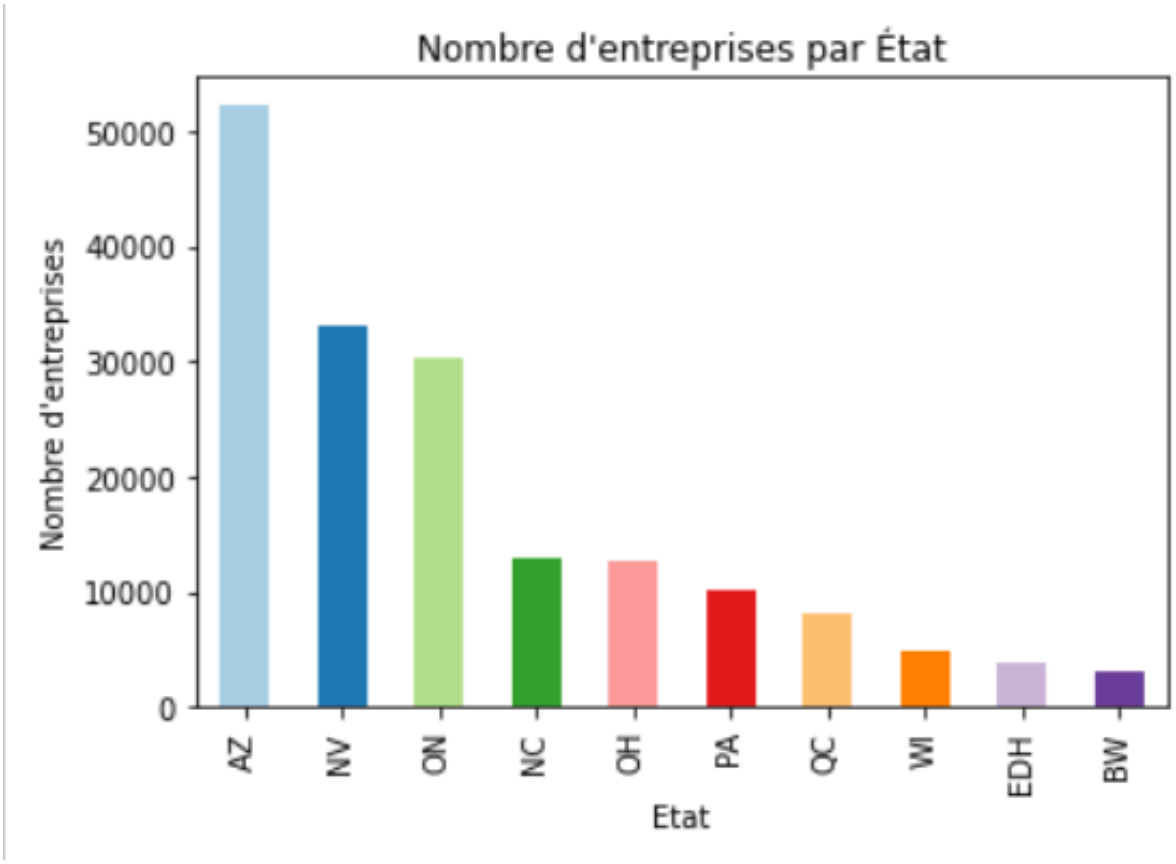


FIGURE 3.1 – Nombre d'entreprises par État

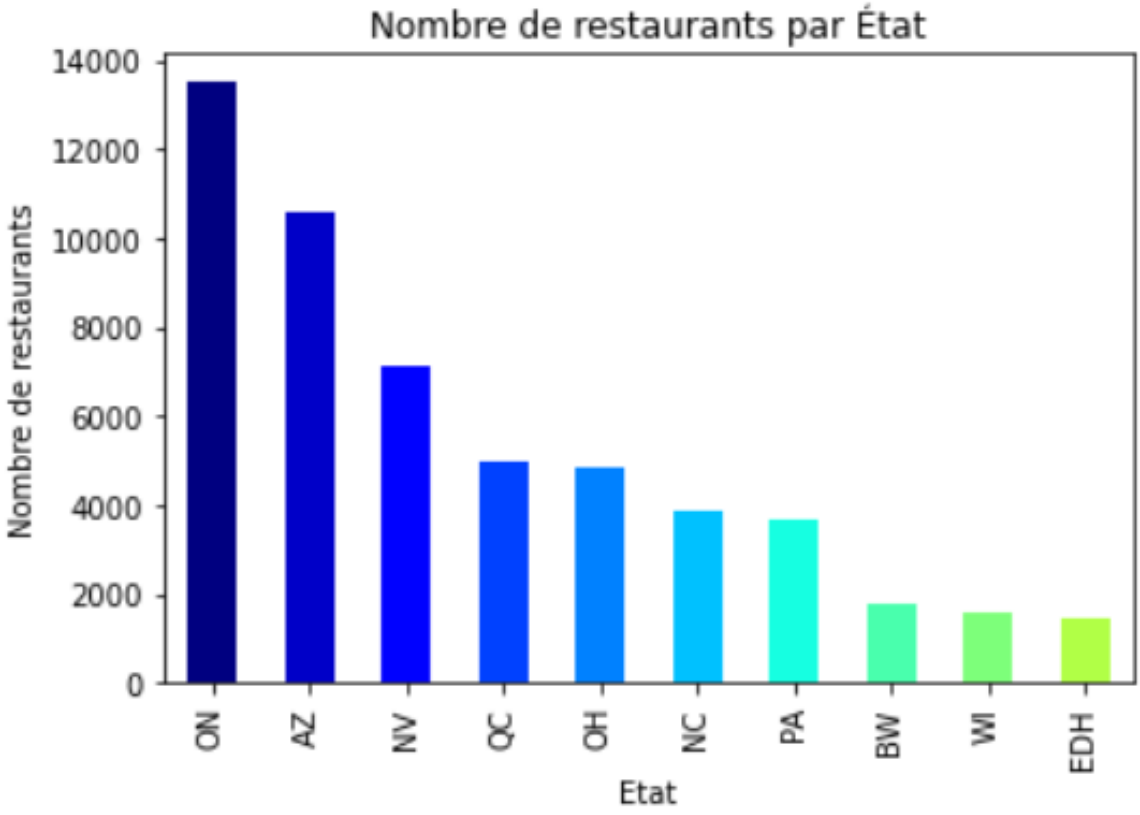


FIGURE 3.2 – Nombre de restaurants par État

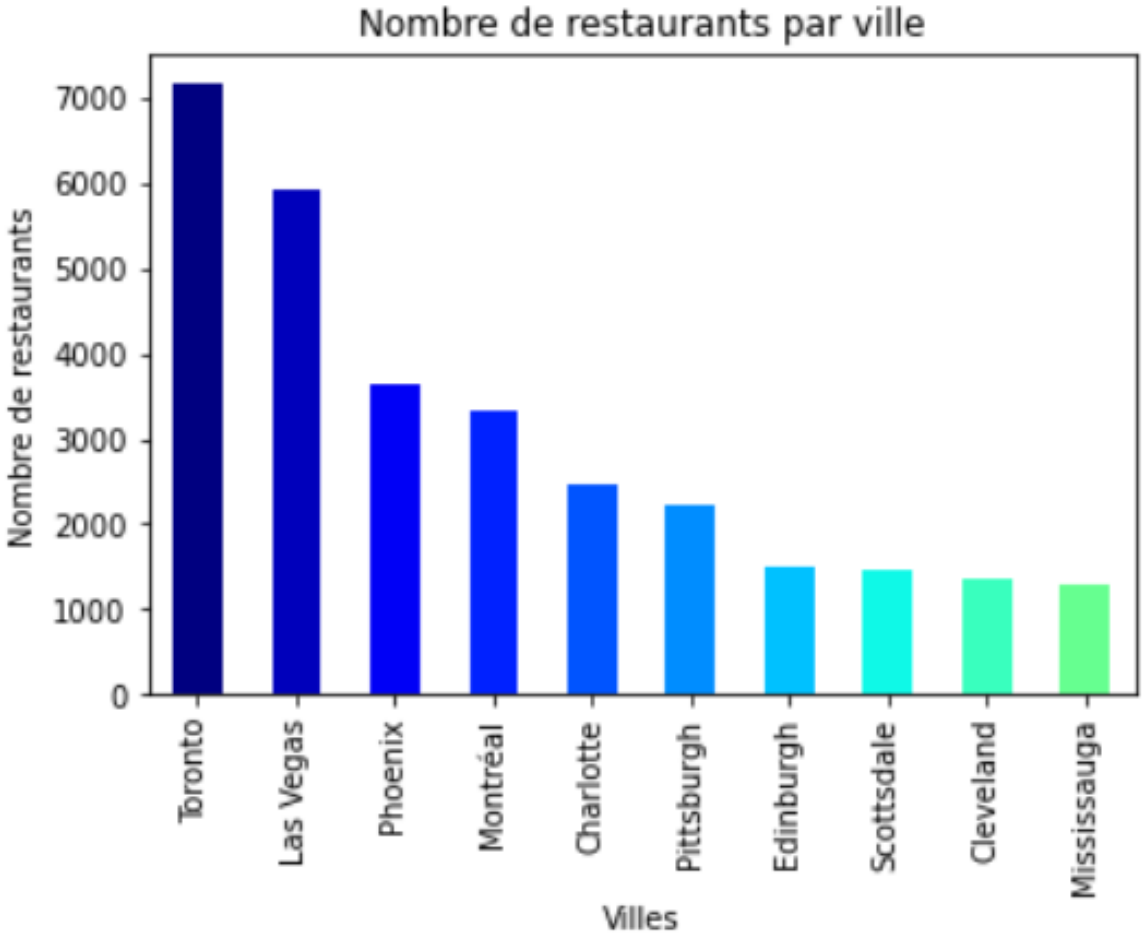


FIGURE 3.3 – Nombre de restaurants par ville

En regardant le business.json, le restaurant a le plus grand nombre dans l'entreprise, et Toronto et L'Ontario a le plus grand nombre de restaurants parmi la ville et l'état respectivement.

Nous avons donc décidé d'utiliser les données des restaurants à Las Vegas car c'est là qu'il y a le plus de restaurants aux États-Unis et le plus grand nombre de critiques dans l'ensemble

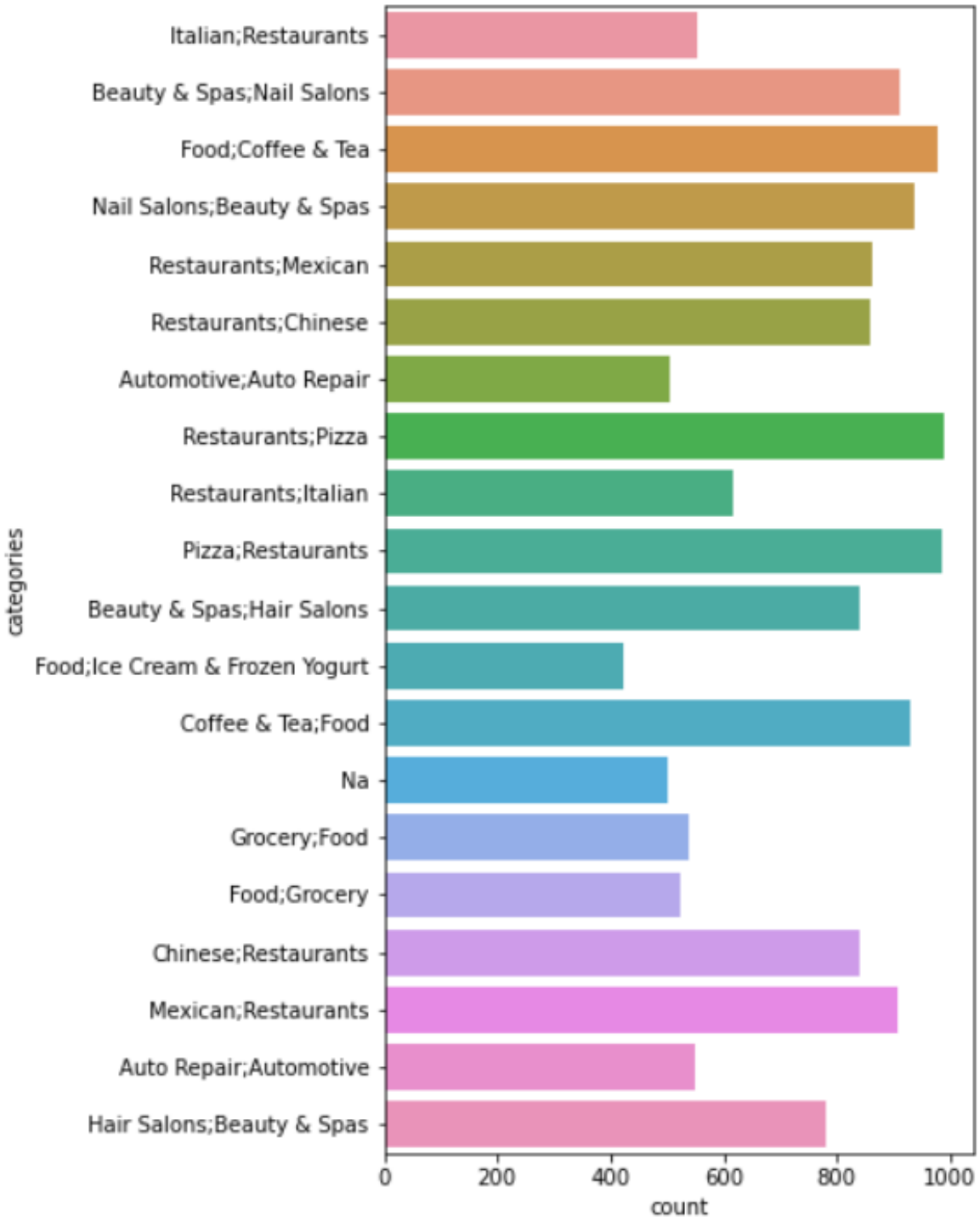


FIGURE 3.4 – Catégories des restaurants et les menus qui s'y trouvent

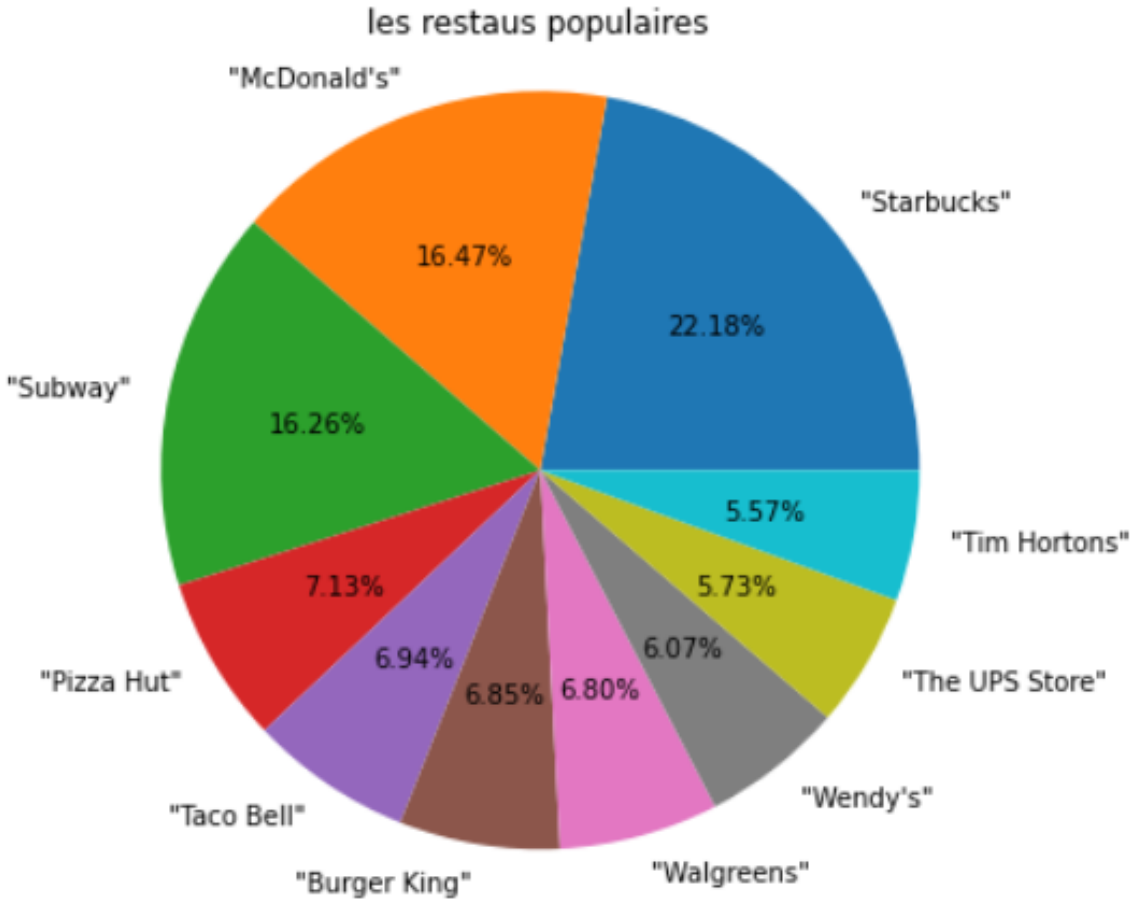


FIGURE 3.5 – Les majeurs noms de restaurants qui se trouvent dans notre base de données et leur contribution

Ensuite, nous examinons les données des utilisateurs de Yelp, où nous pouvons voir que le nombre d'utilisateurs augmente depuis ses débuts, et significativement de 2010 à 2011 ; cependant, il a diminué depuis 2014.

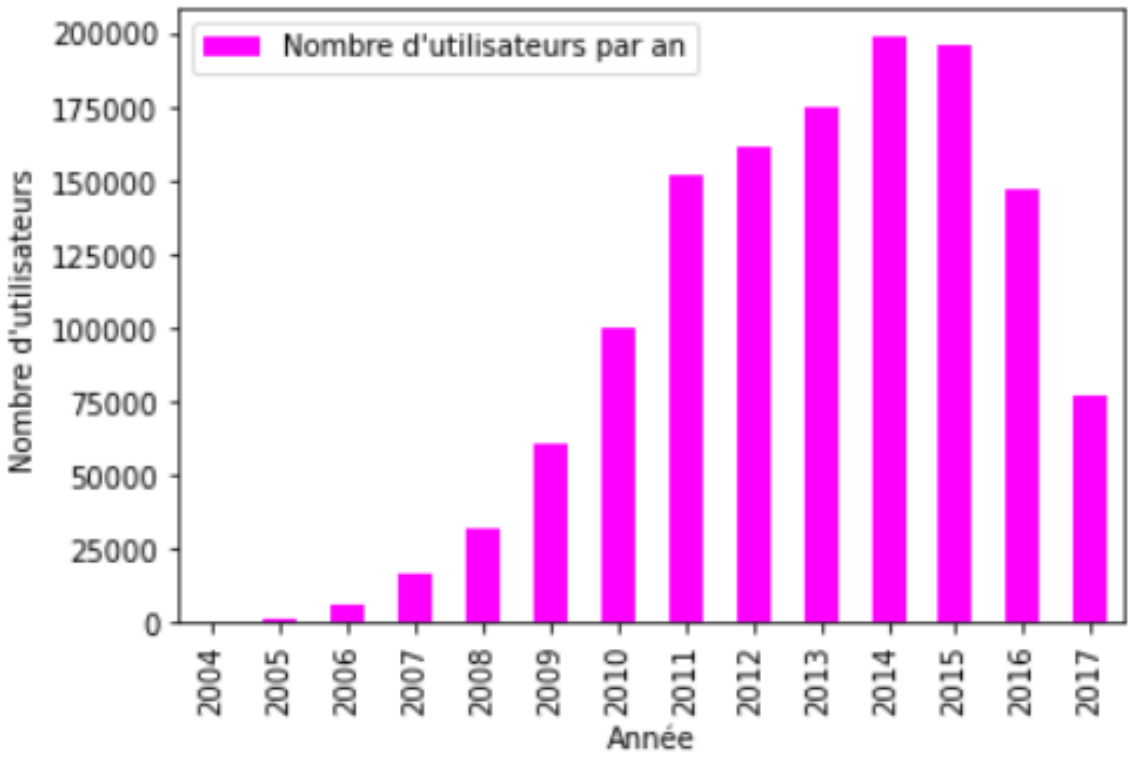


FIGURE 3.6 – Users par an

Les visualisations montrent également que les Stars de notation 1 à 5, la note 4 est la plus populaire donnée par les utilisateurs qui ont la plus grande part de distribution parmi tous les niveaux, tandis que les stars 1 et 2 ont un nombre relativement plus petit de tous.

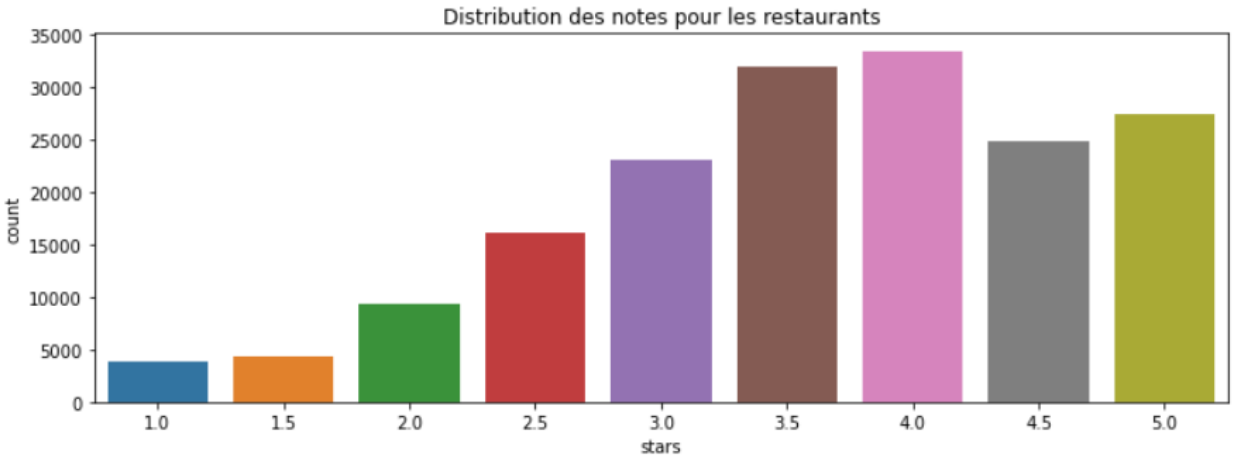


FIGURE 3.7 – Etoiles données aux restaurants

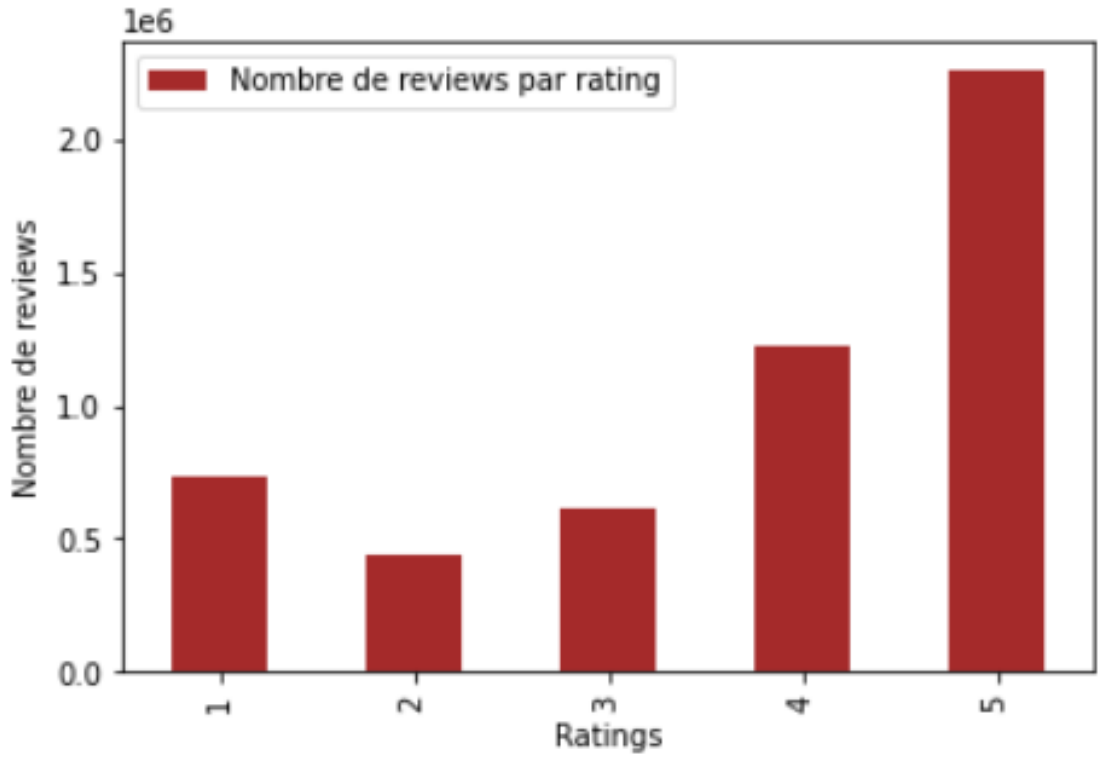


FIGURE 3.8 – Reviews par rating

3.1.1 Préparation et nettoyage des données à utiliser

On se lance tout d'abord de la table business : Pour notre travail, plusieurs champs sont insignifiants . on aura pas besoin du nom du proprietaire du business ni du quartier où se trouve ce dernier ou son adresse exacte,long/lat, qu'il est ouvert ou non !

On ne garde donc que l'Id, la ville state, stars et le nombre de reviews avec les categories

	business_id	name	city	state	stars	review_count	categories
0	FYWN1wneV18bWNgQjJ2GNg	"Dental by Design"	Ahwatukee	AZ	4.0	22	Dentists;General Dentistry;Health & Medical;Or...
1	He-G7vWjzVUysIKrfNbPUQ	"Stephen Szabo Salon"	McMurray	PA	3.0	11	Hair Stylists;Hair Salons;Men's Hair Salons;Bl...
2	KQPW8lFf1y5BT2MxiSZ3QA	"Western Motor Vehicle"	Phoenix	AZ	1.5	18	Departments of Motor Vehicles;Public Services ...
3	8DShNS-LuFqpEWIp0HxijA	"Sports Authority"	Tempe	AZ	3.0	9	Sporting Goods;Shopping
4	PfOCPjBrlQAnz__NXj9h_w	"Brick House Tavern + Tap"	Cuyahoga Falls	OH	3.5	116	American (New);Nightlife;Bars;Sandwiches;Ameri...

FIGURE 3.9 – La nouvelle table Business

La table des **Reviews** est comme suit :

	review_id	user_id	business_id	stars	date	text
0	vkVSCC7xljjrAI4UGfnKEQ	bv2nCi5Qv5vroFiqKGopiw	AEx2SYEUJmTxVVB18LICwA	5	2016-05-28	Super simple place but amazing nonetheless. It...
1	n6QzIUObkYshz4dz2QRJTw	bv2nCi5Qv5vroFiqKGopiw	VR6GpWIda3SfvPC-lg9H3w	5	2016-05-28	Small unassuming place that changes their menu...
2	MV3CcKScW05u5LVfF6ok0g	bv2nCi5Qv5vroFiqKGopiw	CKC0-MOWMqoeWf6s-szl8g	5	2016-05-28	Lester's is located in a beautiful neighborhoo...
3	IXvOzsEMYtiJI0CARmj77Q	bv2nCi5Qv5vroFiqKGopiw	ACFtxLv8pGrrxMm6EgjreA	4	2016-05-28	Love coming here. Yes the place always needs t...
4	L_9BTb55X0GDtThi6GlZ6w	bv2nCi5Qv5vroFiqKGopiw	s2I_Ni76bjJNK9yG60iD-Q	4	2016-05-28	Had their chocolate almond croissant and it wa...

FIGURE 3.10 – Reviews par rating

Le text est le commentaire donné par le user après sa visite au restaurant.

La table **User** contient le nom du client , le nombre des reviews qu'il a présenté et la première fois qu'il a enregistré un commentaire :yelpingsince .

	user_id	name	review_count	yelping_since
0	JJ-aSuM4pCFPdkfoZ34q0Q	Chris	10	2013-09-24
1	uUzsFQn_6cXDh6rPNGbIFA	Tiffy	1	2017-03-02
2	mBneaEEH5EMyxaVyqS-72A	Mark	6	2015-03-13
3	W5mJGs-dcDWRGEhAzUYtoA	Evelyn	3	2016-09-08
4	4E8--zUZO1Rr1IBK4_83fg	Lisa	11	2012-07-16

FIGURE 3.11 – User

Chapitre 4

Réalisation et implémentation d'algorithmes

4.1 Introduction

Dans ce chapitre nous présenterons les diffèrents outils utilisés dans la réalisation du projet. Puis nous allons expliquer les algorithmes implémentés. Ensuite, nous présenterons

4.2 Outils utilisés

4.2.1 Outils de codage

— Python : Python est un langage de programmation interprété, multi-paradigme et multiplateformes. Il favorise la programmation impérative structurée, fonctionnelle et orientée objet.[?]

FIGURE 4.1 – Logo du langage Python

Python est connu par ses bibliothèques riches en fonctionnalités. Dans notre projet,nous nous sommes appuyées sur 4 bibliothèques principales : Pandas, Scikit Learn, Beautiful Soup et Django.

FIGURE 4.2 – Logos des principales bibliothèques de Python utilisées

— Pandas : Pandas est une bibliothèque open-source permettant la manipulation et l'analyse de données de manière simple et intuitive en Python.[?]

— Scikit Learn : Scikit-learn est le principal package de machine learning en python, il possède des dizaines de modèles dont la régression logistique. En tant que package de machine learning, il se concentre avant tout sur l'aspect prédictif du modèle de régression logistique, il permettra de prédire très facilement mais sera pauvre quant l'explication et l'interprétation du modèle. Par contre il est extrêmement efficace pour valider la qualité prédictive des modèles, ajuster les hyper paramètres et passer en production des modèles.[?]

— Numpy : NumPy est une bibliothèque pour le langage de programmation Python, ajoutant la prise en charge de grands tableaux et matrices multidimensionnels, ainsi qu'une grande collection de fonctions mathématiques de haut niveau pour opérer sur ces tableaux. [?]

— Surprise : Simple Python RecommendatIon System Engine. Surprise fait partie de Python scikit pour la construction et l'analyse des systèmes de recommandations qui interagissent avec le rating Data explicitement. Surprise est à la fois utile et simple car il peut former un modèle qui sert des recommandations en utilisant des données annotées simples qui incluent des champs pour les évaluations des utilisateurs, les évaluations des éléments, le nombre total d'utilisateurs, le nombre d'éléments, les évaluations et l'échelle d'évaluation, qui sont toutes nécessaires pour construire un système de recommandation simple.. [?]

4.2.2 Gradio

Gradio est le moyen le plus rapide de faire la démonstration de votre modèle d'apprentissage automatique avec une interface Web conviviale afin que tout le monde puisse l'utiliser, n'importe où !

FIGURE 4.3 – Logo de l'interface Gradio

Nous nous sommes basés sur cette interface pour visualiser notre sortie en lui entrant un utilisateur ou un item.

4.3 Implémentation des algorithmes

4.3.1 FILTRAGE COLLABORATIF :

Le filtrage collaboratif peut se faire autant en gardant toutes les informations relatives aux usagers similaires en mémoire, ou encore en créant un modèle de préférences à partir de ces informations.
Avec un système où les informations sont gardées en mémoire, les trois mesures de similarité les plus populaires sont la mesure du cosinus (eq.1) (utilisée ici avec deux vecteurs d'achats, x et y) , la corrélation de Pearson (eq.2) (où les vecteur x et y représentent les cotes attribuées aux produits par les utilisateurs X et Y, et x et ÿ représentent les moyennes de leurs cotes) [20] Et la mesure msd (eq.3).
La corrélation de Pearson est utile lorsque le système de recommandations utilise de l'information explicite alors que la mesure du cosinus est utilisée lorsqu'il y a de l'information implicite comme un historique de restaurants déjà visités.

$$\text{cosine\_sim}(u, v) = \frac{\sum_{i \in I_{uv}} r_{ui} \cdot r_{vi}}{\sqrt{\sum_{i \in I_{uv}} r_{ui}^2} \cdot \sqrt{\sum_{i \in I_{uv}} r_{vi}^2}}$$

FIGURE 4.4 – Eq. cosine similarity

$$\text{pearson\_sim}(u, v) = \frac{\sum_{i \in I_{uv}} (r_{ui} - \mu_u) \cdot (r_{vi} - \mu_v)}{\sqrt{\sum_{i \in I_{uv}} (r_{ui} - \mu_u)^2} \cdot \sqrt{\sum_{i \in I_{uv}} (r_{vi} - \mu_v)^2}}$$

FIGURE 4.5 – Eq. pearson

$$\text{msd}(u, v) = \frac{1}{|I_{uv}|} \cdot \sum_{i \in I_{uv}} (r_{ui} - r_{vi})^2$$

FIGURE 4.6 – Eq. msd

Pour récapituler, nous avons mis en œuvre un filtrage collaboratif en suivant les étapes suivantes :

— Les valeurs de similarité entre les restaurants sont mesurées en observant tous les critères communs entre le restaurant cherché et le reste des restaurants

existants dans la base de données.
— Les méthodes du cosine, pearson et msd sont ensuite exploités de manière à comparer les similitudes des restaurants avec celui cherché.
— Tester le modèle en cherchant des restaurants similaires à des entrées désirées.
— Affichage des dix restaurants les plus semblables à notre entrée.

4.3.2 Méthodes du Collaborative Filtering :

4.3.2.1 Memory-based KNN

L'une des méthodes les plus populaires de filtrage collaboratif est l'algorithme des k plus proche voisin (kNN). kNN est un algorithme de classification où k est un paramètre qui spécifie le nombre de voisins utilisés. En utilisant kNN, le système doit se base sur une mesure de similitude pour faire la distinction entre des utilisateurs qui sont près et ceux qui sont éloignés. Selon la problématique étudiée, cette mesure de similitude peut être issue de la distance euclidienne, la mesure du cosinus, la corrélation de Pearson, etc. Ensuite, lorsque le système est interrogé sur une nouvelle information, par exemple une recherche de restaurants qui pourraient plaire à un individu, il va aller trouver les k utilisateurs qui se trouvent le plus près de l'utilisateur cible. Finalement, un vote majoritaire se fait pour décider quel produit sera recommandé. Par exemple, dans le cas d'un système à information explicite, le système pourrait recommander le produit ayant la meilleure cote. Alors que pour l'information implicite, le système pourrait recommander le produit le plus populaire au sein du voisinage.

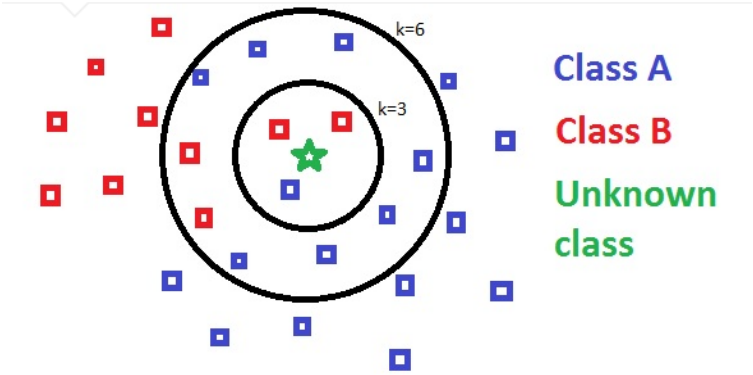


FIGURE 4.7 – KNN collaborative filtering

Pour les FC basées sur l'utilisateur et sur les éléments, nous implémentons K basé sur la mémoire technique du plus proche voisin pour trouver la similarité utilisateur/restaurant. KNN est un algorithme de filtrage collaboratif, considérant les notes moyennes de chaque utilisateur où le le voisin le plus proche ou les objets les plus proches sont plus similaires.

4.3.2.2 User-based collaborative filtering

Pour le filtrage collaboratif basé sur l'utilisateur, nous essayons de créer une matrice où nous regardons à la similarité d'une paire d'utilisateurs. Si l'utilisateur A a attribué un score X au restaurant A, tandis que B également évalué un score similaire sur le même restaurant, ces deux utilisateurs pourraient avoir un score de similarité plus élevé, et nous pouvons recommander le restaurant B à l'utilisateur B où l'utilisateur A classé restaurant 5 étoiles puisque B pourrait aussi être intéressé.

	Restaurant A	Restaurant B	Restaurant C
User A	3	4	?
User B	?	4	5
User C	5	5	?

FIGURE 4.8 – User-based collaborative filtering

4.3.2.3 Item-based collaborative filtering

Le filtrage collaboratif basé sur les éléments tient compte du score de similarité entre les restaurants et recommande des restaurants similaires à l'utilisateur. Par exemple, si l'utilisateur A aime le restaurant A, alors nous pouvons recommander le restaurant B à l'utilisateur puisque le restaurant B a un score de similarité très élevé avec le restaurant A.

Restaurant X	Restaurant A	Restaurant B	Restaurant C
4	3	/	/
2	/	4	5
3	5	5	4

FIGURE 4.9 – Item-based collaborative filtering

4.3.2.4 Model-based SVD

SVD est une technique de réduction matricielle qui diminue la dimension de la matrice, à mapper chaque utilisateur et chaque restaurant respectivement. Ainsi, il nous aide à mieux comprendre la relation entre les utilisateurs et les éléments à mesure qu'ils deviennent directement comparables (Malaheb, 2016). La matrice peut être très clairsemée car tous les utilisateurs ne donnent pas notes à chaque restaurant, SVD peut nous aider à ignorer les cellules vides dans la matrice et seulement gardez ceux qui ont de la valeur. SVD peut aider à trouver le RMSE minimum en optimisant le dimension spatiale des jeux de données. Nous avons

également utilisé la grille de recherche pour trouver le paramètres en testant un certain nombre de paramètres (Scikit Learn 0.22.2, 2019).

$$\min_{U,V,\Sigma} \sum_{ij \in A} \left(A_{ij} - [U\Sigma V^T]_{ij}\right)^2$$

FIGURE 4.10 – SVD collaborative filtering

4.3.2.5 Rating prediction based on sentiment analysis

Dans la deuxième partie, nous essayons de prédire la notation d'un restaurant en fonction de analyse de sentiment sur les avis des utilisateurs pour les restaurants. Analyse des sentiments, sens que nous devons connaître les émotions d'un utilisateur envers un restaurant en regardant les critiques. Par exemple, « bon » et « délicieux » signifient positif, tandis que « affreux » et "mauvais" signifiant négatif. Pour calculer le score de sentiment, nous avons utilisé la **polarité** pour déterminer le score de sentiment.
Polarity of Sentence = Sum of polarity of all the words in a sentence/the total number of words in the sentence

Si le score de sentiment était supérieur à zéro, alors une polarité initiale de 1 a été donnée à cet avis ou conseil particulier, ce qui signifie qu'il s'agit d'un Sentiment positif. Si le score de sentiment était inférieur à zéro, alors une polarité initiale de 0 a été attribué à cet avis particulier, ce qui signifie qu'il s'agit d'un sentiment négatif.

>= 3.5	Positive sentiment, i.e restaurant is amazing
2.0 < r < 3.5	Neutral Sentiment, i.e restaurant is average
<= 2.0	Negative Sentiment, i.e restaurant is not so good

FIGURE 4.11 – SVD collaborative filtering

Pour tester les performances de nos modèles, nous avons utilisé les algorithmes suivants pour entraîner nos modèle : Viz. SVM, Arbre de décision, Régression logistique, Bayes naïf, KNN, Aléatoire Forêt et Perceptron multicouche

4.3.2.6 Collaborative Filtering avec l'analyse de sentiment

Celle-ci est une approche d'inférence de notation pour intégrer le filtrage collaboratif à l'analyse des sentiments à partir des avis des utilisateurs. Les auteurs de cette technique incorporent avec Part-of-Speech (POS) le marquage qui est une technique utile pour extraire des mots plus utiles tels que des adjectifs et les verbes de l'opinion des utilisateurs, Negation Tagging qui peut identifier le sens de mots qui pourraient être affectés par des mots comme "Non", et généralisation des fonctionnalités pour collecter des fonctionnalités existantes et précieuses à partir des avis. Ensuite, ils créent une opinion dictionnaire en attribuant différents poids à l'importance du mot a, cela le dictionnaire reflète la force des mots d'opinion avec la classe de sentiment c (positif/négatif), avec f comme la fréquence des mots se produit dans ce classe sentimentale. Par conséquent, des opinions globales sont plus importantes pour décider de la la force de l'examen, et les fonctionnalités de double pondération et l'étiquetage de négation sont prouvé pour augmenter la précision.

$$OS(a_n, c) = \frac{F(a_n, c)}{\sum_{c_i \in C} F(a_n, c_i)}$$

FIGURE 4.12 – Relation mathématique de la pondération

L'approche utilisée :

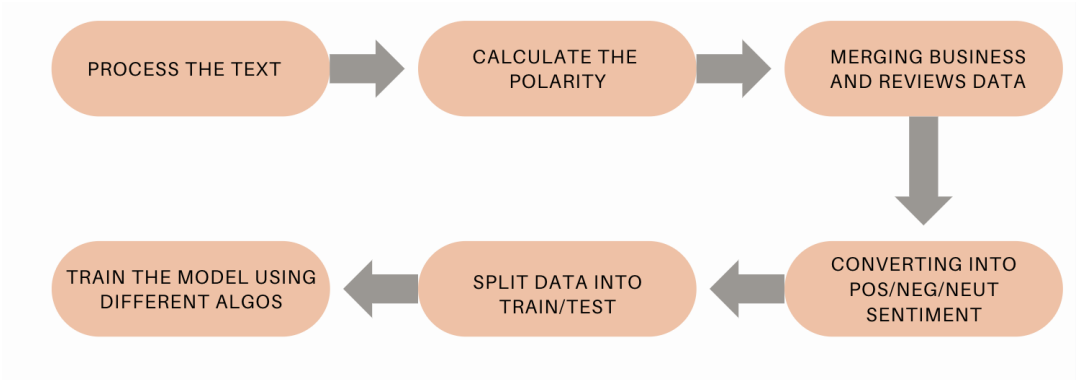


FIGURE 4.13 – collaborative filtering et analyse de sentiments

Chapitre 5

Résultats et évaluations

5.1 RMSE

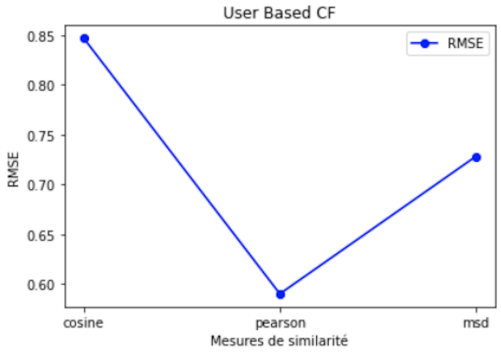
La racine de l'erreur quadratique moyenne «Root Mean Square Error» est une mesure qui permet de calculer la différence entre une cote réelle (r) et une cote prédite (f) pour un ensemble de recommandations n. Cette mesure est utilisée surtout pour les systèmes qui utilisent des cotes comme information explicite. D'autres variantes de cette mesure sont la moyenne d'erreur absolue et la moyenne normalisée d'erreur absolue.

$$RootMeanSquareError = \sqrt{\frac{\sum_{i=1}^{n}(r_i - \widehat{r}_i)^2}{n}}$$

Figure 5.1 – RMSE

Il peut devenir intéressant de regarder seulement l'erreur quadratique moyenne, car cette mesure met beaucoup l'emphase sur l'écart entre la cote prédite et la cote réelle. Ainsi, dans un environnement où on s'intéresse surtout à la différence entre les cotes, cette mesure est essentielle [20]. Toutefois, pour un système où l'écart n'a que très peu d'importance, par exemple un système binaire, d'autres mesures seraient plus adéquates. Dans ce même ordre d'idée, il est intéressant d'utiliser la moyenne normalisée d'erreur absolue lorsqu'il est nécessaire de comparer les résultats obtenus d 'un système avec des systèmes où les cotes ne suivent pas la même échelle de gradation.

5.2 Évaluation de user-based par RMSE

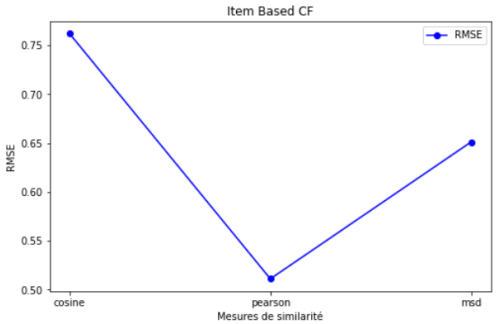



```
Computing the cosine similarity matrix...
Done computing similarity matrix.
Accuracy for  cosine True
RMSE: 0.8468
Computing the pearson similarity matrix...
Done computing similarity matrix.
Accuracy for  pearson True
RMSE: 0.5902
Computing the msd similarity matrix...
Done computing similarity matrix.
Accuracy for  msd True
RMSE: 0.7283
Computing the cosine similarity matrix...
Done computing similarity matrix.
Accuracy for  cosine False
RMSE: 0.7621
Computing the pearson similarity matrix...
Done computing similarity matrix.
Accuracy for  pearson False
RMSE: 0.5111
Computing the msd similarity matrix...
Done computing similarity matrix.
Accuracy for  msd False
RMSE: 0.6512
```

FIGURE 5.2 – résultats similarity user-based

5.3 Évaluation de item-based par RMSE

En ce qui concerne le CF basé sur les articles, nous sommes en mesure de générer une liste de restaurants similaires en fonction du score de similarité avec le restaurant interrogé.




```
Recommendations for Restaurant kkEqZmVvVkgmCaOqE13mDg on priority basis:

51665    "Sushi Bay"
Name: name, dtype: object
44231    "Big Sushi"
Name: name, dtype: object
95156    "Sekai Sushi"
Name: name, dtype: object
41316    "Sushi Bong"
Name: name, dtype: object
159161    "Sushi Kee"
Name: name, dtype: object
148991    "Sushi Kee"
Name: name, dtype: object
33561    "Kokoro Sushi"
Name: name, dtype: object
8682    "Nippon Sushi"
Name: name, dtype: object
40989    "Hana Sushi"
Name: name, dtype: object
69471    "Kanda Sushi Bar"
Name: name, dtype: object
```

FIGURE 5.3 – résultats similarity item-based

En ce qui concerne la comparaison entre le CF basé sur l'user et le CF basé sur les items ou les élements, il y a certains avantages et des inconvénients pour les deux.
Pour le CF basé sur l'utilisateur, premièrement, il suppose que l'utilisateur conservera les mêmes goûts préférés à l'avenir, mais il est probable que l'utilisateur change son goût préféré restaurants en réalité. Deuxièmement, il a une matrice très clairsemée puisque tous les utilisateurs ne l'évalueront pas de nombreux res-

taurants. Cela peut également avoir un coût plus élevé lors de la recherche de K voisins les plus proches lorsque nous avoir un grand nombre d'utilisateurs.
Enfin, il a un problème de démarrage à froid pour les nouveaux utilisateurs car ils ne le font pas avoir une cote au dossier. Pour le CF basé sur les éléments, il peut générer des recommandations de résultats pour utilisateurs, même pour les nouveaux utilisateurs, mais il ne peut pas capturer les intérêts des utilisateurs qui évoluent au fil du temps.

5.4 Comparaison CF item-based utilisant rating vs text

Nous avons également comparé les performances de différentes métriques de similarité sur le CF basé sur l'utilisateur/l'élément. Nous pouvons voir que la similarité du coefficient de corrélation de Pearson montre une relativement meilleure performances avec le taux d'erreur le plus faible, et le CF basé sur les items a un score légèrement meilleur sur évaluation.
Enfin, nous avons comparé les performances sur le CF basé sur les items entre le CF basé sur les items en utilisant le classement et les FC en utilisant des textes. L'utilisation uniquement de la notation montre une précision moindre sur les résultats que d'utiliser des textes. Par exemple, si nous voulons générer des restaurants similaires de Delhi Indian Cuisine, qui est un restaurant indien, nous pouvons voir qu'en utilisant la notation, il y a quelques restaurants d'autres cuisines telles que chinoise, japonaise et mexicaine, tandis que les résultats en utilisant des textes, nous avons surtout d'excellents restaurants indiens en haut de la liste.

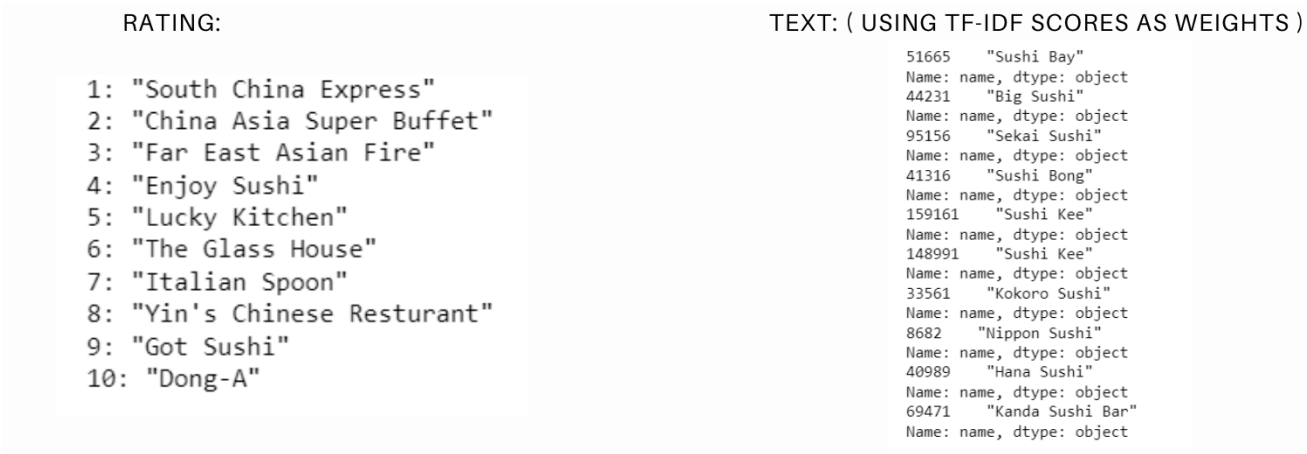


FIGURE 5.4 – Comparing Item-based CF (Ratings vs Text)

5.5 Model-based

Pour la Model-based SVD , nous avons reçu le meilleur score rmse à 0,9790 et les meilleurs paramètres sont 'n_epochs' : 20,'lr_all' : $0,005 et 'reg_all'$: $0,2, on ous avons essayle$s $pour" lr_all" et [0.2, 0.4, 0.6] pour" reg_all"$.

```
0.9790796725004685
{'n_epochs': 20, 'lr_all': 0.005, 'reg_all': 0.2}
```

Figure 5.5 – SVD

5.6 Evaluation du modèle de classification

Les mesures d'évaluation que nous avons utilisées pour la partie 2 sont l'exactitude, la précision, le rappel et le score F1.

TEl que leur relations sont :

$$\text{Accuracy} = \frac{Vrai\ positif + Vrai\ négatif}{Total}$$

$$recall = \frac{Vrai\ Positif}{Vrai\ Positif + Faux\ Négatif}$$

$$precision = \frac{Vrai\ Positif}{Vrai\ Positif + Faux\ Positif}$$

$$F1\ Score = 2 \times \frac{recall \times precision}{recall + precision}$$

5.6.1 L'arbre de décision

```
Evaluations for  Decision Tree Classifier
       PRECISION    RECALL  F1_SCORE  ACCURACY
Train   0.503466  0.680239  0.461295  0.829583
Test    0.501662  0.586146  0.458969  0.832252
```

Figure 5.6

5.6.2 La regression logistique

```
Evaluations for  Logistic Regression
       PRECISION    RECALL  F1_SCORE  ACCURACY
Train   0.333836  0.408589  0.303602  0.829055
Test    0.334798  0.513857  0.306284  0.832651
```

FIGURE 5.7 – La regression logistique

5.6.3 Gaussian Naive Bayes

```
Evaluations for  Gaussian Naive Bayes Classifier
       PRECISION    RECALL  F1_SCORE  ACCURACY
Train   0.337562  0.408620  0.312937  0.827410
Test    0.340295  0.439222  0.318684  0.830984
```

FIGURE 5.8 – Gaussian Naive Bayes

5.6.4 K-Nearest Neighbors

```
Evaluations for  KNN Classifier
       PRECISION    RECALL  F1_SCORE  ACCURACY
Train   0.402698  0.600575  0.425604  0.837021
Test    0.345581  0.369086  0.339555  0.805695
```

FIGURE 5.9 – KNN

5.6.5 Le perceptron multi-couches

```
Evaluations for  Multi-layer Perceptron Classifier
       PRECISION    RECALL  F1_SCORE  ACCURACY
Train        1.0  0.829242  0.906651  0.829242
Test         1.0  0.832760  0.908749  0.832760
```

FIGURE 5.10 – Le perceptron multi-couches

5.7 Accuracy

En regardant la comparaison de tous les modèles, nous pouvons voir La regression Logistique et Le perceptron multi-couches posent les meilleures performances

parmi toutes en précision, tandis que KNN a les performances les plus faibles.

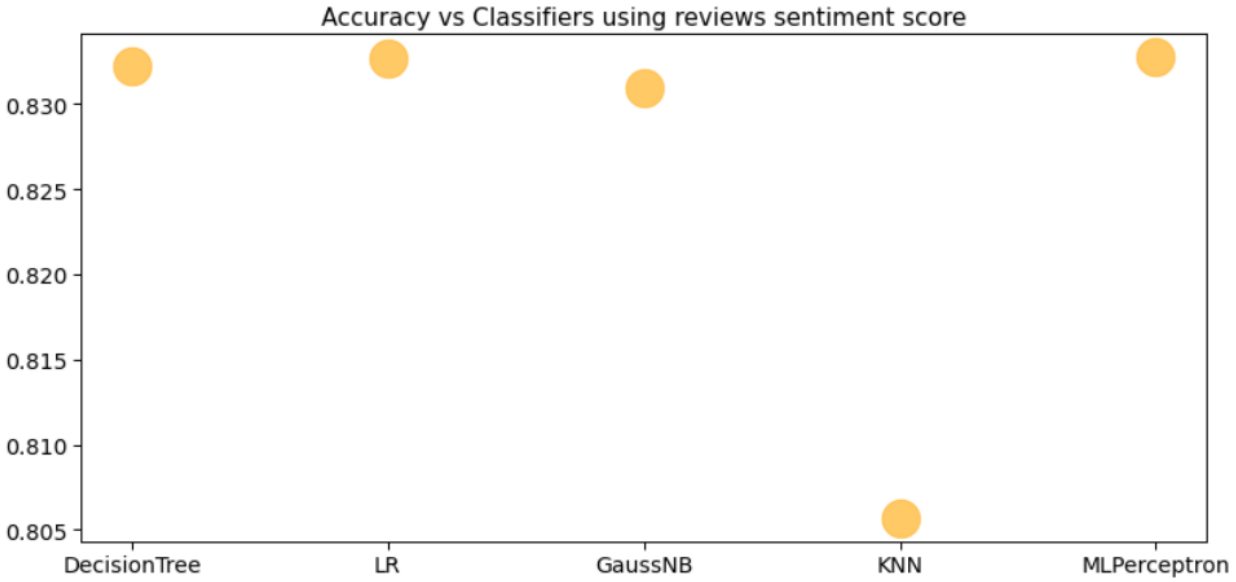


FIGURE 5.11 – comparaison finale des modèles

5.8 Visualisation du résultat sur l'interface Gradio

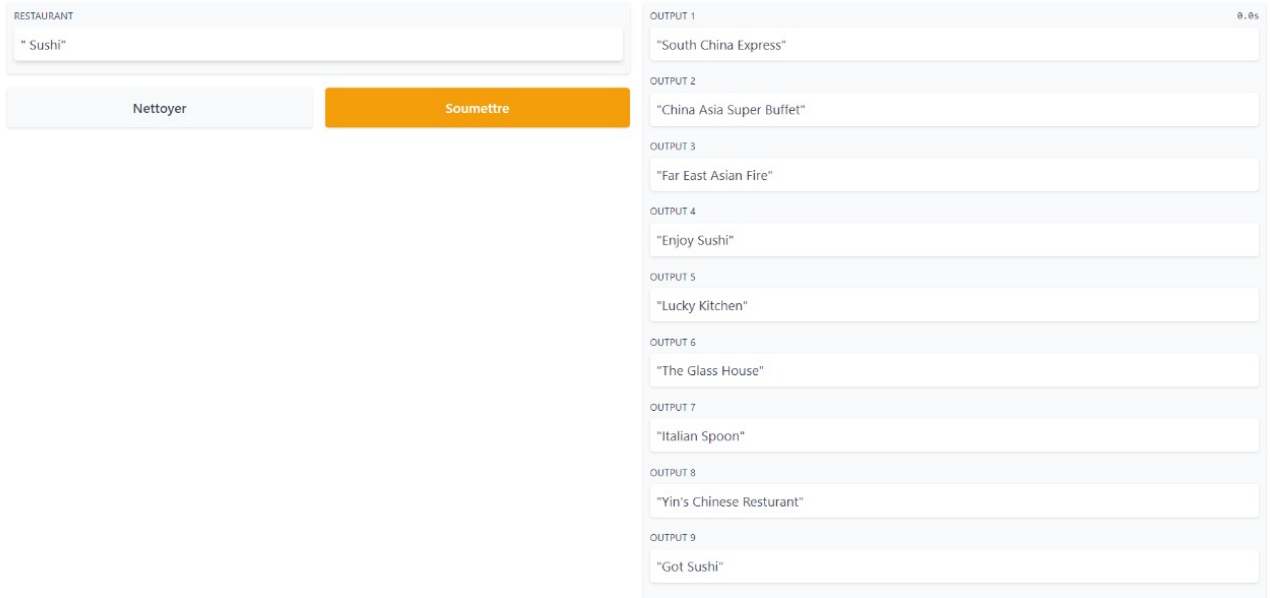


FIGURE 5.12 – Les recommandations sur interface

Chapitre 6

Conclusion

Dans notre projet, nous avons mis en place un filtrage collaboratif pour les systèmes de recommandation et de notation et prédiction des restaurants sur la base des avis des utilisateurs et des données avec différents modèles. Nous avons aussi étudié différentes approches pour combiner le filtrage collaboratif avec l'analyse des sentiments pour améliorer le système de recommandation. Sur la base de notre observation, nous pouvons voir qu'en utilisant différents ensembles de données (notations/avis), prétraitement des données (parsité), modèles de classification (forêt aléatoire/arbre de décision), et les techniques CF (item/user/model/knn) peuvent faire une énorme différence dans les résultats d'évaluation tels que l'exactitude et la précision/rappel.
La limitation majeure qui pourra être soulignée est celle du processing du text ; le NLP étant un domaine si vaste et florissant, il est toujours difficile de prétraiter le texte. DE mieux le texte est prétraité et capturé efficacement, meilleurs sont les résultats, donc il sont d'autres bonnes approches que nous pourrions envisager de mettre en œuvre et d'améliorer dans le avenir. Nous pourrions peut être envisager l'utilisation d'algorithmes d'apprentissage profond pour le NLP et cela pourrait conduire à une meilleure précision.
De plus, nous pourrions utiliser plus de fonctionnalités comme la disponibilité du restaurant, de l'emplacement, des catégories et d'autres fonctionnalités liées aux affaires à améliorer notre modèle de prédiction. Nous pourrions également essayer d'utiliser différentes techniques de prétraitement de texte NLP et des modèles tels que les modèles de radicalisation (stemming) , d'indexation sémantique, d'unigramme, de bigramme et de trigramme... Nous peut également essayer de mettre en œuvre des algorithmes de classification en utilisant des techniques de validation croisée pour trouver un meilleur équilibre entre biais et variance....