


République Algérienne Démocratique et Populaire

Ministère de l'Enseignement Supérieur et de la Recherche Scientifique

Université AMO de Bouira

Faculté des Sciences et des Sciences Appliquées

Département d'Informatique

Mémoire de Master

en Informatique

Spécialité : *GSI*

Thème

Etude de performances d'un système de prédiction et de recommandation et mise en pratique des algorithmes de Machine Learning

---

Encadré par

— Dr.Chouiref Zahira

Réalisé par

— Hamadache Lyza

— Senani Fatima

2021/2022

Remerciements

Tout d'abord nous tenons à remercier le bon dieu qui nous a donné la santé, le courage et la patience pour accomplir et bien mener notre travail.

Nous tenons à remercier du fond du cœur notre promotrice Mme Zahira Chouiref, pour l'aide qu'elle nous a apporté, ses conseils précieux, son encouragements qui nous ont été vraiment bénéfiques et pour la qualité de son encadrement exceptionnel. Merci d'être quelqu'un sur qui nous pouvons toujours compter.

Nos remerciements s'adresse également aux membres du jury qui nous font honneur d'avoir accepter de juger notre travail.

Un grand merci à nos maman et nos papa, pour leurs amours, leurs patiences, leurs encouragements, leurs conseils ainsi que leur soutient moral et financier. Sans nos parents on ne sera jamais devenu ce qu'on est aujourd'hui. On remercie également nos frères, sœurs et toutes la famille.

Nous voudrions exprimer nos reconnaissances à Mr.Hayi Mohamed Yacine pour ses conseils et son aides.

Nos remerciements sont destinés de même à nos professeurs du département informatique, pour leurs conseils, critiques et leurs efforts fournis durant les cinq années d'étude.

Nous tenons à remercié également nos amis et collègues d'être à nos coté tout au long de ce travail.

Dédicaces

Je remercie Allah de m'avoir aidé et donné le courage et la santé d'entamer et de terminer ce travail.

Je dédie ce travail spécialement :

A l'être la plus chers de ma vie, mes chers parents qui ont été toujours présent à mes côtés par leurs prières, que dieu leur procure une longue vie avec une bonne santé.

A mon cher père, pour ses encouragement, soutiens, surtout pour son amour et son sacrifice afin que rien n'entrave le déroulement de mes études.

A ma chère mère qui a été à mes côtés et ma soutenu durant toute ma vie, pour sa patience et son amour.

A mes adorables, mon frère « Houssem » et mes sœurs « Nariman et Chahira », qu'ils trouvent dans ce travail l'expression de mes grands attachements, je vous souhaite une vie plaine de bonheur et de sucées.

A mes ami(e)s, tout particulièrement ma chére amie Lyza, pour sa compréhension, pour commémorer tout ce que nous avons vécu ensemble merci d'avoir cru en moi et d'être à mes cotés.

A tous ceux qui de près ou de loin, ont collaboré a la réalisation de ce travail.

A tous ceux que m'aiment et que j'aime.

Fatima

Dédicaces

Je remercie Allah de m'avoir donnée le courage et la force de mener ce travail à terme et la patience d'aller jusqu'au bout du rêve.

Je dédie ce travail à :

Mon père, tu as toujours été pour moi un exemple du père respectueux et honnête. Je voudrais te remercier pour ton amour, ton soutien et tes sacrifices que dieu te garde pour nous Papa.

Ma très chère maman, je suis fière d'être ta fille. Je voudrais te remercier pour ton amour, encouragement et ton soutien tu m'as tout donnée sans rien demander que dieu te garde pour nous maman.

Ma petite sœur Lina, merci d'être la pour moi, sache que je te promet d'être toujours la pour toi petite sœur.

Mon petit frère Anis ,tu es notre raison de vivre, tu as apporter que du bonheur à notre famille.

Ma grand mère, que dieu te garde pour nous.

Ma très cher amie et sœur Wissem, tu es toujours présente a mes coté, merci d'être une amie si merveilleuse, ton amitié est une véritable chance.

Mon cousin Mahmoud, je te souhaite de tout mon cœur de réussir dans ta vie professionnel.

Hichem, Asma, Nassim, mes ami(es), toutes ma famille et tout les étudiants du M2 informatique.

Lyza

Résumé

Les systèmes de recommandations (SR) sont devenus un outil essentiel face aux quantités massives de données qui ne cessent de croître chaque jour depuis l'avènement d'Internet. La plupart des solutions SR sont basées sur l'analyse des préférences des utilisateurs et leurs notations. Le but de la recommandation est de prédire les évaluations manquantes d'un utilisateur, ou d'une autre façon recommander à un utilisateur des éléments que ses amis apprécient en utilisant la prédiction.

Dans notre travail nous avons appliqué différents algorithmes (KNN, SVD, NMF) sur des Data-Sets différents en développant les approches : prédiction de note (rating), recommandation avec Prédiction, un système de recommandation et proposés une approche Hybride et un système sensible au contexte.

Ces différentes approches ont été validé en utilisant deux techniques de validation : Split-validation et cross-validation et les évaluées en utilisant différentes métriques d'évaluation, à savoir : Test Statistique, mesure d'évaluation, métriques d'erreur, métrique sensible au classement et la Complexité.

Mots clés : Systèmes de recommandation, Systèmes de Prédiction, Contextuel, KNN, SVD, NMF,Test Statique, Mesure d'évaluation, Métriques d'erreur, Métrique sensible au classement, la Complexité.

Abstract

Recommender systems have become an essential tool in the face of the massive amounts of data that have been growing every day since the Internet event. Most SR solutions are based on the analysis of user preferences and their ratings. The purpose of the recommendation is to predict a user's missing ratings, or otherwise recommend items to users that their friends like using these predictions.

In our work we have applied different algorithms (KNN, SVD, NMF) on different Data-Sets by developing approaches : rating prediction, recommendation with Prediction, a

recommendation system and proposed a Hybrid approach and a sensitive system to the context.

These different approaches have been validated using two validation techniques : Split-validation and cross-validation and evaluated using different evaluation metrics, namely : Test Statistic, Evaluation measure, Error metrics, Rank sensitive metric and Complexity.

Key words : Recommender Systems, Prediction Systems, Contextual, KNN, SVD, NMF, Test Statistic, Evaluation Measure, Error Metrics, Rank Sensitive Metric, Complexity.

ملخص

أصبحت أنظمة التوصية (SR) أداة أساسية في مواجهة الكميات الهائلة من البيانات التي تتزايد كل يوم منذ ظهور الإنترنت. تعتمد معظم حلول SR على تحليل تفضيلات المستخدم وتقييماته. الغرض من التوصية هو التنبؤ بالتقييمات المفقودة للمستخدم ، أو التوصية بعناصر لمستخدم يحبها أصدقاؤه باستخدام التنبؤ.

في عملنا ، قمنا بتطبيق خوارزميات مختلفة (KNN، SVD، NMF) على مجموعات بيانات مختلفة من خلال تطوير مناهج: التنبؤ بالتصنيف ، والتوصية بالتنبؤ ، ونظام التوصية ، واقترح نهجًا هجينًا ونظامًا حساسًا للسياق.

تم التحقق من صحة هذه الأساليب المختلفة باستخدام طريقتين: التحقق من صحة الانقسام والتحقق المتبادل وتم تقييمها باستخدام مقاييس تقييم مختلفة ، وهي: إحصاء الاختبار ، ومقياس التقييم ، ومقاييس الخطأ ، ومقياس حساس للترتيب ، والتعقيد.

{الكلمات الرئيسية}: أنظمة التوصية ، أنظمة التنبؤ ، المحتوى ، KNN، SVD ، NMF ، الاختبار الثابت ، قياس التقييم ، مقاييس الخطأ ، مقياس حساس للترتيب ، التعقيد.

Table des matières

Table des figures

Liste des tableaux

Liste des abréviations

IA	Intelligence Artificielle
ML	Machine Learning
KNN	K-Nearest Neighbor
SVD	Singular Value Decomposition
NMF	Non negative mMatrix Factorization
CF	Filtrage Collaboratifs
SE	Système d'Exploitation
DS	Data-Set
SR	Système de Recommandation
RMSE	Root Mean Squared Error
MSE	Mean Squared Error
MAE	Mean Absolute Error
Anova	Analysis Of Variance

Introduction générale

Avec l'avancée technologique spectaculaire de l'intelligence Artificielle (IA) dans plusieurs domaines, l'exploitation ou l'analyse de données volumineuses est devenue une tâche essentielle. Plusieurs techniques de l'IA ont été développées pour faciliter la recherche et l'extraction d'informations pertinentes.

Le domaine de la cinématographie est particulièrement touché par le problème de la surcharge d'informations, car il contient une quantité très importante et étendue d'information, comme des genres infinis de films et de séries. Profiter de ces larges sélections devient très compliqué pour les utilisateurs qui doivent passer beaucoup de temps à trouver des films qui leur conviennent et reflètent leurs préférences. L'un des principaux domaines de recherche liés au problème de la surcharge d'informations aujourd'hui est le domaine des systèmes de prédiction et de recommandation.

La logique des systèmes de prédiction ou autrement dit un système de prédiction repose sur l'intuition d'outils d'analyse de données passées ou présentes. Ces outils nous permettent de nous projeter dans l'avenir et de prendre les meilleures décisions pour assurer sa pérennité.

Les systèmes de recommandation sont capables de fournir des recommandations basées sur les préférences et les besoins des utilisateurs. Ils se sont avérés fournir des résultats très satisfaisants et fournir un excellent soutien aux utilisateurs. Les systèmes de recommandation sont rapidement devenus très populaires dans divers domaines en raison de leurs hautes performances.

L'aspect clé du SR est l'utilisation des algorithmes de Machine Learning (apprentissage automatique), ces derniers jouent un rôle de plus en plus important car ils déterminent l'information qui serait essentielle et pertinente.

Problématiques

— Comment prédire la note d'un item donné par un utilisateur en fonction de son utilisateur similaire ?

— Pouvons-nous recommander un item en fonction des préférences et des notes précédentes des utilisateurs ?

— Pouvons-nous recommander des items évalués par un utilisateur aux autres utilisateurs similaires ?

— Comment recommander des items en fonction de sa description ?

Objectifs

Nos objectifs sont :

— Appliquer les algorithmes KNN, SVD et NMF.

— Développer un système de prédiction, recommandation et système Sensible au contexte.

— Évaluation des systèmes en utilisant différentes métriques d'évaluation (Test Statistique, Mesure d'évaluation, Métriques d'erreur, Métriques sensible aux classement et la Complexité).

Contributions

— Implémentation de trois algorithmes KNN, SVD et NMF sur deux larges Data-Sets du domaine de la cinématographie.

— Développer une approche Hybride en combinant deux filtrage (filtrage collaboratif (en utilisant le SVD)+ filtrage basé sur le contenu(en utilisant la technique TF-IDF).

— Intégrer le contexte dans l'approche hybride pour créer un système sensible au contexte.

— Mesurer la performance du système en utilisant Anova, Precision, Recall, F1-Score, Accuracy, RMSE, MAE, MSE, NDCG et la complexité afin de voir lequel de nos algorithmes est plus performant et adapté au domaine utilisé.

Organisation du manuscrit

Notre travail est réparti sur quatre chapitres : Le premier chapitre aborde les concepts de base de l'apprentissage automatique ainsi que l'analyse prédictive et les différentes approches des systèmes de recommandation .

Le deuxième chapitre présente l'étude des algorithmes implémentés ainsi que leur mode de fonctionnement.

Le troisième chapitre décrit l'approche proposée qui consiste à nos deux systèmes à savoir : système de prédiction et recommandation.

Le dernier chapitre sera entièrement consacré pour l'évaluation des algorithmes implémentés en utilisant différentes métriques d'évaluation afin de voir les performances fournit par les différentes études.

Chapitre 1

L'analyse prédictive et les systèmes de recommandation

1.1 Introduction

Dans ce chapitre, nous allons définir l'analyse prédictive ainsi que systèmes de recommandation avec ses différents types essentiels existant tels que le filtrage collaboratifs et sensible au contexte et enfin nous avons l'exemple de Netflix et son utilisation des systèmes de recommandation.

1.2 Analyse prédictive

L'analyse prédictive, également connue sous le nom de logique prédictive, est une méthode analytique et statistique qui s'appuie sur l'analyse de données actuelles et historiques pour créer des hypothèses et des prédictions sur des événements futurs[8]. L'analyse prédictive vise à identifier des modèles dans les données pour comprendre la probabilité d'un élément.

1.2.1 De la prédiction à la production

Ce système fonctionne en enregistrant l'historique du profil, en collectant passivement des données, et les caractéristiques des informations saisies (recherches, notes, etc.) et des vidéos visionnées, qui sont taguées par des mots-clés (genres, acteurs, récompenses, éléments spécifiques, etc.) Grâce notamment à la recherche de toutes les données récoltées,

comme Netflix par exemple a pu identifier ses futures émissions à succès. La société a déjà commencé à financer et à produire des séries originales, alliant notamment succès et très bons retours de la communauté[9].

1.2.2 Prédiction étendue à l'aide de l'apprentissage automatique

Le monde de l'intelligence artificielle a commencé à produire ses propres célébrités qui défendent une excellente idée et offrent de vraies prédictions sur l'avenir de l'IA, certaines nous promettant un avenir utopique qui croît de façon exponentielle. On ne peut pas parler d'intelligence artificielle sans évoquer les modèles prédictifs, qui permettent d'utiliser les données pour faire de meilleures prédictions et améliorer l'efficacité opérationnelle. Grâce à l'intelligence c, nous assistons actuellement à l'intégration accélérée du machine learning (ML)dans notre quotidien[10].

La prédiction est largement utilisée comme solution pour faire des prédictions sur des données indisponibles ou futures afin de faire des prédictions statistiques sur l'occurrence d'événements dans le domaine de l'apprentissage automatique[11]. Avec cette révolution Peut-on avoir confiance aux prédictions d'une IA ?

1.2.3 Méthodes de l'analyse prédictive

Il existe devers domaine dans lequel l'analyse prédictive est utilisé comme :

1. Prédiction de note (rating) par exemple dans le domaine de la cinématographie[12].
2. Prédire le comportement des utilisateur par exemple dans le domaine du marketing[12].
3. Prédire une maladie comme par exemple dans le domaine de la santé[12].
4. Prédire le phénomène de fraude comme par exemple dans le domaine des finance[12].

1.2.4 Améliorer la performance obtenue par l'apprentissage automatique

Les algorithmes d'apprentissage automatique créent des modèles prédictifs à partir de données historiques et les utilisent pour prédire de nouvelles données[13]. La question la plus fréquemment posée lors du développement d'un modèle est de savoir comment obtenir de meilleures prédictions.

1.3 Le système de recommandation : un cerveau à la place du cœur

un système de recommandation est un système de filtrage de données qui classe les objets en fonction de leur pertinence pour les utilisateurs. Pour ce faire, il construit un modèle afin de répondre et satisfaire les besoins des utilisateurs . Les systèmes de recommandation ont envahi de nombreux domaines : ils sont désormais utilisés par les plateformes de divertissement, les réseaux sociaux, Netflix, mais ils apparaissent également dans des domaines plus spécialisés[14].

1.3.1 Fonctionnement du système de recommandation

Il est composé de trois étapes essentiels :

Collecte de données

Cela collecte des informations pertinentes sur l'utilisateur telles que les avis, les évaluations, l'historique des commandes, les actions du panier, l'historique des recherches, etc[15].

Apprentissage

Appliquez des algorithmes d'apprentissage pour filtrer et exploiter les caractéristiques des utilisateurs à partir des données collectées[15].

Phase de recommandation

recommande ou prédit la préférence de l'utilisateur[15]. La figure 1.1 illustre le fonctionnement du système de recommandation :

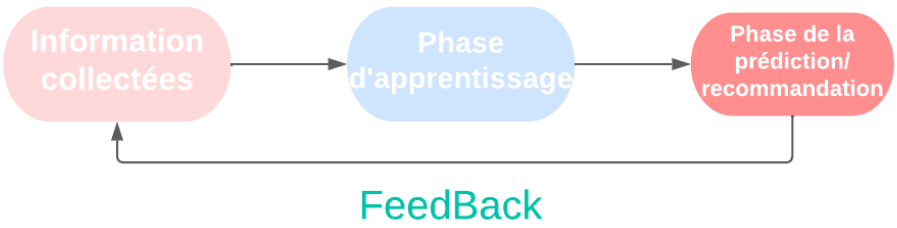

FIGURE 1.1 – Fonctionnement du système de recommandation

1.3.2 Classification des systèmes de recommandation

On pourra distinguer plusieurs systèmes de recommandations :

Approche basée sur le contenu

Le filtrage basé sur le contenu analyse les propriétés des éléments pour générer des prédictions. Avec ce type de filtrage, la décision de sélection est entièrement basée sur le contenu. Son fonctionnement est caractérisé par le contenu informationnel à filtrer, et il se compose de trois techniques :[16]

1. L'analyseur de contenu : Ce module est conçu pour effectuer un pré-traitement afin d'extraire les informations pertinentes, la description sera utilisée comme entrée pour d'autres modules.[16]

2. Apprentissage du profil : Pour créer des profils d'utilisateurs, ce module collecte des données représentant les préférences des utilisateurs[16].

3. Composant de filtrage : Ce module peut filtrer les recommandations qui seront présentées à l'utilisateur en faisant correspondre les représentations du profil de l'utilisateur avec les éléments candidats[16].

Cette figure 1.2 explique l'approche basée sur le contenu :

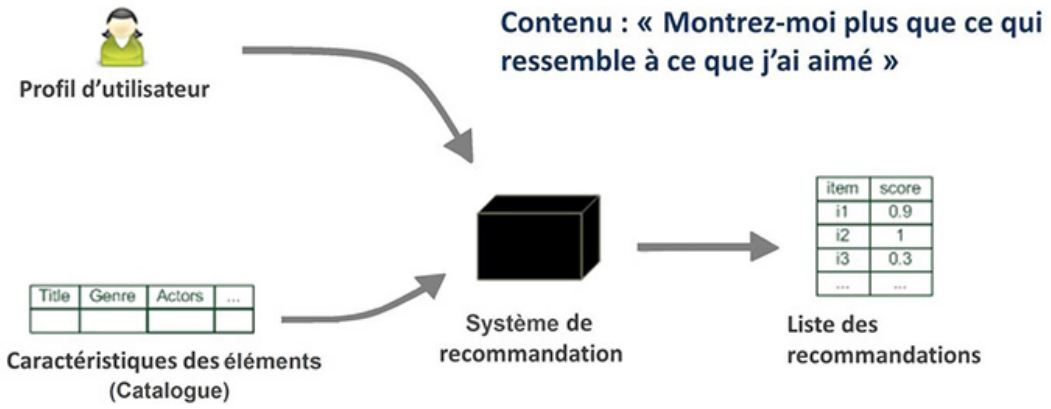

FIGURE 1.2 – Approche basé contenu [1]

Approche basée sur le filtrage collaboratif

L'algorithme de filtrage collaboratif est basé sur l'idée que si deux clients avec des historiques de notation similaires, ils se comporteront de la même manière à l'avenir. Par exemple, s'il y a deux utilisateurs très probables, et que l'un d'eux a regardé un film et lui a donné un bon score, cela suggérerait que le deuxième utilisateur se comporterait de la même manière. Il s'agit d'une approche utile car elle ne repose pas sur des informations supplémentaires sur l'élément (par exemple, acteur, réalisateur, genre) ou sur l'utilisateur (par exemple, données démographiques) pour générer des recommandations. Les recommandations produites par cette approche peuvent être des recommandations ou des prédictions spécifiques, et le FC peut être classée en deux approches différentes : les techniques basées sur la mémoire et les techniques basées sur un modèle[17].
La figure 1.3 explique l'approche basée sur le filtrage collaboratif :

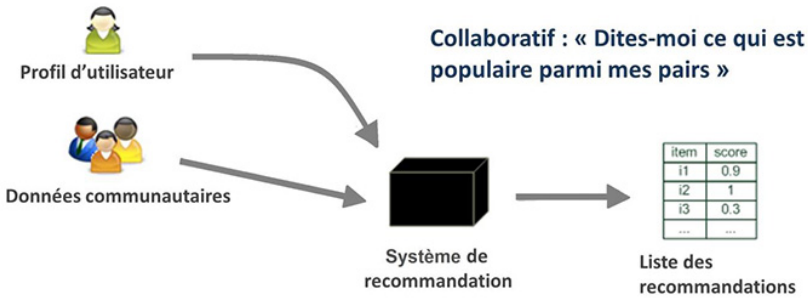

FIGURE 1.3 – Approche basée sur le filtrage collaboratif [1]

— Techniques basées sur la mémoire : les méthodes de filtrage collaboratif basées sur la mémoire peuvent être divisées en deux parties principales : le filtrage collaboratif basé sur l'utilisateur et le filtrage collaboratif basé sur les éléments. Lorsque les recherches d'utilisateurs trouvent des utilisateurs similaires, en fonction d'évaluations similaires, et recommandent des produits que ces utilisateurs aiment. Les filtrages basés sur les éléments, d'autre part, recherchent les utilisateurs qui ont aimé cet élément, puis recherchent différents éléments que ces utilisateurs ont aimés, et c'est ainsi que ces éléments sont utilisés pour les recommandations[17].

1. Filtrage basé sur les utilisateurs (user-user) : L'objectif principal du filtrage collaboratif basé sur l'utilisateur est d'identifier les utilisateurs ayant des valeurs de notation similaires et de leur fournir de nouveaux éléments avec les

notes les plus élevées en fonction de leurs préférences[17]. Autrement dit, cela fonctionne en définissant d'abord les utilisateurs qui sont similaires à l'utilisateur actuel, puis en calculant une valeur prédite pour chaque élément candidat à la recommandation.

2. Filtrage basé sur les items (items-items) : Ce type de filtrage est le même que (utilisateur-utilisateur), la seule différence entre les deux est que (éléments-éléments) utilise la note donnée par cet utilisateur pour prédire la note de l'utilisateur pour l'élément. Fondamentalement, l'algorithme commence à trouver des utilisateurs similaires en fonction des vues et des préférences. En d'autres termes, il essaie de déterminer à quel point un film est similaire à un autre[17].

La figure 1.4 est une explication de la technique basée sur la mémoire :

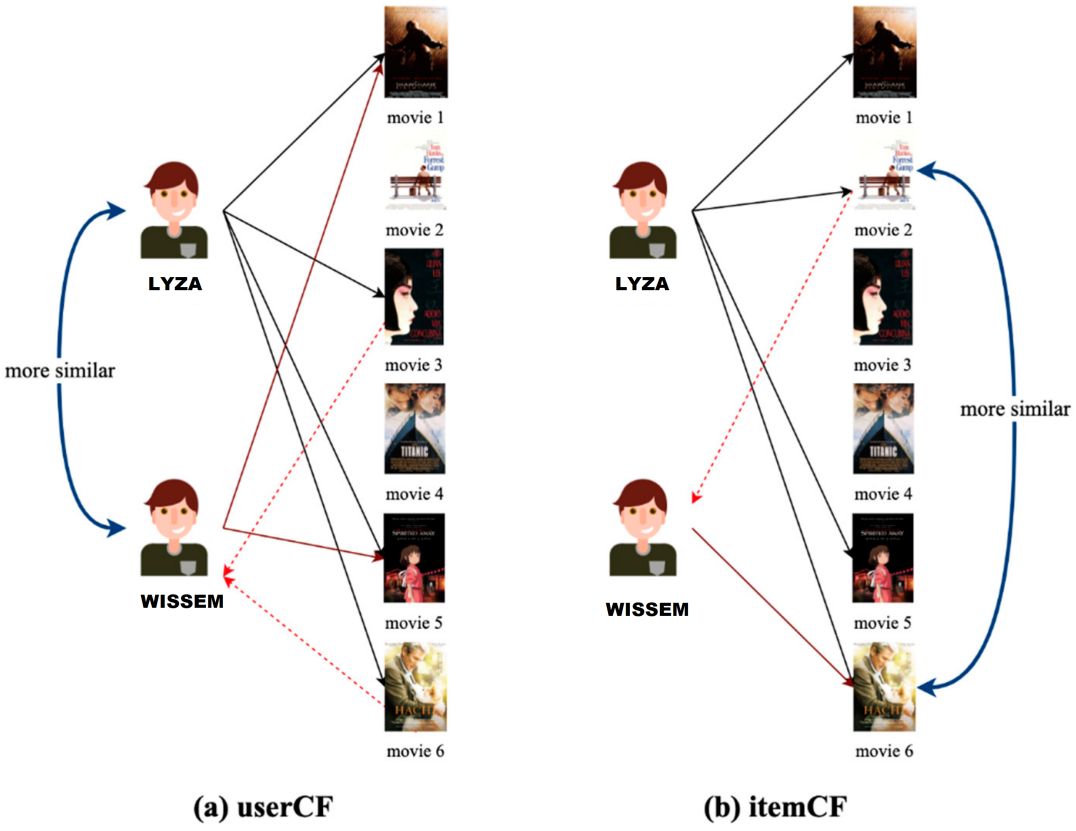

FIGURE 1.4 – Technique basée sur la mémoire

— Technique basée sur le modèle : Ce filtrage apprend un modèle descriptif reliant les utilisateurs, les documents et les votes. D'un point de vue probabiliste, le processus

de filtrage prédit la valeur d'un vote donné en fonction du profil de l'utilisateur ou de ses votes précédents[18]. Ces types de modèles sont utiles pour faire des recommandations et afficher plus rapidement des résultats similaires aux modèles basés sur la mémoire. Les techniques basées sur des modèles sont basées sur la factorisation matricielle (MF), qui est très populaire car il s'agit d'une méthode d'apprentissage non supervisée pour la réduction de la dimensionnalité[17].

Approche de filtrage hybride

Un système de recommandation hybride est une combinaison entre deux ou plusieurs approches de type différent ou du même type. L'objectif de cette hybridation est d'exploiter les avantages et de surmonter certains inconvénients de chaque approche. Parmi les inconvénients existants on note : le démarrage à froid, le problème de l'éparpillement et la scalability[19].

Approche sensible aux contexte

Les systèmes de recommandation sont devenus un domaine de recherche à part entière, les chercheurs ont commencé à se concentrer sur les problèmes de recommandation, en s'appuyant sur le concept de "notes" pour exprimer les préférences des utilisateurs. Étant donné que les gens disposent d'appareils mobiles à tout moment et en tout lieu, l'utilisation des capacités de ces appareils intelligents présente une opportunité importante d'améliorer la qualité des produits recommandés aux utilisateurs. Récemment, de nouveau SR sont apparu qui sont les systèmes de recommandation contextuels. Ces derniers, prennent également en compte les informations contextuelles (par exemple, l'heure, le lieu, le compagnon social et l'humeur) associés aux préférences collectées. Dans de cette façon, les systèmes de recommandation contextuels peut discriminer l'intérêt qu'un utilisateur peut avoir pour un élément particulier dans le contextes et situations différents[20].

- Le contexte

Le contexte est un concept complexe. Il a été évoqué dans plusieurs sources souvent d'une manière complètement indépendante. Parmi la consultation de plusieurs références sur le terme "contexte" (Dey 2001) a tenté d'englober la définition des entités relatives au contextes « toute information qui peut être utilisée pour caractériser la situation d'une entité. Toute entité est une personne, ou un objet qui est considéré significatif à l'interac-

tion entre l'utilisateur et l'application, incluant l'utilisateur et l'application lui-même≫[21].

D'après la définition de dey le contexte influence l'interaction entre l'utilisateur et l'application afin d'affecter les préférences, souhait et l'intérêt de l'utilisateur et ses décisions. Une autre analyse de Brezilloon (Mostefaoui, Pasquier-Rocha et al. 2004) faite sur les définitions du terme contexte mène à conclure que la plupart des définitions sont des réponses aux questions de notre étude : "Qui ?" Avec qui je suis, "Quoi ?" quel est le genre de film a regarder, "Où ?" l'endroit pour voir un film, "Quand ?" Le temps pour voir un film, "pourquoi ?" L'intérêt de regarder un film et "comment ?" L'appareil utilisé pour regardé un film[21].

- La sensibilité au contexte

Le concept de sensibilité définit la capacité du système à s'adapter aux changements de l'environnement.

En se basant sur l'approche orientée service (Miraoui 2009) ont défini un système sensible au contexte comme suit : ≪Un système est dit sensible au contexte s'il peut changer automatiquement les formes des services ou déclencher un service comme réponse au changement de la valeur d'une information ou d'un ensemble d'informations qui caractérisent le service.≫ Cette définition fournit un système sensible au contexte car elle explique la sensibilité des interactions du système aux transitions contextuelles en déclenchant des changements de service ou de forme de service (Ameyed, Miraoui et al. 2015)[21].

Bien qu'il y ait beaucoup de travail dans le domaine de la sensibilisation au contexte, le domaine est loin d'être développé. En effet, il reste encore plusieurs sujets à approfondir et à étudier[21].

- La Recommandation bidimensionnels Vs Contextuel

Alors que les systèmes de recommandation traditionnels reposent uniquement sur les utilisateurs et les éléments de recommandation, les systèmes de recommandation contextuels prennent en compte le contexte de l'utilisateur, ce qui affecte les intérêts/besoins de l'utilisateur[21].

A. **Un système de recommandation bidimensionnels 2D :**

   Le processus de recommandation commence généralement par la spécification d'un ensemble initial d'évaluations, qui sont explicitement fournies par l'utilisateur. Une fois ces notes initiales spécifiées, le système de recommandation tente d'estimer une

fonction de la note R : [20]

$$R : Utilisateur * Film = Note(Rating) \tag{1.1}$$

Pour les paires (utilisateur, film) qui n'ont pas été évaluées par un utilisateur. Une fois que la fonction R a été estimée pour l'ensemble de l'espace utilisateur x, le système de recommandation peut recommander le film le mieux noté pour chaque utilisateur. Nous appelons ces systèmes des systèmes traditionnels ou bidimensionnels (2D) car ils ne prennent en compte que les dimensions de l'utilisateur et du film dans le processus de recommandation[20].

B. **Un système de recommandation Contextuel**

La plupart des recherches effectuées se sont concentrées sur la recommandation de films aux utilisateurs sans tenir en compte les informations contextuelles telles que l'appareil de regarder ou la compagnie d'autres personnes pour regarder un film[20]. Par la suite les chercheurs ont explorés le domaine des systèmes de recommandation contextuels en incorporant les informations contextuelles disponibles au cours du processus de recommandation dans des catégories explicites ou implicites, grâce à la modélisation et à la prédiction des goûts et préférences des utilisateurs. Ces préférences et goûts à long terme sont souvent exprimés sous la forme d'évaluations et sont basés non seulement sur les item et les utilisateurs, mais aussi sur le contexte[20].

$$R : Utilisateur * Film * Contexte = Note(Rating) \tag{1.2}$$

Contrairement aux modèles traditionnels, (en plus des informations sur l'utilisateur et l'élément) pour estimer les préférences de l'utilisateur pour les éléments non vus[20].

Pour conclure, les systèmes de recommandation contextuels tentent d'incorporer ou d'utiliser une information supplémentaires qui est le contexte (temps,humeur,compagnie...etc) contrairement aux modèles traditionnels.

- Les Limitation des systèmes Contextuels

- L'utilisateur est à l'origine du facteur contextuel en renseignant manuellement ces données, chaque fois il doit remplir le contexte dans lequel il se trouve cela peut être relativement complexe et fastidieux[22].
- Les études ont montré que la collecte d'informations personnelles est une pratique de plus en plus critiquée par les utilisateurs, et la collecte d'informations contextuelles repousse encore les limites de la vie privée des utilisateurs[22].
- Bien que l'ajout d'une dimension de contexte au modèle de préférence de l'utilisateur augmente la quantité d'informations disponibles, il augmente également la quantité d'informations manquantes[22].
- Outre l'aspect de la confidentialité, il existe de nombreuses failles dans l'étape de récupération des informations (difficulté de récolte d'information) et sans facteurs contextuels le système ne serait pas en mesure de fournir des suggestions contextuelles[22].

1.3.3 Moteur de recommandation : la transparence optimale

Un moteur de recommandation est un algorithme qui produit des recommandations personnalisées ou, lorsque le choix peut être multiple, guide les utilisateurs de manière personnalisée vers des éléments d'intérêt ou d'utilité[23]. La prémisse principale pour développer l'utilisation des algorithmes est que les utilisateurs et les régulateurs qui leur font confiance y seront placés. Jusqu'à présent, les algorithmes étaient considérés comme des "boîtes noires". Ces boîtes noires doivent être ouvertes et des cadres réglementaires se construisent en ce sens afin de créer les conditions d'une plus grande transparence[24].

La transparence s'organise autour de 2 piliers : l'interprétabilité et l'explicabilité.

L'interprétabilité : Vise à comprendre comment les algorithmes prennent des décisions. L'objectif est de rendre interprétable la logique de décision pour le développement et la mobilisation d'algorithmes[24].

L'explicabilité : Vise à comprendre pourquoi les algorithmes prennent des décisions. Son ciblage est plus granulaire que l'interprétabilité, car il peut fournir des informations ciblées sur les variables essentielles à une décision particulière[24].

1.3.4 Les systèmes de recommandations dans le domaine de la cinématographie

Il existe plusieurs plateformes de recommandation dans le domaine de la cinématographie comme Netflix.
On été surpris de voir à quel point système de recommandation fonctionne dans l'infrastructure de Netflix. Netflix possède l'un des systèmes de recommandation les plus sophistiqués au monde, et son système de recommandation prend en compte non seulement les informations sur les utilisateurs, mais également les différents éléments qu'ils consomment chaque jour. Plusieurs algorithmes peuvent être utilisés pour modéliser les systèmes de recommandation. Un dernier mot : parmi les algorithmes qui ont faits partie de la recette du succès de Netflix[25] :Singular Value Decomposition (SVD), Restricted Boltzmann Machines (RBM)

1.4 Conclusion

Ce chapitre a été consacré à la présentation des notions de base comme l'analyse prédictive qui est utilisée pour prédire les futurs résultats d'une activité ou d'un comportement, ensuite nous avons abordé la notion des systèmes de recommandation qui est devenu indispensable ces dernières années car ils sont conçu pour aider les utilisateurs à trouver des ressources qui les intéressent.

Chapitre 2

Les algorithmes au cœur du raisonnement

2.1 Introduction

Pour réaliser un modèle dans un projet machine Learning, nous avons besoin d'utiliser plusieurs techniques et méthodes. Les algorithmes font partie des notations les plus utilisées aujourd'hui dans le domaine de l'apprentissage automatique car ils permettent de résoudre des problèmes complexes dans la vie réelle. Dans ce chapitre, on allons aborder les concepts de base du Machine Learning en se basant sur ces types d'apprentissage ensuite nous nous focalisons sur ce qui se cache derrière cette fameuse boite noire et enfin nous présentons comment mesurer et augmenter les performances des modèles grâce aux métriques d'évaluation.

2.2 Machine Learning

Bien que l'apprentissage automatique ne soit pas nouveau, sa définition précise en déroute encore beaucoup. Plus précisément, il s'agit d'une science moderne consistant à découvrir des modèles et à faire des prédictions à partir de données basées sur des statistiques, l'exploration de données, la reconnaissance de modèles et l'analyse prédictive[26].D'une façon plus simple le Machine Learning consiste à écrire un programme qui ne sait rien faire au début, mais qui va apprendre à faire quelque chose avec le temps et l'expérience. Un peu comme un être humain qui apprend à faire du vélo : au début il y arrive pas du tout, mais à force d'en faire et bien il y arrive de mieux en mieux, jusqu'au moment où il le fait super bien[2]. La figure ci dessous 2.1 est un exemple sur le Machine Learning :

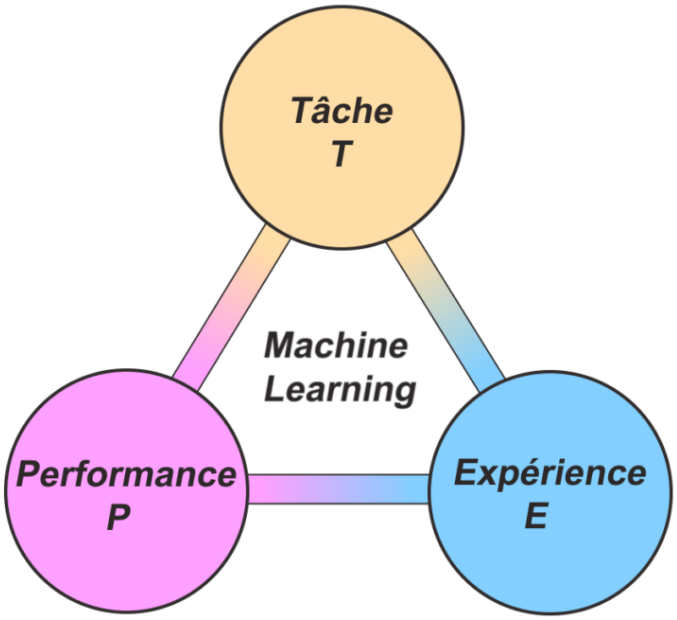

FIGURE 2.1 – Apprendre à faire T = Améliorer P grâce à E [2]

2.2.1 Le grand défi de l'apprentissage machine

Pour renforcer l'apprentissage automatique, les modèles actuels sont entraînés avec des données de base, puis tous les modèles sont testés avec de nouvelles données. La recherche scientifique a produit plusieurs techniques qui permettent aux modèles de mieux se généraliser à de nouvelles données pour des prédictions précises. Le but est donc de permettre aux machines de prédire l'avenir avec des prédictions plus précises : c'est un défi ![27]

Parmi les divers défis qui existe :

Apprendre des données massives

À mesure que la technologie progresse, la quantité de données que nous traitons augmente chaque jour. L'attribut principal des données est la quantité. Par conséquent, le traitement d'une telle quantité d'informations est un énorme défi[28].

Apprentissage de différents types de données

Il existe aujourd'hui une grande variété de données. La variété est également un attribut majeur des méga-données. Structurées, non structurées et semi-structurées sont trois types de données différents qui entraînent en outre la génération de données hétérogènes, non linéaires et de grande dimension. Apprendre à partir d'un si grand ensemble de données est un défi et entraîne en outre une augmentation de la complexité des données[28].

Apprentissage de données streamées à grande vitesse

Il existe diverses tâches, y compris l'achèvement des travaux dans un délai précis. La vitesse est également un attribut important du Big Data. Si la tâche n'est pas terminée dans le délai spécifié, l'effet du traitement peut être pire précieux, voire sans valeur. Par conséquent, il est très nécessaire et difficile de traiter les méga-données en temps opportun. Pour surmonter ce défi[28].

2.2.2 Les tâches de l'apprentissage artificiel

Des techniques d'apprentissage automatique sont nécessaires pour améliorer l'exactitude des modèles prédictifs. Selon la nature du problème traité, il existe différentes approches qui varient selon le type et le volume des données. Dans cette section, nous discutons des catégories de l'apprentissage automatique :[29]

— Apprentissage supervisé : Pour cet apprentissage, nous disposons de données d'entrée (caractéristiques) et de résultats attendus (étiquettes). Il nous permet de faire des prédictions basées sur un modèle en se basant sur des données historiques pour l'algorithme choisi. Parmi les algorithmes existants, on peut distinguer les algorithmes de classification (prédiction non numérique) et les algorithmes de régression (prédiction numérique). Selon le problème à résoudre, on utilise l'un de ces deux modèles[30].

— Apprentissage Non-supervisé : Ce type d'apprentissage automatique est souvent utilisé pour découvrir des structures et des modèles dans les données. À partir des données historiques dont nous disposons, nous essayons de voir ce que nous pouvons

apprendre des données, sans oublier de vérifier les conclusions tirées avec des experts du domaine. Avec cet apprentissage, nous avons toujours des fonctionnalités, mais pas d'étiquettes car nous n'essayons pas de prédire quoi que ce soit[30].

— Apprentissage Sumi-supervisé : Il existe d'autres classifications basées sur le type de méthode d'apprentissage, comme par exemple l'effet "apprentissage semi-supervisé"[31]. L'apprentissage semi-supervisé décrit un flux de travail spécifique dans lequel des étiquettes sont automatiquement générées à l'aide d'un algorithme d'apprentissage non supervisé, et ces étiquettes peuvent être introduites dans un algorithme d'apprentissage supervisé[32].

— Apprentissage par renforcement : L'apprentissage par renforcement est un type d'apprentissage automatique dont le but est d'apprendre par étapes successives comment obtenir la meilleure solution. Dans un tel problème, on dit que "une personne" interagit avec "l'environnement" pour arriver à une solution optimale. L'apprentissage par renforcement est fondamentalement différent des autres apprentissages en termes d'itération et d'interaction : la personne teste plusieurs solutions (explorations) pour trouver la meilleure politique (à partir des résultats de ses explorations) en observant les réponses de l'environnement et son comportement (variables)[33].

La figure 2.2 montre différents types d'apprentissage automatique :

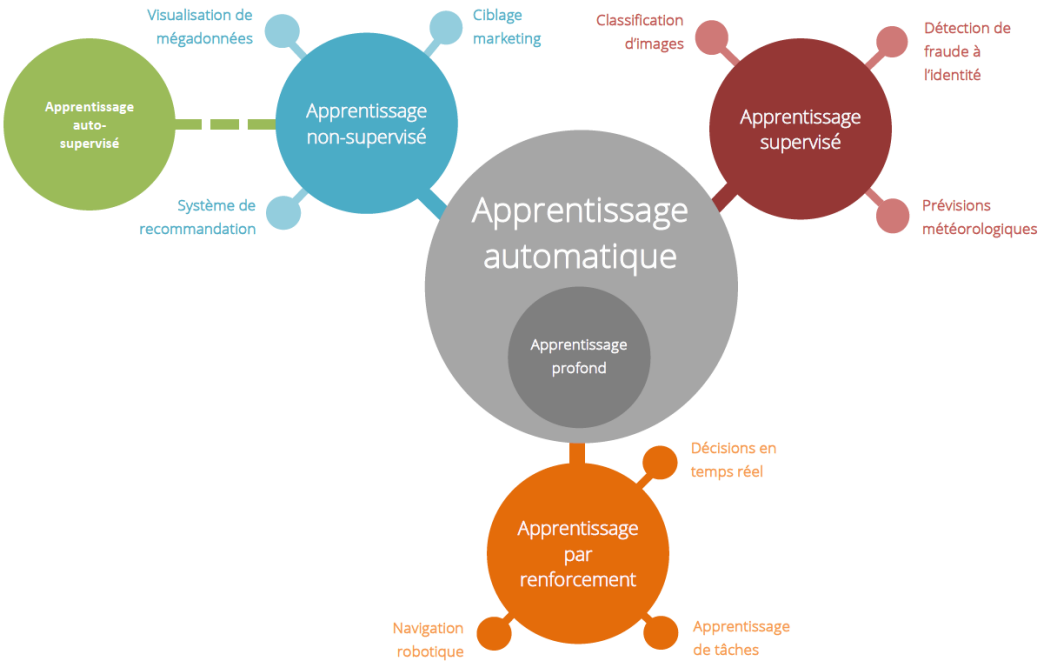

FIGURE 2.2 – Les types de Machine Learning[3]

2.3 L'étude des algorithmes ou 'La boîte noire'

Les algorithmes d'apprentissage automatique ne sont pas nouveaux, mais ce n'est que récemment qu'il a été possible d'appliquer des calculs mathématiques complexes[34]. L'apprentissage automatique est souvent utilisé pour prendre en charge les moteurs de recommandation qui fournissent des recommandations basées sur l'historique des utilisateurs.

2.3.1 Boîte noire : Fondation de Machine Learning

L'apprentissage automatique est très complexe et son fonctionnement dépend de la tâche à accomplir et de l'algorithme utilisé. Au cœur d'un modèle d'apprentissage automatique, une examination des données et une reconnaissance des modèles sont faites par un ordinateur, puis utilise ces informations pour mieux exécuter la tâche qui lui est assignée. Toute tâche qui repose sur un ensemble de points de données ou de règles peut être automatisée à l'aide de l'apprentissage automatique, y compris des tâches extrêmement complexes. Selon la situation, les algorithmes d'apprentissage automatique nécessitent plus ou moins d'intervention humaine ou de renforcement[35]. Cet algorithme va brasser

l'historique des données en combinant plusieurs critères, il pourra gérer des dizaines voire des centaines de critères[36].

Alors que le ML fait référence à une large catégorie d'algorithmes qui peuvent prendre un ensemble de données et l'utiliser pour identifier des modèles, découvrir des informations et/ou faire des prédictions[35].

La figure (2.1) montre la classification de quelques algorithmes de Machine Leraning :

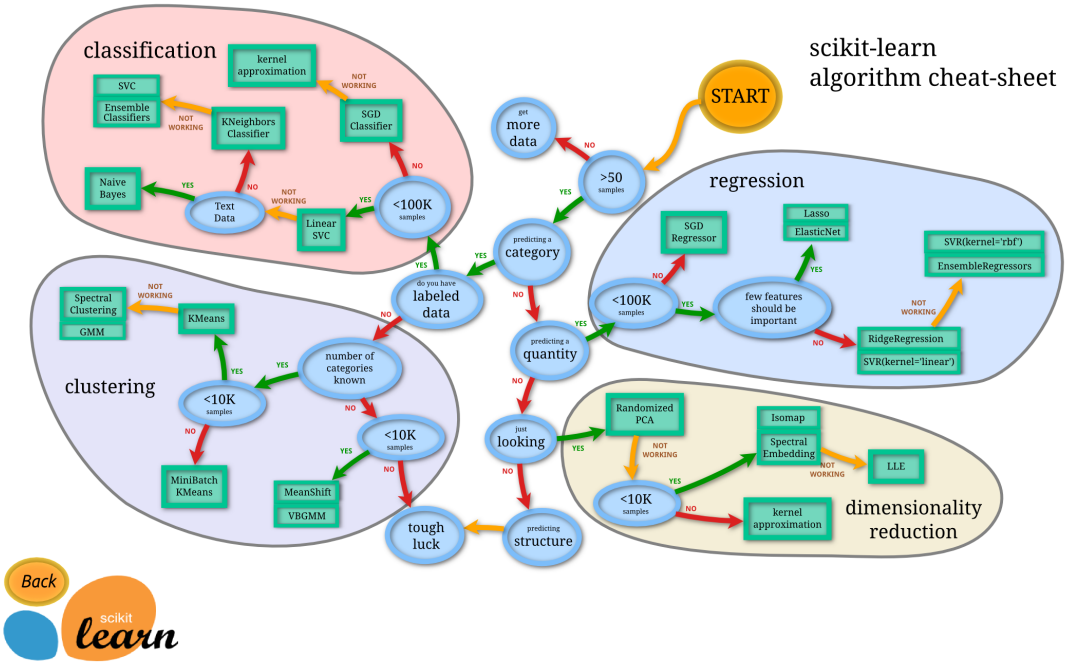

FIGURE 2.3 – Les différents algorithmes d'apprentissage automatique[4]

2.3.2 Quelques algorithmes de l'apprentissage artificiel

Le concepteur d'apprentissage artificiel fournit un ensemble complet d'algorithmes, chacun conçu pour résoudre un type spécifique de problème d'apprentissage automatique. Certains des algorithmes les plus courants sont décrits ci-dessous :

- **L'arbre de décision**

  C'est un algorithme d'apprentissage supervisé, un outil d'aide à la décision ou d'exploration de données utilisé pour représenter un ensemble de choix sous la

forme graphique d'un arbre. C'est l'une des méthodes les plus populaires pour les problèmes de classification de données, et les hiérarchies de test des modèles d'arbres de décision permettent de prédire les résultats. Les décisions possibles sont situées aux extrémités des branches ("feuilles" de l'arbre) et sont prises en fonction des décisions prises à chaque étape. Les arbres de décision fonctionnent en appliquant de manière itérative des règles logiques très simples (généralement une séparation des données via des "hyperplans" : plans à plus de 2 dimensions), chaque règle étant choisie en fonction du résultat de la règle précédente. Il est utilisé dans la médecine, la sécurité et l'intelligence économique[37].

- **Le boosting de gradient**

  C'est un algorithme supervisé et le Boosting est une méthode qui permet de convertir des apprenants faibles en apprenants forts. Dans un environnement boosté, chaque arbre est adapté à une version modifiée du premier jeu de données. L'amplification de gradient est une amélioration de l'apprentissage automatique qui s'appuie fortement sur la prédiction du modèle suivant pour réduire l'erreur de prédiction lorsqu'elle est mélangée avec des modèles précédents. L'idée principale est d'établir des résultats cibles pour le modèle suivant afin de minimiser les erreurs. Les méthodes de gradient boosting sont utilisées pour renforcer les modèles qui produisent des prédictions faibles. Il peut être trouvé dans la zone de recherche d'informations[38].

- **K-mean**

  C'est un algorithme non supervisé. Le clustering ne tente pas d'apprendre la corrélation entre l'ensemble observé de caractéristiques et la valeur à prédire. K-means est une méthode de clustering non hiérarchique. Il permet de regrouper les observations d'un ensemble de données en grappes distinctes (un petit ensemble de données). Par conséquent, des données similaires se retrouveront dans le même cluster. De plus, une seule observation peut être trouvée dans un groupe à la fois (exclusivité d'adhésion). Par conséquent, la même observation ne peut pas appartenir à deux clusters différents. Il est utilisé dans plusieurs domaines tels que : la démographie et l'exploration de données[39].

- **Apriori**

  Il appartient au type d'algorithme non supervisé. L'algorithme Apriori a été le premier algorithme proposé pour extraire fréquemment des ensembles d'éléments.

L'algorithme Apriori est une série d'étapes pour trouver l'ensemble d'éléments le plus fréquent dans une base de données. Cette technique d'exploration de données suit de manière itérative les étapes de jointure et d'élagage jusqu'à ce que l'ensemble d'éléments le plus fréquent soit atteint. Le seuil de support minimum est donné dans la question, ou supposé par l'utilisateur. Le principe Apriori stipule que si un ensemble d'éléments est fréquent, alors tous ses sous-ensembles doivent également être fréquents. On le trouve dans le domaine médical[40].

- **Q-learning** C'est un algorithme d'apprentissage par renforcement, il cherche à trouver la meilleur méthode pour atteindre un objectif définit en cherchant à obtenir un maximum de récompenses[41].
- **Naive Bayes** C'est un algorithme d'apprentissage semi-supervisé, cette algorithmes est utilisé pour reconnaitre les classes d'objets sur un data-set étiquettes[41].

2.4 Les algorithmes utilisés dans cette étude

Chaque data scientiste utilise l'apprentissage automatique pour analyser ses données et produire des modèles utiles, un bon modèle doit prendre en considération de nombreux aspects tels que les métriques, le temps de formation, la complexité du modèle, le nombre de paramètres et le nombre de fonctionnalités, ce qui nécessite évidemment une connaissance mathématique pour comprendre tous ces notions.[42]. Voici les algorithmes utilisés :

2.4.1 Méthode du K plus proche voisin KNN

La méthode "k-plus proche voisin" est l'une des méthodes d'apprentissage supervisé les plus simples disponibles pour les cas de régression et de classification[43].

"k-Nearest Neighbors" ou k-Nearest Neighbors en anglais (d'où le nom knn) est une méthode dans laquelle le modèle mémorise les observations dans l'ensemble d'apprentissage et est utilisé pour classer les données dans l'ensemble de test[43].

En effet, cet algorithme est qualifiée comme paresseux (Lazy Learning) car il n'apprend rien pendant la phase d'apprentissage. Pour prédire la classe d'une nouvelle donnée

d'entrée, il recherche ses K voisins les plus proches (en utilisant la distance euclidienne ou de Manhattan) et choisit la classe de la plupart des voisins[43].

On calcule la distance Euclidienne par :

$$\sqrt{\sum_{i=1}^{n}(x_i - y_i)^2} \tag{2.1}$$

Et la distance Manhattan par :

$$\sum_{i=1}^{n}|x_i - y_i| \tag{2.2}$$

La figure 2.4 est une explication du KNN :

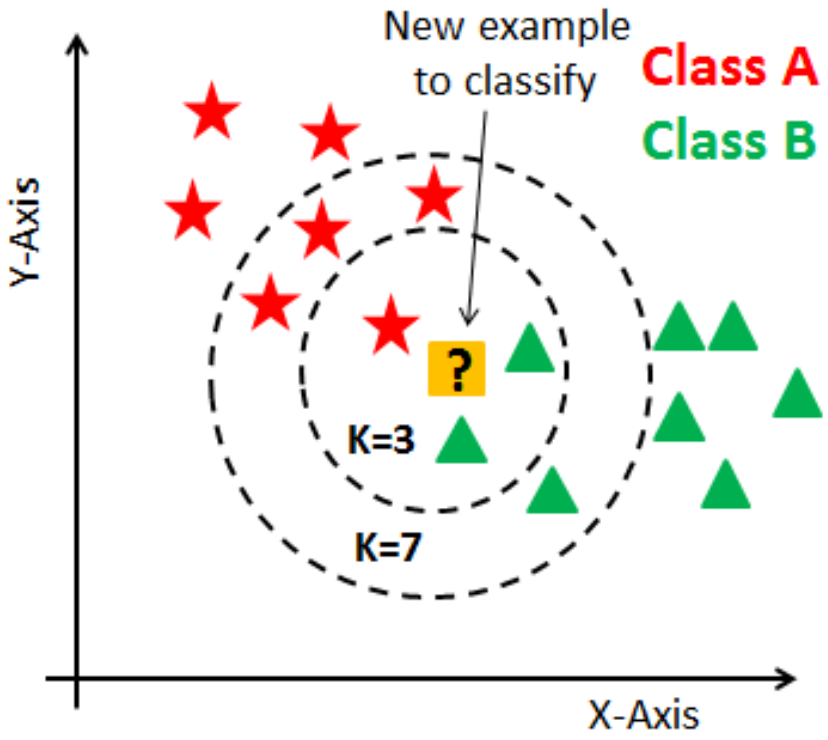

FIGURE 2.4 – Explication du knn [5]

Pour faire des prédictions, l'algorithme K-NN sera basé sur l'ensemble des données. En effet, pour les observations que nous voulons prédire et qui ne font pas partie du jeu de données, l'algorithme recherchera les K instances du jeu de données les plus proches de nos observations. Ensuite pour ces K voisins, l'algorithme calcule la valeur de la variable y pour l'observation que l'on souhaite prédire en fonction de leur variable de sortie (output

variable) y[44].

Pour appliquer cette méthode, les étapes à suivre sont les suivantes :

- On fixe le nombre de voisins k[43].
- On détecte les k-voisins les plus proches des nouvelles données d'entrée que l'on veut classer[43].
- On attribue les classes correspondantes par vote majoritaire[43].

Mais, comment choisit-on ce paramètre k lors de l'implémentation de l'algorithme ?

∗ On fait varier k[43].

∗ Pour chaque valeur de k, on calcule le taux d'erreur de l'ensemble de test[43].

∗ On garde le paramètre k qui minimise ce taux d'erreur test[43].

Le contre-coût est qu'il doit garder toutes les observations en mémoire afin de faire des prédictions. Il faut donc faire attention à la taille de l'ensemble d'entraînement. Aussi, le choix de la méthode de calcul de la distance et du nombre K de voisins peut ne pas être évident. Il faut essayer plusieurs combinaisons et ajuster l'algorithme pour obtenir des résultats satisfaisants[44].

2.4.2 Méthode de la NMF

NMF "Non-negative Matrix Factorization" est un algorithme d'apprentissage non supervisé pour l'extraction de caractéristiques[45]. Il est utile lorsqu'il existe de nombreux attributs et que les attributs sont ambigus ou peu prévisibles. En combinant des attributs, NMF peut générer des modèles. Elle est parfois considérée comme une méthode de classification non supervisée plutôt que comme une réduction de dimensionnalité[46].

L'algorithme est basé sur un objectif simple : décomposer une matrice positive en produit de deux matrices positives comme suit :

Soit V une matrice de dimensions $M \times \mathrm{N}$ à coefficients réels positifs ou nuls[46].
La NMF est la détermination d'une factorisation approchée :

$$V \approx \mathbf{W}\, H = \hat{V} \tag{2.3}$$

où W et H sont des matrices de dimensions respective.

$M \times$K et $K \times$N dont tous les coefficients sont des réels positifs ou nuls, et où l'opérateur "≈" désigne une « approximation » à définir[46].

L'ordre du modèle K est habituellement choisi tel que :

MK + KN ≪ MN, ce qui fait de la NMF une technique de réduction de la dimensionnalité[46]. La figure 2.5 explique le principe du NMF :

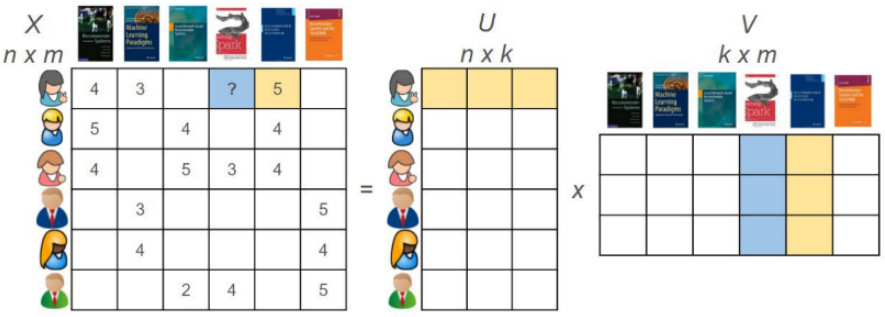

FIGURE 2.5 – Explication du NMF [6]

L'algorithme produit donc deux matrices : une matrice dont le nombre de lignes est le nombre d'assurés présents dans notre base de données, appelée par la suite "matrice des assurés", et une matrice dont le nombre de colonnes est notre prétraité. Le nombre d'étiquettes sera appelé "Matrix Group Behavior" (Matrix GA)[46].

La matrice de groupe de comportement est une représentation de chacune de nos anciennes variables, nos étiquettes de comportement, sur la nouvelle variable. Ces nouvelles variables révéleront alors des corrélations entre comportements : si deux descriptions comportementales sont bien représentées par la même variable, c'est qu'elles apparaissent souvent ensemble dans notre matrice d'origine. Par conséquent, nous avons nommé ces nouvelles variables « groupes d'actes »[46].

De son côté, la matrice des assurés renseignera sur la façon dont chacun de nos assurés est représenté par notre nouvelle variable (notre groupe programme). Chaque assuré s'est vu attribuer un poids dans chaque ensemble de procédures. Il s'agit de la projection de

notre assuré sur notre nouvel espace de faible dimension[46].

En machine learning, avant d'appliquer un algorithme, il est nécessaire de définir une métrique d'erreur que l'algorithme utilisera pour savoir s'il s'améliore. Dans le cas des algorithmes NMF, deux mesures d'erreur sont généralement choisies : la distance euclidienne et la similarité de Kullback-Leibler (ou entropie relative)[46].

L'algorithme NMF a de nombreuses caractéristiques. Ils sont particulièrement efficaces dans le cas de matrices creuses. De plus, ils ont la capacité d'obtenir une base non orthogonale. Cette particularité peut rendre les mots significatifs dans les deux ensembles de comportements. Ces propriétés rendent l'algorithme NMF largement utilisé. Cela montre qu'ils sont très efficaces dans différents domaines[46].

2.4.3 Méthode de la SVD

SVD (Singular Value Decomposition), la diagonalisation d'une matrice est souvent utile pour calculer les puissances de cette matrice ou comprendre son comportement. Malheureusement, toutes les matrices ne sont pas diagonalisables et cette procédure ne peut pas être appliquée aux matrices rectangulaires. La décomposition en valeurs singulières est une alternative à la diagonalisation dans certains cas[47].

soit M une matrice $N \times$N à coefficients complexe.Alors M peut s'écrire comme : [47]

$$M = \mathbf{U} \ \ \Sigma \ V \tag{2.4}$$

U et V sont deux matrice orthogonales d'ordre n.
Σ est une matrice diagonale d'ordre n dont les coefficients sur la diagonale sont des valeurs singulières de M[47].
Cette décomposition s'appelle décomposition en valeurs singulières de M[47].

Plus précisément, la décomposition en valeurs singulières (SVD) est l'un des algorithmes d'apprentissage non supervisé [48] Il fait partie des algorithmes de factorisation

matricielle qui réduit le nombre d'entités dans un jeu de données en réduisant la dimension spatiale de N à K (où K<N). Dans le cadre des systèmes de recommandation, SVD est utilisé comme technique de filtrage collaboratif. Il utilise une structure matricielle où chaque ligne représente un utilisateur et chaque colonne représente un élément. Les éléments de cette matrice sont les notes que l'utilisateur attribue à l'élément[49].

La figure 2.6 montre le fonctionnement du SVD :

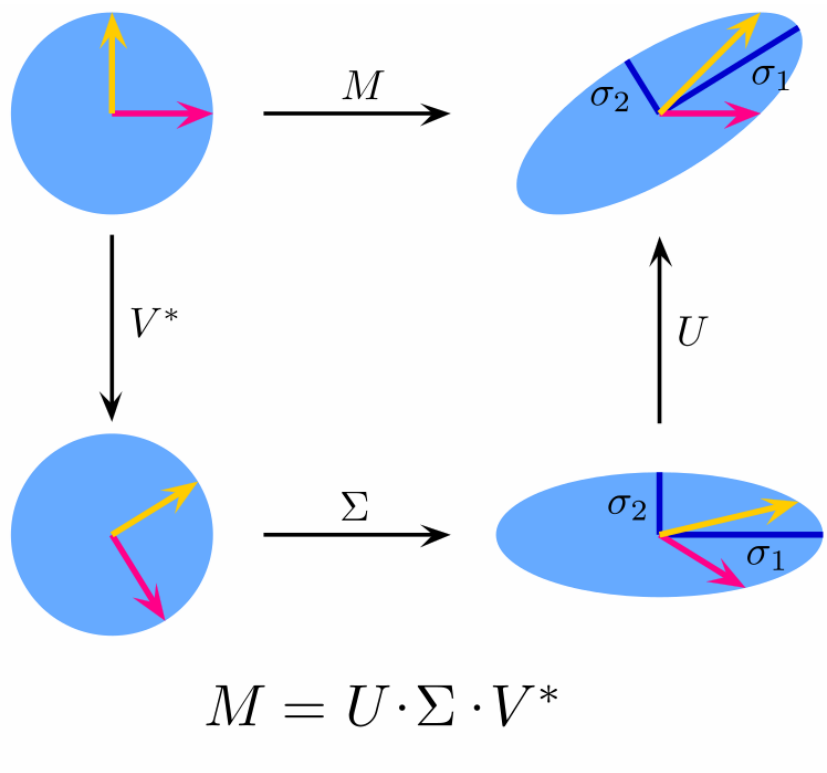

FIGURE 2.6 – Explication du SVD [7]

Illustration de la décomposition de la valeur singulière $M = \mathbf{U}\ \Sigma\ V^*$ d'une vraie matrice[50].

- La matrice V contient un ensemble de vecteurs de base orthonormés de K^n, dits « d'entrée » ou « d'analyse »[50].

- La matrice U contient un ensemble de vecteurs de base orthonormés de K^m, dits « de sortie »[50].

-La matrice Σ contient dans ses coefficients diagonaux les valeurs singulières de la matrice M, elles correspondent aux racines des valeurs propres de $M * M$[50].

2.4.4 Résolution des problèmes grâce aux Algorithmes de Machine Learning

Quel est l'intérêt d'utiliser le Machine Learning pour une variété de cas d'utilisation de base ? Qu'est-ce qui rend les utilisateurs heureux ? Comment l'apprentissage automatique hérite-t-il de nos préjugés ?[51]. L'un des plus grands avantages de l'apprentissage automatique est sa capacité à automatiser et à accélérer la prise de décision, ainsi qu'à augmenter le délai de rentabilisation. Cela commence par une meilleure visibilité commerciale et une collaboration renforcée.[51]

Par exemple, plus on utilise Netflix, plus l'algorithme nous permet de personnaliser notre contenu d'actualités, et lorsque on est sur Netflix, on reçoit de nouveaux épisodes/films à regarder. Ce système puissant collecte non seulement nos informations, mais les compare également à celles d'autres utilisateurs qui ont aimé les mêmes séries/films. Ces suggestions sont plus pertinentes par rapport à nos intérêts et notre satisfaction augmente avec notre utilisation.

En intégrant l'apprentissage artificiel, les systèmes de recommandation peuvent fonctionner plus rapidement et plus intelligemment[51].

2.5 Évaluation des performance des modèles de l'apprentissage artificiel

L'apprentissage artificiel est un domaine de la science qui peut produire des résultats intéressants et de très haut niveau en créant un modèle avec de bons résultats pour une tâche donnée, bien que de bons modèles soient développés, ces derniers font parfois face à plusieurs problèmes. Pour surmonter ces obstacles et problèmes, nous pouvons utiliser des métriques essentielles dans l'apprentissage automatique, qui aident à améliorer, mesurer et quantifier les modèles.

2.5.1 Mesurer les performances des modèles

Bien que la préparation des données et la formation d'un modèle d'apprentissage automatique soient des étapes primordiale dans un pipeline ML, il est tout aussi important de mesurer les performances de ce modèle formé. La façon dont un modèle se généralise à

des données invisibles définit des modèles d'apprentissage automatique adaptatifs et non adaptatifs. En utilisant différentes métriques pour l'évaluation des performances, nous devrions être en mesure d'améliorer la puissance prédictive globale du modèle avant de le déployer sur des données invisibles. Parmi les mesures d'évaluation : **Tests statistiques**

A. **Anova :** L'analyse de la variance (ANOVA) est une technique statistique paramétrique qui permet de déterminer si les moyennes de trois groupes ou plus sont significativement différentes. Il examine l'influence de divers facteurs en comparant des groupes (échantillons) selon leurs moyennes respectives[52].

Métriques d'évaluation sensibles au classement

A. **NDCG**

A Normalized Discounted Cumulative Gain (NDCG) est une mesure de la qualité du classement. L'objectif est de classer les éléments pertinents plus haut que les éléments non pertinents. Généralement, il est utilisé pour mesurer les performances des systèmes de recommandation[53].

$$NDCG = \frac{DCG}{IDCG} \tag{2.5}$$

Avec :

DCG de l'ordre recommandé.

iDCG de l'ordre idéal.

Mesures d'évaluation

A. **Précision**

C'est le nombre de résultats corrects trouvés, cette mesure permet de détecter la qualité des résultats renvoyés par le système[54].

$$Precision = \frac{TP}{TP + FP} \tag{2.6}$$

B. **Recall**

Le rappel est en fait le nombre de vrais positifs qui ont été rappelés (trouvés), c'est-à-dire il représente la probabilité qu'un élément pertinent soit sélectionné[54].

$$Recall = \frac{TP}{TP + FN} \tag{2.7}$$

C. **F1-Score**

F1-Score, qui combine parfaitement précision et rappel, est intéressant en tant que

métrique.[55] Il est calculé selon la formule suivante :

$$F1 - Score = 2 * \frac{Precision * Recall}{Precision + Recall} \tag{2.8}$$

D. **Accuracy**

L'Accuracy décrit le pourcentage de prédictions correctes[56].

$$Accuracy = \frac{TP + TN}{TP + TN + FN + FP} \tag{2.9}$$

E. **P@K**

P@K spécifie la proportion des items recommandés par le top-K[57].

$$Precision@K = \frac{1}{|U|}\Sigma_u \frac{TP}{TP+FP} \tag{2.10}$$

avec :

-$|U|$ Le nombre d'utilisateurs.

-TP et FP Le nombre de vrai positifs et faux positifs.

F. **R@K**

La R@K spécifie la proportion d'éléments de test extraits de la liste des recommandations top-K[57].

$$Recall@K = \frac{1}{|U|}\Sigma_u \frac{TP}{TP+FN} \tag{2.11}$$

avec :

-$|U|$ Le nombre d'utilisateurs.

-TP et FN Le nombre de vrai positifs et faux négatifs.

Métrique d'erreur

A. **MAE**

Une métrique qui nous indique la différence absolue moyenne entre les valeurs prédites et réelles dans l'ensemble de données. Plus la MAE est faible, plus le modèle s'adapte à l'ensemble de données[58].

$$MAE = \frac{1}{N}\Sigma|(y - \hat{y})| \tag{2.12}$$

Avec :

-N nombre d'observation.

-Y sortie réelle.

-ŷ sortie prédite.

B. **MSE**

C'est la moyenne arithmétique des écarts au carré entre la prédiction du modèle et l'observation[59].

$$MSE = \frac{1}{N}\Sigma_{i=1}^{N}(y_i - \hat{y})^2 \tag{2.13}$$

Avec :

-N nombre d'observation.

-Y sortie réelle.

-ŷ sortie prédite.

C. **RMSE**

Il s'agit d'une règle de notation quadratique qui mesure également la marge d'erreur moyenne. C'est la racine carrée de la moyenne de la différence au carré entre la valeur prédite et la valeur réelle observée[58].

$$RMSE = \sqrt{MSE} = \sqrt{\frac{1}{N}\Sigma_{i=1}^{N}(y_i - \hat{y})^2} \tag{2.14}$$

La Complexité

La complexité mesure la qualité et la vitesse d'un algorithme et permet de faire une comparaison entre plusieurs algorithmes selon le temps d'exécution[54].

2.5.2 L'augmentation des performances par rapport à la baseline

L'amélioration des performances du modèle commence par la beseline, un élément qui permet de comparer un modèle comparant à un autre modèle. Il peut être divisé en deux types :

— Après avoir construit un algorithme d'apprentissage automatique, nous comparons ses performances avec les performances d'un nouvel algorithme déjà créez[60].

— Comparez les performances du modèle avec les connaissances métier de l'expert[60].

Des étapes supplémentaires doivent être suivies pour augmenter l'amélioration du modèle, telles que :

1. Éviter l'overfitting : L'overfitting est l'incapacité du modèle à se généraliser aux données de test, car il a appris à partir des données d'entraînement par cœur. Nous pouvons certainement avoir des modèles qui sont très bons pour prédire l'ensemble de données d'entraînement. Les performances dans l'ensemble de données de test peuvent être inférieures, ce qui signifie que le modèle est overfit[60].
2. Fournir davantage d'échantillons de données : comme pour les humains, plus un algorithme est formé, plus il est susceptible d'améliorer les performances. Une façon d'améliorer les performances du modèle consiste à fournir à l'algorithme davantage d'échantillons de données d'apprentissage. Plus le modèle contient de données d'apprentissage, mieux il peut identifier correctement les observations[61].
3. Validation croisée : La validation croisée est une technique de formation et d'évaluation de modèles qui divise les données en partitions et forme plusieurs algorithmes sur ces partitions. Cette technique améliore la robustesse du modèle en préservant les données lors de l'apprentissage. Elle peut être un outil efficace pour former des modèles avec des ensembles de données plus petits[61].
4. Utiliser le Feature Engineering : une technique largement utilisée qui peut considérablement améliorer les performances du modèle[60]. Il s'agit d'une transformation des données brutes afin de les utiliser comme données d'apprentissage[62].

2.6 Conclusion

Dans ce chapitre, nous avons détaillé les concepts de base nécessaire pour la réalisation de notre travail. En commençant par définir le ML et présenter quelques algorithmes, par la suite nous avons mentionné les principaux algorithmes qui vont nous accompagner tout ou long de ce projet et enfin nous avons expliqué l'importance de mesurer un modèle en citant les métriques d'évaluation.

Chapitre 3

L'étude de l'approche proposée

3.1 Introduction

Après avoir expliquer la notion de ML et ses différents algorithmes, dans ce chapitre nous allons décrire les travaux applicatifs ainsi que les détails d'implémentation réalisés pendant ce travail. En premier lieu, nous allons décrire le contenu de nos Data-sets, ensuite nous allons présenter quelques étapes suivi durant la phase du pré-traitement et en dernier on présentons l'étude de l'approche proposée tout en détaillant les différentes étapes liées à la réalisation.

3.2 La collecte et le pré-traitement des données

Une de nos préoccupations était de trouver et de choisir un bon data-set, lors de nos recherches nous avons constaté la difficulté de trouver un jeu de données pertinent et qui répond a nos espérances. Pour que nos résultats soient significatifs, notre jeu de donnée doit être facilement accessible, disponible, déjà utilisé et son négliger la taille du data-set qui est un élément très important. Nous avons choisis deux data-sets de films contenant un nombre important d'évaluations des utilisateurs.

La collecte des données est une étape fondamentale pour la construction d'un modèle car ceux-ci s'appuient sur les données récoltées auprès des utilisateurs. Ces données récoltées vont nous aider à construire un profil d'utilisateur qui sera ensuite utilisé par différents algorithmes. Ces données sont divisées en deux, d'une part les données explicites

définit comme des données fournies de manière explicite par l'utilisateur (par exemple demander à un utilisateur de classer une collection de films en fonction de sa préférence) tant dit que les données implicites sont des données récoltées de manière implicites par l'utilisateur (comme par exemple le nombre de fois d'avoir regarder tel film).

Toutes ces données récoltées soit d'une manière explicite ou implicite sont utilisées pour le remplissage des data-sets et ces data-sets vont être exploités par nos algorithmes pour la construction de notre système de prédiction et de recommandation. Le choix de nos data-sets s'est orienté vers des data-sets qui possèdent des données explicites qui sont des données sous la forme d'évaluations (Le vote donné par un utilisateur pour un film), leurs principales avantages c'est qu'elles sont faciles à interprétées car elles sont données d'une manière précise de la part de l'utilisateur. En revanche, nous n'avons pas choisi les données implicites comme par exemple le temps passé à regarder un film ou le nombre de fois de l'avoir vu car ces données sont difficiles à récupérer et à interpréter de plus elles peuvent être non précises parce-qu'elles sont récoltées de plusieurs sources différentes et ne sont pas fournit par l'utilisateur. Le choix des données explicites (l'avis des utilisateurs sur les films visionnés) également vont nous aidé dans la construction de notre système de recommandation car notre système ne tient pas en compte les informations démographiques (sexe, l'age, adresse, le vrai nom, activités,etc.) pour prendre des décision. On peut constater que les caractéristiques socio-démographique des utilisateurs ne sont pas finalement assez importantes pour la constructions d'un système de recommandation, le plus important c'est l'avis et les préférences de l'utilisateur. Cependant, nous avons aussi pris en considération des informations liées au contexte à savoir : L'appareil utilisé, la compagnie, Les saisons,etc.

3.2.1 Le contenu des deux Data-sets

Dans notre étude nous avons utilisé deux Data-Sets : - Data-Set01 :

Lien : https ://www.kaggle.com/datasets/rounakbanik/the-movies-dataset

Source : Kaggle.

- Data-Set02 :

Lien : https ://grouplens.org/datasets/movielens/latest

Source : Movielens.

	Fichier	**Description**	**Attribut**
Data-Set01	ratings-small.csv	Contient les votes des films attribué par l'utilisateur.	-userId. -movieId. -rating. -timestamp.
	movies-metadata.csv	Contient les informations sur les films.	-adult. -genres. -id -title. ...etc (24 attributs)
	links.csv	Contient des identifiants(Id).	-movieId. -imdbId. -tmdbId.
	keywords.csv	Contient les caractéristique des films.	-id. -keywords.
	credits.csv	Contient des données sur la production des films.	-cast. -crew. -id.
Data-Set02	ratings.csv	Contient les votes des films.	-userId. -movieId. -rating. -timestamp.
	movies.csv	Contient les informations sur les films.	-movieId. -title. -genres.
	tags.csv	Contient les tags appliqués aux films.	-movieId. -userId. -tag. -timestamp.

TABLE 3.1 – Fichiers des deux Data-Set

3.2.2 Data-set 1 : Description des fichiers utilisés

1. Fichier "ratings.csv" :
   - userId : L'identifiant de l'utilisateur.
   - movieId : L'identifiant du film.
   - rating : La note attribuer par l'utilisateur.
   - timestamp : La date avec l'heur du vote de l'utilisateur.
2. Fichier "movies-metadata.csv" :

movies-metadata.csv	
Attributs	**Description**
genres	Catégorie des films.
id	L'identifiant des films.
title	Titres des films.

TABLE 3.2 – Description du fichier movies-metadata

3. Exemple d'un extrait du Data-Set 1 :
   La figure 3.1 représente des exemples d'utilisateurs qui ont noté des films tirés du fichier "ratings". Les colonnes représentent : userId, movieId et le rating.

	userId	movieId	rating
0	1	31	2.5
1	1	1029	3.0
2	1	1061	3.0
3	1	1129	2.0
4	1	1172	4.0

FIGURE 3.1 – Extrait du DS-1

3.2.3 Data-set 2 : Description des fichiers utilisés

1. Fichier "ratings.csv" :

- userId : L'identifiant de l'utilisateur.
- movieId : L'identifiant du film.
- rating : La note attribuer par l'utilisateur.
- timestamp : La date et l'heur du vote de l'utilisateur.

2. Fichier "movies.csv" :
   - movieId : L'identifiant du film.
   - title : Titre du film.
   - genres : Le genre du film.
3. Fichier "tags.csv" :
   - movieId : L'identifiant du film.
   - userId : L'identifiant de l'utilisateur.
   - tag : Description du film.
   - timestamp : La date et l'heur du vote de l'utilisateur.
4. Exemple d'un extrait du Data-Set 2 :
   La figure 3.2 représente des exemples de films tirés du fichier "movies". Les colonnes représentent : movieId, titre et le genre , et Les lignes représentent les informations de chaque film.

	movieId	title	genres
0	1	Toy Story (1995)	Adventure|Animation|Children|Comedy|Fantasy
1	2	Jumanji (1995)	Adventure|Children|Fantasy
2	3	Grumpier Old Men (1995)	Comedy|Romance
3	4	Waiting to Exhale (1995)	Comedy|Drama|Romance
4	5	Father of the Bride Part II (1995)	Comedy

FIGURE 3.2 – Extrait du DS-2

3.3 L'exploration de données et le pré-traitement : Un guide pratique

3.3.1 Exploration de données (EDA)

Après avoir choisi les deux Data-Sets, nous allons explorer ce qu'ils contiennent. Cette analyse va nous permettre de comprendre la structure de nos données et de tirer des statistiques qui pourront nous servir à l'interprétation des résultats.

- Data-Set 1 :

  Distribution des votes

  La figure 3.3 expose la manière dont les utilisateurs votent pour les films. On remarque que le plus grand nombre de note (rating) est à partir de 3 ce qui signifie qu'il existe un grand nombre d'utilisateurs qui votent que pour les films qu'ils ont aimé et les films de mauvaise qualité(note"rating" $\prec$ 3) sont probablement vite ignorés par les utilisateurs, alors on peut conclure qu'il y a moins de chance de regarder un film mal noté ou de le recommander.

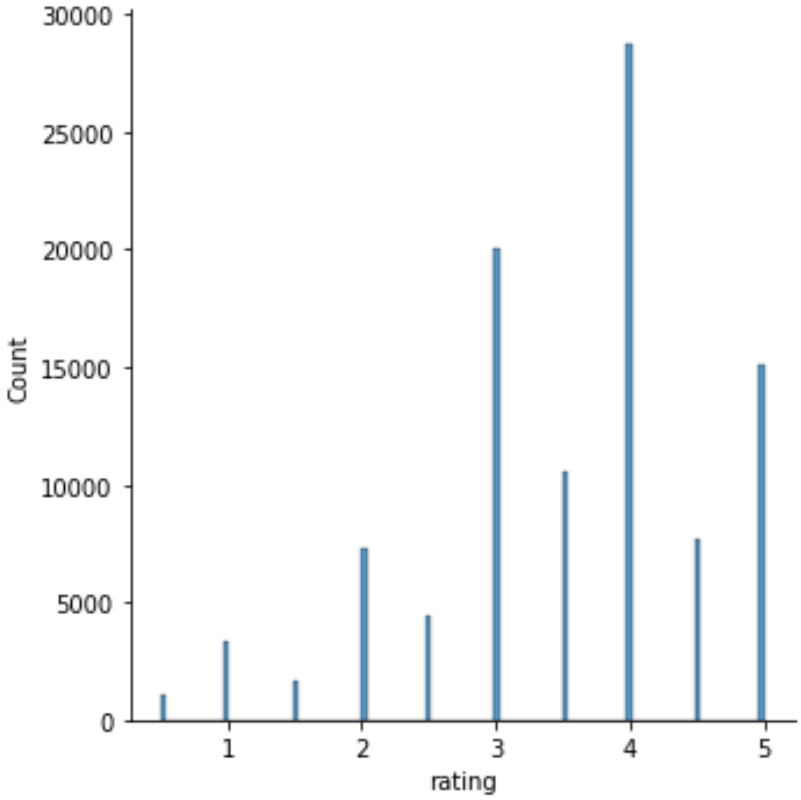

FIGURE 3.3 – Distribution des votes

Quelques opérations sur le premier Data-Set

1. La distribution des genres

   La figure 3.4 montre la distributions des genres dans notre DS, on remarque que les deux genre "Drama et Comédie" sont plus fréquent comparant aux autres catégories.

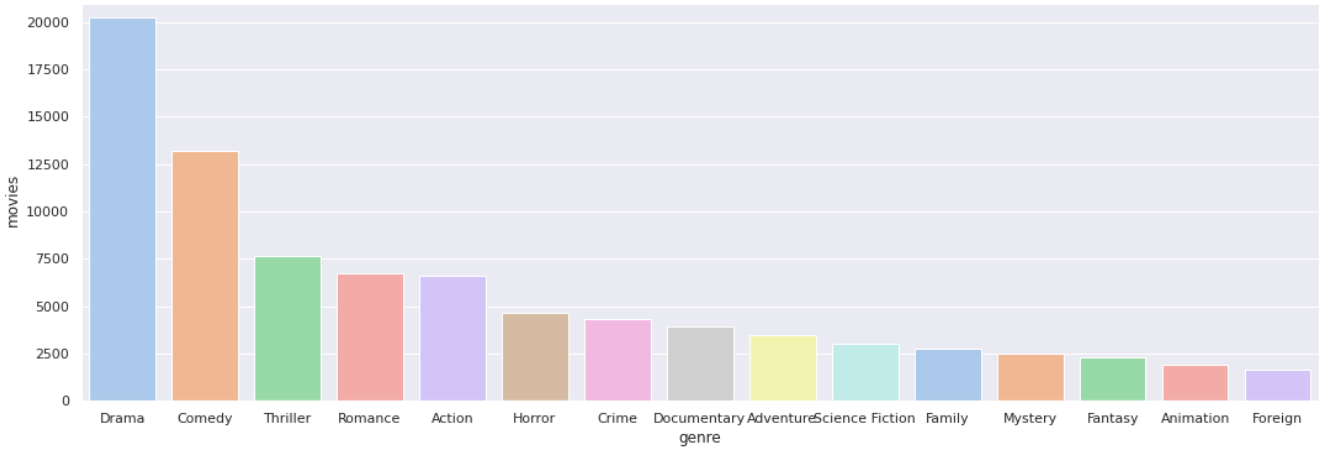

FIGURE 3.4 – Distribution des genres

2. Fichier "movies-metadata.csv"

   La figure 3.5 est une représentations des NAN (not a number) des attributs du fichier "movies-metadata" de notre DS-1. Les attributs nécessaire pour notre étude dans ce fichier(id,titre) n'ont pas un manque d'informations (les cases vides).

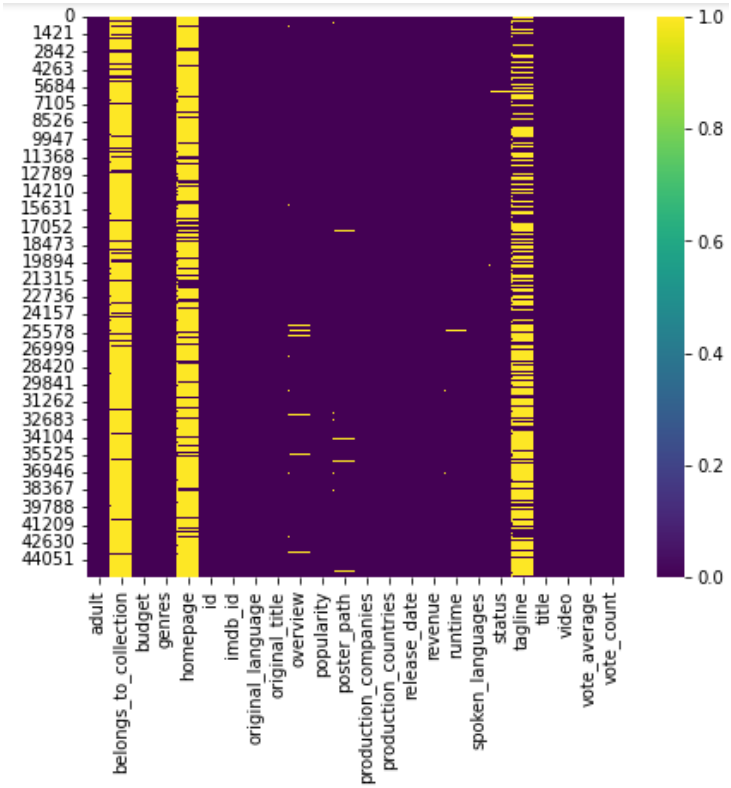

FIGURE 3.5 – La distribution des NAN

- Data-Set 2 :

Distribution des votes

La figure 3.6 montre la distribution des votes dans notre DS-2, on remarque que le nombre de vote est vraiment bas (entre 0 et 2) ce qui explique que les utilisateurs votent rarement pour les films qui n'ont pas aimé.

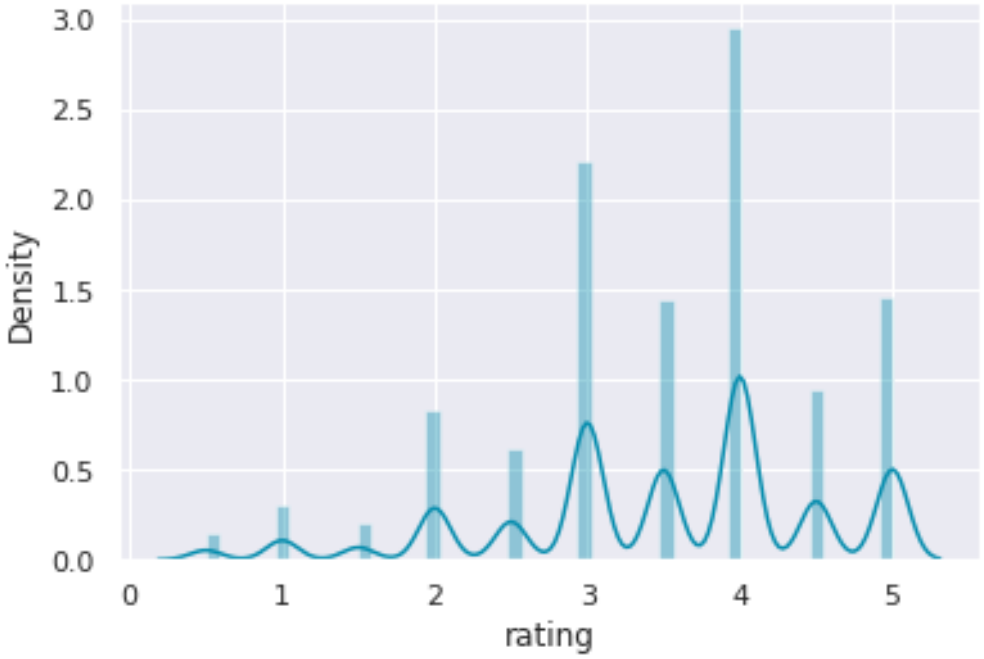

FIGURE 3.6 – Distribution des votes

Quelques opérations sur le deuxième Data-Set

1. La distribution des NAN dans le fichier "tags.csv"

   Cette figure 3.5 représente le nombre de valeur manquantes dans le fichier "tags" de notre DS-2. On remarque qu'il n y a pas un manque d'information ce qui est assez bien.

```
tag.isna().sum()

userId       0
movieId      0
tag          0
timestamp    0
dtype: int64
```

FIGURE 3.7 – Distribution des NAN

2. Relation entre le genre et la note (rating) :

   Le diagramme à barres suivant nous montre la relation entre le genre et le vote attribuée par l'utilisateur.

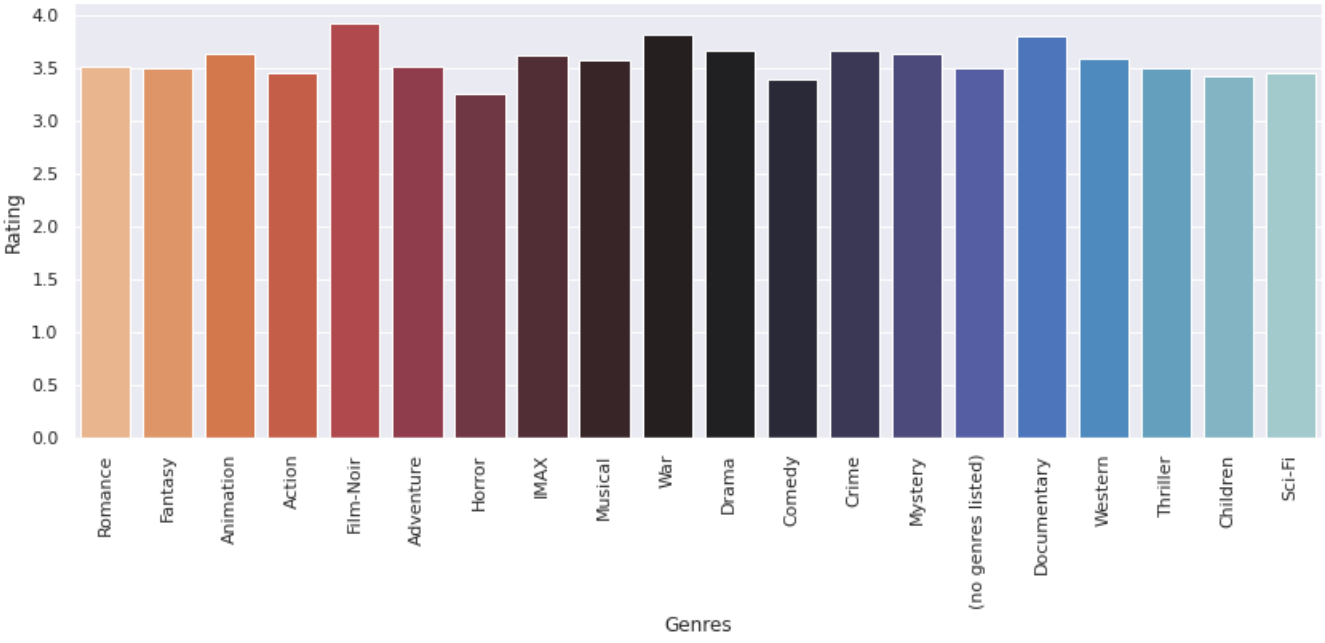

FIGURE 3.8 – Relation entre le genre et la note (rating)

3.3.2 Le pré-processing

Le principal objectif de la préparation des données est de s'assurer que les informations préparées pour l'analyse sont exactes et cohérentes. Afin qu'un algorithme puisse utiliser les données du Data-set, celles-ci doivent subir quelques transformations, on peut résumer notre pré-traitement dans les étapes suivantes :

1. La suppression des colonnes inutiles :

Le fichier "movies-metadata" du premier data-set contient 24 colonnes mais notre travail consiste à utiliser que 3 colonnes qu'on note (id, title, tagline) et les autres seront supprimées car ils ne sont pas utile.

Pour le deuxième data-set nous avons uniquement supprimé la colonne "timestamp" du fichier "tags".

2. Data Cleaning :

   Le nettoyage des données (Data Cleaning) est le processus de réparation ou de suppression des données incorrectes, mal formatées, en double ou incomplètes dans un jeu de données. Lors de la combinaison de plusieurs sources de données, il existe de nombreuses possibilités que les données soient dupliquées ou mal étiquetées. Dans notre cas par exemple nous avons le "id" du fichier "movies-metadata" qui contient des dates (Ex :1997-08-20) du coup nous avons essayé de réparer cette mal formation de données.

3. Combiner les fichiers et les stocker :

   Après l'application des étapes précédentes, nous avons fait une jointure de deux fichiers afin d'avoir un seul jeu de données :

   - Data-Set 1 : fichier "ratings" avec "movies-metadata".
   - Data-Set 2 : fichier "ratings" avec "movies".

   Ensuite nous allons les stocker pour pouvoir appliquer les différents algorithmes et réaliser les différentes analyses sur ce nouveau jeu de données.

3.4 Approches proposées

Dans cette section, nous présentons notre proposition que nous avons développer pour le système de prédiction et de recommandation. Notre objectif est d'aider les utilisateurs en recommandant des films en fonction de leurs préférences et de leurs attentes, pour cela nous avons développé trois différentes approches et proposer deux approches différentes qui sont :

A) Évaluer un système prédictif en utilisant KNN, SVD, NMF et KNNBasic sur les deux DS.

B) Évaluer un système de recommandation avec prédiction en utilisant SVD, NMF le DS-2.

C) Évaluer un système de recommandation en utilisant KNN, SVD, NMF sur les deux DS.

D) Proposer une approche hybride(Filtrage collaboratif en utilisant le SVD + Filtrage basée sur le contenu(technique du RF-IDF)) sur le DS-2.

E) Proposer une approche contextuel (intégrer le contexte dans l'approche Hybride).

3.4.1 Architecture proposées

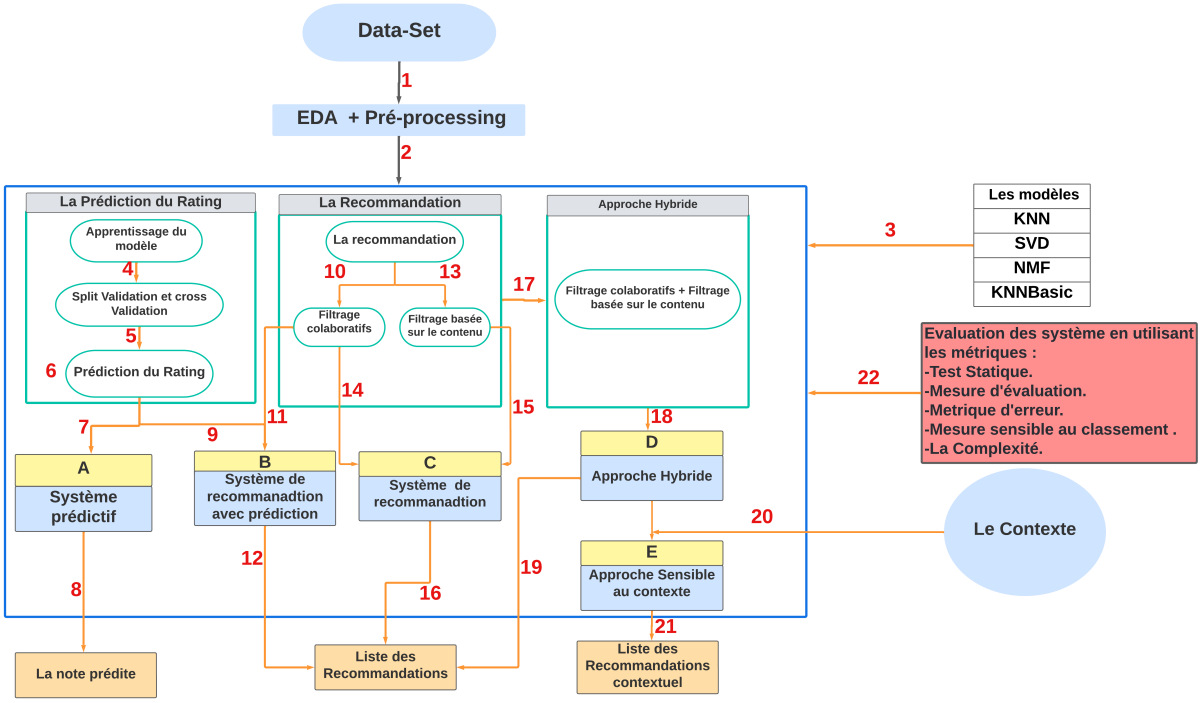

FIGURE 3.9 – Architecture de nos approches

La figure 3.9 représente l'architecture de l'approche proposées et ses étapes se résument comme suit :

1. Visualiser Les DS.
2. Exploration de données et le pré-traitement (pour plus de détaille voir la section 3.3).

3. Utilisation des algorithmes pour chaque phase.

   A) Système prédictif.

      4. Apprentissage du modèle en utilisant KNN, SVD, NMF et KNNBasic.
      5. Utiliser le Split et Cross Validation.

6. Prédire la note (rating).

8. Retourner la valeur prédite.

B) Système de recommandation avec prédiction.

9. Utiliser la prédiction des notes.

11. Recommander les films en utilisant le SVD et le NMF.

12. Obtenir la liste des recommandation avec prédiction.

C) Système de recommandation.

10. Filtrage collaboratif en utilisant le SVD.

13. Filtrage basée sur le contenu en utilisant la technique TF-IDF.

14. Liste des recommandation pour chaque filtrage.

D) Approche hybride.

17. Combiner les deux types de filtrage(filtrage collaboratif + contenu). 19. Liste des recommandation.

E) Proposer une approche contextuelle.

20. Intégrer le contexte à l'approche hybride.

21. Liste des recommandation contextuels.

22. L'évaluation en utilisant différentes métriques :

- Test Statistiques.
- Mesure d'évaluation.
- Métriques d'erreurs. - Métriques sensible au classement.
- La complexité.

3.4.2 Approche du système prédictif

Pour cette approche nous avons choisi d'utiliser les fichiers combiner des deux Data-Set en suivant les étapes suivante :

1. **L'apprentissage du modèle :**

   Cette phase est consacré pour l'apprentissage et la validation de nos modèle de machine Learning en utilisant deux type de validation différentes : Split et Cross Validation.

   Split Validation

   Split Validation ou Train-Test Split est une technique utilisée pour évaluer les performances d'un modèle. Cette technique consiste à décomposer de manière aléatoire l'ensemble de données : une partie de données pour l'entrainement et une autre pour le test. Dans notre approche nous avons utilisé cette technique pour diviser nos deux jeux de données au hasard en 80% (35995 lignes pour le DS01 et 80658 lignes pour le DS02) pour l'entraînement de notre modèle de Machine Learning et les 20%(8999 lignes pour le DS01 et 20165 lignes pour le DS02) qui reste permettra de tester notre modèle.

   Cross Validation

   La Cross-Validation est une technique permettant de tester les performances d'un modèle prédictif de ML. Parmi les méthode du Cross Validation les K-Folds, cette dernière consiste à séparer l'ensemble de données en k sous-ensembles différents de même tailles appelés (folds). Dans notre cas nous avons utilisé cette technique qui va diviser les deux jeux de données en 3 trois sous-ensemble différents (3-folds, 6-folds et 9 folds) afin de trouver le meilleur sous-ensemble.

2. **Prédiction du la note (rating) :**

   Après avoir diviser nos data-sets nous allons mettre en œuvre des modèles de ML pour la prédiction et l'analyser sur les votes attribués par l'utilisateur car actuellement, c'est devenu une tendance de prédire la note du film en fonction des données collectées qui relient l'utilisateur aux films. Ce types d'approches utilisent les algorithmes d'apprentissage pour apprendre de nouvelles évaluations.

   Notre système de prédiction repose sur l'intuition que les utilisateurs ayant eu un comportement similaire dans le passé (personnalisation des films selon l'historique de visualisation et l'opinion attribué) vont avoir tendance à se comporter de manière

similaire dans le futur. Nous avons choisi de prédire la note qu'un utilisateur va attribuer à un film qui n'a pas encore été évalué car l'opinion de l'utilisateur est très important pour classer les films s'ils sont de bonne ou de mauvaise qualité.

L'exemple applicatifs du Data-Set 1 :

Dans notre DS01, nous avons l'utilisateur 450 qui a vu les films suivants "The Million Dollar Hotel", "Boogie Nights", "Murder She Said","The Good Thief", "Terminator 3 : Rise of the Machines" et il les a noté respectivement 4.0, 4.5, 4.5, 4.0, 5.0. D'une autre part nous avons l'utilisateur 16 qui a le même historique de visualisation que l'utilisateur 450 et presque attribué les même notes respectivement aux films visualisé comme le montre le tableau suivant :

Utilisateur 450	
Films	Rating
The Million Dollar Hotel	4.0
Boogie Nights	4.5
Murder She Said	4.5
The Good Thief	4.0
Terminator 3 : Rise of the Machines	5.0

$\Longleftrightarrow$

Utilisateur 16	
Films	Rating
The Million Dollar Hotel	4.0
Boogie Nights	4.5
Murder She Said	4.0
The Good Thief	4.0
Terminator 3 : Rise of the Machines	?

TABLE 3.3 – Deux profiles d'utilisateurs similaires

D'après le tableau ci dessous 3.3, on peut constater qu'on peut prédire la note que l'utilisateur 16 va attribuer au film 'Terminator 3 : Rise of the Machines' car ces deux utilisateurs sont similaires et on peut même constater qu'il est fort possible que la note prédite soit entre 3 et 5.

L'exemple applicatifs du Data-Set 2 :

Le tableau 3.4 montre les deux utilisateurs choisi du deuxième Data-Set :

Dans notre exemple, On a l'utilisateur 66 et l'utilisateur 414 qui ont vu et évaluer les même films(comme le montre le tableau 3.4), d'après ce tableau on peut déduire que ces deux utilisateurs ont le même historique de visualisation et attribué presque le même vote c'est pour cela notre système a le pouvoir de prédire la note que l'utilisateur 66 va attribuer au film "Jumanji(1995)" sachant que l'utilisateur 414 a attribué la note de 3.0 au film "Jumanji(1995)".

Utilisateur 414	
Films	Rating
Get Shorty(1995)	4.0
Collateral(2004)	4.0
Shaun of the Dead(2004)	4.0
incredibles, The(2004)	4.0
Jumanji(1995)	3.0

$\Longleftrightarrow$

Utilisateur 66	
Films	Rating
Get Shorty(1995)	4.0
Collateral(2004)	4.5
Shaun of the Dead(2004)	5.0
incredibles, The(2004)	4.0
Jumanji(1995)	?

TABLE 3.4 – Deux profiles d'utilisateurs similaires

Maintenant après avoir citer les exemples applicatifs de nos DS nous allons construire à l'aide des algorithmes de l'apprentissage automatique 4 modèles qui vont prédire le vote manquant des utilisateurs 16 et 66 que note :

KNN : Déroulement de l'algorithme KNN

Input : Données du DS, Fonction de calcul de distance, Nombre k :

- Calculer toutes les distances d'une observation A (film ou utilisateur) avec toute les autres observations (film ou utilisateur) du jeu de données DS.
- Retenir les K observations du DS les plus proches de A en utilisant les distances les plus petites renvoyées par la fonction de calcul de distance.
- Prendre les valeurs de r_y [1] des K observations retenues :

1. La valeur de la distance

- Pour une classification, calculer le mode de y retenu.

Output : Retourner la valeur obtenu dans la dernière étape comme étant la valeur qui a été prédite par le KNN pour l'observation A.

SVD :Déroulement de l'algorithme SVD

Input : La matrice Utilisateur-Film-Note (Rating) (A matrice utilité m [2]*n [3]) :

-Décomposer la matrice A en trois autres matrices :

- **U est une matrice singulière gauche orthogonale m*r** représentant la relation entre les utilisateurs et les facteurs latent(Note"Rating").
- **S est une matrice diagonale r*r** décrit la force de chaque facteur latent.
- **V est une matrice diagonale droite singulière r*n** indique la similitude entre les films et les facteurs latents.

-Multiplier à nouveau U*S*V.

Output : Prédire la note (rating) du film que l'utilisateur n'a pas évalué.

NMF : Déroulement de l'algorithme NMF

Input : X une matrice de taille M(Les films)*N(Les utilisateurs) :

-Recherche de deux matrices W (de taille M*K) et H (de taille K*N) tel que :

- Le rang de factorisation K(Entier choisi relativement) est plus petit que min(M,N).
- W et H ne contiennent que des valeurs positives ou nulles.
- X $\approx WH$

-Le produit de ces deux matrices W*H est les valeurs des notes prédites.

Output : Les notes prédites pour un film donné.

2. Les films
3. Les utilisateurs

KNNBasic :

KNNBasic (algorithme de voisinage de base) appelé aussi KNN basé sur l'utilisateur est un algorithme de filtrage collaboratif, il a le même fonctionnement que le KNN (expliqué précédemment). Il est dédié spécialement pour la prédiction du la note (rating). Dans notre étude nous l'avons utilisé pour prédire la note inconnue des deux utilisateurs 16 et 66 en utilisant la similitude entre leur utilisateurs similaires. La note (rating) fournit par le KNN basé sur l'utilisateur peut être une valeur réelle contrairement aux KNN classique qui fournit que des nombres entiers.

3. **Résultats des prédictions**

Data-Set 1 :

Les figures suivantes 3.10 et 3.11 représentent les résultats de prédiction obtenus des algorithmes utilisés pour l'utilisateur 16 du premier DS pour le film "Terminator 3 : Rise of the Machines" :

Résultats du KNN :

Model	Predictions
KNN	4

FIGURE 3.10 – Résultats de prédiction du KNN

D'après le tableau 3.3 l'utilisateur 450 a évalué le film "Terminator 3 : Rise of the Machines" avec une note de 5.0. La figure ci dessus montre la note de prédiction fournie par le KNN. On remarque que la note est de 4 ce qui est assez proche de la note donnée par l'utilisateur 450.

Résultats du SVD, NMF et KNNBasic :

On remarque par l'observation de la figure 3.11 que les prédictions des trois algorithmes sont pertinentes et proches de la note donnée par l'utilisateur 450.

Modeles	Predictions
KNNBasic	4.274194
SVD	4.839304
NMF	4.169566

FIGURE 3.11 – Résultats de prédiction du SVD, NMF KNNBasic

Discussion :

En comparant tout les résultats entre la figure 3.10 et 3.11, la meilleure note a été obtenue par le SVD ce qui signifie que c'est l'algorithme le plus performant par rapport aux KNN, NMF et KNN basée sur l'utilisateur . On peut constater également que les notes prédites ne dépassent pas la note maximale donné par un utilisateur(5 étoile) ce qui confirme la performance de notre système.

Data-Set 2 :

Le tableau 3.5 résume les résultats de prédiction pour l'utilisateur 66 du DS02 :

Jumanji (1995)	
La note de l'utilisateur 414 est 3	
Prédiction du Film pour l'utilisateur 66	
Algorithme	Prédiction
KNN	5.0
SVD	3.73
NMF	3.60
KNNBasic	3.64

TABLE 3.5 – Prédiction d'évaluation de l'utilisateur 66

D'après le tableau 3.5 on remarque que l'algorithme qui fournit la note la plus proche à celle de l'utilisateur 414 est le NMF avec une note de 3.60. Cependant le KNN fournit une prédiction de 5.0 ce qui est un peu loin du la note (rating) donnée par l'utilisateur 414. Par conséquent, on peut déduire que le NMF est l'algorithme le plus performant dans notre deuxième jeux de donnés.

3.4.3 Approche de Recommandation avec Prédiction

Recommander un film uniquement basé sur son genre, acteur ou réalisateur sans l'intégration des évaluations reste une mauvaise idée mais avec l'ajout de la prédiction de notation de l'utilisateur, elle serait d'une grande aide pour le système de recommandation car la plupart des prédictions de notation dans les recommandations sont basées sur les préférences et sur le comportement historique des utilisateurs similaires. Pour réaliser cette recommandation avec la prédiction nous avons utilisé les deux algorithmes de factorisation SVD et le NMF pour le DS-2.

Les étapes suivies dans l'approche

1. Afficher en utilisant une fonction l'historique des meilleurs films que l'utilisateur 66 a noté.
2. Prédire la note (rating) en utilisant le SVD et le NMF.
3. Créer une fonction pour obtenir les résultats des recommandations en utilisant la prédictions du la note (rating).
4. Afficher les notes prédite et les films recommandées pour l'utilisateur 66.

Recommandation basée sur FC par facteurs latents

En se basant sur les notes prédites, il est raisonnable d'inspecter visuellement les films recommandés et les films les mieux notés de l'utilisateur dans le passé en utilisant les deux algorithmes SVD et NMF (expliqué dans les sections précédentes). Pour voir l'historique de l'utilisateur 66, examinons ses 11 films les mieux notés, on peut constater que cet utilisateur bénéficie d'un large éventail de genres plus précisément le genre Drama. C'est pour cela on va l'aider à choisir les bons films en fonction des prédictions acquises qui sont stockées dans le même ordre et dans le même format que l'historique de cet utilisateur et aussi comparer les deux algorithmes et de tirer le plus performant.

Résultats des recommandations avec prédiction

Films précédemment notés par l'utilisateur 66

La figure 3.12 représente l'historique de l'utilisateur 66 :

userId	movieId	rating	title	genres
66	2762	5.0	Sixth Sense, The (1999)	Drama|Horror|Mystery
66	6708	5.0	Matchstick Men (2003)	Comedy|Crime|Drama
66	4226	5.0	Memento (2000)	Mystery|Thriller
66	5673	5.0	Punch-Drunk Love (2002)	Comedy|Drama|Romance
66	5747	5.0	Gallipoli (1981)	Drama|War
66	5902	5.0	Adaptation (2002)	Comedy|Drama|Romance
66	5952	5.0	Lord of the Rings: The Two Towers, The (2002)	Adventure|Fantasy
66	922	5.0	Sunset Blvd. (a.k.a. Sunset Boulevard) (1950)	Drama|Film-Noir|Romance
66	6711	5.0	Lost in Translation (2003)	Comedy|Drama|Romance
66	535	5.0	Short Cuts (1993)	Drama
66	1704	5.0	Good Will Hunting (1997)	Drama|Romance
66	1748	5.0	Dark City (1998)	Adventure|Film-Noir|Sci-Fi|Thriller

FIGURE 3.12 – Films précédemment notés

Films recommandés pour l'utilisateur 66 par l'algorithme SVD

userId	movieId	predictions_rating	title	genres
66	750	4.847987	Dr. Strangelove or: How I Learned to Stop Worr...	Comedy|War
66	1204	4.818106	Lawrence of Arabia (1962)	Adventure|Drama|War
66	904	4.806510	Rear Window (1954)	Mystery|Thriller
66	4973	4.805142	Amelie (Fabuleux destin d'Amélie Poulain, Le) ...	Comedy|Romance
66	858	4.793754	Godfather, The (1972)	Crime|Drama
66	898	4.775077	Philadelphia Story, The (1940)	Comedy|Drama|Romance
66	1104	4.745057	Streetcar Named Desire, A (1951)	Drama
66	1223	4.733733	Grand Day Out with Wallace and Gromit, A (1989)	Adventure|Animation|Children|Comedy|Sci-Fi
66	741	4.727226	Ghost in the Shell (Kôkaku kidôtai) (1995)	Animation|Sci-Fi
66	48516	4.721196	Departed, The (2006)	Crime|Drama|Thriller

FIGURE 3.13 – Films recommandés

La figure 3.13 montre la liste des films recommandés pour l'utilisateur 66 en utilisant le SVD. En comparant les résultats avec les films regardés et notés par l'utilisateur 66 (Figure 3.17) on peut constater clairement que les genres de films recommandés ressemble beaucoup aux préférences de l'utilisateur 66.

Films recommandés pour l'utilisateur 66 par l'algorithme NMF

userId	movieId	predictions_rating	title	genres
66	7767	5.000000	Best of Youth, The (La meglio gioventù) (2003)	Drama
66	148881	5.000000	World of Tomorrow (2015)	Animation|Comedy
66	31364	4.999693	Memories of Murder (Salinui chueok) (2003)	Crime|Drama|Mystery|Thriller
66	158966	4.994077	Captain Fantastic (2016)	Drama
66	78836	4.990233	Enter the Void (2009)	Drama
66	6442	4.987143	Belle époque (1992)	Comedy|Romance
66	26258	4.977983	Topo, El (1970)	Fantasy|Western
66	27397	4.975043	Joint Security Area (Gongdong gyeongbi guyeok ...	Crime|Drama|Mystery|Thriller|War
66	26171	4.971606	Play Time (a.k.a. Playtime) (1967)	Comedy
66	1248	4.965959	Touch of Evil (1958)	Crime|Film-Noir|Thriller

FIGURE 3.14 – Films recommandés

La figure 3.14 nous montre les résultats des recommandations pour l'utilisateur 66 en appliquant l'algorithme NMF. Les résultats des films recommandés semble être similaires aux préférences de cet utilisateur car on retrouve les mêmes genres de films recommandés que les genres préférés par l'utilisateur 66.

3.4.4 Approche de Recommandation

La raison pour laquelle les utilisateurs ont besoin de recommandations fiables est simple : étant donné la disponibilité d'un nombre presque illimité de choix (différents genres de films et de différentes qualités), l'utilisateur a besoin d'être guidé vers le prochain meilleur film qui répond à ses besoins ou à ses préférences. Le système de recommandation vise à nous aider également à trouver le plus facilement possible un film qui va plaire à l'utilisateur. En revanche, lorsqu'un nouveau utilisateur s'inscrit au système et il n'a pas encore fourni des informations à son sujet (ces préférences,noter les films,etc) le système peut ne pas être en mesure de générer des recommandations personnalisées pour cette utilisateur, on appelle ce type de problème "le démarrage à froid" c'est l'un des principaux défis des systèmes de recommandation. Une des solutions typiques pour ce problème est de recommander des films sans la prédiction c'est-à-dire présenter aux utilisateurs les films populaires, permettant au système d'obtenir des informations initiales sur l'utilisateur qui peut être utilisé ensuite pour générer plus de recommandations personnalisé. Il existe une autre solution proposée par Netflix cette solution consiste à fournir à de nouveaux utilisateurs un mois gratuit afin de collecter des données sur les

préférences des utilisateurs par la suite ces données seront utilisées pour les recommandations personnalisées.
Pour la construction de notre système de recommandation sans prédiction, nous avons utilisé deux types de filtrage :

1. Filtrage collaboratif item-item :

   L'approche que nous proposons ici consiste à recommander des films aux utilisateurs selon leurs préférences. Plus précisément notre approche utilise les données et l'historique des films évalués ou consultés par l'utilisateur pour une recommandation, ce qui est évident que les préférences d'un utilisateur restent similaires. Cette technique a été réalisé avec trois algorithmes :

   KNN

   Pour recommander des films aux utilisateurs l'algorithme du k-plus proche voisin s'appuie sur la similarité des caractéristiques des films en suivant les étapes suivantes :

   (a) Calculer la "distance" entre le film cible et tous les autres films de la base de données.

   (b) Classer les distances de la plus petite à la plus grande.

   (c) Renvoyer les K films voisins les plus proches comme des recommandations de films les plus similaires.

   Résultat de la recommandation du KNN

   - Data-Set 1 :

Recommandation pour le film : Rocky III	
K = 5	
Titre Recommandé	Genres
The Long Walk Home	Drama-History
School Daze	Comedy-Drama
The Skulls	Crime-Drama-Thriller
Harley Davidson and the Marlboro Man	Action-Thriller
Hunk	Fantasy-Comedy

TABLE 3.6 – Recommandation fournit par le KNN

Le tableau 3.6 illustre les recommandations obtenues pour le film "Rocky III" qui est du genre "Drama" en utilisant l'algorithme KNN dans le DS01. À première vue, la recommandation obtenue semble assez bonne car l'algorithme recommande 3 films sur 5 du même genre que le film de départ.

- Data-Set 2 :

Recommandation pour le film : Pulp Fiction (1994)	
K = 5	
Titre Recommandé	Genres
Silence of the Lambs, The (1991)	Crime-Horror-Thriller
Shawshank Redemption, The (1994)	Crime-Drama
Seven (a.k.a. Se7en) (1995)	Mystery-Thriller
Forrest Gump (1994)	Comedy-Drama-Romance-War
Usual Suspects, The (1995)	Crime-Mystery-Thriller

TABLE 3.7 – Recommandation fournit par le KNN

Le tableau en haut 3.7 représente les recommandations pour le film "Pulp Fiction (1994)" en utilisant l'algorithme KNN dans le DS02. On remarque que les genres des films recommandés sont fortement liés aux genres du film de départ et on remarque également que la date de sortie du film d'entrée influence de manière très importante les recommandations.

SVD

Cette méthode est basée sur la description des films les mieux notée. On a appliqué cette méthodes en suivant les étapes ci dessous :

(a) Afficher la description du film le plus populaire par rapport au note (rating).

(b) Afficher le nom du film le plus populaire par rapport au note (rating).

(c) Créer de la matrice de corrélation de user-item.

(d) Afficher le nombre de la matrice d'utilité.

(e) Transposer la matrice utilité.

(f) Utiliser la corrélation basée sur les similitudes entre les goûts des utilisateurs pour calculer le coefficient PearsonR pour chaque paire de films dans la matrice transposée.

(g) On prend le film le plus populaire , ensuite nous allons extraire les valeurs de corrélation avec tous les autres films calculés par PearsonR .

(h) Filtrer les films le plus corrélé à notre film de départ et afficher les films recommandés.

Résultat de la recommandation du SVD

- Data-set 1 :

Recommandation pour le film : Rocky III	
K-SVD=4	
Titre Recommandé	Genres
Blood Diamond	Drama-Thriller-Action
Rocky IV	Drama
Rocky V	Drama
Sweet Sixteen	Horror

TABLE 3.8 – Recommandation fournit par le SVD

Les résultats du tableau 3.8 nous donne un aperçu sur les films recommandés en appliquant le SVD. Comme film de départ pour le DS01 on a choisi "Rocky III" du genre "Drama", on constate que la suite des films recommandés ont beaucoup

de similitudes communes d'ailleurs même les saisons suivante du film sont parmi les recommandations.

- Data-set 2 :

Recommandation pour le film :Pulp Fiction (1994)	
K-SVD = 5	
Titre Recommandé	Genres
2001 : A Space Odyssey (1968)	Adventure-Drama-Sci-Fi
Big Lebowski, The (1998)	Comedy-Crime
Fight Club (1999)	Action-Crime-Drama-Thriller
Léon : The Professional (a.k.a. The Professional) (Léon) (1994)	Action-Crime-Drama-Thriller
Pulp Fiction (1994)	Comedy-Crime-Drama-Thriller

TABLE 3.9 – Recommandation fournit par le SVD

Les résultats du tableau 3.9 présente un aperçu sur les films recommandés en appliquant le SVD mais cette fois si pour le DS02. On lisant le tableau on constate que tous les films recommandés sont des films des années 90 comme le film d'entrée sans oublier que les genres des films recommandés ressemblent beaucoup au film de départ qui est du genre "Comedy-Crime-Drama-Thriller".

NMF

Dans cette partie nous construisons une fonction 'syst-recommendation' en suivant les mêmes étapes pour les deux DS :

(a) Création d'une matrice qui contient les utilisateurs en colonnes et les films en lignes et pour les valeurs nous avons mis des statistiques comme l'évaluation des utilisateurs.

(b) Création d'une fonction de recommandation en appelant l'algorithme NMF depuis sklearn.decomposition.

(c) Normaliser la matrice générée dans le but de réduire les cas de doublant.

userId	1	2	3	4	5	6	7	8	9	10	...	662	663	664	665	666	667	668	669	670	671
title																					
!Women Art Revolution	0.0	0	0.0	0	0.0	0.0	0	0.0	0	0	...	0	0.0	0.0	0	0	0	0	0	0	0.0
'Gator Bait	0.0	0	0.0	0	0.0	0.0	0	0.0	0	0	...	0	0.0	0.0	0	0	0	0	0	0	0.0
...And God Created Woman	0.0	0	0.0	0	0.0	0.0	0	0.0	0	0	...	0	0.0	0.0	0	0	0	0	0	0	0.0
00 Schneider - Jagd auf Nihil Baxter	0.0	0	0.0	0	0.0	0.0	0	0.0	0	0	...	0	0.0	0.0	0	0	0	0	0	0	0.0
10 Items or Less	0.0	0	0.0	0	0.0	0.0	0	0.0	0	0	...	0	0.0	0.0	0	0	0	0	0	0	0.0

5 rows × 671 columns

FIGURE 3.15 – Matrice utilités

Résultat de la recommandation du NMF

- Data-set 1 :

Recommandation pour le film : Rocky III	
K-NMF = 4	
Titre Recommandé	Genres
Rocky III	Drama
Blood Diamond	Action-Crime-Drama-Thriller
The Driver	Drama-Thriller-Action
Sweet Sixteen	Horror

TABLE 3.10 – Recommandation fournit par le NMF

L'application de l'algorithme NMF pour le DS-1 retourne les résultats du tableau 3.10. ce dernier nous expose les recommandations obtenues pour le film "Rocky III". En observant les résultats du tableau on peut constater que les 3 premiers films recommandés sont du même genre que le film "Rocky III" qui est du genre "Drama".

-Data-set 2 :

Recommandation pour le film :Pulp Fiction (1994)	
K-NMF=6	
Titre Recommandé	Genres
Pulp Fiction (1994)	Comedy-Crime-Drama-Thriller
Schindler's List (1993)	Drama-War
Silence of the Lambs, The (1991)	Crime-Horror-Thriller
Usual Suspects, The (1995)	Crime-Mystery-Thriller
Shawshank Redemption, The (1994)	Crime-Drama
Venom (1982)	Horror-Thriller

TABLE 3.11 – Recommandation fournit par le NMF

Le tableau 3.11 présente également les résultats de l'application de l'algorithme NMF pour le DS02. L'observation du tableau nous montre que les genres et l'année de sorties des films recommandés sont quasiment les même que le film "Pulp Fiction (1994)" sachant que notre film de départ est un film des année "90" et du genres "Comedy-Crime-Drama-Thriller" .

2. Filtrage basée sur le contenu TF-IDF :

   Dans cette approche nous avons utilisé les concepts de fréquence de terme (TF) et de fréquence de document inverse (IDF) qui sont des statistique qui montrent l'importance de chaque mot spécifique dans un document, ils sont utilisés dans le systèmes de recommandation ainsi que dans le filtrage basés sur le contenu. L'importance globale de chaque mot pour les documents dans lesquels ils apparaissent est égale à TF * IDF. Dans notre cas les documents représentent la description des films de notre deuxième data-set. Cette technique a été appliqué pour le TF-IDF seul et en intégrant aux TF-IDF les deux algorithmes SVD et NMF afin d'améliorer les résultats ce recommandation.

   TF-IDF

   Voici les étapes qui permet de réaliser cette phase :

   (a) Entrer la description du films.

(b) Calculer la matrice TF-IDF pour chaque sac de mots du film.

(c) Calculer la similarité cosinus.

(d) Filtrage des films par ordre décroissant selon les scores de similarité (de plus grand au plus petit).

(e) Créer une fonction et recommander les films.

Résultats des recommandation TF-IDF seul

Recommandation pour le film : Pulp Fiction (1994)		
TF-IDF seul		
Titre Recommandé	Genres	Similarité
Big Lebowski, The (1998)	Comedy-Crime	0.317
Reservoir Dogs (1992)	Crime-Mystery-Thriller	0.294
City of God (Cidade de Deus) (2002)	Action-Adventure-Crime-Drama-Thriller	0.232
Kiss Kiss Bang Bang (2005)	Comedy-Crime-Mystery-Thriller	0.215

TABLE 3.12 – Recommandation TF-IDF seul

En observant les résultats du tableau (3.12) on peut constater que les valeurs de similarité entre le film de départ et les films recommandés par la technique du TF-IDF seul sont vraiment basse malgré la ressemblance en niveaux du genres.

TF-IDF avec le SVD :

Pour réaliser cette phase nous avons suivi les étapes suivantes :

(a) Calculer la matrice TF-IDF pour chaque description du film.

(b) Faire appel à SVD qui va réduire la dimension de la matrice TF-IDF :

- Une qui exprime l'importance des concepts par rapport aux autres sac de mots des films.
- Une qui exprime les concepts par rapport aux termes (mots) importants.
- Une troisième qui donne l'importance de chaque concept.

(c) Appel à la mesure de similarité qui va trouver les films similaires par rapport aux sac des mots des films .

(d) Créer une fonction et récupérer les films recommandé.

Résultats des recommandation TF-IDF avec le SVD

Recommandation pour le film : Pulp Fiction (1994)		
TF-IDF avec SVD		
Titre Recommandé	Genres	Similarité
Patton (1970)	Drama-War	0.929
Wild at Heart (1990)	Crime-Drama-Mystery-Romance-Thriller	0.924
Babe (1995)	Children-Drama	0.921
Billy Madison (1995)	Comedy	0.916
Pretty Woman (1990)	Comedy-Romance	0.866

TABLE 3.13 – Recommandation TF-IDF avec SVD

En premier lieu, pour la méthode utilisée TF-IDF avec le SVD on constate d'après le tableau 3.13 que les valeurs de similarité sont assez élevées et le genre du film de départ (Comedy-Crime-Drama-Thriller) ressort dans tout les films. Par conséquent, nous pouvons conclure que cette méthode est très pertinente.

TF-IDF avec le NMF :

L'intégration du NMF dans le TF-IDF a pour but de décrire la variété des tags (description des films), la réalisation de cette recommandation se fait en suivant les étapes ci-dessous :

(a) Prendre les tags (description des films).

(b) créer le TF-IDF qui sera une matrice de M lignes et N colonne où M est le nombre de tag et N est le nombre mots.

(c) intégrer la mesure de similarité.

(d) Créer une fonction qui retourne la liste des films recommandés.

Résultats des recommandation TF-IDF avec le NMF

Recommandation pour le film : Pulp Fiction (1994)		
TF-IDF avec NMF		
Titre Recommandé	Genres	Similarité
Accused, The (1988)	Drama	0.997
Bonnie and Clyde (1967)	Crime-Drama	0.997
Down with Love (2003)	Comedy-Romance	0.997
Auntie Mame (1958)	Comedy-Drama	0.991

TABLE 3.14 – Recommandation TF-IDF avec NMF

Pour la méthode utilisée avec le NMF, d'après le tableau 3.14 on constate que les valeurs de similarité des films recommandés dépassent les 0.90 ce qui signifie que les genres de ces films sont presque les mêmes que le genre du film de départ sachant que le film de départ est du genre "Comedy-Crime-Drama-Thriller".

Discussion sur les différentes recommandations obtenus

1. Approche du système prédictifs

   Malgré la présence d'un très grand nombre de films dans notre DS, les recommandations obtenues par les deux algorithmes pour l'utilisateur 66 semblent très intéressantes car en comparant l'historique du genres de films visionnée par cet utilisateur on va trouver que le genre de films recommandé sont quasiment similaires à ses préférences.

2. Approche de Recommandation

   Les recommandations basées sur le filtrage collaboratif item-item semblent relativement pertinentes. Le genre de films recommandés par les trois algorithmes pour le premier data-set semblent fortement liés au genre du film de départ mais les titres recommandés par le KNN sont carrément différents comparant les deux algorithmes SVD et NMF, par conséquent les résultats de la recommandation du SVD et le NMF ont des titres communs. Dans le deuxième Data-Set on remarque que les genres recommandés semblent fortement liés au film du départ et aussi la date de sortie du film d'entrée semble influencer de manière très importante les recommandations, ce qu'on peut comprendre à partir de ces résultats que les utilisateurs apprécient les films de la mêmes époques. Sur toutes ces observations on peut conclure que notre

système de recommandation est performant pour les deux DS.

Pour le filtrage basé sur le contenu, en raison de la baisse du taux de similarité cette technique reste pauvre. Pour améliorer les résultats de similarités de cette technique nous avons ajouter les deux algorithmes (SVD et NMF).

3.4.5 Approche hybride (filtrage collaboratif + filtrage basé contenu)

Un système de recommandation est dit hybride lorsqu'il est construit en combinant deux ou plusieurs techniques de recommandation différentes. Nous nous focalisons sur l'agrégation de la recommandations collaboratives avec la recommandation basées sur le contenu pour plusieurs raisons, l'une des raisons principales est de traiter les insuffisances de chaque technique et profiter de leurs points forts. Par exemple le FC ne peut recommander un film que s'il a été évalué par un certain nombre d'utilisateurs et le filtrage contenu nécessite la description des films, la combinaison entre eux va déterminer les films les plus proches (similaires) aux autres films en appliquant un filtrage sur le contenu, puis appliquer un filtrage collaboratif en se basant sur la qualité des films à partir des évaluations des utilisateurs ce type d'hybridation combine les deux points fort de ces deux types de filtrage.[63]

La figure 3.16 détaille l'approche hybride :

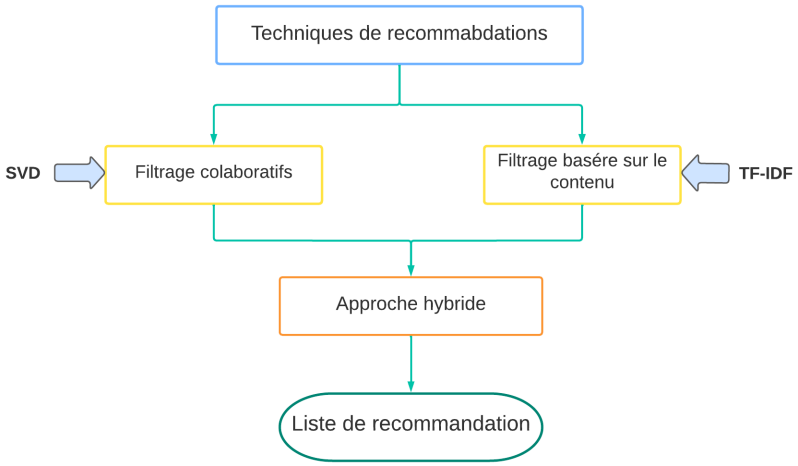

FIGURE 3.16 – Approche hybride

Résultats de l'approche hybride

L'approche que nous avons proposé dans cette partie consiste à combiner les deux type de filtrage afin de comparer et de voir quelle est la meilleur technique qui répond aux besoin des utilisateurs. Donc nous avons effectuer d'une manière indépendante les deux techniques, en commençant par construire notre SR basée sur le contenu en utilisant le TF-IDF ensuite nous avons sélectionné le film "Toy Story (1995)" et on a obtenu des recommandation a ce film à l'aide de "cosine_ similarity". Par la suite, nous avons appliqué le filtrage collaboratifs en utilisant le SVD en sélectionnant le même film "Toy Story (1995)" et on a obtenu également des recommandation. Enfin, nous avons combiner les des deux type de filtrage et voici les résultats obtenu :

	content	collaborative	hybrid
Crimson Tide (1995)	4.269133e-01	0.999419	0.713166
American History X (1998)	5.041704e-01	0.749700	0.626935
Peacemaker, The (1997)	1.705880e-01	0.986360	0.578474
Friday the 13th Part 2 (1981)	2.322706e-01	0.902592	0.567431
Silence of the Lambs, The (1991)	1.709359e-01	0.931091	0.551014

FIGURE 3.17 – Résultats de l'approche hybride

Discussion

La figure 3.17 montre les résultats de la combinaison des deux techniques de recommandation collaborative et basée content qui est l'approche hybride ainsi que les valeurs de la

similarité cosinus qui sont une quantité numérique indiquant la similarité entre le film de départ et le film recommandé. Globalement, en analysant toutes les valeurs de similarité entre le film de départ et les films recommandés de la figure on constate que les meilleures valeurs de similarité sont fournit par le filtrage collaboratif comparant l'approche hybride.

3.4.6 Approche sensible au contexte

Solution proposée

1. **Schéma de la solution proposée**

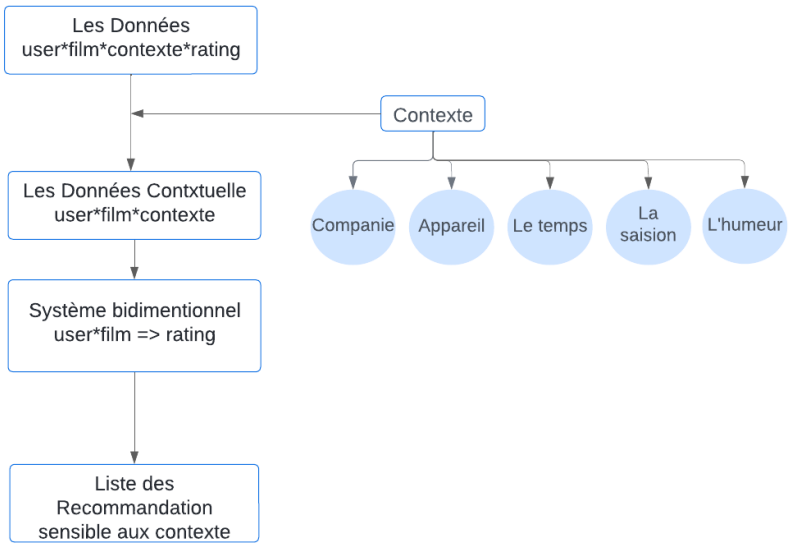

FIGURE 3.18 – Structure du système contextuel

2. **Data-Set utilisé**

   La recommandation dans cette partie de notre projet sera basée sur un jeux de donnée qui contient des informations contextuels. Ces informations ne sont pas disponibles sur la collection de référence Movielens ni sur les données publics.

   — Description du DS :

   Le jeux de donnée qu'on a utilisé "ContextualRecommendationFormResult.csv" contient des informations sur l'utilisateur plus précisément il contient des informations sur le genre de films qu'il désire regarder en fonction d'un contexte précis (l'humeur de l'utilisateur, la compagnie, l'appareil utilisé, etc.)

— Extrait du Data-set :

La figure 3.19 nous montre un extrait du DS du contexte :

	Timestamp	Name	Genre Preference in Morning -: (Choose all that apply)	Genre Preference during Afternoon -: (Choose all that apply)	Genre Preference in Evening -: (Choose all that apply)
0	2020/10/17 12:28:01 AM GMT+5:30	Raj Aryan Sharma	Comedy	Drama;Sci-Fi	Romance
1	2020/10/17 12:36:16 AM GMT+5:30	Punit Jain	Romance;Comedy;Drama	Comedy;Crime And Thriller;Documentary	Romance;Comedy;Drama;Sci-Fi
2	2020/10/17 12:48:16 AM GMT+5:30	Rajneesh	Comedy;Sci-Fi	Romance;Action	Drama

FIGURE 3.19 – Extrait de notre DS

3. **EDA**

Après avoir décrit et présenté notre DS nous allons l'explorer afin de voir son contenu.

Le contexte de compagnie

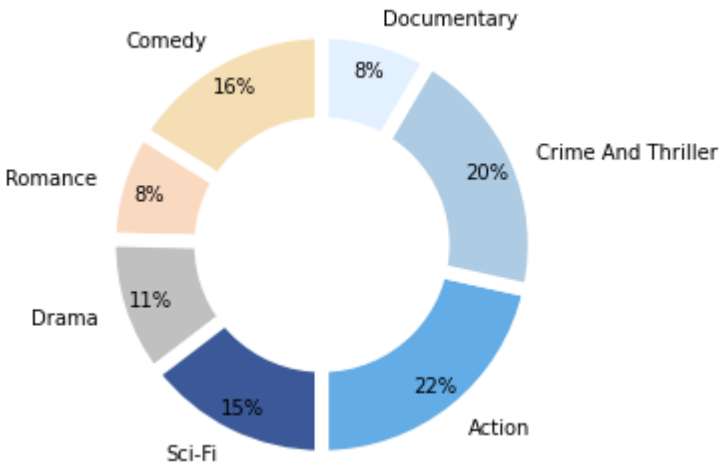

FIGURE 3.20 – Préférence de l'utilisateur en compagnie avec ses amis

Le schéma de la figure 3.20 montre le genre de films que l'utilisateur regarde en compagnie avec ses amis. On constate qu'il aime bien regarder le genre "Crime et Thriller" avec ses amies comparant les autres catégories.

Le contexte de l'appareil utilisé

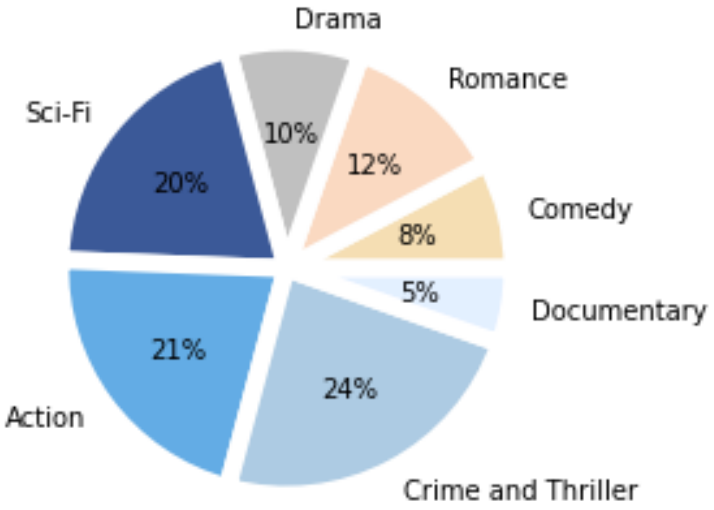

FIGURE 3.21 – Préférence de l'utilisateur en utilisant le téléphone portable

Le cercle relativiste de la figure 3.21 montre le genre de films que l'utilisateur préfère en utilisant son téléphone portable, on remarque qu'avec son smart-phone il préfère visionner le genre "Crime et Thriller".

4. **Résultats des Recommandations**

- Résultats de la Recommandation 2D :

	Title	Prediction
933	Delicatessen	4.515313
2061	The Matrix	4.423896
5362	The Day of the Jackal	4.221012
522	Terminator 2: Judgment Day	4.218496
7418	The Book of Eli	4.184696
1528	Labyrinth	4.128720
6919	Inside	4.021620
6922	Forgetting Sarah Marshall	3.989316
557	Jack & Sarah	3.983613
8308	Stories We Tell	3.956464

FIGURE 3.22 – Résultats de la recommandation Bidimensionnelle

La figure 3.22 représente les résultats de la recommandation avec prédiction(2D). Ces résultats sont obtenu en appliquant l'approche hybride (Filtrage Collaboratif + Contenu) sur notre Data-set-1. Le filtrage collaboratif est conçu à l'aide de l'algorithme SVD et le filtrage basé sur le contenu est conçu en utilisant la technique TF-IDF (plus de détails dans la partie 3.4.5).

- Résultats de la Recommandation contextuelle :

	Title	Prediction
0	Border	4.250000
1	Uri:The Surgical Strike	4.250000
2	Delicatessen	3.765313
3	The Matrix	3.673896
4	The Day of the Jackal	3.471012
5	Terminator 2: Judgment Day	3.468496
6	The Book of Eli	3.434696
7	Labyrinth	3.378720
8	Inside	3.271620
9	Forgetting Sarah Marshall	3.239316
10	Jack & Sarah	3.233613
11	Stories We Tell	3.206464

FIGURE 3.23 – Résultats de la recommandation Bidimensionnelle

La figure 3.23 est une représentation des résultats obtenu par la recommandation avec prédiction en intégrant le contexte. Pour avoir ces résultats nous avons utilisé

les colonnes du jeux de données "ContextualRecommendationFormResult.csv". Le but de notre système est de recommander à l'utilisateur des films qu'il préfère dans des contextes bien précis à titre d'exemple (Regarder un film avec un ami dans une tablette, Regarder un film en famille).

5. **Discussion**

   Après avoir obtenu les résultats de la recommandation avec prédiction des deux systèmes (2D et contextuels) pour l'utilisateur 16 nous avons analysé chaque résultats nous pouvons conclure les points suivants :

   - On remarque que les titres recommandés du système 2D sont différents par rapport au système contextuels.

   - Les notes prédites pour les mêmes titres par les deux systèmes sont différentes.

   - L'estimation de la note prédite pour le système 2D répond mieux aux attentes de notre utilisateur sachant que l'utilisateur préfère des films recommandés avec une estimation des votes [4] prédites plus de 4.

3.5 Étude comparative des travaux existants avec nos approches proposées

Le tableau 3.15 montre les travaux connexes ainsi notre approches proposées :

3.6 Conclusion

Dans cette partie, nous avons présenté la méthodologie utilisée, tout d'abord nous avons définit les différents data-sets ainsi que leur EDA et leur Pré-processing, ensuite nous avons détaillé de manière précise notre approche afin de bien expliquer et de justifier les solutions que nous avons proposé. Le chapitre suivant sera consacré à présenter les outils utilisés ainsi que les résultats et l'évaluation de notre étude.

4. Notre clé de prédiction des notes est de : 5 étoiles "Adorer",4 étoiles "apprécier", 3 étoiles "aimer", 2 étoiles "détester", 1 étoile "haïr"

Critère de comparaison / Travaux connexes	Algorithmes utilisés	Types de validation slpit/cross	Métriques d'évaluations	Systèmes réalisés
Pawel Herman (2020)[64]	KNN Baseline	Split validation	RMSE MAE	Prédition de note (rating)
Zahabiya Mhowwala et all (2020)[65]	Random Forest XGBoost	Cross validation	RMSE MAE MSE	Prédiction de note (rating)
Muhammad Sanwal et all (2021)[66]	SVR Random Forest Matrix Factorization Artificial Neural Network	Split validation	MAE MSE	SR hybride et prédiction de note (rating)
Pirunthavi Sivakumaret et all (2020)[67]	Random Forest Naïve Bayers	Split validation	Mean values	Prédiction de note (rating)
Marwa Hussien Mohamed et all (2020)[68]	Basic CF SVD with clustering Association rule with clustering	Split validation	Precision Recall F-mesure	SR
Les approches proposées				
Un système prédictif (prédiction de note"rating")	KNN SVD NMF KNNBasic	Split et Cross validation	Precision Recall F-mesure P@k R@K	
Un système de recommandation sans prediction	KNN SVD NMF		RMSE MAE MSE	
Un système de recommandation avec prediction	SVD NMF		Anova	
Un système hybride	SVD la technique TF-IDF		NDCG	
Un sytème contextuel	SVD la technique TF-IDF		la complexité	

TABLE 3.15 – Tableau comparatif

Chapitre 4

Implémentation et Evaluation

4.1 Introduction

Dans ce chapitre nous allons présenter les outils de développement ainsi que les bibliothèques utilisées, ensuite nous allons décrire les différentes métriques d'évaluations que nous avons appliqué sur les algorithmes et pour conclure ce chapitre, nous allons procéder à une discussion sur les résultats obtenus.

4.2 Outils d'implémentation

Dans cette section, nous allons présenter les outils utilisés dans notre implémentation.

4.2.1 Présentation du Language de programmation

Python est l'un des langages de programmation open source. C'est un langage à la fois simple et puissant, il permet d'écrire des scripts de haut niveau, portables et dynamiques, et grâce à ses nombreuses librairies, on peut travailler sur des projets ambitieux.[64]

Caractéristique du langage

Nous avons remarqué ces dernières années que les développeurs préfèrent le langage python comparant aux autres langages, cela grâce à ses différents avantages et caractéristiques. On peut citer les points suivants : [65]

- C'est un langage facile à apprendre et son code est plus lisible, il est donc plus facile à

maintenir.[65]

- La richesse de ses librairies.[65]

- Création de code complexe en peu de lignes.[65]

- Une très grande variété d'applications.[65]

Les bibliothèques utilisées

Si Python s'est imposé comme le meilleur langage de programmation pour développer des algorithmes d'apprentissage automatique, c'est grâce à ses différentes bibliothèques de science des données. Voici les bibliothèques utilisées dans notre étude :[64]

Numpy : NumPy est le package fondamental pour le calcul scientifique en Python. Il s'agit d'une bibliothèque Python conçue pour manipuler des matrices ou des tableaux multidimensionnels, la bibliothèque peut effectuer des opérations rapides sur des tableaux, y compris des fonctions mathématiques qui opèrent sur ces tableaux.[66]

Pandas : Il s'agit d'une bibliothèque de langage de programmation Python entièrement dédiée à la science des données. Il facilite la manipulation des données que vous souhaitez analyser et offre de nombreuses fonctionnalités natives très utiles, telles que la création de grandes trames de données à partir de ces sources et la génération de graphiques basés sur les résultats de l'analyse.[67]

Scikit-Learn : Scikit-learn est une bibliothèque Python pour l'apprentissage automatique, utile pour les algorithmes de classification, de régression ou de clustering. Cette bibliothèque d'apprentissage automatique pour Python complète d'autres bibliothèques comme NumPy.[64]

Matplotlib : Matplotlib est une bibliothèque complète pour créer des visualisations statiques, animées et interactives en Python. Il offre une grande variété de graphiques, en particulier des graphiques de haute qualité. Ces graphiques peuvent également être enregistrés dans des for-

mats raster tels que PNG, JPEG.[68]

Surprise : Le nom Surprise signifie "Simple Python Recommender Engine", et c'est un scikit Python pour créer et analyser des systèmes de recommandation qui traitent des données de notation explicites, et il fournit également des outils pour évaluer, analyser et comparer des algorithmes de performance.[69]

4.2.2 Environnement de développement

Dans ce qui suit nous allons présenter l'environnement du développement que nous avons utilisé pour réaliser notre travail.

Google Colab : Google Colab ou Colaboratory est un service cloud fourni par Google (gratuitement), basé sur Jupyter Notebook et destiné à la formation et à la recherche en machine learning. La plateforme permet de former des modèles de machine learning directement dans le cloud.[70]

Jupyter Notebook : Jupyter Notebook est une application Web open source permettant de créer et de partager des documents contenant du code (qui peut être exécuté directement dans le document), des équations, des images et du texte. Avec cette application, le traitement des données, la modélisation statistique, la visualisation des données et l'apprentissage automatique sont possibles.[70]

4.3 Métriques d'évaluation

Lors de la construction d'un modèle d'apprentissage automatique, la première chose qui vient à l'esprit est de savoir comment construire un modèle précis et bien l'adapter.

4.4 Résultats et discussion

I. **Tests Statistique**

Nous avons appliqué le test Anova sur la colonne "rating" du fichier "ratings.csv" des deux DS en divisant la note (rating) en trois catégories (mauvais, moyen et excellent) ensuite nous avons appliqué notre test d'Anova et le tableau suivant montre les résultats obtenus en appliquant le test statistique Anova :

Data-Set	f-stat	p-value
DS1	101.86	$6.40 * 10^{-45}$
DS2	144.18	$2.96 * 10^{-63}$

TABLE 4.1 – Résultats de l'Anova pour les DS

L'application du test statistique Anova nous a indiqué dans le tableau 3.1 que les valeurs des p-value sont inférieures aux seuils (seuil=0.05) pour les deux Data-Sets, ce qui signifie que les différences entre certaines notes sont statistiquement significatives.

II. **NDCG**

Data-Set	KNN	SVD	NMF	KNNBasic
DS1	0.986	0.965	0.962	0.964
DS2	0.979	0.959	0.956	0.962

TABLE 4.2 – Résultats de la NDCG pour les DS

Nous observons d'après le tableau 4.2 pour le DS-1 que les valeurs du NDCG pour les quatre modèles se varient entre [0.96,0.98] ce qui signifie que nos modèle ont une grande capacité de classer les films les plus pertinents en premiers.
Pour le DS-2, nous remarquons également que les valeurs du NDCG sont élevées ce qui confirme la pertinence de nos modèles au niveau du classement.

III. **Split Validation**

Voici les résultats obtenus en utilisant le Split Validation :

- Métriques d'évaluation du KNN

-Data-set 1 :

Le schéma 4.1 suivant nous montre les résultats des métriques d'évaluation du KNN.

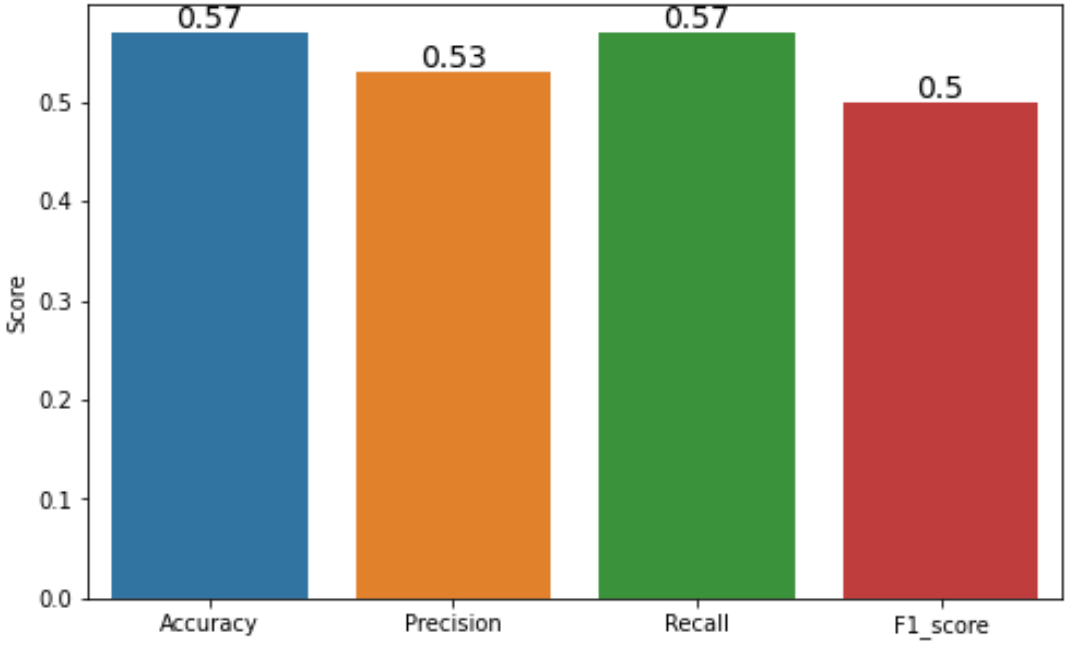

FIGURE 4.1 – Les métriques d'évaluation du DS-1

Pour le DS-1, nous constatons que la valeur de l'accuracy est de 0.57 ce qui signifie que nos prédictions sont bonnes à 57% ce qui est acceptable, pour la précision nous avons une valeur de 0.53 ce qui veut dire que 53% des résultats sont des résultats positifs réels, en ce qui concerne le *recall* nous avons une valeur de 0.57 ce qui signifie que notre modèle à détecter 57% d'échantillons positifs, la combinaison entre la *précision* et le *recall* nous a donnée un pourcentage de 50%.

-Data-set 2 :

Le schéma 4.2 exposé ci-dessous montre quelquesrecall résultats des métriques appliquées sur l'algorithme KNN pour le DS-2.

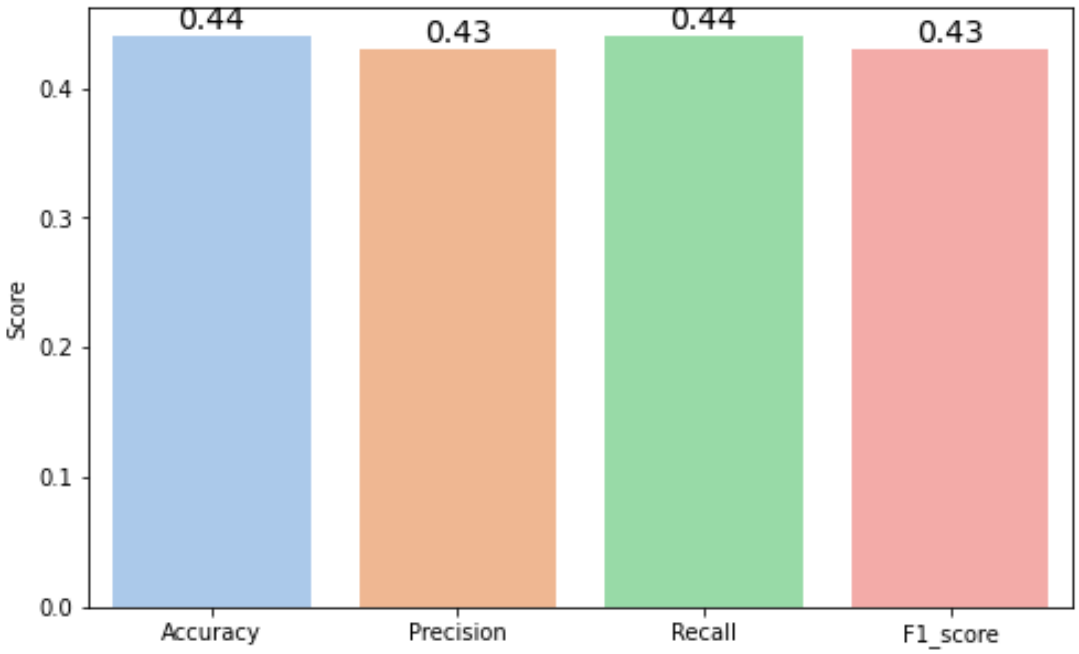

FIGURE 4.2 – Les métriques d'évaluation du DS-2

Nous remarquons que les résultats de l'accuracy, precision, recall et F1-score sont entre 0.43 et 0.44 ce qui est assez satisfaisant.

- Résultats des Mesures d'évaluation :

  - Data-set 1 :

  Le graphe suivant 4.3 représente une comparaison des résultats des performances des trois algorithmes utilisés dans le premier Data-Set en appliquant le mécanisme du Train-Test Split :

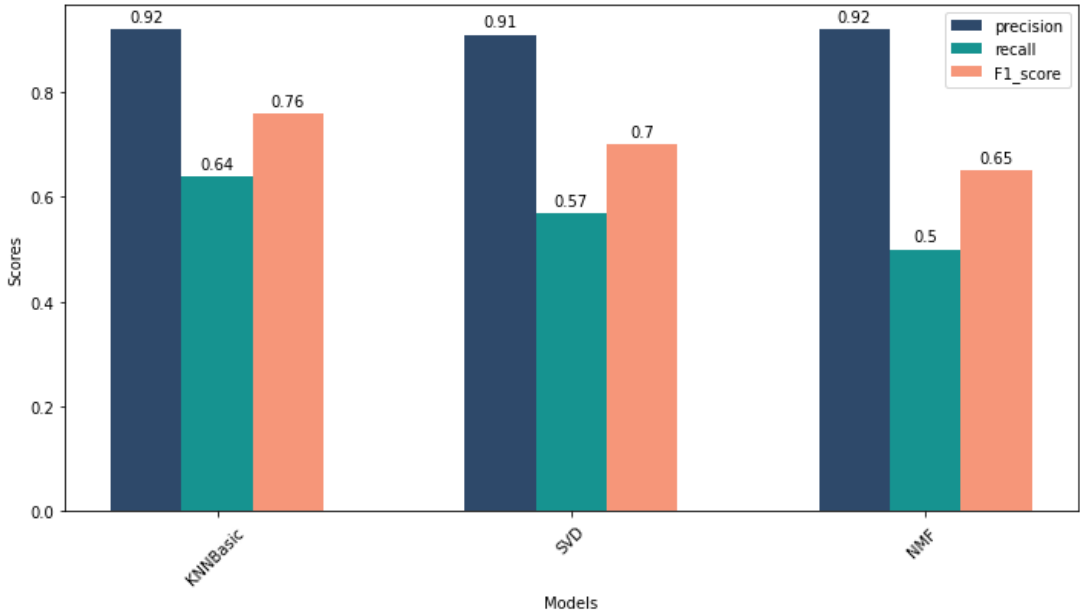

FIGURE 4.3 – Mesures d'évaluation du Data-Set 1

Pour évaluer nos algorithmes, le calcul des mesures d'évaluation est une phase très importante. La figure ci-dessus nous montre les résultats d'évaluation des trois algorithmes utilisés. Nous remarquons que le NMF et le KNNBasic nous donne la même précision avec une valeur de 0.92 et le SVD un peu moins avec une valeur de 0.91 ce qui signifie que 92% des résultats du KNN basés sur l'utilisateur et NMF et 91% des résultats du SVD sont des résultats positifs réels. En ce qui concerne le Recall et le F1-score, nous observons clairement que le meilleur taux est fournit par le KNN basé sur l'utilisateur, mais globalement on peut dire que les performances des trois algorithmes sont quasiment similaires.

-Data-set 2 :

Les résultats du graphe 4.4 sont fournit par le DS-2. Nous avons obtenu des résultats de performance différents par rapport aux résultats des performances du DS-1 :

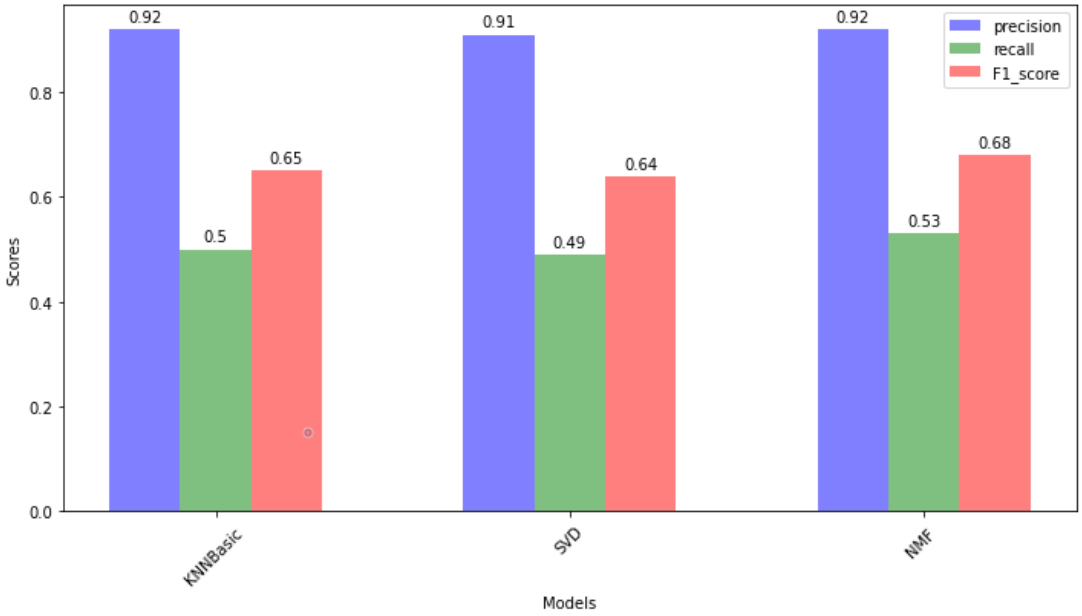

FIGURE 4.4 – Mesures d'évaluation du Data-Set 2

Nous constatons qu'on observant le graphe que les valeurs moyennes de la précision, recall et F-mesure du KNNBasic, SVD et NMF sont respectivement(0.92, 0.91, 0.92), (0.5, 0.49, 0.53), (0.65, 0.64, 0.68), par conséquent on peut constater que les deux algorithmes qui fournissent le plus grand nombre de valeurs positives réelles sont KNNBasic et NMF comparant au SVD. Pour la détection des échantillons positifs, la meilleure valeur est fournit par le NMF. Globalement en analysant le graphe on peut constater que l'algorithme dominant est le NMF.

Discussion

Les résultats de la mise en œuvre montrent que les trois algorithmes(KNNBasic, SVD, NMF) ont fournit des précisions qui dépassent les 90% pour les deux ensembles de données donc nous pouvons conclure que les résultats des trois algorithme sont performants.

- Résultat des métriques d'erreur :

  (a) KNN :

  Le tableau 4.3 expose les résultats des métriques d'erreur de l'algorithme KNN :

KNN		
Data-Set	RMSE	MAE
1	0.77	0.48
2	1.11	0.74

TABLE 4.3 – Métriques d'erreur du KNN

Nous remarquons dans le tableau précédent que les valeurs du RMSE et MAE du DS-1 sont respectivement 0.77 et 0.48, ce qu'on peut tirer de ces résultats c'est que notre modèle est bien adapté à notre ensemble de données. En ce qui concerne le DS-2, les valeurs des métriques d'erreur sont assez élevées ce qui confirme que le KNN n'est pas bien adapté au DS-2.

(b) Les modèles SVD, NMF et KNN basé sur l'utilisateur :

- Data-Set 1 :

La figure 4.5 représente un graphe de comparaison des algorithmes SVD, NMF et KNNBasic selon la valeur de RMSE et MAE pour le DS-1.

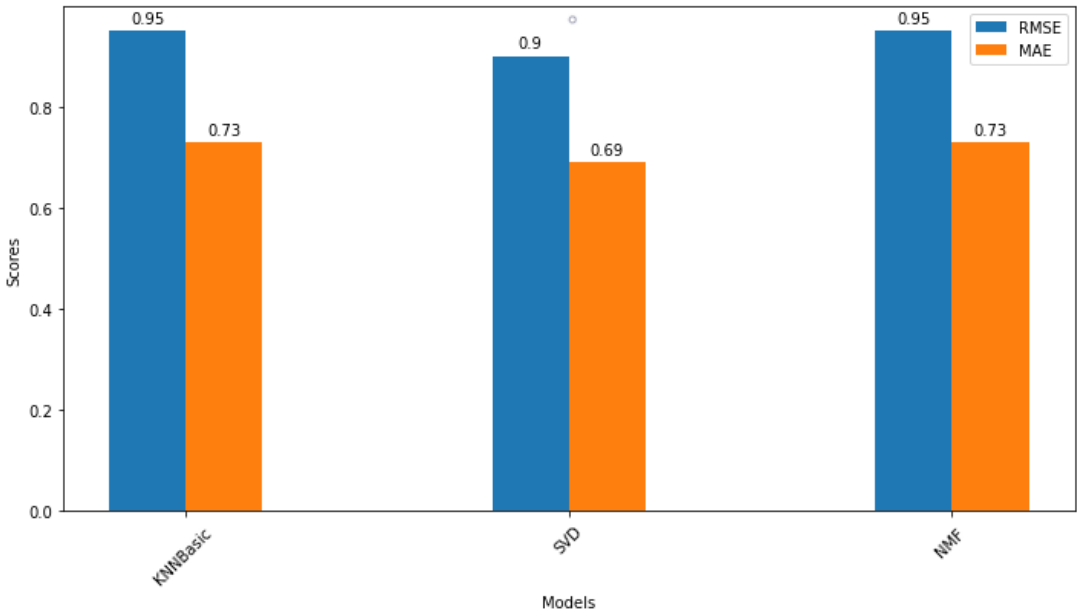

FIGURE 4.5 – Métriques d'erreurs du Data-Set 1

La figure ci-dessus montre le test d'erreur de chaque algorithme étudié pour le DS-1, le MAE et le RMSE sont des scores orientés négativement donc les valeurs inférieures sont les meilleures. La plus petite valeur du RMSE et MAE est renvoyée par le SVD en le comparant aux KNNBasic et NMF ce qui explique que c'est l'algorithme le mieux adapté à notre jeux de données.

- Data-Set 2 :

La figure en dessous 4.6 représente une comparaison des algorithmes SVD, NMF et KNNBasic selon la valeur de RMSE et MAE pour le DS-2.

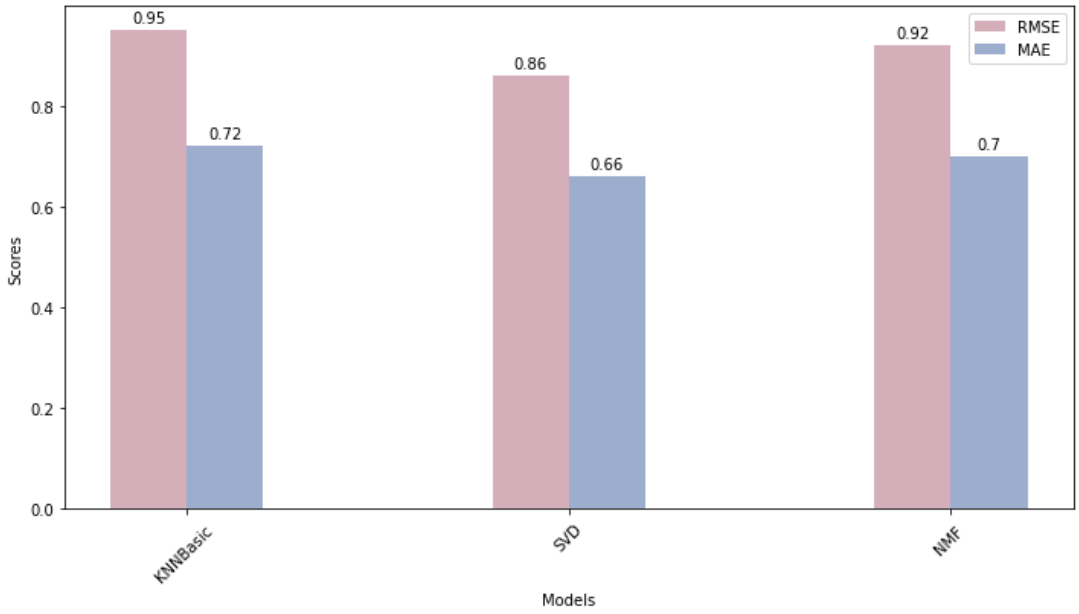

FIGURE 4.6 – Métriques d'erreurs du Data-Set 2

Pour Le DS-2, en analysant la figure, nous remarquons que le taux du RMSE et MAE obtenu par le KNNBasic est 0.95 et 0.72, pour le SVD 0.96 et 0.66 et pour le NMF 0.92 et 0.7 respectivement, ce qui implique que l'algorithme le moins adapté au DS-2 est le KNNBasic.

IV. Cross Validation

Nous avons également testé les performances de nos modèles en utilisant la Cross validation.

- **Résultats et desscussion de la technique des K-fold :**

- Résultats des métriques d'erreurs du KNN :

Les tableaux 4.4 suivants résument l'approche des K-fold du Cross-Validation pour le KNN :

Data-Set 01	
K-fold	Accuracy
3	0.301
6	0.338
9	0.242

Data-Set 2	
K-fold	Accuracy
3	0.271
6	0.219
9	0.212

TABLE 4.4 – K-fold KNN

Nous avons évalué le KNN avec la mesure Accuracy sur trois paramètres différents de validation et nous avons obtenu les résultats dans le tableau ci-dessus. Nous remarquons que la meilleure valeur d'accuracy est obtenue par le sous ensemble 6 avec une valeur de 0.338 dans le DS01 ce qui montre que notre modèle est performant quand nous décomposons notre ensemble de données en 6-folds. Pour le DS02, nous remarquons que les valeurs d'accuracy sont basses pour la décomposition en 6 et 9 folds comparant la décomposition en 3 folds ce qui signifie que le meilleur sous ensemble est le 3-folds.

- Résultats des métriques d'erreurs des modèles KNNBasic, SVD et NMF :

Le tableau 4.5 résume l'approche des K-fold du Cross-Validation pour les 3 modèles d'apprentissage automatique :

RMSE Data-Set 1			
K-fold	KNNBasic	SVD	NMF
3	0.9425	0.8718	0.9517
6	0.9621	0.9039	0.9343
9	0.9625	0.9052	0.9456

MAE du Data-Set 1			
K-fold	KNNBasic	SVD	NMF
3	0.8674	0.6722	0.7363
6	0.8630	0.6953	0.7201
9	0.8660	0.6989	0.7216

TABLE 4.5 – Résultats des K-fold DS-1

Nous avons évalué chaque algorithme avec 3 paramètres : 3 validation croisée, 6 validation croisée et 9 validation croisée. En comparant les meilleures valeurs de RMSE et MAE pour le KNNBasic,SVD et NMF. Nous remarquons que le SVD fournit les meilleurs résultats comparant le KNNBasic et le NMF, il atteint une valeur du MAE à 0.6722 et du RMSE à 0.8718 pour k-fold = 3.

RMSE Data-Set 2			
K-fold	KNNBasic	SVD	NMF
3	0.9505	0.8656	0.9205
6	0.9469	0.8520	0.9194
9	0.9456	0.8730	0.9216

MAE du Data-Set 2			
K-fold	KNNBasic	SVD	NMF
3	0.7292	0.6614	0.7038
6	0.7283	0.6551	0.7031
9	0.7323	0.6706	0.7059

TABLE 4.6 – Résultats des K-fold DS-2

Pour le DS-2, nous avons utilisé aussi 3 flods : 3 validation croisée, 6 validation croisée et 9 validation croisée. Nous remarquons que le SVD fournit les meilleurs résultats en le comparant avec les deux autres algorithmes avec des valeurs de 0.8520 et 0.6551 respectivement pour le RMSE et le MAE en utilisant k-fold = 6. D'après les résultats ci-dessus, la cross validation k-fold est la meilleure méthode possible pour valider l'exactitude d'un modèle prédictif.

- Mesures d'évaluation du SVD

  - La relation entre la Précision et le Recall :

  La figure 4.7 décrit le croisement des deux mesures d'évaluation Précision@K et le Recall@K pour le DS-1.

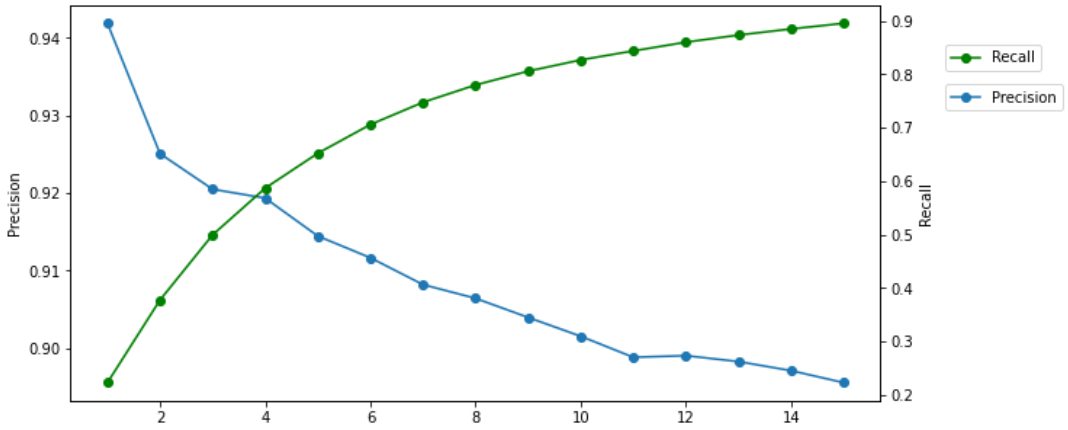

FIGURE 4.7 – Relation entre Précision@K et le Recall@K du DS-1

D'après la figure précédente, nous remarquons que la valeur optimale de k pour le DS1 est de 4 avec une P@K=0.92 ce qui représente que 92% de nos recommandations sont pertinentes et avec R@K=0.58 ce qui signifie que 58% des nombres de films totals pertinents vont apparaitre dans les 4 premières recommandations.

La figure 4.8 représente le croisement des deux mesures d'évaluation Précision@K et le Recall@K pour le DS02.

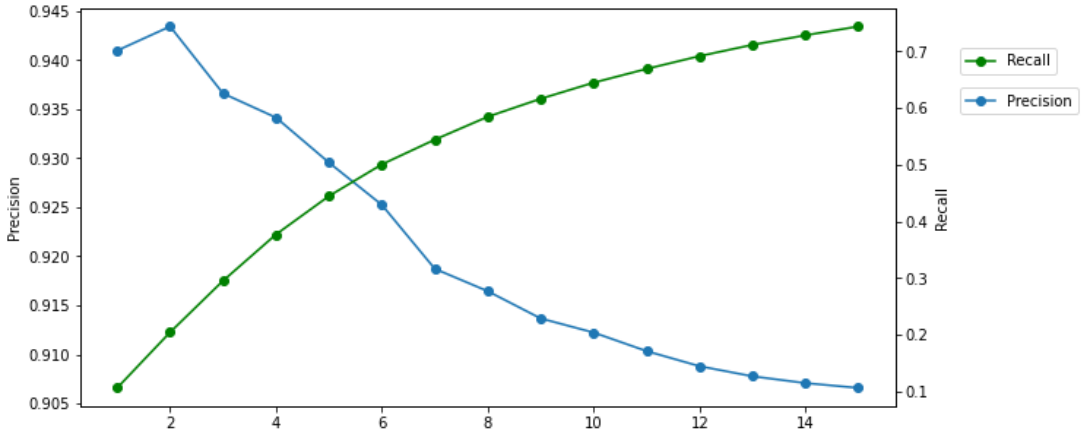

FIGURE 4.8 – Relation entre Précision@K et le Recall@K du DS-2

Pour le deuxième jeux de donnée, le croisement des deux métrique d'évaluation se fait dans le K=5 avec une P@K=0.927 et R@K=0.45 ce qui implique que les meilleurs K-recommendation sont dans k= 5.

- Mesures d'évaluation du NMF

  - La relation entre Précision@K et le Recall@K :

  La figure 4.9 nous montre les valeurs et la relation entre la Précision@K et le Recall@K en fonction du K du DS-1.

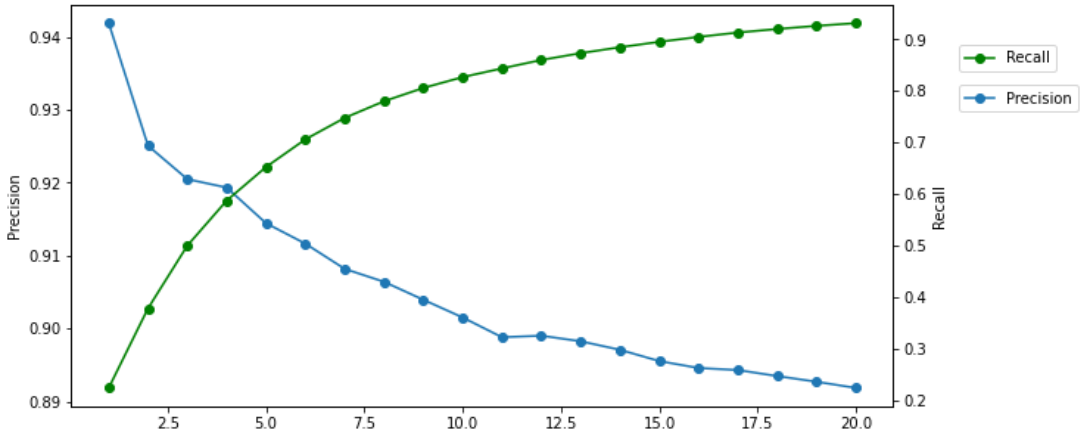

FIGURE 4.9 – Les valeurs de Précision@K et le Recall@K du DS-1

Dans le schéma précédent, nous remarquons que chaque fois que la valeur de K grandi le P@K démunie et le R@K augmente donc la détermination de leur valeur optimale sera indiqué par leur croisement ce qui signifie que les k-top recommandations seront dans le k=4.

La figure 4.10 montre les valeurs et la relation entre la Précision@K et le Recall@K en fonction du K du DS-2.

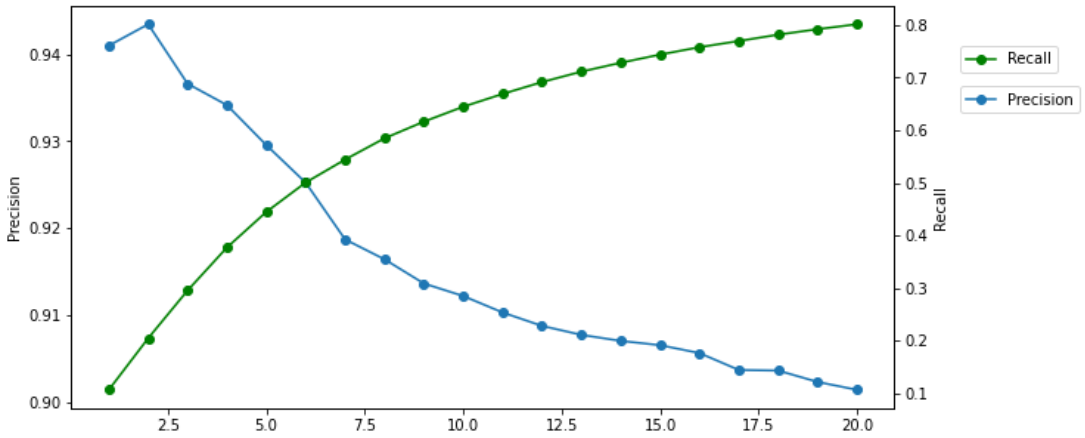

FIGURE 4.10 – Les valeurs de Précision@K et le Recall@K du DS-2

Pour le DS2, les deux graphes se croisent dans les environs k=6 ce qui montre que nos top recommandations vont être dans les 6 premiers films.

- **La complexité**

  Le tableau 4.7 montre le temps d'exécution de chaque algorithme :

Data-Set	KNN	SVD	NMF
DS-1	0.97	39.88	38.09
DS-2	0.72	57.9	74.35

TABLE 4.7 – Résultats de complexité pour les DS

Nous remarquons que la complexité entre le KNN et les deux autres algorithmes est assez différente. Nous retrouvons des valeurs qui ne dépassent pas 1 pour le KNN alors que le SVD et NMF sont élevées.

V. **Discussion des résultats finaux**

Les trois algorithmes KNNBasic, SVD et NMF nous ont donnée d'excellent résultats pour les deux méthodes (split et cross validation) pour les deux Data-Set. En comparant les résultats du Cross avec les résultats que nous avons calculés dans le Split on constate qu'il y a une différence entre les méthodes, d'après les résultats discutés auparavant La validation croisée k-fold est la meilleure méthode possible. D'un autre coté, l'analyse des performance du KNN nous montre que les résultats obtenu du

Split sont mieux que le Cross validation ce qui est assez rare mais les résultats du KNN reste assez basse en le comparant aux autres algorithmes.

4.5 Évaluation du système 2D et sensible au contexte

-Métriques d'erreurs

Le tableau 4.8 montre l'évaluation des deux systèmes en utilisant les métriques d'erreur :

Système bidimensionnel			
K-fold	RMSE	MAE	MSE
2	0.8962	0.6898	0.7916
4	0.9003	0.6861	0.7997
5	0.8973	0.6931	0.7947

Système sensible au contexte			
K-fold	RMSE	MAE	MSE
2	0.8951	0.6883	0.7785
4	0.8946	0.6875	0.8101
5	0.8878	0.6831	0.7993

TABLE 4.8 – Évaluation du système 2D et sensible au contexte

Nous avons cité dans les discussions précédentes que les métriques d'erreurs sont des métriques orientées négativement, donc en comparant les résultats des deux systèmes en utilisant 3 folds différents, nous remarquons que le système contextuel fournit les meilleurs résultats donc on peut conclure que le système contextuel est plus performant.

4.6 Discussion globale sur les prédictions et les recommandations

A) Les valeurs de prédiction des notes n'ont pas dépassé 5.0 (la note maximale qu'un utilisateur peut attribuer) ce qui signifie que notre système est performant.

B) Le SVD a tendance à donner les meilleures prédictions et recommandations dans notre études.

C) L'observation globale sur les titres recommandés par le SVD et le NMF renvoient quelques titres communs ce qui assure la performance des films recommandés par

le système.

D) Les méthodes de filtrage collaboratif nous ont permis de faire des recommandations pertinentes vu que le genre de films recommandé correspond à l'historique de l'utilisateur.

E) Notre système de recommandation semble recommande la même suite de genre d'où on garantie le bon fonctionnement du système.

F) Les résultats de l'approche hybride proposée sont assez bien .

G) Globalement, les résultats obtenus par les algorithmes utilisés sont cohérents et les recommandations et les prédictions fournit sont très satisfaisants.

H) Le système contextuel proposé est plus précis et performant comparant avec le système 2D.

4.7 Conclusion

Dans ce dernier chapitre, nous avons présenté l'environnement du travail en citant ses différentes bibliothèques. Par la suite, nous avons détaillé les résultats et les performances de chaque Data-Set ainsi que chaque type de validation et enfin, nous avons présenté une discussion globale sur notre travail.

Conclusion générale et perspectives

Les systèmes de recommandation se développent de plus en plus comme toutes les sciences. Ces systèmes là sont très bénéfiques car ils apportent une grande aide dans le quotidien des utilisateurs qui peuvent consulter du contenu personnalisé selon leurs goûts et envies. Mais la question qui se pose et revient toujours est-ce-que ces systèmes sont performants et répondent aux besoins et aux préférences des utilisateurs ?

Notre projet s'inscrit justement dans ce cadre, il s'agit de réaliser une étude de performances d'un systèmes de prédiction et de recommandation en appliquant les différents algorithmes de Machine Learning.

Le travail présenté dans ce mémoire consiste à utiliser les différentes techniques de Machine Learning. Plus précisément, nous avons implémenté, évalué et comparé quatre algorithmes, l'un est basé sur les K-plus proches voisins, les deux autres sur la factorisation matricielle (SVD et NMF) et le dernier le KNNBasic. L'évaluation de ces quatre algorithmes nous a permis de déterminer que ces algorithmes qui retournent les meilleurs résultats.

Nous avons présenté dans un premier lieu l'apprentissage artificiel et son utilisation massive dans la Data Science et nous avons présenté une vue globale sur l'analyse prédictive et les systèmes de recommandation à savoir leurs définitions, notations et leurs classifications.

Par la suite, dans le deuxième chapitre nous avons entamé le terme de la boite noire qui représente l'étude faite sur les différents algorithmes utilisés durant ce travail.

En ce qui concerne le troisième chapitre, nous avons expliqué en détail l'approche que nous avons proposé pour les deux système en implantant les algorithmes choisi sur deux différents jeux de données afin de montrer si le choix de ces algorithmes dépend du jeu de données et de sa taille.

Le dernier chapitre était complètement consacré pour l'évaluation des performances de quelques algorithmes en utilisant différentes métriques dans le but de trouver les algorithmes optimaux pour nos Data-Set.

Ce projet a fait l'objet d'une expérience intéressante, très bénéfique pour nous car nous avons enrichi nos connaissances théoriques et pratiques. Mais nous avons été confrontés à différentes difficultés, par exemple au niveau de l'implémentation des algorithmes (SVD et NMF), cependant c'était une véritable expérience et une chance pour la découverte de ce domaine.

Comme travaux futur, ce

Sa sera intéressant comme point de vue lié à ce travail d'améliorer les approches proposées en utilisant le Deep Learning.
en intégrant d'autres dimensions du contexte et utiliser le Deep Learning ensuite appliquer nos approches proposées sur d'autres domaines.M

Bibliographie

[1] Les systèmes de recommandation : une catégorisation. https ://interstices.info/les-systemes-de-recommandation categorisation/.

[2] Guillaume Saint-Cirgue. Comment fonctionne le machine Learning. *machinelearnia*, mars 2019.

[3] https ://www.coe.int/fr/web/artificial intelligence/glossary. consulté le 02/01/2022.

[4] https ://scikit-learn.org. consulté le 10/01/2022.

[5] https ://medium.com/@kenzahari/. consulté le 13/01/2022.

[6] Farnaz Ghasemi Toudeshki. Factorisation matricielle non negatuve (NMF) pour les systèmes de recommandation. *univ toulouse*, 11 mars 2020.

[7] https ://www.wikiwand.com/. Singular value decomposition. consulté le 29/05/2022.

[8] Clément Côme. Qu'est-ce que l'analyse prédictive ? *Kobia Devançond l'avenir*, 10 août 2021.

[9] Olivier Dumons. L'algorithme de netflix, un cerveau à la place du cœur. *Le monde*, 19 août 2019.

[10] https ://course.elementsofai.com/fr/6/1. consulté le 05/01/2022.

[11] DJAFRI Laouni. Analyse de données massives -Big Data- pour la prédiction. *univ de Sidi Bel Abbes*, 2020.

[12] https ://www.talend.com/fr/resources/analyse-predictive/. *consulté le 25/06/2022.*

[13] Judith Hurwitz and Daniel Kirsch. Machine Learning. *IBM Limited Edition*, https ://www.ibm.com/downloads/cas/GB8ZMQZ3 2018.

[14] Benjamin R. Systèmes de recommandation. *Olcya*, 22 septembre 2018.

[15] Systèmes de recommandation : ce que vous devez savoir. *affde*, 26 juin 2021.

[16] Idir Benouaret. Un système de recommandation contextuel et composite pour la visite personnalisée de sites culturels. *HAL archives ouvertes*, 18 avril 2018.

[17] Yasmine BELHADRI TIDIANE CISSE. Comprendre l'algorithme de recommandation de Netflix. https ://www.lesmondesnumeriques.net/wp-content/uploads/2019/02/Comprendre-lalgorithme-de-recommandation-de-Netflix-1.pdf Univ PEM.

[18] Houda Oufaida Omar Nouali . Le filtrage collaboratif et le web 2.0 vol 11, page 13 à 45 . *cairn.info*, 2008.

[19] Kilian Bourhis Bruno Canitia Assaad Kenaan, Khalid Benabdeslem. Approches hybrides pour la recommandation dans le domaine du pneumatique ? page 133 à 144. *Université Lyon 1.*

[20] Darine AMEYED. Modélisation et spécification formelle de contexte et sa prédiction dans les systèmes diffus : Une approche basée sur la logique temporelle et le modèle stochastique. *UNIV Montreal*, 8 fevrier 2017.

[21] Halima NEFZI. PROPOSITION D'UNE NOUVELLE APPROCHE DE RECOMMANDATION CONTEXTUELLE EN SE BASANT SUR LA MÉTHODE D'ANALYSE HIÉRARCHIQUE DES PROCÉDÉS (AHP). *UNIV TUNIS*, 17 fevrier 2018.

[22] Amaury l'Huillier. Modéliser la diversité au cours du temps pour comprendre le contexte de l'utilisateur dans les systèmes de recommandation. *HAL OPEN*, 10 janvier 2019.

[23] IKRAM GAGAOUA. Les systèmes de recommandation pour l'apprentissage et la formation. *domoscio*, https ://domoscio.com/fr/ 25 fevrier 2021.

[24] les challenges associés au développement des algorithmes d'exploitation des données en vie réelle Les défis posés par les algorithme. https ://gt2.ariis.fr.

[25] Tidiane CISSE et Yasmine BELHADRI. Comprendre l'algorithme de recommandation de Netflix. 2 fevrier 2019.

[26] https ://tdoct.com/2020/01/29/definition-rapide-du ml/. *Définition rapide du machine Learning (ML)*, 29 janvier 2020.

[27] Nonvikan Karl-Augustt ALAHASSA. Mon Projet en 800 mots. https ://ivado.ca/mon-projet-de-recherche-en-800-mots/ Université de Montréal.

[28] enzotripoli. Quels sont les défis de l'apprentissage automatique dans l'analyse des mégadonnées. *smartrgpd*, 23 juin 2021.

[29] Judith Hurwitz and Daniel Kirsch. Machine Learning. *IBM Limited Edition*, 2018.

[30] Zakia messaoudi. Dans l'Ia. *spiria*, 22 janvier 2020.

[31] Saidi Amaria Lakhdari Salsabil. Étude des techniques d'apprentissage semi- supervisé par regroupement. *UNIV TLEMCEN*, 11 septembre 2017.

[32] George Lawton. Apprentissage supervisé et non supervisé : les différencier et les combiner. *lemagit*, https ://www.lemagit.fr/ 14 octobre 2020.

[33] https ://dataanalyticspost.com/Lexique/apprentissage-par-renforcement/. consulté le 02/01/2022.

[34] intelligence artificielle. Apprentissage automatique (Machine Learning) – Définition, fonctionnement et secteurs d'application. *La redaction*, https ://intelligence-artificielle.com/machine-learning-definition/ 8 avril 2022.

[35] What is Machine Learning ? Hewlett Packard Enterprise. *hpe*, https ://www.hpe.com/.

[36] LAURENT. Le machine learning est une des formes de l'intelligence artificielle. *thegoodlife*, 27 JUIN 2019.

[37] Leo Breiman. ARBRE DE DECISION Page 5 à 32 . *Random forests. Machine Learning*, https ://link.springer.com/article/10.1023/A :1010933404324.

[38] Data Science Team. Gradient Boosting – Ce que vous devez savoir. *datascience*, 2020.

[39] Younes Benzaki. Tout ce que vous voulez savoir sur l'algorithme K-Means. *mrmint*, 18 Avril 2018.

[40] https ://fr.myservername.com/volume-testing-tutorial. consulté le 11/01/2022.

[41] Kassidy Kelley et Gaétan Raoul. Machine Learning : les 9 types d'algorithmes les plus pertinents en entreprise. *mrmint*, 8 juin 2020.

[42] https ://moncoachdata.com/blog/mathematiques-du-machine learning/. 6 domaines Mathématiques du Machine Learning.

[43] Kenza Harifi. Bien comprendre l'algorithme des K-plus proches voisins (Fonctionnement et implémentation sur R et Python. *medium*.

[44] Younes Benzaki. Introduction à l'algorithme K Nearst Neighbors (K-NN). *mrmint*.

[45] Pascal Fernsel Delf Lachmund Tobias Boskamp Johannes Leuschner, Maximilian Schmidt. Supervised non-negative matrix factorization methods for MALDI imaging applications. *OXFORD ACADEMIC*, 01 juin 2019.

[46] Romain Gauchon. Une nouvelle méthode de classification permettant de cibler des actions de prévention. *Actuaris Addactis group*, univ Lyon 1.

[47] Décomposition en valeurs singulières. *bibmath.net*.

[48] Patrick Luboobi. Foundations of Machine Learning : Singular Value Decomposition (SVD). *medium*, 5 fevrier 2018 https ://medium.com/the-andela-way/foundations-of-machine-learning-singular-value-decomposition-svd-162ac796c27d.

[49] Vaibhav Kumar. Singular Value Decomposition (SVD) et Its Application In Recommender System. *towardsdatascience*, https ://towardsdatascience.com/ 2022.

[50] https ://www.techno-science.net/glossaire-definition/Decomposition-en-valeurs singulieres.html. Décomposition en valeurs singulières - Définition et Explications.

[51] https ://www.oracle.com/dz/data-science/machine-learning/what-is-machine learning/. Qu'est-ce que le machine learning.

[52] Acervo Lima. ANOVA à un facteur. *acervolima*, 2022.

[53] Moussa Taifi PhD. MRR vs MAP vs NDCG : Rank-Aware Evaluation Metrics And When To Use Them. *medium*, 25 novembre 2019 https ://medium.com/swlh/rank-aware-recsys-evaluation-metrics-5191bba16832.

[54] Zahira Chouiref et Mohamed Yassine Hayi. Toward Preference and Context-Aware Hybrid Tourist Recommender System Based on Machine Learning Techniques. *iieta*, 4 Avril 2022.

[55] Charles Tremblay et Clément Côme. F1-score, la synthèse entre precision et recall. *kobia*, 17 novembre 2021.

[56] Marie-Jeanne Vieille. Mesurer la performance d'un modèle : Accuracy, recall et precision. *lovelyanalytics*.

[57] Wenshan Guo et Longya Ran Ninghua Sun, Tao Chen. Enhanced Collaborative Filtering for Personalized E-Government Recommendation. *mdpi*, 20 Decembre 2021 https ://www.mdpi.com/2076-3417/11/24/12119/pdf ?version=1639983960.

[58] JJ. MAE and RMSE — Which Metric is Better ? *medium*, 23 mars 2016.

[59] JY Baudot. *Les indicateurs d'écarts*, http ://www.jybaudot.fr/Stats/indicecarts.html.

[60] Antoine Krajnc. Comment évaluer et améliorer son modèle de Machine Learning. *microsoft*, 12 Janvier 2020.

[61] Luis Quintanilla. améliorez votre modèle ML. 2 Decembre 2021.

[62] JDN. Feature Engineering définition techniques en machine learning. *journaldunet*, 25 fevrier 2022 https ://www.journaldunet.fr/web-tech/guide-de-l-intelligence-artificielle/1501337-feature-engineering-definition/.

[63] G.and Tuzhilin Adomavicius. Toward the next generation of recommender systems : A survey of the state-of-the-art and possible extensions Page 734 à 749. *ieeexplore*, 2005 https ://ieeexplore.ieee.org/document/1423975.

[64] Bastien L. Python : tout savoir sur le principal langage Big Data et Machine Learning. *lebigdata*, 5 juin 2022.

[65] https ://python.doctor/. Apprendre le langage de programmation python.

[66] Acervo Lima. Pourquoi Numpy est-il plus rapide en Python ? *geeksforgeeks, Python*, 2022.

[67] Adriano R. Pandas : la bibliothèque Python dédiée à la Data Science. *datascientest*, 10 janvier 2022.

[68] Acervo Lima. Tracé simple en Python avec Matplotlib. *geeksforgeeks, Python*, 2022.

[69] Nicolas Hug. A python scikit for recommender systems. *surpriselib*, http ://surpriselib.com/.

[70] Henri Michel. Google Colab : Le guide Ultime. Data Science, IA, Programmation.

[71] Pawel Herman. Predicting movie ratings using KNN. *KTH VETENSKAP OCH KONST*, 8 juin 2020 http ://www.diva-portal.org/smash/get/diva2 :1464572/FULLTEXT01.pdf.

[72] Sujala D.Shetty Zahabiya Mhowwala, A.Razia Sulthana. Movie Rating Prediction using Ensemble Learning Algorithms. *IJACSA*, aout 2020.

[73] Cafer ÇALIŞKAN2 Muhammad Sanwal. A Hybrid Movie Recommender System and Rating Prediction Model. *WOAS*, juillet 2021.

[74] Kamalanathan Abishankar E.M.U.W.J.B. Ekanayake Yanusha Mehendran Piruntha vi Sivakumar, Vithusia Puvaneswaren Rajeswaren. Movie Success and Rating Prediction Using Data Mining Algorithms. *JISIT*, 2020.

[75] Mohamed Khafagy Marwa Hussien Mohamed, Mohamed Hasan Ibrahim. Two recommendation system algorithms used SVD and association rule on implicit and explicit data sets. *ResearchGate*, 13 October 2021.

Annexe A

Fichier "movies-metadata.csv" :

movies-metadata.csv	
Attributs	**Description**
Adult	Le film est pour les adulte ou non.
belongs-to-collection	Information sur les films.
budget	Cout du film.
genres	Catégorie des films.
homepage	Lien vers les films.
id	L'identifiant des films.
imdb-id	L'identifiant du imdb.
original-language	Langue originale des films.
original-title	Titre originale des films.
overview	Résumé des films.
popularity	popularité des films.
poster-path	L'URL des images d'affiches dans le film.
production-company	La maison de production.
production-countries	Le pays producteur.
release-date	La date de la réalisation des films.
revenue	Chiffres d'affaires des films.

runtime	La durée du film en minutes.
spoken-languages	Liste des langues parlées dans le film.
status	le statut du film (Released, To Be Release, Annoncé, etc).
tagline	La description des films.
title	Titres des films.
video	Indique s'il y a une vidéo présente du film avec TMDB.
vote-average	La note moyenne du film.
vote-count	Le nombre de votes des utilisateurs compté par TMDB.

TABLE A.1 – Description du fichier movies-metadata