

République Algérienne Démocratique et Populaire
Ministère de l'enseignement Supérieur et de la Recherche Scientifique
Université de Mohamed El Bachir El Ibrahimi de Bordj Bou Arréridj
Faculté des Mathématiques et d'Informatique
Département d'informatique



MEMOIRE

Présenté en vue de l'obtention du diplôme

Master en informatique

Spécialité : Technologie de l'information et de la communication

THEME

Les Systèmes De Recommandation Basés Sur La Méthode Explicable

Présenté par :

MAATOUG Marwa

CHEMALI Sifeddine

Soutenu publiquement le : 03/07/2025

Devant le jury composé de :

Président : SAIFI Abdelhalim

Examineur : BEN ABID Sonia

Encadré par : ATTIA Abdelouahab

2024/2025

Dédicace

Tous les mots ne sauraient exprimer la gratitude, l'amour, Le respect, la reconnaissance, c'est tout simplement que je dédie ce mémoire. À mes parents car ils m'ont inculqué un esprit de persévérance et qu'ils m'ont toujours poussé et motivé dans mes études. Sans eux, certainement je ne serais pas à ce niveau. Ils ont tout fait pour mon bonheur et ma réussite. Aucune dédicace ne puisse exprimer ce que je leur dois. Que dieu leur réserve la bonne santé et une longue vie. À mes frères et sœurs, qu'ils restent pour moi des exemples de persévérance, de courage et de générosité. Et à tous ceux qui me sont chers et qui ont contribué à mes encouragements.

MAATOUG MARWA

Dédicace

À nos parents, frères et sœurs et amis.

Nous exprimons notre profonde gratitude.

CHEMALLI Sifeddine

Remerciement

A l'issue de ce travail, je remercie, en premier lieu, le bon Dieu de m'avoir donné la force et le courage de le mener à terme. Je voudrai présenter mes remerciements à ma directrice de mémoire « Dr. ATTIA Abdelouahab ». Je voudrai également lui témoigner ma gratitude pour sa patience et son soutien qui m'ont été précieux afin de mener mon travail à bon port. Et pour ses précieux conseils et son aide durant toute la période du travail. Aussi je remercie messieurs les membres de jury et nos professeurs pour la qualité de l'enseignement qu'ils nous ont prodigué au cours de ces cinq années Passées au département d'informatique. Enfin je tiens, également à exprimer ma sincère reconnaissance et ma profonde gratitude à Ma famille, qui m'a toujours donné la possibilité de faire ce que je veux et qui ont toujours cru en moi. Et à tous ceux qui ont contribué de près ou de loin à la réalisation de ce mémoire.

Résumé

Ce mémoire s'intéresse à l'intégration des méthodes explicables (XAI – eXplainable Artificial Intelligence) dans les systèmes de recommandation, afin de concilier performance algorithmique et transparence des décisions. Les systèmes de recommandation, bien que largement utilisés, fonctionnent souvent comme des « boîtes noires », rendant leurs décisions difficiles à interpréter. Cette opacité peut nuire à la confiance des utilisateurs et poser des problèmes éthiques ou réglementaires.

Dans ce travail, nous avons implémenté plusieurs approches de recommandation explicables, notamment la régression linéaire avec TF-IDF, les forêts aléatoires, et les règles d'association à l'aide de l'algorithme FP-Growth. Les résultats obtenus sur un jeu de données réel (avis d'hôtels) ont permis d'analyser le compromis entre précision et interprétabilité. Nous avons également exploré l'apport de techniques comme LIME et SHAP pour expliquer les prédictions de modèles complexes.

Les conclusions mettent en évidence l'intérêt d'une IA explicable dans les systèmes de recommandation, tant pour renforcer la confiance des utilisateurs que pour répondre aux exigences de transparence dans des domaines critiques tels que la santé, la finance ou l'éducation.

Mots-clés : Système de recommandation, intelligence artificielle explicable (XAI), régression linéaire, forêt aléatoire, FP-Growth, LIME, SHAP, interprétabilité.

Abstract

This thesis explores the integration of explainable artificial intelligence (XAI) techniques into recommendation systems, with the goal of combining predictive performance and decision transparency. Although recommendation systems are widely used, they often operate as “black boxes,” making their decisions difficult to interpret. Such opacity can undermine user trust and raise ethical and legal concerns.

We implemented several explainable recommendation approaches, including linear regression with TF-IDF, random forests, and association rules using the FP-Growth algorithm. The models were tested on a real-world dataset (hotel reviews), allowing us to evaluate the trade-off between accuracy and interpretability. Additionally, we examined the use of LIME and SHAP to generate meaningful explanations for complex models.

The findings highlight the importance of explainability in recommendation systems, both to strengthen user confidence and to comply with transparency requirements in sensitive fields such as healthcare, finance, and education.

Keywords: Recommender system, explainable artificial intelligence (XAI), linear regression, random forest, FP-Growth, LIME, SHAP, interpretability.

Table des matières

Dédicace	2
Dédicace	3
Remerciement	4
Résumé	5
Abstract	6
Table des matières	7
Liste de figure	10
Liste des tableaux	12
List d abréviations.....	13
Introduction générale	14
1 Chapitre : État de l’art.....	16
1.1 Introduction	16
1.2 Définition d’un système de recommandation	16
1.3 Techniques de recommandation.....	16
1.3.1 Le filtrage basé sur le contenu	17
1.3.2 Le filtrage collaboratif.....	18
1.3.3 Les systèmes hybrides	19
1.4 Contextes d’utilisation.....	21
1.5 Évaluations des systèmes de recommandation	21
1.6 Limites des systèmes de recommandation	23
1.6.1 Le problème du démarrage à froid.....	23
1.6.2 Le problème de parcimonie	24
1.7 Conclusion	24
2 Chapitre : La méthode explicable.....	25
2.1 Introduction	25
2.2 Méthodes explicables (XAI)	25
2.3 Méthodes et techniques explicables	26

2.3.1	Méthodes intrinsèquement interprétables	26
2.3.2	Méthodes post-hoc (modèle-agnostiques).....	27
2.3.3	Autres méthodes explicables	28
2.4	Pourquoi l'intelligence artificielle explicable est-elle importante ?.....	28
2.5	Comment fonctionne l'intelligence artificielle explicable (XAI) ?.....	29
2.6	Les avantages de l'intelligence artificielle explicable (XAI).....	30
2.7	Conclusion	31
3	Chapitre : XAI dans les systèmes de recommandation	32
3.1	Introduction	32
3.2	Pourquoi rendre les recommandations explicables?	32
3.3	Méthodes utilisées pour l'explication	33
3.3.1	Les règles d'association	33
3.3.2	LIME (Local Interpretable Model-agnostic Explanations)	34
3.3.3	SHAP (SHapley Additive exPlanations).....	34
3.3.4	Modèles basés sur le contenu interprétable.....	34
3.3.5	Approches visuelles ou textuelles.....	34
3.4	Études de cas / Exemples d'applications.....	35
3.5	Limites et défis des systèmes de recommandation explicables (XAI)	37
3.6	Évaluation des systèmes de recommandation explicables	38
3.6.1	Types d'évaluation	39
3.6.2	Évaluer la qualité des explications.....	40
3.7	Conclusion	42
4	Chapitre : Mise en œuvre du système de recommandation explicable	43
4.1	Introduction	43
4.2	Environnement expérimental	44
4.2.1	Google Colab	44
4.2.2	Python.....	44
4.2.3	Bibliothèque Surprise	44
4.3	Les method utilise	45

4.3.1	TF-IDF (Term Frequency – Inverse Document Frequency).....	45
4.3.2	Régression linéaire avec TF-IDF.....	48
4.3.3	Random Forest avec TF-IDF.....	50
4.4	Présentation du jeu de données	53
4.5	Régression Linéaire avec TF-IDF (explicable)	53
4.5.1	Importation des bibliothèques	53
4.5.2	Prétraitement des avis.....	54
4.5.3	Analyse des résultats	56
4.6	Random Forest avec TF-IDF	57
4.6.1	Importation des bibliothèques	58
4.6.2	Prétraitement des avis.....	58
4.6.3	Analyse des résultats	60
4.7	Comparaison entre Régression linéaire et Random Forest	62
4.8	Conclusion	62
	Conclusion générale.....	64
	Reference.....	65

Liste de figure

Figure1.1 : les techniques de system de recommandation.....	16
Figure1.2 : Un système de recommandation basé sur le contenu.....	16
Figure1.3 : Un système de recommandation collaboratif.....	17
Figure1.4 : Le système de recommandation hybride.....	20
Figure2.1 : La méthode explicable (XAI).....	25
Figure 4.1 : Importation des bibliothèques.....	53
Figure 4.2 : Chargement de fichier CSV.....	54
Figure 4.3 : Suppression des lignes.....	54
Figure 4.4 : Transformation des avis.....	54
Figure 4.5 : Conversion des notes en valeurs numériques (ex. : de 1 à 10).....	54
Figure 4.6 : Séparation en données d'entraînement (80%) et de test (20%).....	54
Figure 4.7 : Création et entraînement du modèle de régression linéaire.....	55
Figure 4.8 : Prédiction des notes à partir des avis test.....	55
Figure 4.9 : Évaluation de MSE, R ²	55
Figure 4.10 : Les hôtels reels.....	55
Figure 4.11 : hôtels prédits par le modèle.....	55
Figure 4.12 : Evaluation de la méthode.....	56
Figure 4.13 : Résultat des hôtels.....	57
Figure 4.14 : Importation des bibliothèques.....	57
Figure 4.15 : Chargement des données.....	58

Figure 4.16: Suppression des lignes.....	58
Figure 4.17 : Vectorisation des descriptions (TF-IDF).....	58
Figure 4.18 : Conversion de la note (Rating) en type numérique.....	58
Figure 4.19 : Division du jeu de données.....	59
Figure 4.20 : Création et entraînement du modèle Random Forest.....	59
Figure 4.21 : Prédiction sur les données de test.....	59
Figure 4.22 : Evaluation du modèle.....	59
Figure 4.23 : Extraction des meilleurs hôtels (réels).....	59
Figure 4.24 : Extraction des meilleurs hôtels (prédits).....	60
Figure 4.25 : Evaluation de MSE, R ²	60
Figure 4.26 : Résultat des hôtels.....	61

Liste des tableaux

Tableau 4.1 : Les avantage et les limites de TF-IDF.....	48
Tableau 4.2 : Les Avantages et les limites de Régression linéaire avec TF-IDF.....	50
Tableau 4.3 : Les avantage et les limites de Random Forest avec TF-IDF.....	52
Tableau 4.4 : Comparaison entre les deux approches.....	62

List des abréviations

IA : Intelligence Artificielle.

XAI : eXplainable Artificial Intelligence (Intelligence Artificielle Explicable)

ML : Machine Learning (Apprentissage automatique)

NLP : Natural Language Processing (Traitement automatique du langage naturel)

TF-IDF: Term Frequency - Inverse Document Frequency

RMSE : Root Mean Squared Error (Erreur quadratique moyenne)

MAE : Mean Absolute Error (Erreur absolue moyenne)

R^2 : Coefficient de détermination

SHAP : SHapley Additive exPlanations

LIME : Local Interpretable Model-Agnostic Explanations

SVD : Singular Value Decomposition (Décomposition en valeurs singulières)

FP-Growth : Frequent Pattern Growth (Croissance des motifs fréquents)

RGPD : Règlement Général sur la Protection des Données

CSV : Comma-Separated Values (Fichier de valeurs séparées par des virgules)

UI : User Interface (Interface utilisateur)

UX : User Experience (Expérience utilisateur)

Introduction générale

À l'ère du numérique, les utilisateurs sont confrontés quotidiennement à une abondance de contenus, de produits et d'informations. Pour faire face à cette surcharge informationnelle, les systèmes de recommandation se sont imposés comme des outils incontournables, capables de proposer des contenus personnalisés en fonction des préférences, des comportements et des besoins des individus. Ils sont aujourd'hui omniprésents dans des domaines variés tels que le commerce électronique, les réseaux sociaux, la musique, les plateformes de streaming, ou encore les environnements éducatifs.

Cependant, malgré leur efficacité prouvée, la plupart de ces systèmes reposent sur des modèles complexes et opaques, qualifiés de "boîtes noires". Cette opacité soulève plusieurs problématiques :

- Comment justifier une recommandation donnée à un utilisateur ?
- Comment s'assurer que le système ne reproduit pas de biais injustes ?
- Comment gagner la confiance des utilisateurs dans des domaines critiques comme la santé, la finance ou l'éducation ?

Face à ces enjeux, l'intégration de méthodes explicables, regroupées sous le terme XAI (eXplainable Artificial Intelligence), constitue une avancée majeure. Ces techniques visent à rendre les décisions des algorithmes compréhensibles et interprétables pour les utilisateurs finaux, les développeurs et les régulateurs. Elles permettent d'expliquer pourquoi et comment une recommandation a été produite, offrant ainsi une plus grande transparence, une meilleure acceptabilité, et une conformité avec les exigences éthiques et légales (comme le RGPD).

Ce mémoire s'inscrit dans cette problématique actuelle et ambitieuse. Il vise à développer, comparer et évaluer plusieurs modèles de recommandation intégrant l'explicabilité, en s'appuyant sur un jeu de données réel (avis d'hôtels) et des approches issues de l'intelligence artificielle interprétable.

Les objectifs principaux de ce travail sont les suivants :

- Présenter les fondements théoriques des systèmes de recommandation et de l'intelligence artificielle explicable.
- Implémenter différents modèles explicables : régression linéaire avec TF-IDF, Random Forest, et FP-Growth.
- Appliquer les techniques LIME et SHAP pour analyser et expliquer les recommandations.
- Comparer les performances et le niveau d'interprétabilité de chaque approche.
- Discuter les apports, les limites et les perspectives de l'intégration de XAI dans les systèmes de recommandation.

Ce mémoire est structuré comme suit :

- Chapitre 1 : présente l'état de l'art des systèmes de recommandation, leurs techniques principales, leurs avantages et leurs limites.
- Chapitre 2 : introduit les méthodes explicables (XAI), leurs principes, leurs catégories et leur importance croissante.
- Chapitre 3 : décrit l'implémentation des modèles de recommandation explicables et l'intégration de techniques comme LIME et SHAP.
- Chapitre 4 : expose les résultats expérimentaux, les comparaisons entre modèles et les analyses des performances et de l'interprétabilité.
- Conclusion générale : résume les apports du travail, ses limites, et propose des perspectives de recherche futures.

1 Chapitre : État de l'art

1.1 Introduction

Ce chapitre présente les concepts fondamentaux des systèmes de recommandation, en abordant les principales approches classiques telles que le filtrage collaboratif, le filtrage basé sur le contenu, ainsi que les systèmes hybrides. Nous détaillerons également les techniques avancées, les limites connues de ces approches, ainsi que les contextes d'utilisation typiques. L'objectif est de fournir une base théorique solide pour mieux comprendre les choix méthodologiques développés dans les chapitres suivants.

1.2 Définition d'un système de recommandation

Un système de recommandation est un outil de filtrage de l'information conçu pour proposer à un utilisateur des éléments susceptibles de l'intéresser, sur la base de ses préférences passées, de son profil, ou de comportements similaires d'autres utilisateurs. Ces systèmes visent à réduire la surcharge informationnelle tout en améliorant l'expérience utilisateur par la personnalisation des contenus. Ils sont présents dans de nombreux domaines : commerce électronique, musique, streaming vidéo, réseaux sociaux, etc. [1]

1.3 Techniques de recommandation

Les techniques de recommandation peuvent être classées de différentes manières. Parfois plusieurs termes sont utilisés pour désigner une même méthode ou approche. La classification la plus utilisée repose sur trois types : filtrage basé sur le contenu, filtrage collaboratif et filtrage hybride.

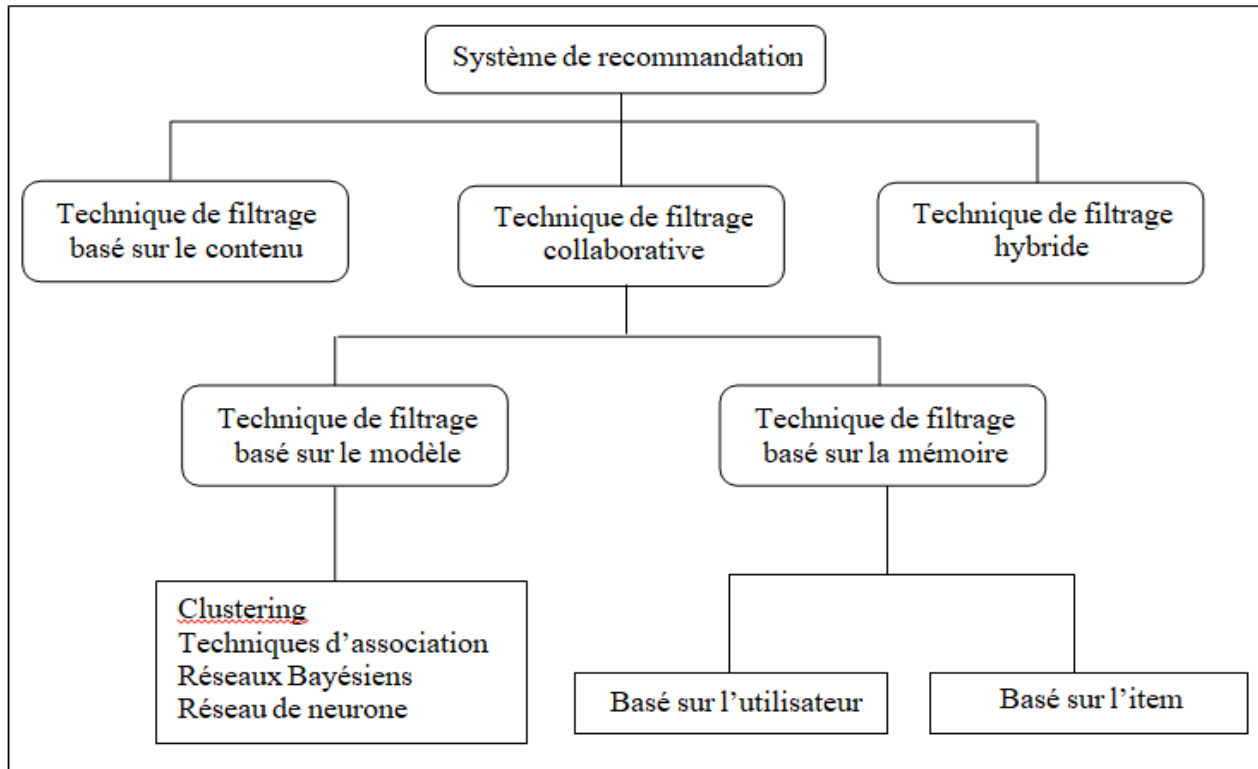


Figure1.1 : les techniques de system de recommandation

1.3.1 Le filtrage basé sur le contenu

Le filtrage basé sur le contenu recommande des items similaires à ceux qu'un utilisateur a déjà appréciés, en se basant sur les caractéristiques intrinsèques des items (attributs, texte, métadonnées).[2]

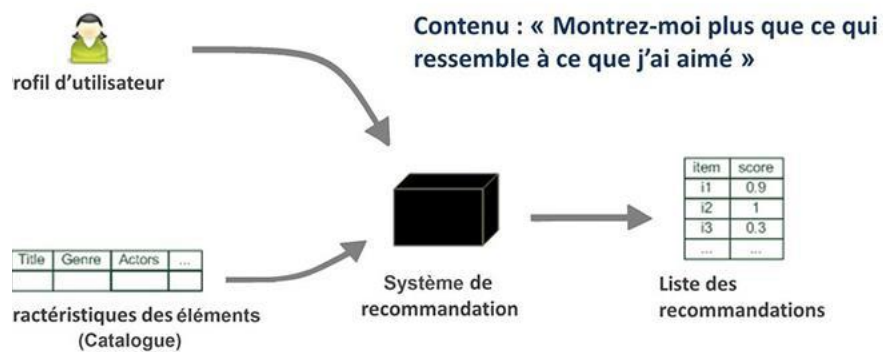


Figure1.2 : Un système de recommandation basé sur le contenu

Exemple : Recommander un livre du même genre ou du même auteur qu'un livre déjà aimé.

Avantages :

- Ne dépend pas des autres utilisateurs.
- Particulièrement utile pour les utilisateurs avec des goûts uniques.

Limites :

- Sur-spécialisation possible.
- Nécessite des données bien structurées sur les items.

1.3.2 Le filtrage collaboratif

Le filtrage collaboratif repose sur l'idée que les utilisateurs ayant manifesté des intérêts similaires dans le passé auront des préférences similaires à l'avenir. On distingue deux approches principales :

- **User-based filtering** : la prédiction est fondée sur la proximité entre les utilisateurs (utilisateurs voisins).
- **Item-based filtering** : la prédiction est basée sur la similarité entre les items eux-mêmes.[3]

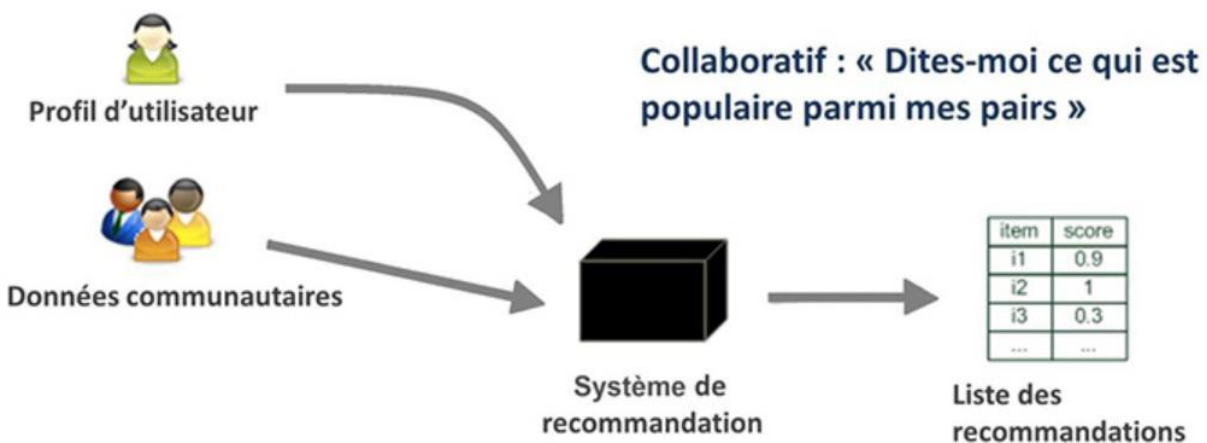


Figure1.3 : Un système de recommandation collaboratif

Avantages :

- Ne nécessite pas de connaissance explicite sur les items.
- Capable de capturer des préférences complexes.

Inconvénients :

- Problème de démarrage à froid.
- Sensible à la parcimonie (sparse data).

1.3.3 Les systèmes hybrides

Face aux forces et faiblesses respectives du filtrage collaboratif et du filtrage basé sur le contenu, de nombreux systèmes modernes optent pour leur combinaison, donnant naissance aux systèmes de recommandation hybrides.

L'objectif est de tirer profit des avantages de chaque approche tout en atténuant leurs limites (comme le démarrage à froid, la parcimonie ou la sur-spécialisation).[4]

En règle générale, l'hybridation suit une logique en deux étapes :

- ✓ Appliquer séparément plusieurs techniques de recommandation (collaboratif, contenu, etc.) pour générer des recommandations candidates ;
- ✓ Fusionner ces résultats à l'aide de méthodes telles que la pondération, la cascade, la commutation, ou encore la fusion de caractéristiques, afin de produire des recommandations finales plus pertinentes.

De manière plus large, certains systèmes hybrides construisent des profils utilisateurs orientés contenu, qu'ils comparent ensuite pour former des groupes d'utilisateurs similaires – une approche qui active ainsi indirectement le filtrage collaboratif.

La typologie la plus reconnue des méthodes hybrides a été proposée par Burke (2002), qui distingue sept principales formes d'hybridation :

a. Pondération (Weighted)

Les résultats de plusieurs systèmes de recommandation sont combinés par une fonction de pondération (ex. : moyenne pondérée), chaque méthode contribuant proportionnellement à la note finale.

b. Commutation (Switching)

Le système sélectionne dynamiquement l'approche la plus appropriée selon des critères contextuels (ex. : présence ou non d'historique utilisateur).

c. Technique mixte (Mixed)

Les recommandations issues de différentes méthodes sont présentées simultanément à l'utilisateur.

d. Combinaison de caractéristiques (Feature Combination)

Les sorties d'un système (ex. : notes estimées) sont utilisées comme caractéristiques d'entrée dans un autre modèle, généralement basé sur le contenu.

e. Cascade

Les techniques sont appliquées séquentiellement : une première méthode génère un ensemble de candidats, puis une seconde les affine ou les trie.

f. Augmentation de caractéristiques (Feature Augmentation)

Contrairement à la cascade, ici la sortie du premier modèle est transformée en nouvelle variable d'entrée pour le second.

g. Méta-niveau (Meta-level)

Le premier modèle produit un modèle intermédiaire (profil utilisateur, modèle factorisé...) qui est ensuite utilisé comme entrée par un second algorithme.

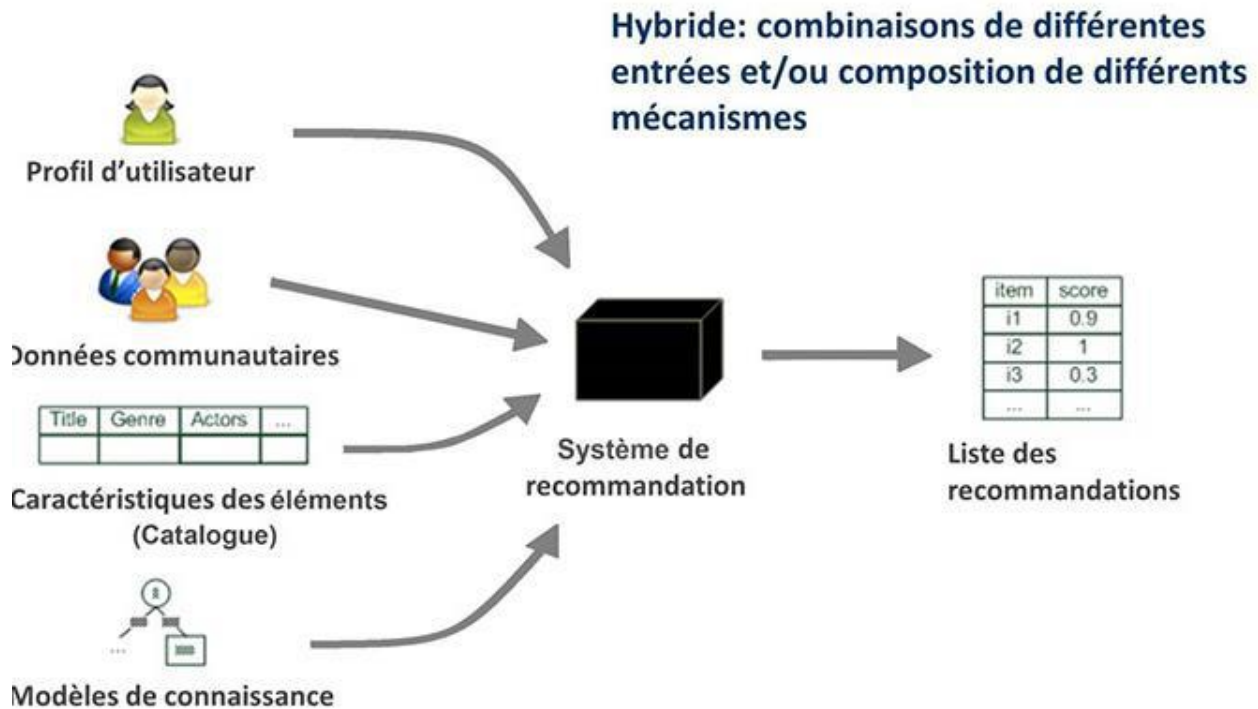


Figure1.4 : Le système de recommandation hybride

Avantages :

- Meilleure robustesse.
- Réduction du démarrage à froid et de la sur-spécialisation.

1.4 Contextes d'utilisation

Un système de recommandation est utile partout où l'utilisateur fait face à un grand choix d'options. Il peut être intégré à des sites de commerce en ligne, des plateformes de streaming, des réseaux sociaux, des systèmes éducatifs, ou encore des outils de diagnostic médical.

1.5 Évaluations des systèmes de recommandation

L'évaluation d'un système de recommandation a pour objectif de mesurer sa performance au regard des finalités qui lui sont assignées.

Le choix des métriques d'évaluation dépend donc directement des objectifs visés (précision, couverture, diversité, etc.).

Dans une procédure classique d'évaluation, l'ensemble de données est divisé en deux sous-ensembles :

- Un ensemble d'entraînement, utilisé pour construire le modèle,
- Un ensemble de test, destiné à évaluer la qualité des prédictions.

Pour chaque utilisateur u appartenant à l'ensemble de test, le système tente de prédire la note qu'il attribuerait à un item i . Cette prédiction est effectuée à partir des notes connues de u (sur d'autres items) et éventuellement des données issues des utilisateurs de l'ensemble d'entraînement.[5]

➤ **Métriques de précision prédictive (Prediction Accuracy)**

Ces métriques mesurent l'écart entre les notes prédites par le système et les notes réelles attribuées par les utilisateurs.

➤ **MAE (Mean Absolute Error) – Erreur absolue moyenne**

Elle calcule la moyenne des **valeurs absolues** des différences entre les notes prédites et réelles.

$$MAE = \frac{1}{N} \sum_{(u,i)} |\hat{r}_{ui} - r_{ui}| \quad (1.1)$$

- \hat{r}_{ui} : note prédite par le système pour l'utilisateur u et l'item i .
- r_{ui} : note réelle donnée par l'utilisateur.
- N : nombre total de prédictions.

Plus la valeur du MAE est **faible**, plus la prédiction est **précise**.

➤ **RMSE (Root Mean Squared Error) – Erreur quadratique moyenne**

Cette métrique pénalise davantage les erreurs importantes que le MAE, car elle élève l'erreur au carré.

$$RMSE = \sqrt{\frac{1}{N} \sum_{(u,i)} (\hat{r}_{ui} - r_{ui})^2} \quad (1.2)$$

Une valeur faible de RMSE indique une meilleure qualité de prédiction, tout en révélant les cas où le modèle se trompe fortement.

➤ **R² (coefficient de détermination)**

Mesure la proportion de la variance des notes réelles qui est expliquée par le modèle.

$$R^2 = 1 - \frac{\sum_{(u,i)} (\hat{r}_{ui} - r_{ui})^2}{\sum_{(u,i)} (r_{ui} - \bar{r})^2} \quad (1.3)$$

- \bar{r} : moyenne des notes réelles

Un score proche de 1 indique un modèle très performant, alors qu'un score proche de 0 signifie que le modèle est peu explicatif.

1.6 Limites des systèmes de recommandation

1.6.1 Le problème du démarrage à froid

Il survient lorsqu'un nouvel utilisateur ou un nouvel item entre dans le système. En l'absence d'historique, le système ne peut proposer de recommandations fiables.

✓ **Solutions possibles :**

- Utiliser le contenu (filtrage basé sur le contenu).
- Demander des préférences initiales à l'utilisateur.
- Exploiter des données démographiques.

1.6.2 Le problème de parcimonie

Les matrices utilisateur-item sont souvent sparse : chaque utilisateur n'interagit qu'avec une petite fraction des items. Cela complique l'identification de similarités fiables.

Exemple : sur Amazon, un utilisateur actif n'interagit qu'avec $<1\%$ des produits.

✓ **Conséquences :**

- Faible recouvrement entre utilisateurs.
- Moindre performance des modèles collaboratifs.

✓ **Solutions :**

- Factorisation matricielle.
- Enrichissement des données (ex. : avis textuels).
- Méthodes hybrides.

1.7 Conclusion

Ce chapitre a permis de poser les bases théoriques nécessaires à la compréhension des systèmes de recommandation, en présentant leurs principales méthodes, avantages et défis. Ces éléments sont essentiels pour aborder ensuite la question de l'explicabilité, qui vient compléter ces techniques en rendant leurs décisions plus transparentes.

2 Chapitre : La méthode explicable

2.1 Introduction

L'intelligence artificielle explicable (XAI – *eXplainable Artificial Intelligence*) regroupe un ensemble de techniques conçues pour rendre les décisions des systèmes d'IA compréhensibles et interprétables par les humains.

Dans des secteurs sensibles comme la santé, la finance ou la justice, il ne suffit pas qu'un système d'IA fournisse une réponse précise ; il est également crucial de comprendre les raisons et le processus ayant conduit à cette décision.

Les approches XAI répondent ainsi à des enjeux de transparence, de confiance et de responsabilité.

Elles permettent, par exemple, d'expliquer pourquoi un algorithme a recommandé un produit, refusé une demande de crédit ou posé un diagnostic, en mettant en évidence les facteurs clés de la prédiction.

2.2 Méthodes explicables (XAI)

L'intelligence artificielle explicable (XAI) désigne l'ensemble des méthodes, techniques et processus visant à rendre les résultats produits par des algorithmes de machine learning compréhensibles, interprétables et justifiables pour les utilisateurs humains.

Elle permet non seulement d'expliquer le fonctionnement d'un modèle, mais aussi d'analyser son impact, d'identifier d'éventuels biais, et d'évaluer des critères fondamentaux tels que la précision, la transparence, ou encore l'équité des décisions algorithmiques.

L'objectif de la XAI est de favoriser la confiance entre les systèmes d'IA et leurs utilisateurs, en particulier dans les contextes où les décisions automatisées ont des conséquences importantes (ex. : santé, finance, justice). En ce sens, elle constitue un élément clé d'une IA responsable et d'une gouvernance éthique des technologies d'apprentissage automatique.

À mesure que les modèles d'IA deviennent plus complexes, il devient de plus en plus difficile pour les humains — y compris les ingénieurs ou data scientists — de comprendre ou retracer le cheminement logique ayant conduit à une décision. Ces modèles, souvent qualifiés de "boîtes noires", produisent des résultats à partir de calculs internes inaccessibles ou opaques, rendant leur interprétation quasiment impossible sans outils spécialisés.

L'IA explicable s'impose ainsi comme une réponse nécessaire à cette opacité, en fournissant des explications intelligibles sur le comment et le pourquoi des prédictions, renforçant ainsi l'acceptabilité, la sécurité et la redevabilité des systèmes d'IA.[6]

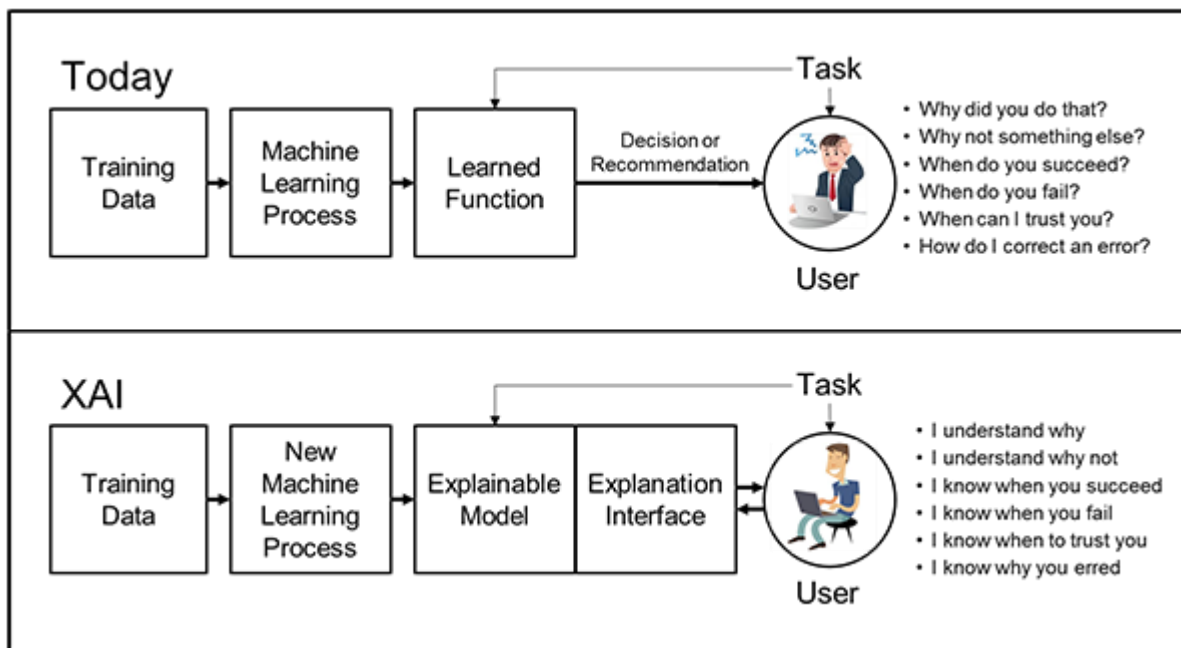


Figure2.1 : La méthode explicable AI (XAI)

2.3 Méthodes et techniques explicables

Les méthodes XAI se classent en deux grandes catégories :

2.3.1 Méthodes intrinsèquement interprétables

Ces méthodes reposent sur des modèles simples et transparents, dont le comportement peut être compris directement :

- **Arbres de décision** : leur structure hiérarchique permet de suivre logiquement les règles menant à une prédiction.[7]
- **Règles d'association** (ex. : Apriori, FP-Growth) : elles identifient des cooccurrences fréquentes dans les données, générant des règles du type :[9]
« Si un utilisateur a aimé A et B, alors il aimera probablement C. »
- **Régression linéaire ou logistique** : chaque coefficient représente l'impact d'une variable sur la prédiction finale.[7]

Avantage : transparence directe et facilité d'interprétation.

Limite : ces modèles ont souvent une capacité limitée à représenter des relations complexes.

2.3.2 Méthodes post-hoc (modèle-agnostiques)

Ces méthodes s'appliquent à des modèles opaques comme les forêts aléatoires ou les réseaux de neurones. Elles sont dites "agnostiques" car elles peuvent expliquer n'importe quel modèle, quelle que soit sa structure.[6]

- **LIME (Local Interpretable Model-Agnostic Explanations)** :

LIME crée un modèle simple (souvent une régression linéaire) autour d'une instance donnée, en générant des perturbations de ses caractéristiques. Il identifie ensuite les caractéristiques les plus influentes dans cette prédiction locale.[7]

→ Utilisé pour : texte, images, données tabulaires.

→ Exemple : expliquer pourquoi un utilisateur a reçu une certaine recommandation de film ou d'hôtel.

- **SHAP (SHapley Additive exPlanations)** :

SHAP repose sur la théorie des jeux coopératifs et attribue à chaque caractéristique une contribution numérique à la prédiction, à travers les valeurs de Shapley.[8]

→ Il fournit des explications locales (par instance) mais aussi globales (par modèle).

→ Avantage : rigueur théorique et cohérence.

→ Limite : calcul coûteux, surtout pour les grands modèles.

2.3.3 Autres méthodes explicables

Anchors : Génèrent des règles conditionnelles fiables ("Si telle condition est vraie, alors la prédiction sera X").[7]

Counterfactual Explanations : Cherchent à répondre à la question "Que faudrait-il changer pour obtenir une autre prédiction ?".[7]

2.4 Pourquoi l'intelligence artificielle explicable est-elle importante ?

L'intelligence artificielle explicable (XAI) joue un rôle fondamental dans l'adoption responsable et durable des systèmes d'IA par les organisations. Il ne suffit pas d'exploiter les résultats produits par des modèles d'apprentissage automatique sans en comprendre le processus décisionnel sous-jacent. Il est essentiel pour toute organisation de pouvoir contrôler, auditer et justifier les décisions générées par ces systèmes. C'est précisément ce que permet la XAI : elle fournit les outils nécessaires à une comprise humaine des algorithmes complexes, notamment ceux issus du machine learning, de l'apprentissage profond ou des réseaux de neurones.[10]

Ces derniers sont souvent considérés comme des modèles "boîtes noires", en raison de la difficulté à interpréter leur fonctionnement interne. Cela représente un défi majeur, d'autant plus que ces modèles peuvent intégrer ou amplifier des biais algorithmiques (liés au genre, à l'origine ethnique, à l'âge, ou à d'autres caractéristiques sensibles) présents dans les données d'entraînement.[11]

De plus, les performances des systèmes d'IA peuvent évoluer négativement au fil du temps, en raison de l'écart entre les données d'entraînement et les données réelles en production (phénomène de dérive ou model drift). Il devient donc crucial de mettre en place une surveillance continue des modèles afin d'assurer leur fiabilité, leur conformité et leur robustesse.[12]

L'IA explicable contribue également à :

- Renforcer la confiance des utilisateurs finaux,
- Améliorer l'auditabilité des décisions,
- Permettre une évaluation claire de l'impact des modèles sur les activités de l'entreprise,
- Réduire les risques liés à la non-conformité réglementaire, à la sécurité des données et à la réputation.

En ce sens, la XAI est au cœur de ce que l'on appelle l'IA responsable, une approche méthodologique qui vise à concevoir, déployer et gérer des systèmes d'intelligence artificielle de manière éthique, équitable, transparente et explicable.

Adopter une IA responsable implique d'intégrer des principes éthiques dans le développement d'applications et de processus décisionnels, en créant des systèmes fondés sur la confiance, la transparence et la redevabilité.

2.5 Comment fonctionne l'intelligence artificielle explicable (XAI) ?

L'intelligence artificielle explicable permet aux organisations de comprendre, retracer et interpréter les décisions prises par les algorithmes d'apprentissage automatique. En rendant transparents les mécanismes internes des modèles, la XAI offre la possibilité de diagnostiquer les erreurs, de corriger les dérives et de renforcer la confiance des utilisateurs vis-à-vis des systèmes intelligents.[6]

Grâce à ces méthodes, les entreprises peuvent non seulement observer le processus décisionnel sous-jacent d'un modèle d'IA, mais aussi l'ajuster ou le contrôler lorsque cela est nécessaire. Cette capacité à expliquer les résultats est particulièrement utile pour améliorer l'expérience utilisateur, en rassurant les clients finaux sur la légitimité et la cohérence des décisions automatisées.

Une question centrale se pose alors : quel niveau de confiance peut-on raisonnablement attendre d'un système d'IA ? Et comment peut-on s'assurer que ce système est capable de corriger ses erreurs et de s'adapter aux changements dans les données ou les comportements utilisateurs ?

Dans un contexte où l'intelligence artificielle devient de plus en plus sophistiquée, il est essentiel que les processus de machine learning soient compris, interprétés et contrôlés pour garantir l'exactitude, la fiabilité et la stabilité des prédictions.

Cela suppose de distinguer plusieurs concepts fondamentaux :

- La différence entre IA conventionnelle (centrée sur la performance) et IA explicable (centrée sur la compréhension),
- Les différentes méthodes et techniques permettant de transformer un modèle opaque en système interprétable,
- Et enfin, la distinction entre interprétation (accès aux mécanismes internes du modèle) et explication (capacité à rendre compte des décisions de manière intelligible pour un humain).

2.6 Les avantages de l'intelligence artificielle explicable (XAI)

L'intégration de l'intelligence artificielle explicable (XAI) dans les processus décisionnels offre de nombreux bénéfices, tant sur le plan technique que stratégique. Elle constitue un levier essentiel pour fiabiliser, optimiser et gouverner efficacement les systèmes basés sur l'IA.

1. Renforcement de la confiance et de l'adoption des systèmes d'IA

La XAI permet de garantir l'interprétabilité des modèles, en rendant leurs décisions compréhensibles, traçables et justifiables. Cette transparence favorise une plus grande confiance des utilisateurs et facilite la mise en production des modèles dans des environnements critiques. Elle permet également aux équipes techniques et métiers de mieux évaluer les modèles, d'identifier les biais éventuels et de valider les prédictions selon des critères objectifs.

2. Amélioration continue des performances

Grâce à l'explicabilité, il devient possible de suivre, analyser et ajuster dynamiquement le comportement des modèles. Une surveillance régulière facilite la détection des dérives (concept drift) et la réduction des erreurs.

Les retours sur les explications fournies permettent d'affiner les efforts de développement, d'optimiser les performances métier, et de renforcer l'efficacité opérationnelle des systèmes IA.

3. Réduction des risques et des coûts liés à la gouvernance

La XAI joue un rôle majeur dans la maîtrise des risques réglementaires, juridiques et éthiques. En rendant les décisions algorithmiques explicites, elle facilite la conformité aux normes en vigueur (ex. : RGPD, principes éthiques de l'IA), tout en réduisant les coûts liés aux audits manuels, aux contrôles qualité, et aux erreurs potentielles.

De plus, l'explicabilité contribue à identifier et atténuer les biais involontaires, ce qui est essentiel pour garantir l'équité des systèmes et préserver la réputation de l'organisation.

2.7 Conclusion

L'intelligence artificielle explicable (XAI) représente une avancée majeure pour rendre les systèmes d'IA plus transparents, compréhensibles et fiables. Dans le contexte des systèmes de recommandation, elle permet non seulement d'améliorer la confiance des utilisateurs, mais aussi de mieux justifier les suggestions fournies. Grâce à des techniques telles que LIME, SHAP ou encore les règles d'association, il devient possible de mettre en lumière les facteurs influençant chaque recommandation. Toutefois, malgré leurs apports, ces méthodes posent encore certains défis en termes de performance, de complexité et d'interprétation, ce qui ouvre la voie à de nombreuses perspectives de recherche.

3 Chapitre : XAI dans les systèmes de recommandation

3.1 Introduction

L'intégration de l'intelligence artificielle explicable (XAI) dans les systèmes de recommandation répond à un besoin croissant de transparence et de confiance dans les décisions automatisées. Alors que les algorithmes de recommandation classiques sont souvent perçus comme des "boîtes noires", les approches explicables permettent d'expliquer les raisons sous-jacentes à chaque suggestion faite à l'utilisateur. Cette intégration rend les systèmes plus interprétables, améliore l'acceptation des recommandations, et facilite la détection des biais ou des erreurs. Elle devient ainsi un enjeu majeur dans des domaines sensibles comme la santé, la finance ou l'éducation, où comprendre une recommandation est aussi important que sa pertinence.

3.2 Pourquoi rendre les recommandations explicables?

Les systèmes de recommandation ont gagné en importance dans de nombreux domaines tels que le commerce électronique, les plateformes de streaming, la santé ou encore l'éducation. Toutefois, bon nombre de ces systèmes fonctionnent comme des « boîtes noires », c'est-à-dire qu'ils fournissent des résultats sans expliquer comment ou pourquoi une recommandation particulière a été faite. Cette opacité peut poser plusieurs problèmes, notamment en termes de confiance, transparence et adoption par l'utilisateur.[6]

1. Renforcer la confiance de l'utilisateur

Une recommandation dont on comprend les raisons est plus facilement acceptée. En fournissant des explications, l'utilisateur a le sentiment que le système « sait ce qu'il fait » et peut juger par lui-même de la pertinence de la suggestion. Cela favorise une meilleure adoption du système.

2. Améliorer la transparence du système

Les recommandations explicables permettent d'ouvrir la boîte noire des algorithmes. Cette transparence est cruciale, notamment dans les domaines sensibles comme la santé, la finance ou les services publics, où les décisions doivent être justifiées.

3. Détecter les biais ou les erreurs

En visualisant les facteurs ayant conduit à une recommandation, les utilisateurs ou les concepteurs peuvent identifier plus facilement les erreurs, les comportements biaisés ou les failles du modèle (ex. : surreprésentation de certains types d'items).

4. Aider à l'apprentissage utilisateur

Les explications permettent à l'utilisateur de mieux comprendre ses propres préférences. Il peut découvrir des critères qu'il n'avait pas explicitement exprimés mais que le système a détectés de manière implicite.

5. Répondre aux exigences réglementaires

Avec les nouvelles lois sur la protection des données (comme le RGPD en Europe), les systèmes automatisés doivent souvent justifier leurs décisions, notamment lorsqu'ils ont un impact significatif sur les individus. L'intégration de méthodes explicables permet de répondre à ces exigences légales.

3.3 Méthodes utilisées pour l'explication

L'intégration de l'explicabilité dans les systèmes de recommandation repose sur plusieurs techniques développées dans le champ de l'intelligence artificielle explicable (XAI). Ces méthodes permettent de justifier les recommandations en identifiant les facteurs influents dans la prise de décision du modèle. On peut distinguer plusieurs grandes familles de méthodes explicatives, chacune adaptée à un type d'algorithme ou d'application.[14]

3.3.1 Les règles d'association

Les règles d'association, comme celles issues des algorithmes Apriori ou FP-Growth, sont naturellement explicables. Elles identifient des relations fréquentes entre des items dans les transactions des utilisateurs. Par exemple, une règle de type « les utilisateurs qui ont consulté l'hôtel A ont également aimé l'hôtel B » peut être directement présentée à l'utilisateur comme justification d'une recommandation.[15]

3.3.2 LIME (Local Interpretable Model-agnostic Explanations)

LIME est une méthode d'explication locale, indépendante du modèle utilisé. Elle génère des perturbations autour d'une prédiction donnée, puis ajuste un modèle simple (ex. : régression linéaire) pour expliquer cette prédiction. LIME peut être utilisé pour expliquer pourquoi un utilisateur a reçu une certaine recommandation, en mettant en avant les mots-clés (dans les descriptions d'items) ou les attributs qui ont le plus influencé la décision.[7]

3.3.3 SHAP (SHapley Additive exPlanations)

SHAP repose sur la théorie des jeux coopératifs pour attribuer une « valeur » à chaque caractéristique d'entrée (feature), en fonction de sa contribution à une prédiction. Il est particulièrement utile pour les modèles complexes comme les forêts aléatoires ou les réseaux de neurones. Dans le contexte d'un système de recommandation, SHAP peut indiquer que la localisation, la note ou le type de chambre ont joué un rôle majeur dans la suggestion d'un hôtel.[8]

3.3.4 Modèles basés sur le contenu interprétable

Les systèmes de recommandation basés sur le contenu utilisent les attributs des items (titre, catégorie, auteur, etc.) pour proposer des suggestions. Ces attributs peuvent également servir à générer des explications. Par exemple, un système peut expliquer qu'un film est recommandé « car il appartient au même genre que ceux que vous aimez » ou « car il est du même réalisateur ».[16]

3.3.5 Approches visuelles ou textuelles

Des interfaces interactives ou des annotations textuelles peuvent être utilisées pour expliquer les recommandations à l'utilisateur final. Ces méthodes visent à rendre les explications compréhensibles, même pour des personnes non expertes. Par exemple, des barres de score, des bulles d'explication, ou des phrases naturelles peuvent résumer les raisons de la suggestion.[17]

3.4 Études de cas / Exemples d'applications

Pour illustrer l'intérêt et l'impact des méthodes explicables (XAI) dans les systèmes de recommandation, plusieurs cas concrets peuvent être analysés. Ces exemples montrent comment l'intégration de techniques XAI améliore l'interprétabilité, la confiance des utilisateurs, et même les performances du système dans certains contextes critiques.

1. Recommandation d'hôtels avec explication basée sur LIME

Dans un système de recommandation hôtelière, les avis utilisateurs (sous forme textuelle) sont utilisés pour prédire une note globale. Un modèle comme Random Forest peut prédire la note d'un hôtel à partir de ces descriptions. Cependant, sans explication, l'utilisateur ne sait pas pourquoi tel ou tel hôtel est recommandé.

L'intégration de LIME permet de générer des explications locales, par exemple :

« Cet hôtel vous est recommandé car les clients ont particulièrement apprécié sa propreté, sa localisation centrale et la qualité de son service. »

Ainsi, même si le modèle est complexe, LIME extrait les facteurs clés ayant influencé la prédiction, ce qui rend la recommandation compréhensible.

2. Recommandation de films avec SHAP (MovieLens Dataset)

Le jeu de données MovieLens est souvent utilisé dans la recherche sur les systèmes de recommandation. En combinant un modèle de type XGBoost avec SHAP, il est possible d'expliquer les recommandations de films en fonction des préférences explicites d'un utilisateur.

Par exemple, SHAP peut indiquer que la prédiction d'une note élevée pour un film est principalement due à :

- La présence de l'acteur préféré de l'utilisateur
- Le genre du film (science-fiction ou comédie romantique)
- La similarité avec d'autres films déjà notés positivement

L'explication visuelle des scores SHAP permet de mieux comprendre les critères dominants dans la recommandation.

3. Recommandation de produits sur Amazon – Règles d'association

Sur des plateformes e-commerce comme Amazon, l'utilisation de règles d'association (via Apriori ou FP-Growth) permet de recommander des produits fréquemment achetés ensemble. L'avantage est que la justification peut être directement exprimée, par exemple :

« Les clients ayant acheté ce téléphone ont aussi acheté cette coque de protection. »

Cette méthode est naturellement explicable, car elle repose sur des corrélations simples et compréhensibles entre items.

4. Recommandation dans le domaine médical

Dans les systèmes de recommandation médicale (ex. : recommandations de traitements ou d'exams), la nécessité d'explication est cruciale pour des raisons éthiques et légales. Les techniques XAI comme SHAP ou LIME sont utilisées pour justifier pourquoi un médicament ou un protocole est recommandé à un patient donné. Les explications peuvent s'appuyer sur :

- Les symptômes observés
- Les antécédents médicaux
- Les traitements antérieurs

Cette approche renforce la confiance des professionnels de santé dans les outils de support à la décision.

5. Recommandation de cours en ligne (E-learning)

Dans les plateformes d'apprentissage en ligne comme Coursera ou Khan Academy, les systèmes de recommandation orientent les apprenants vers des cours adaptés à leur niveau et à leurs préférences. Un modèle explicable peut dire :

« Ce cours est recommandé parce que vous avez terminé avec succès d'autres cours similaires et que vous avez exprimé un intérêt pour le sujet de l'intelligence artificielle. »

Ici, l'explication permet à l'apprenant de comprendre la logique derrière la suggestion, ce qui renforce son engagement.

3.5 Limites et défis des systèmes de recommandation explicables (XAI)

Malgré les avantages qu'offrent les approches explicables dans les systèmes de recommandation, leur mise en œuvre soulève plusieurs limitations et défis techniques, méthodologiques et humains [13]. Voici les principaux obstacles identifiés dans la littérature et la pratique :

1. Compromis entre explicabilité et performance

Un des défis majeurs réside dans le trade-off (compromis) entre la performance du modèle et sa capacité à être expliqué. Les modèles les plus performants (comme les réseaux de neurones profonds ou les systèmes hybrides complexes) sont souvent les plus difficiles à interpréter. À l'inverse, les modèles linéaires ou à base de règles sont facilement explicables mais peuvent offrir une précision inférieure.

2. Explications parfois approximatives ou peu fidèles

Les méthodes XAI comme LIME ou SHAP fournissent des explications dites "post-hoc", c'est-à-dire produites après la prédiction. Ces explications ne garantissent pas toujours une correspondance exacte avec le raisonnement interne du modèle. Il existe donc un risque de générer des interprétations trompeuses ou approximatives.

3. Complexité computationnelle

Certaines techniques explicables, notamment SHAP, sont coûteuses en temps de calcul, surtout lorsqu'elles sont appliquées à des ensembles de données volumineux ou à des modèles très complexes. Cela peut limiter leur utilisation en temps réel, notamment dans des systèmes de recommandation interactifs.

4. Difficile à généraliser à tous les types de données

Les approches XAI sont souvent adaptées à des types de données spécifiques :

- LIME est efficace pour des données textuelles ou tabulaires, mais moins adapté aux graphes.
- Les règles d'association sont efficaces pour des données transactionnelles mais ne conviennent pas toujours aux systèmes basés sur l'apprentissage profond.

Il est donc complexe de concevoir une méthode d'explication universelle qui s'applique à tous les types de systèmes de recommandation.

5. Compréhension par l'utilisateur final

Même si une explication est techniquement correcte, elle n'est pas toujours compréhensible ou utile pour l'utilisateur final. Le niveau de technicité des explications doit être adapté au profil des utilisateurs. Il faut trouver un équilibre entre la simplicité, la précision et la transparence.

6. Protection de la vie privée

Certaines explications peuvent révéler des données sensibles ou personnelles utilisées dans le modèle. Cela pose un enjeu éthique, notamment dans les domaines de la santé ou de la finance. Il faut donc veiller à ne pas violer la confidentialité des utilisateurs lors de la génération d'explications.

7. Absence de standardisation

Il n'existe pas encore de cadre normatif universel pour évaluer la qualité d'une explication. Ce manque de standard rend difficile la comparaison entre différentes approches XAI, et pose un problème d'évaluation rigoureuse dans un contexte académique ou industriel.

3.6 Évaluation des systèmes de recommandation explicables

L'évaluation des systèmes de recommandation explicables repose en grande partie sur les mêmes métriques que celles utilisées pour les systèmes de recommandation classiques, notamment en ce qui concerne la qualité des recommandations. Toutefois, dans le cas de la XAI, il est

également nécessaire de prendre en compte la qualité, l'interprétabilité et la pertinence des explications générées.

3.6.1 Types d'évaluation

Les chercheurs identifient trois approches principales pour évaluer les systèmes de recommandation, qu'ils soient explicables ou non :

a) Études utilisateurs (user studies)

Les utilisateurs sont invités à interagir directement avec le système, à effectuer des tâches spécifiques, puis à donner leur avis sur la pertinence des recommandations et la clarté des explications.

Cette méthode permet d'obtenir des retours qualitatifs riches, mais présente plusieurs limitations :

- Difficulté à recruter un échantillon représentatif.
- Taille souvent réduite des panels.
- Résultats parfois non généralisables à l'ensemble des utilisateurs.[17]

b) Évaluation en ligne (online evaluation)

Elle se déroule sur un système opérationnel en production et repose sur l'analyse des comportements réels des utilisateurs (clics, achats, navigation...).

Des indicateurs tels que le taux de clics (CTR) ou le temps passé sont utilisés pour mesurer l'impact des recommandations.

Toutefois, ce type d'évaluation n'est possible que si :

- Le système dispose d'un nombre suffisant d'utilisateurs actifs,
- Des données de suivi sont disponibles,
- Le système est déjà déployé.

Contrairement aux études utilisateurs, l'évaluation en ligne est souvent non intrusive : les utilisateurs ne sont pas conscients qu'ils participent à une étude, ce qui rend leur comportement plus naturel.[17]

c) Évaluation hors ligne (offline evaluation)

Elle est la plus répandue, car elle ne nécessite pas l'intervention directe des utilisateurs. Elle repose sur des ensembles de données historiques, divisés en jeux d'entraînement et de test. Deux approches peuvent être distinguées dans le cas des systèmes explicables :

- L'analyse du taux d'explicabilité (proportion de recommandations accompagnées d'une explication, indépendamment de sa qualité).
- L'évaluation de la qualité des explications, selon leur format (textuel, visuel, verbal) et leur fonction (expliquer pourquoi, pourquoi pas, comment, quoi...).[17]

3.6.2 Évaluer la qualité des explications

L'évaluation de la composante explicable d'un système va au-delà de la simple mesure de la précision des recommandations. Elle implique de prendre en compte la compréhension, l'utilité perçue et la satisfaction de l'utilisateur vis-à-vis des explications fournies. Plusieurs dimensions peuvent être analysées :

a) Modèle mental de l'utilisateur

Un modèle mental est la représentation interne qu'un utilisateur se fait du fonctionnement du système. Les explications fournies par la XAI aident à structurer cette compréhension. Une méthode courante consiste à interroger directement l'utilisateur pour qu'il décrive, selon lui, comment le système fonctionne. Des entretiens qualitatifs permettent d'analyser en profondeur sa perception et sa logique de raisonnement.[18]

b) Performance de la tâche homme-machine

L'un des objectifs de la XAI est d'aider l'utilisateur à mieux accomplir ses tâches. L'évaluation consiste alors à mesurer si, grâce aux explications, l'utilisateur parvient à :

- Mieux comprendre les prédictions.
- Repérer les erreurs.
- Justifier les choix faits par le système.

Des tests permettent d'étudier la capacité des utilisateurs à formuler des questions (type "pourquoi", "pourquoi pas") et à recevoir des réponses adaptées.[18]

c) Satisfaction utilisateur

La satisfaction est souvent évaluée à travers :

- La compréhensibilité de l'explication.
- Son utilité perçue.
- Et parfois des indicateurs indirects comme la curiosité, la confiance, ou l'engagement.

Des outils comme le questionnaire de satisfaction ou la méthode du ratio de validité du contenu (Content Validity Ratio – CVR) permettent d'obtenir des mesures quantitatives de cette dimension.[18]

d) Confiance dans le système

La confiance est un facteur affectif et cognitif essentiel dans l'acceptation de l'IA. Elle dépend de plusieurs éléments :

- Les compétences techniques de l'utilisateur,
- Sa perception de la transparence du système,
- L'efficacité perçue des recommandations.

La confiance peut être mesurée par des enquêtes avant, pendant et après l'utilisation du système, à travers des échelles d'attitude, des scores de fiabilité perçue, ou des entretiens.[19]

3.7 Conclusion

L'intégration de méthodes explicables (XAI) dans les systèmes de recommandation représente une avancée significative vers des modèles plus transparents, compréhensibles et dignes de confiance. En permettant aux utilisateurs de comprendre les raisons derrière les recommandations, XAI améliore non seulement l'acceptabilité du système, mais aussi son efficacité dans des contextes critiques. Cette approche ouvre la voie à des recommandations plus responsables et éthiques, tout en renforçant l'interaction homme-machine.

4 Chapitre : Mise en œuvre du système de recommandation explicable

4.1 Introduction

Ce chapitre est consacré à l'évaluation expérimentale de notre système de recommandation. Il a pour objectif d'analyser les performances et l'efficacité des algorithmes mis en œuvre à travers un processus de validation rigoureux. En menant des expérimentations et en étudiant les résultats obtenus, nous obtenons des informations précieuses sur les capacités et les limites des algorithmes utilisés.

L'évaluation est réalisée dans un environnement contrôlé en s'appuyant sur des outils et des méthodes appropriés. Nous avons utilisé Google Colab comme plateforme expérimentale, bénéficiant ainsi de ressources de calcul suffisantes et de fonctionnalités collaboratives. Les expérimentations sont développées en Python, un langage reconnu pour son efficacité en traitement de données et en implémentation d'algorithmes.

Pour faciliter l'évaluation, nous avons recours à la bibliothèque Surprise, un outil puissant dédié à la construction et à l'évaluation de systèmes de recommandation. Elle offre un large éventail d'algorithmes et de métriques d'évaluation permettant une analyse fine des performances.

Ce chapitre décrit également en détail le jeu de données utilisé pour les expérimentations. Il s'agit de données d'interactions utilisateur-item, qui constituent la base sur laquelle les recommandations sont générées. La compréhension des caractéristiques de ce jeu de données est essentielle pour interpréter correctement les résultats expérimentaux.

Enfin, nous présentons la méthodologie de validation adoptée, incluant les stratégies de partitionnement des données, les protocoles d'évaluation et les métriques utilisées pour mesurer la précision et la qualité des recommandations.

Ce chapitre se termine par une analyse détaillée et une discussion des résultats obtenus lors des expérimentations. Les performances de chaque algorithme de recommandation sont examinées

à travers les différents indicateurs d'évaluation, mettant en lumière leurs points forts ainsi que leurs limites. Cette évaluation expérimentale permet de mieux cerner les capacités réelles du système de recommandation et constitue une base solide pour envisager des pistes d'amélioration ou de futures recherches.

4.2 Environnement expérimental

L'environnement expérimental constitue un élément essentiel pour assurer la précision, la reproductibilité et l'efficacité des évaluations menées. Dans cette section, nous présentons les principaux outils utilisés pour la mise en œuvre et l'évaluation du système de recommandation, à savoir : Google Colab, le langage Python, et la bibliothèque Surprise.

4.2.1 Google Colab

Google Colab a été utilisé comme plateforme d'exécution des expériences. Il s'agit d'un environnement de développement cloud proposé par Google, offrant des ressources de calcul performantes, notamment l'accès à des GPU/TPU, tout en permettant un travail collaboratif. Cette plateforme facilite l'expérimentation sur des ensembles de données volumineux et assure un partage facile des notebooks entre chercheurs et contributeurs.[20]

4.2.2 Python

Python est le langage de programmation retenu pour le développement des algorithmes et l'analyse des résultats. Il est largement plébiscité dans le domaine de l'intelligence artificielle en raison de sa simplicité, de sa lisibilité, et de son riche écosystème de bibliothèques (comme NumPy, Pandas, scikit-learn, etc.). Ces atouts en font un choix privilégié pour la mise en œuvre de modèles d'apprentissage automatique et le traitement de données. [21]

4.2.3 Bibliothèque Surprise

La bibliothèque Surprise (Simple Python Recommendation System Engine) est spécifiquement conçue pour la création et l'évaluation de systèmes de recommandation. Elle propose un ensemble d'algorithmes standards (collaboratifs et basés sur les facteurs latents), des outils de gestion de

données, ainsi que diverses métriques de validation. Son utilisation permet d'implémenter rapidement des modèles de recommandation tout en bénéficiant d'un cadre structuré pour leur évaluation. [22]

En combinant Google Colab, Python et la bibliothèque Surprise, nous avons pu établir un environnement d'expérimentation stable et reproductible, propice à l'analyse rigoureuse des performances des algorithmes testés. La section suivante détaillera la méthodologie de validation adoptée pour évaluer ces performances.

4.3 Les method utilise

Dans ce travaille on utilise deux méthodes avec une technique de pondération

4.3.1 TF-IDF (Term Frequency – Inverse Document Frequency)

Définition générale

TF-IDF (Fréquence Terme – Fréquence Inverse de Document) est une technique de pondération utilisée pour évaluer l'importance relative d'un mot dans un document, par rapport à l'ensemble d'un corpus (ensemble de documents).

Elle est couramment utilisée pour :

- Extraire les mots-clés d'un texte,
- Représenter des documents sous forme de vecteurs numériques,
- Alimenter des modèles de classification ou de recommandation sur du contenu textuel (avis clients, descriptions, etc.).

Fonctionnement de TF-IDF

TF-IDF est le produit de deux composantes :

1. TF (Term Frequency) – Fréquence du terme

Il mesure combien de fois un mot apparaît dans un document donné. Plus un mot est fréquent dans un document, plus il est supposé être important dans ce document.

$$TF(t, d) = \frac{\text{Nombre d'occurrences du terme } t \text{ dans le document } d}{\text{Nombre total de mots dans } d} \quad (4.1)$$

Exemple : si le mot "hôtel" apparaît 3 fois dans un avis de 100 mots, alors $TF = 3 / 100 = 0.03$

2.IDF (Inverse Document Frequency) – Fréquence inverse du document

Il mesure dans combien de documents du corpus un mot apparaît.

L'idée est que si un mot apparaît dans tous les documents, il n'a aucune valeur distinctive (ex. : "le", "et", "très").

$$IDF(t) = \log \left(\frac{N}{n_t} \right) \quad (4.2)$$

- N = nombre total de documents dans le corpus
- n_t = nombre de documents contenant le terme t

Plus un mot est rare dans l'ensemble du corpus, plus sa valeur IDF est élevée.

3. TF-IDF final

$$TF-IDF(t, d) = TF(t, d) \times IDF(t) \quad (4.3)$$

Le score TF-IDF est donc élevé pour les mots :

- Fréquents dans un document donné (TF élevé)
- Mais rares dans le reste du corpus (IDF élevé)

4. Exemple concret

Prenons trois avis d'utilisateurs sur des hôtels :

- Document 1 : "Cet hôtel est propre et bien situé"
- Document 2 : "Très bon rapport qualité-prix"
- Document 3 : "Propre et calme, personnel accueillant"

Le mot "propre" apparaît dans 2 documents sur 3.

- $TF("propre", Doc1) = 1 / 6 = 0.167$
- $IDF("propre") = \log(3 / 2) = \sim 0.176$
- $TF-IDF("propre", Doc1) = 0.167 \times 0.176 \approx 0.029$

En revanche, un mot rare comme "accueillant", s'il n'apparaît que dans un seul document, aura un IDF plus élevé.

5. Utilisation dans les systèmes de recommandation

Dans les systèmes de recommandation textuels, TF-IDF permet de :

- Transformer un texte (avis, description d'un produit, etc.) en vecteur numérique
- Comparer la similarité entre documents ou profils utilisateurs
- Alimenter des modèles comme la régression linéaire, les forêts aléatoires, ou le filtrage basé sur le contenu

Exemple : recommander un hôtel en fonction des mots-clés que l'utilisateur aime dans ses avis précédents.

6. Avantages et limites

Avantages	Limites
Simple, rapide, efficace	Ignore l'ordre des mots (pas de contexte)
Bon pour les petits corpus	Ne capture pas la polysémie ou synonymie
Très utile pour les modèles linéaires	Inefficace pour les textes très courts

Tableau 4.1 : Les avantages et les limites de TF-IDF

4.3.2 Régression linéaire avec TF-IDF

1. Objectif de la méthode

L'objectif ici est de prédire une note (par exemple, une évaluation utilisateur de 1 à 10) à partir de données textuelles comme des avis clients sur des hôtels. Pour cela, on transforme le texte (non exploitable en l'état par un algorithme) en vecteurs numériques via TF-IDF, puis on applique un modèle de régression linéaire pour faire la prédiction.

2. Étapes de la méthode

a) Prétraitement des données

- Nettoyage des textes (suppression des caractères spéciaux, mise en minuscule, etc.)
- Suppression des mots vides (stop words) comme "le", "de", "et"
- Transformation des commentaires en représentations numériques avec TF-IDF

b) Vectorisation avec TF-IDF

Chaque commentaire client devient un vecteur de poids numériques, où chaque poids représente l'importance du mot (cf. explication TF-IDF). Ainsi, un avis comme :

"Très propre, bon accueil et emplacement central"

c) Application de la régression linéaire

Le vecteur TF-IDF sert d'entrée pour un modèle de régression linéaire, qui apprend à prédire une note numérique (ex. : 8.5/10) en fonction des mots présents dans l'avis.

Le modèle suit une équation de type :

$$\hat{y} = \beta_0 + \beta_1 x_1 + \beta_2 x_2 + \dots + \beta_n x_n \quad (4.4)$$

- \hat{y} : note prédite
- x_i : poids TF-IDF de chaque mot
- β_i : coefficient appris par le modèle, représentant l'impact du mot

d) Évaluation

On évalue la performance du modèle à l'aide de métriques comme :

- **MSE (Mean Squared Error)** : écart quadratique moyen entre notes réelles et prédites
- **R² (coefficient de détermination)** : proportion de la variance expliquée par le modèle

3. Exemple d'interprétation

Supposons que la régression linéaire indique :

Mot clé	Coefficient (β)
"propre"	+1.2
"sale"	-1.5
"calme"	+0.9

"bruyant"	-1.1
-----------	------

Cela signifie que :

- Un commentaire contenant "propre" augmente la note prédite d'environ +1.2 points,
- Tandis que "sale" diminue la note de 1.5 points.

C'est ici que l'aspect explicable est très fort : on peut justifier la prédiction en montrant quels mots ont influencé la note.

4. Avantages et limites

Avantages	Limites
Interprétable facilement (modèle linéaire)	Ne capte que les relations linéaires
Rapide à entraîner	Moins performant sur des données complexes
Facile à expliquer à l'utilisateur (XAI-native)	Ignore le contexte sémantique des mots

Tableau 4.2 : Les Avantages et les limites de Régression linéaire avec TF-IDF

4.3.3 Random Forest avec TF-IDF

1. Objectif général

Cette approche vise à prédire une note (ou score) attribuée par un utilisateur à un item (ex. : un hôtel), à partir de descriptions textuelles comme des avis clients.

Elle combine :

- La vectorisation TF-IDF pour convertir les textes en représentations numériques exploitables,

- Et l'algorithme Random Forest, un modèle d'ensemble puissant basé sur des arbres de décision, pour effectuer la prédiction.

2. Étapes de la méthode

a) Prétraitement et vectorisation TF-IDF

- Nettoyage des textes (mise en minuscules, suppression des caractères inutiles, etc.)
- Suppression des mots fréquents et peu informatifs (stop words)
- Transformation des avis en vecteurs TF-IDF : chaque document devient un vecteur de poids représentant l'importance de chaque mot dans ce texte par rapport au corpus.

Exemple : "Très propre et calme" → vecteur TF-IDF avec des poids pour "propre", "calme", etc.

b) Entraînement du modèle Random Forest

Le modèle Random Forest est ensuite appliqué aux vecteurs TF-IDF pour apprendre à prédire les notes données par les utilisateurs (par exemple de 1 à 10).

Il s'agit d'un modèle d'ensemble qui construit un grand nombre d'arbres de décision sur des sous-échantillons aléatoires des données, et combine leurs résultats pour obtenir une prédiction plus stable et robuste.

c) Prédiction et évaluation

Une fois entraîné, le modèle peut être utilisé pour prédire des notes sur de nouveaux avis textuels.

On l'évalue à l'aide de métriques comme :

- **MSE (Mean Squared Error)** : plus elle est faible, meilleure est la précision.
- **R² (coefficient de détermination)** : indique la proportion de variance expliquée par le modèle.

3. Interprétabilité (XAI)

Contrairement à la régression linéaire, Random Forest n'est pas directement interprétable. Cependant, on peut appliquer des techniques XAI telles que :

- **Feature importance** : calcule l'impact moyen de chaque mot (ou caractéristique TF-IDF) sur les prédictions.
- **SHAP (SHapley Additive exPlanations)** : mesure la contribution exacte de chaque mot à la prédiction d'une note.
- **LIME** : génère des modèles linéaires locaux autour d'une prédiction donnée, pour expliquer le résultat en langage simple.

Exemple d'explication SHAP :

"La note prédite est élevée car les mots 'propreté', 'emplacement', et 'accueil' sont fortement valorisés par le modèle."

4. Avantages et limites

Avantages	Limites
Bonne précision sur des données textuelles	Moins interprétable sans outils XAI
Résistant aux données bruitées	Temps d'entraînement plus élevé
Capte les relations non linéaires complexes	Moins adapté aux petits jeux de données
Fonctionne bien avec les vecteurs TF-IDF	Ne tient pas compte du contexte sémantique des mots

Tableau 4.3 : Les avantages et les limites de Random Forest avec TF-IDF

4.4 Présentation du jeu de données

Le jeu de données utilisé dans ce projet concerne les avis et informations sur des hôtels. Il contient diverses caractéristiques descriptives comme :

- **Hotel Name** : le nom de l'hôtel.
- **Location** : emplacement géographique de l'hôtel.
- **Rating** : note globale attribuée à l'hôtel (de 1 à 10).
- **Review Score** : commentaire textuel synthétisant l'avis client (souvent sous forme de mentions comme « Excellent », « Very Good », etc.).
- **Room Type, Bed Type, Room Price**: descriptions des chambres.

Ce jeu de données a été extrait d'une plateforme de réservation hôtelière et converti en format CSV pour être traité par des outils Python (notamment pandas et scikit-learn).

4.5 Régression Linéaire avec TF-IDF (explicable)

```
import pandas as pd
import numpy as np
from sklearn.feature_extraction.text import TfidfVectorizer
from sklearn.linear_model import LinearRegression
from sklearn.model_selection import train_test_split
from sklearn.metrics import mean_squared_error, r2_score
```

Figure 4.1 : Importation des bibliothèques

4.5.1 Importation des bibliothèques

- Pandas et numpy : manipulation des données.
- TfidfVectorizer : transforme le texte en vecteurs numériques.
- LinearRegression : algorithme de régression linéaire.
- train_test_split : sépare les données pour l'entraînement/test.
- mean_squared_error, r2_score: métriques d'évaluation.

```
data = pd.read_csv('/content/drive/MyDrive/bookhotel.csv', encoding='latin1')
```

Figure 4.2 : Chargement de fichier CSV

Chargés un fichier CSV contenant les informations sur les hôtels, stocké sur Google Drive.

4.5.2 Prétraitement des avis

```
data.dropna(subset=['Review Score'], inplace=True)  
data['Review Score'] = data['Review Score'].astype(str)
```

Figure 4.3 : Suppression des lignes

Nettoyage : on supprime les lignes sans score d'avis et on convertit les avis en texte.

```
tfidf_vectorizer = TfidfVectorizer(stop_words='english')  
tfidf_matrix = tfidf_vectorizer.fit_transform(data['Review Score'])
```

Figure 4.4 : Transformation des avis

Vectorisation TF-IDF : transforme chaque avis en un vecteur basé sur l'importance des mots (en ignorant les mots vides).

```
data['Rating'] = pd.to_numeric(data['Rating'], errors='coerce')
```

Figure 4.5 : Conversion des notes en valeurs numériques (ex. : de 1 à 10).

Conversion des notes en valeurs numériques (ex. : de 1 à 10).

```
X_train, X_test, y_train, y_test = train_test_split(tfidf_matrix, data['Rating'], test_size=0.2)
```

Figure 4.6 : Séparation en données d'entraînement (80%) et de test (20%).

Séparation en données d'entraînement (80%) et de test (20%).

```
lr_model = LinearRegression()
lr_model.fit(X_train, y_train)
```

Figure 4.7 : Création et entraînement du modèle de régression linéaire.

Création et entraînement du modèle de régression linéaire.

```
y_pred = lr_model.predict(X_test)
```

Figure 4.8 : Prédiction des notes à partir des avis test.

Prédiction des notes à partir des avis test.

```
mse = mean_squared_error(y_test, y_pred)
r2 = r2_score(y_test, y_pred)
```

Figure 4.9 : Évaluation de MSE, R²

Évaluation

- MSE : erreur moyenne quadratique.
- R² : mesure la qualité des prédictions (1 = parfait, 0 = aléatoire).

```
top_10_hotels_reels = pd.DataFrame({'Hotel Name': data.loc[y_test.index, 'Hotel Name'], 'Rating réel': y_test.values})
top_10_hotels_reels = top_10_hotels_reels.sort_values(by='Rating réel', ascending=False)
```

Figure 4.10 : Les hôtels reels.

Top 10 hôtels réels (ceux ayant les meilleures notes réelles).

```
top_10_hotels_predits = pd.DataFrame({'Hotel Name': data.loc[y_test.index, 'Hotel Name'], 'Rating prédit': y_pred})
top_10_hotels_predits = top_10_hotels_predits.sort_values(by='Rating prédit', ascending=False)
```

Figure 4.11 : hôtels prédits par le modèle.

Top 10 hôtels prédits par le modèle.

4.5.3 Analyse des résultats

Les performances du modèle de recommandation basé sur la régression linéaire ont été évaluées à l'aide de deux métriques principales :

MSE (Mean Squared Error): 0.24
R2 Score: 0.69

Figure 4.12 : Evaluation de la méthode

Interprétation

- **MSE faible (0.24)** : cela signifie que les erreurs moyennes entre les notes prédites et les notes réelles sont relativement faibles. Le modèle arrive donc à produire des prédictions proches des évaluations réelles des utilisateurs.
- **R² de 0.69** : ce score indique que **69 % de la variance des évaluations** est expliquée par le modèle. Bien que ce ne soit pas parfait, c'est un résultat satisfaisant pour un système de recommandation basé uniquement sur les descriptions textuelles.

Comparaison des Top 10 :

Les tableaux suivants comparent les 10 meilleurs hôtels selon les notes réelles et les 10 meilleurs selon les prédictions du modèle.

Cela montre que le modèle est capable d'identifier des établissements de haute qualité de manière fiable, ce qui constitue un bon indicateur de performance.

Cependant, on remarque aussi que certaines prédictions aboutissent à des valeurs identiques (ex : 9.667241) pour plusieurs hôtels. Cela suggère un manque de finesse dans la répartition des scores, ce qui peut provenir :

- Du modèle de régression utilisé (linéaire), parfois trop simple pour capter les subtilités du langage naturel,
- De la limitation des données textuelles à elles seules comme source d'information.

Top 10 meilleurs hôtels réels :

	Hotel Name	Rating réel
3159	3.Appartement 6Pers#Sèvres-Lecourbe#Necker	10.0
2067	New Leaf Cortijo Apartment Full disabled access	9.8
3201	43 Mezzanine 5Pers Madeleine Galerie La Fayette	9.8
2088	Cabanas de Vendaval	9.7
3260	Greeter-Bel appartement Parisien de 70m2 - St ...	9.7
1014	Casa Rural Finca Las Dulces	9.7
502	Pimalai Resort & Spa - SHA Extra Plus	9.6
3262	Paris-sur-seine-chez-caroline	9.5
3244	High Stay Saint Germain de Pres Paris 6	9.5
1019	Hotel Boutique Molino de Enmedio	9.5

Top 10 meilleurs hôtels prédits :

	Hotel Name	Rating prédit
3262	Paris-sur-seine-chez-caroline	9.667241
2067	New Leaf Cortijo Apartment Full disabled access	9.667241
1549	Apartamentos Balcon del Cielo	9.667241
3260	Greeter-Bel appartement Parisien de 70m2 - St ...	9.667241
3159	3.Appartement 6Pers#Sèvres-Lecourbe#Necker	9.667241
1554	OCEAN HOMES Apartamentos exclusivos en Isla Ca...	9.667241
3201	43 Mezzanine 5Pers Madeleine Galerie La Fayette	9.667241
2549	Apartamentos Balcon del Cielo	9.667241
3244	High Stay Saint Germain de Pres Paris 6	9.667241
502	Pimalai Resort & Spa - SHA Extra Plus	9.667241

Figure 4.13 : Résultat des hôtels

4.6 Random Forest avec TF-IDF

```
import pandas as pd
import numpy as np
from sklearn.feature_extraction.text import TfidfVectorizer
from sklearn.ensemble import RandomForestRegressor
from sklearn.model_selection import train_test_split
from sklearn.metrics import mean_squared_error, r2_score
```

Figure 4.14 : Importation des bibliothèques

4.6.1 Importation des bibliothèques

```
data = pd.read_csv('/content/drive/MyDrive/bookhotel.csv', encoding='latin1')
```

Figure 4.15 : Chargement des données

```
data.dropna(subset=['Review Score'], inplace=True)  
data['Review Score'] = data['Review Score'].astype(str)
```



Figure 4.16: Suppression des lignes

4.6.2 Prétraitement des avis

```
tfidf_vectorizer = TfidfVectorizer(stop_words='english')  
tfidf_matrix = tfidf_vectorizer.fit_transform(data['Review Score'])
```

Figure 4.17 : Vectorisation des descriptions (TF-IDF)

```
data['Rating'] = pd.to_numeric(data['Rating'], errors='coerce')
```

Figure 4.18: Conversion de la note (Rating) en type numérique

```
X_train, X_test, y_train, y_test = train_test_split(tfidf_matrix, data['Rating'],
```

Figure 4.19 : Division du jeu de données.

```
rf_model = RandomForestRegressor(n_estimators=1500)
rf_model.fit(X_train, y_train)
```

Figure 4.20 : Création et entraînement du modèle Random Forest

```
y_pred = rf_model.predict(X_test)
```

Figure 4.21 : Prédiction sur les données de test

```
mse = mean_squared_error(y_test, y_pred)
r2 = r2_score(y_test, y_pred)

print(f"MSE (Mean Squared Error): {mse:.2f}")
print(f"R2 Score: {r2:.2f}")
```

Figure 4.22 : Evaluation du modèle

```
top_10_hotels_reels = pd.DataFrame({
    'Hotel Name': data.loc[y_test.index, 'Hotel Name'],
    'Rating réel': y_test.values
}).sort_values(by='Rating réel', ascending=False)

print("\nTop 10 meilleurs hôtels réels :")
print(top_10_hotels_reels[['Hotel Name', 'Rating réel']].head(10))
```

Figure 4.23 : Extraction des meilleurs hôtels (réels)

```

top_10_hotels_predits = pd.DataFrame({
    'Hotel Name': data.loc[y_test.index, 'Hotel Name'],
    'Rating prédit': y_pred
}).sort_values(by='Rating prédit', ascending=False)

print("\nTop 10 meilleurs hôtels prédits :")
print(top_10_hotels_predits[['Hotel Name', 'Rating prédit']].head(10))

```

Figure 4.24 : Extraction des meilleurs hôtels (prédits)

4.6.3 Analyse des résultats

```

MSE (Mean Squared Error): 0.23
R2 Score: 0.70

```

Figure 4.25 : Evaluation de MSE, R²

MSE: Cela signifie que l'erreur quadratique moyenne entre les prédictions et les notes réelles est faible. C'est un bon indicateur : plus la valeur est proche de 0, plus le modèle est précis.

R² Score : Le modèle explique 70 % de la variance des notes réelles. Cela indique une bonne capacité prédictive. Pour un système de recommandation basé sur du texte (avis clients), ce score est tout à fait satisfaisant.

Comparaison des Top 10 :

Top 10 hôtels selon les notes réelles (Rating réel)

Ce classement reflète les meilleures notes attribuées par les utilisateurs. On y trouve des hôtels tels que :

- Luxurious Apartment next to Arc de Triomphe : note parfaite de 10.0.
- Plusieurs hôtels entre 9.6 et 9.8 comme *Amanecer Cabrerizos*, *Sabino*, *Pimalai Resort*, etc.

Top 10 hôtels selon les notes prédites (Rating prédit)

Le modèle a prédit des notes très proches entre elles (toutes ≈ 9.666), ce qui montre que :

- Le modèle est assez conservateur dans ses prédictions.
- Il tend à lisser les valeurs vers la moyenne des meilleures notes.
- Cependant, plusieurs hôtels bien notés en réel apparaissent aussi dans les prédictions, ce qui est positif :

Ex. : Luxurious Apartment next to Arc de Triomphe, Pimalai Resort, Blue House Mallorca, etc.

Top 10 meilleurs hôtels réels :		
	Hotel Name	Rating réel
3025	Luxurious Apartment next to Arc de Triomphe	10.0
1670	Amanecer Cabrerizos	9.8
2069	Sabino - baskeyrentals	9.7
2943	Amazing Duna Beach Seafront Apartament	9.7
2561	Agroturismo Son Vives Menorca - Adults Only	9.7
2402	Blue House Mallorca	9.6
2366	Apartamento Malpica Area Grande	9.6
1501	Hotel Creu de Tau Art&Spa-Adults only	9.6
499	Pimalai Resort & Spa - SHA Extra Plus	9.6
1367	Apartamento Malpica Area Grande	9.6
Top 10 meilleurs hôtels prédits :		
	Hotel Name	Rating prédit
3025	Luxurious Apartment next to Arc de Triomphe	9.666224
1806	La Torre del Algas	9.666224
1501	Hotel Creu de Tau Art&Spa-Adults only	9.666224
2402	Blue House Mallorca	9.666224
1670	Amanecer Cabrerizos	9.666224
2018	Hotel Boutique Molino de Enmedio	9.666224
499	Pimalai Resort & Spa - SHA Extra Plus	9.666224
2366	Apartamento Malpica Area Grande	9.666224
2943	Amazing Duna Beach Seafront Apartament	9.666224
1381	Cortijo Lagar de Luisa	9.666224

Figure 4.26 : Résultat des hôtels

4.7 Comparaison entre Régression linéaire et Random Forest

Critère	Régression linéaire	Random Forest
Type de modèle	Linéaire	Ensemble d'arbres (non linéaire)
Complexité	Faible	Élevée
Capacité de modélisation	Relations simples	Relations complexes
Score R^2	≈ 0.69	≈ 0.70
Erreur quadratique (MSE)	≈ 0.24	≈ 0.23 (meilleure précision)
Robustesse au bruit	Faible	Bonne (agrégation d'arbres)
Temps d'exécution	Rapide	Plus long (ex. : 1500 arbres)
Interprétabilité directe	Très bonne	Faible sans SHAP ou LIME
Utilisation de TF-IDF	Oui	Oui

Tableau 4.4 : Comparaison entre les deux approches

4.8 Conclusion

Ce chapitre a permis de mettre en œuvre et d'évaluer deux approches de recommandation basées sur des descriptions textuelles : la régression linéaire et le modèle Random Forest, tous deux appliqués à des vecteurs TF-IDF.

Les résultats montrent que les deux modèles offrent une bonne précision, mais que le Random Forest surpasse légèrement la régression linéaire en termes de performance (R^2 et MSE).

En revanche, la régression linéaire reste plus explicable sans outils supplémentaires, tandis que le modèle Random Forest nécessite l'usage de méthodes XAI comme SHAP ou LIME pour en interpréter les décisions.

Ces résultats confirment la pertinence d'intégrer des modèles interprétables ou des outils d'explication dans les systèmes de recommandation, afin d'allier précision et transparence.

Conclusion générale

Ce mémoire a porté sur la conception et l'évaluation d'un système de recommandation basé sur des méthodes explicables, dans un contexte où l'intelligence artificielle joue un rôle croissant dans la prise de décision personnalisée. Face à la critique de la "boîte noire" associée aux algorithmes traditionnels, notre travail s'est attaché à combiner performance et transparence, en intégrant des approches d'intelligence artificielle explicable (XAI) dans le processus de recommandation.

Nous avons exploré différentes méthodes, telles que la régression linéaire, les forêts aléatoires et les règles d'association (FP-Growth), en appliquant des techniques comme TF-IDF pour extraire des informations pertinentes à partir de données textuelles. Les résultats expérimentaux montrent que les algorithmes complexes (comme Random Forest) offrent de meilleures performances prédictives, tandis que les méthodes plus simples (comme la régression linéaire) sont plus faciles à interpréter. Pour répondre à cet équilibre entre efficacité et explicabilité, des outils comme LIME et SHAP peuvent être mobilisés pour expliquer même les modèles complexes.

L'intégration de la XAI dans les systèmes de recommandation ouvre ainsi de nouvelles perspectives, notamment dans les domaines sensibles où la confiance, la transparence et la responsabilité sont primordiales (santé, finance, éducation, etc.).

En somme, ce travail souligne la nécessité croissante de rendre les systèmes intelligents compréhensibles et responsables, tout en maintenant un haut niveau de performance. Il constitue une base solide pour des travaux futurs visant à améliorer les méthodes explicables, à enrichir les jeux de données, ou à adapter les recommandations aux préférences dynamiques des utilisateurs.

Reference

[1]: Ricci, F., Rokach, L., & Shapira, B. (2011). *Introduction to Recommender Systems Handbook*. In F. Ricci, L. Rokach, B. Shapira, & P. B. Kantor (Eds.), *Recommender Systems Handbook* (pp. 1–35). Springer.

https://doi.org/10.1007/978-0-387-85820-3_1

[2] : Carbonell, N., & Doush, I. (2014). *Systèmes de recommandation : principes, méthodes et évaluation*. Techniques de l'ingénieur.

<https://www.techniques-ingenieur.fr/base-documentaire/informatique-telecoms-th9/intelligence-artificielle-42270210/systemes-de-recommandation-h7820/>

[3]: Herlocker, J. L., Konstan, J. A., Terveen, L. G., & Riedl, J. T. (2004). Evaluating collaborative filtering recommender systems. *ACM Transactions on Information Systems (TOIS)*, 22(1), 5–53.

<https://doi.org/10.1145/963770.963772>

[4]: Burke, R. (2002). Hybrid Recommender Systems: Survey and Experiments. *User Modeling and User-Adapted Interaction*, 12(4), 331–370.

<https://doi.org/10.1023/A:1021240730564>

[5]: Herlocker, J. L., Konstan, J. A., Terveen, L. G., & Riedl, J. T. (2004). Evaluating collaborative filtering recommender systems. *ACM Transactions on Information Systems (TOIS)*, 22(1), 5–53.

<https://doi.org/10.1145/963770.963772>

[6]: Adadi, A., & Berrada, M. (2018). Peeking Inside the Black-Box: A Survey on Explainable Artificial Intelligence (XAI). *IEEE Access*, 6, 52138–52160.

<https://doi.org/10.1109/ACCESS.2018.2870052>

[7]: Ribeiro, M. T., Singh, S., & Guestrin, C. (2016). "Why Should I Trust You?": Explaining the Predictions of Any Classifier. *Proceedings of the 22nd ACM SIGKDD International Conference*

on *Knowledge Discovery and Data Mining*, 1135–1144.

<https://doi.org/10.1145/2939672.2939778>

[8]: Lundberg, S. M., & Lee, S.-I. (2017). A Unified Approach to Interpreting Model Predictions. *Proceedings of the 31st International Conference on Neural Information Processing Systems (NeurIPS)*, 4765–4774.

https://proceedings.neurips.cc/paper_files/paper/2017/file/8a20a8621978632d76c43dfd28b67767-Paper.pdf

[9]: Agrawal, R., & Srikant, R. (1994). Fast algorithms for mining association rules. *Proceedings of the 20th International Conference on Very Large Data Bases (VLDB)*, 487–499.

[10]: Gunning, D., & Aha, D. W. (2019). DARPA’s Explainable Artificial Intelligence (XAI) Program. *AI Magazine*, 40(2), 44–58.

<https://doi.org/10.1609/aimag.v40i2.2850>

[11] : Barocas, S., Hardt, M., & Narayanan, A. (2019). *Fairness and Machine Learning: Limitations and Opportunities*.

<https://fairmlbook.org/>

[12]: Breck, E., Polyzotis, N., Roy, S., Whang, S. E., & Zinkevich, M. (2017). The ML Test Score: A Rubric for ML Production Readiness and Technical Debt Reduction. *Google Research*.

<https://storage.googleapis.com/pub-tools-public-publication-data/pdf/45742.pdf>

[13]: Doshi-Velez, F., & Kim, B. (2017). *Towards A Rigorous Science of Interpretable Machine Learning*. arXiv preprint arXiv:1702.08608.

<https://arxiv.org/abs/1702.08608>

[14]: **Zhang, Y., & Chen, X. (2020)**. Explainable Recommendation: A Survey and New Perspectives. *Foundations and Trends® in Information Retrieval*, 14(1), 1–101.

<https://doi.org/10.1561/15000000056>

[15]: Agrawal, R., Imieliński, T., & Swami, A. (1993). Mining association rules between sets of items in large databases. *Proceedings of the 1993 ACM SIGMOD International Conference on Management of Data*, 207–216.

<https://doi.org/10.1145/170035.170072>

[16] : Pazzani, M. J., & Billsus, D. (2007). Content-Based Recommendation Systems. In *The Adaptive Web* (pp. 325–341). Springer.

https://doi.org/10.1007/978-3-540-72079-9_10

[17]: Tintarev, N., & Masthoff, J. (2007). A Survey of Explanations in Recommender Systems. In *Proceedings of the 2007 IEEE International Conference on Data Engineering Workshops (ICDEW)*, 801–810.

<https://doi.org/10.1109/ICDEW.2007.4401070>

[18]: Nunes, I., & Jannach, D. (2017). A Systematic Review and Taxonomy of Explanations in Decision Support and Recommender Systems. *User Modeling and User-Adapted Interaction*, 27(3–5), 393–444.

<https://doi.org/10.1007/s11257-017-9195-0>

[19]: Kunkel, J., Donkers, T., Loepp, B., & Ziegler, J. (2019). Let me explain: Impact of personal and impersonal explanations on trust in recommender systems. *Proceedings of the 24th International Conference on Intelligent User Interfaces*, 379–390.

<https://doi.org/10.1145/3301275.3302310>

[20]: Google Colaboratory. Google colaboratory. [Online]. Available:

<https://colab.research.google.com/>

[21]: Python Software Foundation. About python. [Online]. Available:

<https://www.python.org/about/>

[22] : Surprise. Surprise documentation. [Online]. Available:

<https://surprise.readthedocs.io/en/stable/>