

Rapport de projet

Chouaou Amer Hichem 28604746

Georges Martin 28731590

Lazrak Ali 28605235

Projet encadré par Mr Nicolas Baskiotis

15 Mai 2024

1 Introduction

Notre projet se consacre entièrement à l'explicabilité des systèmes de recommandation, visant à rendre ces systèmes transparents et compréhensibles pour les utilisateurs. Cette démarche vise à améliorer la confiance et l'acceptabilité des recommandations, en permettant aux utilisateurs de comprendre les raisons sous-jacentes qui motivent les suggestions qui leur sont faites.

Les systèmes de recommandation sont un élément central de nombreuses plateformes en ligne, influençant les choix et les décisions des utilisateurs. Cependant, l'utilisation d'algorithmes souvent considérés comme des "boîtes noires" peut obscurcir la compréhension des processus de recommandation, soulevant des préoccupations sur leur transparence et leur fiabilité.

Pour adresser ce problème, notre approche utilise des visualisations de données, des explications textuelles simples, et des résumés des caractéristiques clés pour démontrer clairement comment et pourquoi les recommandations sont générées. En se concentrant sur une base de données de jeux de société, nous explorons des méthodes pour expliciter les recommandations issues du filtrage collaboratif, une technique préférée pour sa capacité à intégrer les interactions et les préférences des utilisateurs de manière significative.

En mettant l'accent sur l'explicabilité, nous visons à équiper les utilisateurs avec les outils nécessaires pour évaluer et comprendre les recommandations, renforçant ainsi leur engagement et leur satisfaction avec les plate-formes qu'ils utilisent. Notre objectif est de lever le voile sur les mécanismes des systèmes de recommandation, contribuant à un environnement numérique plus transparent et responsable.

Dans la première partie de ce rapport, nous présenterons une analyse exploratoire des données de la base de données utilisée. Cette section examinera en détail les distributions des jeux, des utilisateurs et des notes, fournissant ainsi une compréhension claire des données de base. La deuxième partie sera dédiée aux méthodes de recommandation, décrivant les différents algorithmes employés. Enfin, la troisième partie détaillera le protocole expérimental mis en place pour tester ces méthodes de recommandation, en présentant les résultats.

2 Analyse exploratoire du jeu de donnée

Notre base de données est constituée à partir d'un scrapping d'un site web dédié aux jeux de société. Elle contient des informations détaillées que nous avons réparties en deux tables principales.

La première table, **Jeux**, inclut des informations les jeux de société tels que le casting des créateurs, le titre du jeu, le nombre total de notes reçues, la note moyenne attribuée au jeu, une description détaillée, ainsi que les catégories associées à chaque jeu. Cette table offre une vue complète des caractéristiques propres à chaque jeu, ce qui est crucial pour l'analyse des préférences et des tendances au sein de notre système de recommandation.

La seconde table, **Avis**, contient des données relatives aux utilisateurs du site. Elle comprend le nom d'utilisateur, ses différents commentaires sur les jeux, les titres des jeux commentés,

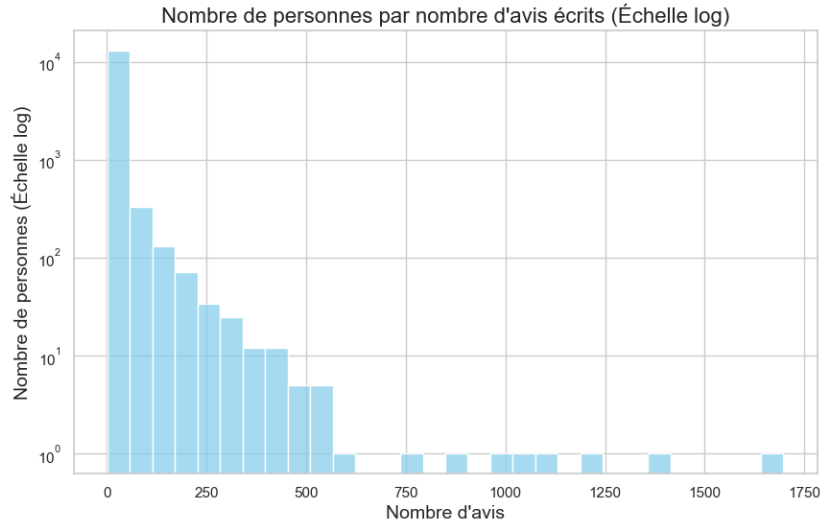


Figure 1

ainsi que les notes que l'utilisateur a attribuées à ces jeux. Cette table est essentielle pour comprendre comment les utilisateurs interagissent avec les jeux, fournissant un aperçu précieux de leurs préférences et de leur engagement envers certains jeux.

La première étape de notre analyse consiste à examiner ces données pour obtenir une vue d'ensemble des statistiques globales. Nous analysons la distribution des notes, le nombre d'utilisateurs, et la variété des jeux disponibles. Cette exploration préliminaire est vitale pour identifier les biais potentiels et déterminer les corrections nécessaires afin de normaliser les données pour leur traitement ultérieur. L'analyse de la distribution du nombre d'avis par jeu, par exemple, montre que la majorité des jeux reçoivent peu d'avis, ce qui peut influencer la précision des recommandations basées sur des données insuffisantes.

2.1 Analyse de la table jeux

Dans un premier temps, nous décidons de travailler sur les différents aspects des jeux afin de voir si une recommandation basée sur le contenu (content based) serait intéressante. Nous avons donc commencé par voir la distribution du nombre de jeux par nombre d'avis. Ce premier histogramme nous montre que la plupart des jeux n'ont que très peu d'avis, ce qui peut influencer les recommandations *figure 1*. Nous nous intéressons maintenant aux différentes distributions de notes.

On remarque dans un premier temps un pic anormal au niveau de la note 0 (*figure 2a*), ce qui suggère que les jeux sans note sont mélangés avec ceux ayant une note de 0. Après vérification et comparaison sans les jeux qui n'ont aucune note, le pic à 0 disparaît. De plus, on peut déduire que les utilisateurs ont tendance à noter les jeux qu'ils apprécient.

Ensuite, nous avons étudié les catégories de jeux *figure 3*. Avec les notes et les descriptions des jeux, ce sont les seuls aspects de notre base de données qui nous permettent de travailler sur les recommandations basées sur le contenu.

Ici, on peut voir que la catégorie avec la note la plus haute est la catégorie "Vêtements", mais elle n'apparaît qu'une seule fois.

De plus, la *figure 4* nous montre qu'il n'y a que très peu de jeux qui ont plusieurs catégories.

Ces analyses sur les notes et catégories montrent qu'il y a peu de matière pour travailler sur une recommandation basée sur le contenu. Nous aborderons ce point plus en détail plus loin dans notre rapport.

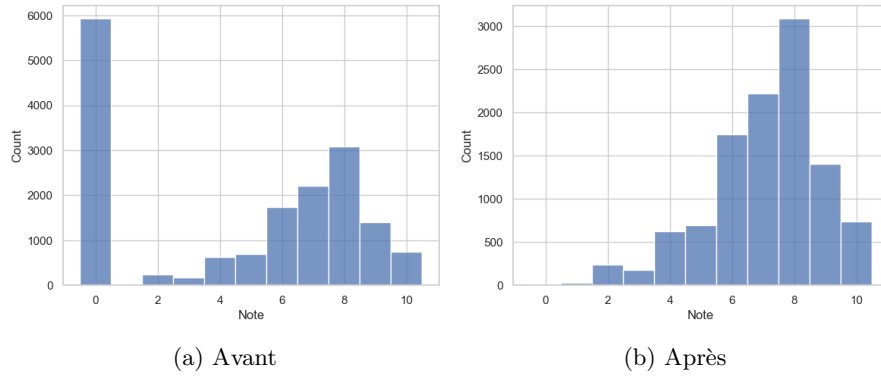


Figure 2: Comparaison du nombre de notes avant et après

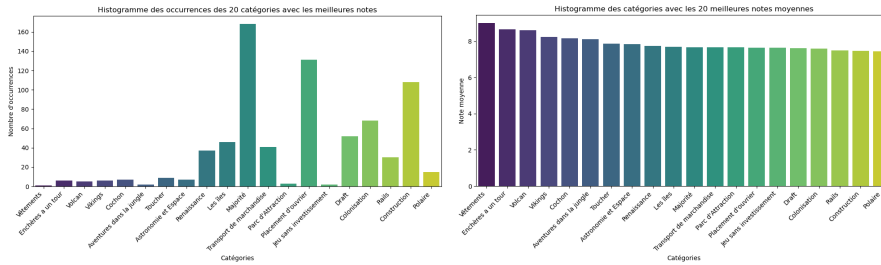


Figure 3: Travail sur les catégories

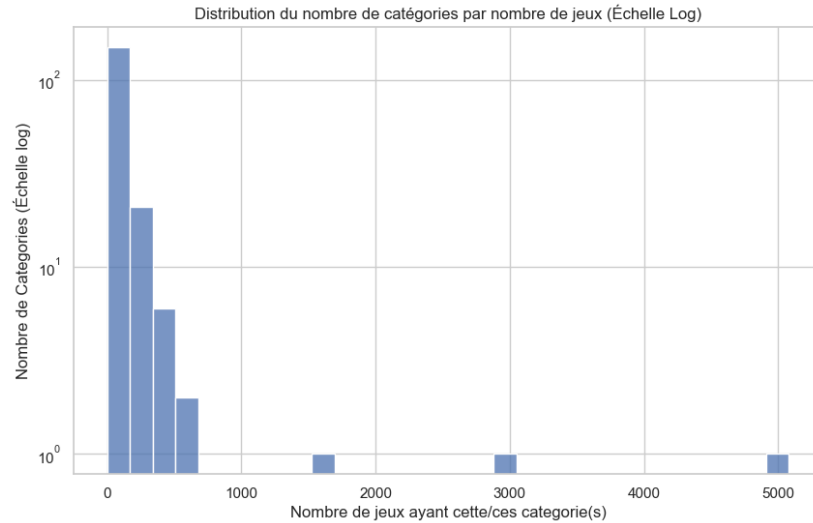


Figure 4

2.2 Analyse de la table utilisateurs

Suite à notre analyse basée sur le contenu, nous avons orienté notre attention vers une étude du filtrage collaboratif (collaborative filtering).

Nous avons donc commencé par examiner le nombre de personnes. Ce premier histogramme nous montre que la plupart des auteurs écrivent très peu de commentaires et que très peu d'auteurs écrivent beaucoup de commentaires *figure 5*.

Ensuite, nous avons analysé la moyenne des notes et l'écart type des avis pour chaque auteur *figure 6*. Ce graphique montre une large dispersion des notes avec la moyenne des notes sur l'axe des Y et l'écart type sur l'axe des X. Les auteurs avec des notes moyennes élevées et un

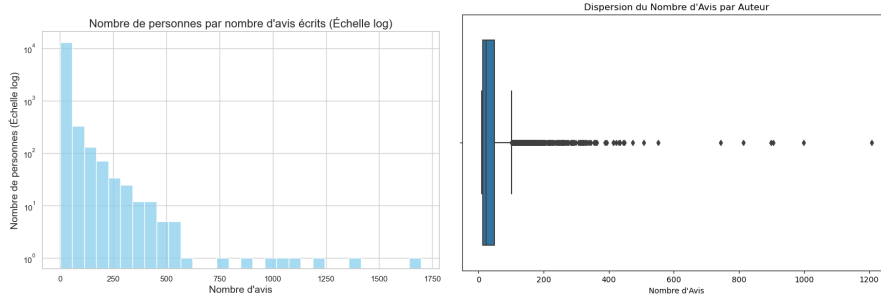


Figure 5: Travail sur les notes et les auteurs

faible écart type ont des évaluations plus cohérentes, tandis que ceux avec un écart type élevé montrent plus de variabilité dans les notes. De plus, on remarque que la plupart des notes se situent entre 6 et 10, il serait donc plus judicieux de prendre les avis et commentaires de ces notes-là.

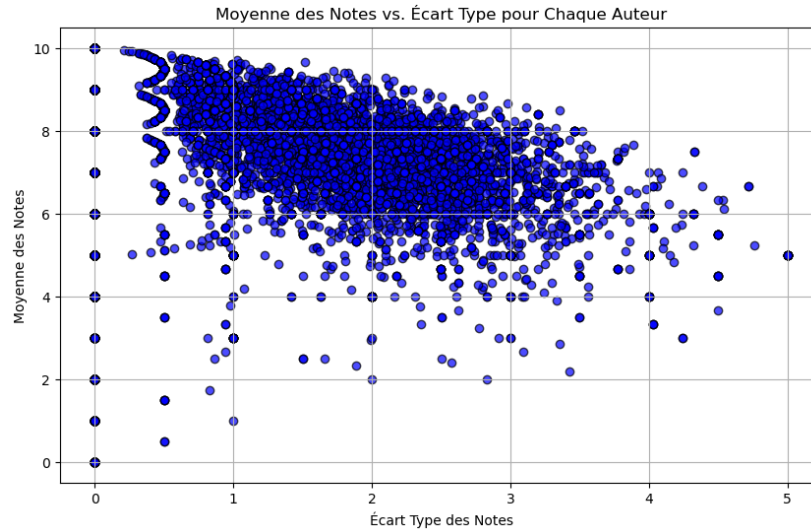


Figure 6

Enfin, nous avons examiné la densité des moyennes des notes et l'écart type pour chaque auteur afin d'avoir une meilleure compréhension de ces données *figure 7*.

La compréhension approfondie des données nous guide vers les étapes spécifiques de pré-traitement. Pour les commentaires des utilisateurs, par exemple, nous procédons à la sélection des commentaires les plus pertinents et à l'élimination des éléments redondants ou non significatifs, tels que les verbes d'action communs et les stop words. Nous appliquons également des techniques de traitement du langage naturel pour extraire des thèmes significatifs des commentaires, ce qui est crucial pour comprendre les opinions et les préférences des utilisateurs (cf partie 4).

Ce processus détaillé de pré-traitement et d'analyse est fondamental pour s'assurer que les recommandations finales soient non seulement précises mais aussi pertinentes et explicables pour l'expérience utilisateur. Cela aide à construire un système robuste capable de s'adapter et d'évoluer en fonction des comportements et des préférences changeants des utilisateurs, tout en étant transparent dans son fonctionnement.

Pour garantir l'efficacité de nos recommandations, il est crucial de commencer par une phase rigoureuse de pré-traitement des données. Ce processus inclut initialement la suppression des

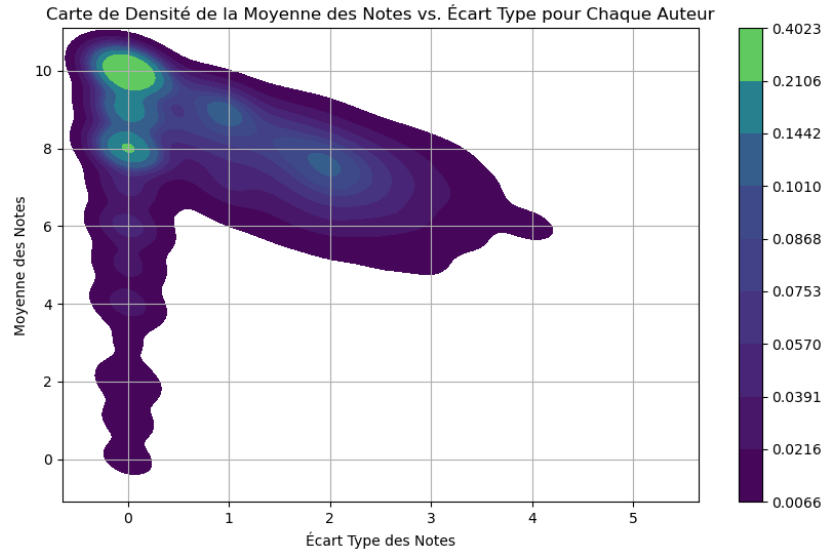


Figure 7

doublons et des colonnes superflues qui ne contribueraient pas à l'efficacité de nos algorithmes de recommandation. Cette étape est essentielle pour assurer une base de données propre et manipulable.

Après le travail de pré-traitement et le processing effectué sur la base de données, nous nous retrouvons avec un total de 13 623 auteurs d'avis, 10 709 jeux et 176 243 avis différents.

3 Méthodes de recommandation

Dans cette partie, nous allons expliquer la différence entre les deux méthodes de recommandation les plus répandues ainsi que discuter des différents résultats obtenus en appliquant ces derniers sur nos données.

3.1 Algorithmes de recommandation

3.1.1 Recommandation basée sur le contenu (content based)

"Content-based filtering (CB) recommends items which are similar to the ones a given user has liked in the past." [KS16]

Le filtrage basé sur le contenu est une méthode de recommandation qui utilise les détails des items eux-mêmes pour recommander d'autres items similaires. Cette technique s'appuie sur une comparaison détaillée des attributs des items pour faire des suggestions. Dans le cadre des jeux de société, cela pourrait signifier l'utilisation de caractéristiques telles que le genre du jeu, la complexité, le nombre de joueurs recommandé, ou même le thème pour aligner les préférences des utilisateurs avec des jeux ayant des attributs similaires. L'approche basée sur le contenu permet de générer des recommandations en exploitant les informations détaillées disponibles sur chaque jeu.

Limitations pour Notre Base de Données Toutefois, l'application du filtrage basé sur le contenu dans notre projet présente des limitations notables, principalement en raison de la nature de notre base de données. Pour une implémentation efficace du filtrage basé sur le contenu, il est essentiel de disposer d'une quantité riche et variée de métadonnées sur chaque item. Dans notre cas, la base de données de jeux de société ne possède pas suffisamment de contenu descriptif détaillé qui permettrait d'expliquer les recommandations.

Dans la partie précédente, nous avons examiné pourquoi les informations disponibles sur les jeux de société ne se prêtent pas efficacement à un filtrage basé sur le contenu. Les limitations principales incluent le manque de données détaillées sur chaque jeu, un nombre insuffisant d'avis par jeu, ainsi qu'une dispersion trop grande des catégories. En effet, pour rendre les catégories utilisables dans un algorithme de filtrage basé sur le contenu, un travail conséquent de consolidation serait nécessaire, ce qui rend l'approche peu pratique.

De plus, les données concernant les créateurs ou le "casting" des jeux ne sont pas non plus propices à cette méthode. Le traitement des types de données nécessaires pour exploiter ces informations est complexe, et la fréquence à laquelle les mêmes créateurs apparaissent sur différents jeux est trop faible pour permettre des recommandations significatives et précises. En outre, l'utilisation des catégories pour faire des recommandations, bien que conceptuellement simple, ne conduirait pas à des suggestions suffisamment personnalisées pour être intéressantes. Une telle approche risquerait de simplifier excessivement les recommandations, les rendant prévisibles et génériques plutôt que pertinentes et adaptées aux préférences individuelles des utilisateurs.

En conséquence, bien que le filtrage basé sur le contenu soit une méthode valable pour de nombreux scénarios, nous avons conclu qu'il ne convient pas idéalement à notre projet en raison des limites spécifiques de notre base de données et des exigences en matière d'explicabilité. Cela nous amène à explorer d'autres méthodes de recommandation qui pourraient être mieux adaptées à nos objectifs et à notre environnement de données.

Cette réalisation nous conduit naturellement à explorer le filtrage collaboratif, une méthode alternative qui pourrait mieux correspondre aux besoins de notre projet en tirant parti des interactions entre utilisateurs.

3.1.2 Recommandation basée sur le filtrage collaboratif (collaborative filtering)

"Collaborative filtering (CF) technique recommends the items to the target user on the basis of past preferences of other users with similar tastes." [KS16]

Le filtrage collaboratif est l'une des approches les plus utilisées dans les systèmes de recommandation. Cette méthode repose sur le principe de recommandations générées en fonction des comportements et des préférences similaires d'autres utilisateurs. Contrairement au filtrage basé sur le contenu, qui nécessite des informations détaillées sur les attributs des items, le filtrage collaboratif tire parti des interactions et des évaluations des utilisateurs pour prédire les intérêts d'un utilisateur en fonction des intérêts des autres.

Notre base de données, riche en avis et interactions des utilisateurs sur différents jeux, se prête particulièrement bien à cette méthode. Avec un volume conséquent de données sur les préférences des utilisateurs, nous sommes en position idéale pour appliquer et tirer profit du filtrage collaboratif. Cette approche nous permet d'utiliser les évaluations existantes pour connecter les utilisateurs à des jeux similaires appréciés par des profils aux goûts similaires.

Parmi les différents types d'algorithmes de filtrage collaboratif, nous avons choisi de nous concentrer principalement sur l'algorithme des k plus proches voisins (kNN) pour notre projet. Le choix de kNN est motivé par sa capacité à identifier les utilisateurs qui ont des préférences similaires basées sur leurs évaluations passées. L'algorithme kNN fonctionne en calculant la similarité entre les utilisateurs ou les items en utilisant des mesures telles que la distance euclidienne ou la similarité cosinus, puis en recommandant des items en fonction des évaluations des "voisins" les plus similaires.

En parallèle, nous explorerons également l'usage de la décomposition en valeurs singulières (SVD), une méthode puissante pour la réduction de dimensionnalité qui aide à améliorer la précision des prédictions en traitant les données manquantes ou en extrayant des caractéristiques latentes des interactions des utilisateurs. L'utilisation de SVD est intéressante pour sa capacité à traiter efficacement de grands ensembles de données et à découvrir des structures sous-jacentes complexes dans les comportements des utilisateurs.

3.2 Algorithmes spécifiques du filtrage collaboratif

3.2.1 SVD

La décomposition en valeurs singulières (ou *SVD de l'anglais Singular Values Decomposition*) est un procédé d'algèbre linéaire et un outil de factorisation de matrice rectangulaire réelles ou complexes. Soit M une matrice de taille $m \times n$, il existe une factorisation de la matrice M de la forme :

$$M = U\Sigma V^*$$

avec U une matrice unitaire $m \times m$, Σ une matrice $m \times n$ dont les coefficients diagonaux sont des réels positifs ou nuls et tous les autres sont nuls, et V^* est la matrice adjointe à V , matrice unitaire $n \times n$.

Comme dit précédemment, nous avons pu commencer à obtenir des premiers résultats sur cet algorithme suite à la mise en place de notre baseline. Nous sommes actuellement en train d'explorer plus en détail ces résultats qui nous seront cruciaux pour la suite.

Pour en savoir plus sur les [SVD](#).

3.2.2 KNN

La méthode des K plus proches voisins (KNN) a pour but de classer des points cibles en fonction de leurs distances par rapport à des points constituant un échantillon d'apprentissage (c'est-à-dire dont la classe est connue a priori).

Utilisée dans le cadre du machine learning, il s'agit d'une généralisation de la méthode du voisin le plus proche (NN). NN est un cas particulier de KNN, où $k = 1$.

Une fois implémentée, cette méthode va nous permettre de mieux expliciter les différentes recommandations aux utilisateurs.

Il y a bien entendu d'autres classifieurs / algorithmes sur lesquels nous comptons travailler tels que les NMF, qui utilisent le score comme description, cherchent les utilisateurs similaires et trouvent les k plus proches profils.

Pour en savoir plus sur les [kNN](#).

3.3 Focus kNN

L'approche kNN s'est révélée particulièrement efficace pour notre jeu de données, en identifiant avec précision les utilisateurs aux préférences similaires à partir de leurs évaluations passées. Cette méthode, qui utilise des mesures de similarité comme la distance euclidienne ou la similarité cosinus, nous permet de recommander des jeux qui correspondent étroitement aux intérêts des utilisateurs en se basant sur les choix de leurs "voisins" les plus proches.

3.3.1 Détails du jeu de données et similarité cosinus

Dans un premier temps, nous avons vérifié les différentes données de notre base de données après traitement des outliers (valeurs extrêmes) et tout le travail de pré-traitement de la partie précédente. Nous avons identifié un total de 2458 auteurs et 2911 jeux dans notre base de données. Notre matrice a une sparsité de 1.64%, ce qui est tout à fait normal.

Par la suite, nous avons calculé la similarité cosinus entre chaque utilisateur. Cette mesure donne la similarité de deux vecteurs à n dimensions en déterminant le cosinus de leur angle. Cette visualisation nous permet non seulement de comprendre les similarités entre les différents utilisateurs, mais sera également essentielle pour expliquer les recommandations spécifiques faites à chaque utilisateur. Elle fournira une base visuelle pour justifier pourquoi certains jeux sont suggérés, en mettant en lumière les connexions entre les préférences des utilisateurs similaires *figure 8*.

Cette visualisation nous permettra non seulement de comprendre les similarités entre les différents utilisateurs mais sera également essentielle pour expliquer les recommandations spécifiques faites à chaque utilisateur. Elle fournira une base visuelle pour justifier pourquoi certains jeux

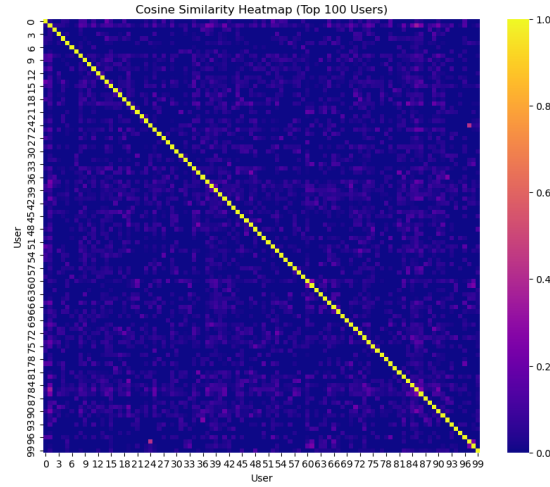


Figure 8

sont suggérés, en mettant en lumière les connexions entre les préférences des utilisateurs similaires.

3.3.2 Calcul MAE & proportions

- EXPLIQUER L'ALGO KNN Nous avons achevé le calcul de la matrice de similarité cosinus entre les utilisateurs, ce qui constitue une étape cruciale pour comprendre et visualiser les similarités dans les préférences des utilisateurs de notre base de données. En plus de la similarité cosinus, nous avons également évalué l'exactitude de nos prédictions à travers l'erreur moyenne absolue (MAE) pour différentes plages de notes. Ces valeurs de MAE nous permettent de mesurer la précision des recommandations fournies par notre système et d'identifier les plages de notes où le système performe mieux ou moins bien.

Les résultats obtenus sont les suivants :

- MAE pour les notes entre 3 et 4 : 2.81
- MAE pour les notes entre 4 et 5 : 2.39
- MAE pour les notes entre 5 et 6 : 1.29
- MAE pour les notes entre 6 et 7 : 1.07
- MAE pour les notes entre 7 et 8 : 0.72
- MAE pour les notes entre 8 et 9 : 0.76
- MAE pour les notes entre 9 et 10 : 1.48
- MAE pour les notes entre 10 et 11 : 1.70

Ces valeurs indiquent une tendance générale où le système semble avoir une meilleure précision dans les plages de notes moyennes (6 à 9), avec une augmentation de l'erreur aux extrémités des échelles de notation. Cela pourrait suggérer que notre système est particulièrement efficace pour recommander des jeux qui reçoivent des évaluations modérément positives ou négatives, mais il pourrait être moins précis avec des évaluations extrêmement hautes ou basses, possiblement en raison de la rareté des données dans ces extrêmes.

Ces insights offrent une base solide pour expliquer les recommandations aux utilisateurs. En démontrant comment le système performe à différents niveaux de notation, nous pouvons mieux justifier et clarifier les recommandations faites, améliorant ainsi la transparence et la confiance dans le système.

L'efficacité de l'approche kNN, en particulier, offre non seulement une précision de recommandation élevée, mais également une explicabilité accrue. Elle permet de tracer clairement les chemins de recommandation, ce qui est essentiel pour renforcer la confiance des utilisateurs dans les recommandations du système, en leur montrant des exemples concrets de recommandations basées sur les préférences de personnes similaires.

L'analyse de la distribution des notes dans notre base de données révèle des tendances intéressantes qui sont essentielles pour comprendre les comportements des utilisateurs et optimiser notre système de recommandation. Les proportions des notes attribuées aux jeux de société sont les suivantes : *figure 9*.

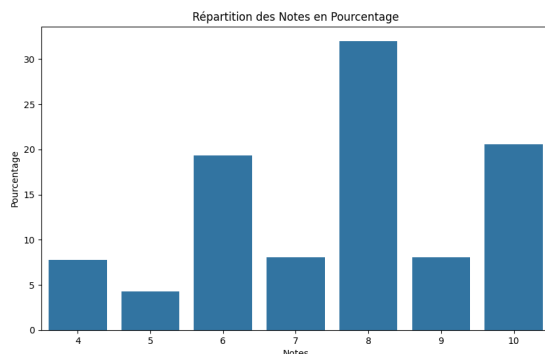


Figure 9

Cette distribution montre une concentration significative de notes autour de **8** et **10**, qui représentent respectivement **32.01%** et **20.56%** de toutes les évaluations. Cela suggère que les utilisateurs sont généralement satisfaits des jeux qu'ils évaluent, avec une tendance à donner des notes élevées. Les notes intermédiaires, telles que **6** et **7**, sont également courantes, indiquant une répartition variée des opinions.

Cette information est particulièrement pertinente car elle montre non seulement la tendance des utilisateurs à évaluer positivement, mais aussi l'importance de bien comprendre la dynamique des notes moyennes et élevées pour mieux cibler les recommandations. En prenant en compte cette distribution, nous pouvons ajuster nos algorithmes pour mieux répondre aux préférences exprimées par les utilisateurs et ainsi améliorer l'efficacité des recommandations.

La compréhension de ces tendances nous aide également à interpréter les valeurs d'erreur moyenne absolue (MAE) obtenues précédemment, en reliant les performances de notre modèle à la fréquence des différentes notes. Cette approche nous permet de fournir des recommandations plus précises et plus personnalisées, renforçant la satisfaction des utilisateurs et leur confiance dans notre système.

3.3.3 Différences des différents kNN

Dans notre évaluation continue des performances de divers algorithmes de filtrage collaboratif basés sur la similarité cosinus, nous avons appliqué plusieurs variantes du modèle kNN (k-Nearest Neighbors) sur notre base de données. Chaque variante a été testée pour mesurer l'erreur moyenne absolue (MAE) entre les notes prévues et réelles, fournissant une perspective comparative sur leur efficacité.

Résultats des Évaluations des Algorithmes kNN

Nous avons choisi d'utiliser les différents algorithmes kNN qu'offre la bibliothèque [Surprise](#) de [Scikit-Learn](#). Cette décision nous permet de pouvoir comparer les différents résultats et ainsi pouvoir choisir le meilleur algorithme.

Les résultats de nos tests montrent que, parmi les différentes variantes de l'algorithme kNN utilisées, kNNWithZScore se distingue comme étant le plus performant *figure 10*. Cette méthode, qui ajuste les évaluations en fonction des moyennes globales des utilisateurs, a démontré une capacité supérieure à améliorer la précision des prédictions, surtout dans les plages de notes intermédiaires. Grâce à cet ajustement, kNNWithZScore offre un meilleur équilibre entre les différentes plages de notes, ce qui suggère une capacité accrue à gérer les biais de notation inhérents à notre base de données.

Ce constat renforce l'argument pour utiliser kNNWithZScore comme l'approche principale dans notre système de recommandation, non seulement pour sa précision mais aussi pour

l'explicabilité accrue qu'il offre. Cela permet de fournir des recommandations plus fiables et justifiables, augmentant ainsi la confiance des utilisateurs dans les suggestions du système.

La qualité des recommandations étant assurée, la prochaine étape consiste à analyser comment expliquer ces recommandations aux utilisateurs.

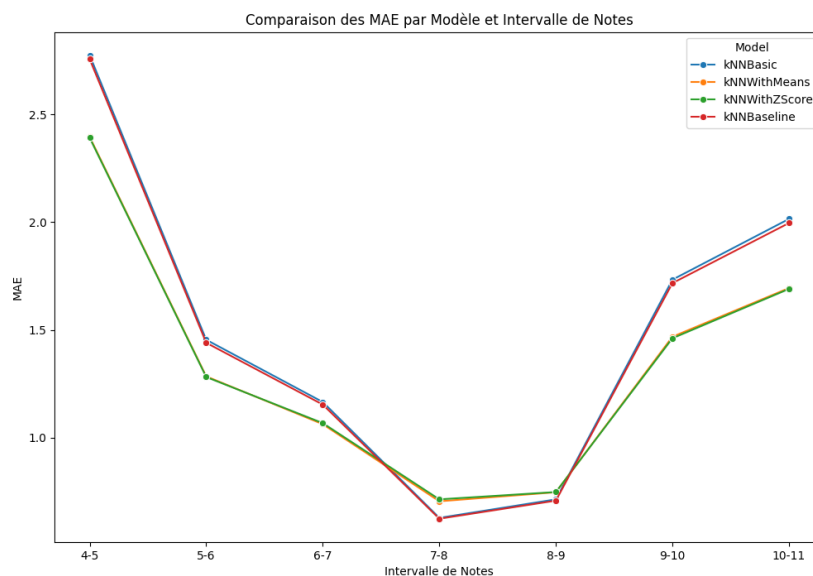


Figure 10

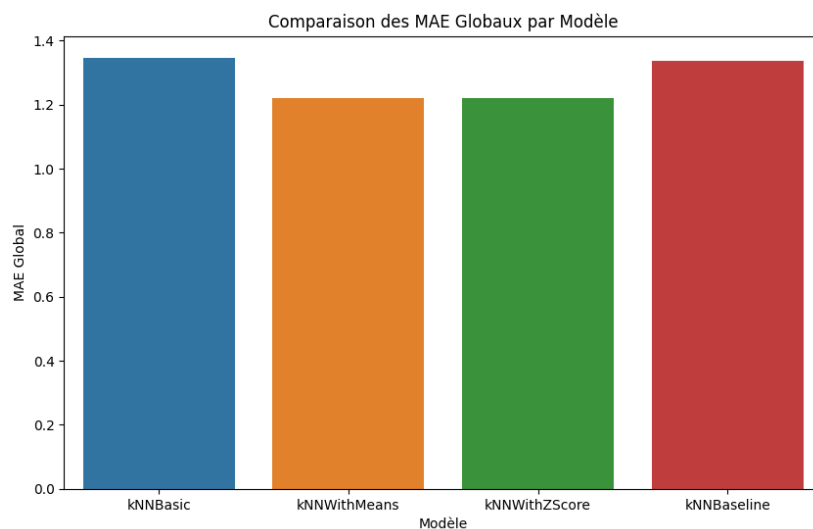


Figure 11

4 Explicabilité des recommandations

L'explicabilité est abordée en comparant deux approches : une première utilisant un modèle de résumé automatique (comme ChatGPT) pour synthétiser les commentaires, et une seconde basée sur des méthodes algorithmiques telles que TF-IDF et le topic modeling. Nous détaillons la méthodologie de sélection des commentaires et l'application des pipelines pour chaque approche.

4.1 Objectif

L’objectif principal de ce protocole est d’évaluer la pertinence et l’utilité des méthodes d’extraction de mots-clés appliquées aux commentaires des utilisateurs pour expliquer les recommandations de jeux. En utilisant les commentaires des utilisateurs proches, nous cherchons à déterminer si les mots-clés extraits peuvent fournir des informations précieuses qui expliquent pourquoi un jeu est recommandé à un utilisateur particulier. Cela permet d’améliorer la transparence et la confiance dans le système de recommandation.

4.2 Protocole Expérimental

Le processus débute par la configuration de l’environnement de développement en Python, avec l’importation de bibliothèques essentielles telles que **pandas** pour la manipulation de données, **spacy** et **nltk** pour le traitement du langage naturel, et **surprise** pour les algorithmes de recommandation. Nous avons utilisé le modèle linguistique français **fr_core_news_sm** de **spaCy**, crucial pour le traitement préliminaire des textes en français.

Le prétraitement des données, discuté dans les parties précédentes du rapport, inclut la suppression des stop-words, la ponctuation, et la lemmatisation ou stemmatisation. Ces techniques préparent les données textuelles pour une analyse plus approfondie et sont essentielles pour l’extraction efficace des mots-clés.

Entraînement de l’Algorithme de Filtrage Collaboratif Nous avons choisi l’algorithme **KNNWithZScore** de la bibliothèque **Surprise** pour implémenter le filtrage collaboratif. Cet algorithme utilise la similarité cosinus pour évaluer les similitudes entre les utilisateurs, permettant ainsi des recommandations plus personnalisées et précises.

Fonctionnalités de Traitement des Commentaires Des fonctions spécifiques ont été développées pour filtrer et traiter les commentaires des utilisateurs afin de les rendre pertinents pour l’analyse. Cela inclut des méthodes pour filtrer les commentaires en fonction du nombre de mots, garantissant que seules les contributions substantielles sont prises en compte. Une fonction récupère et classe les commentaires les plus pertinents des utilisateurs voisins pour des jeux spécifiques, en tenant compte de la fréquence des mots et de la pertinence du contenu.

Méthodes Avancées de Traitement du Langage Naturel

4.2.1 Méthodes d’Extraction de Mots-Clés

Nous avons testé les méthodes d’extraction de mots-clés suivantes :

- **Baseline** (Comptage de mots fréquents) : *Compte les occurrences de chaque mot dans un texte, identifiant les mots les plus fréquents comme mots-clés potentiels.*
- **TF-IDF** : *Modèle statistique qui découvre des sujets cachés dans un corpus de textes en regroupant les mots qui coapparaissent fréquemment.*
- **TextRank** : *Algorithme de graphe qui détermine l’importance des mots dans un texte en utilisant des techniques de classement similaires à celles utilisées par les moteurs de recherche.*

Chaque méthode a été appliquée aux commentaires sélectionnés pour extraire les mots-clés, permettant une comparaison détaillée de leur efficacité.

4.2.2 Évaluation de la Pertinence

Correspondance avec les Commentaires Réels Nous avons comparé les mots-clés extraits à ceux présents dans les commentaires réels des utilisateurs pour évaluer leur pertinence et leur alignement.

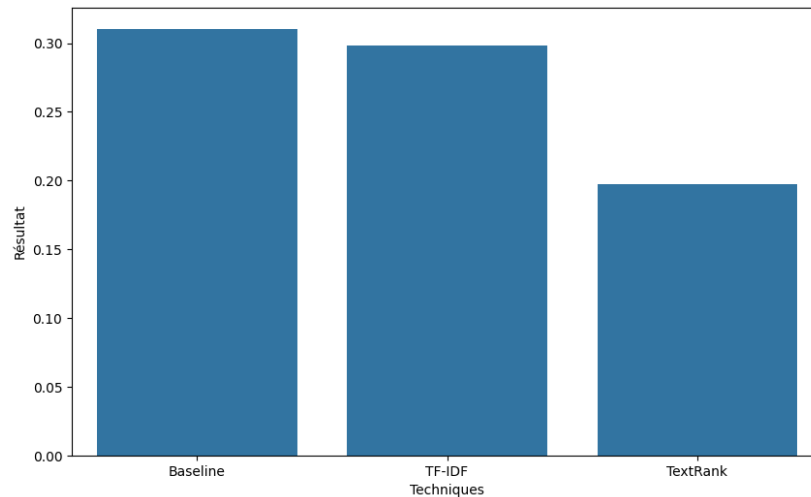


Figure 12: Comparaison des techniques d'extraction de mots

4.2.3 Analyse des Résultats - Taux d'apparition

Baseline (0.3105) Ce taux indique que les mots-clés extraits par la méthode Baseline apparaissent dans environ 31% des cas dans les commentaires testés. Le succès relativement élevé de cette méthode pourrait indiquer que les mots les plus fréquents (qui sont souvent choisis dans la méthode Baseline) sont effectivement pertinents pour le contexte des commentaires. Cela peut être dû à une concentration de sujets ou de termes fréquemment discutés ou mentionnés dans les commentaires.

TF-IDF (0.2986) Un peu moins performante que la méthode Baseline, TF-IDF montre tout de même que près de 30% des mots-clés qu'elle identifie sont présents dans les commentaires. TF-IDF est reconnu pour sa capacité à identifier les mots qui sont importants pour un document mais pas nécessairement communs dans tous les documents (ou commentaires, dans ce cas). Le taux légèrement inférieur par rapport à Baseline pourrait indiquer que TF-IDF capte des termes un peu plus spécifiques qui ne se répètent pas autant à travers les différents commentaires.

TextRank (0.1971) Avec un taux d'apparition de près de 20%, TextRank montre une performance inférieure aux deux autres méthodes. TextRank, étant un algorithme basé sur le graphe qui tente d'extraire des mots-clés en fonction de leur importance structurelle dans le texte, pourrait ne pas aligner aussi efficacement les mots-clés avec le contenu réel discuté, surtout dans les textes où les connexions contextuelles sont moins évidentes ou plus diffuses.

Afin d'avoir une meilleure visibilité de ces résultats voir la *figure 12*.

4.2.4 Interprétation et Recommandations

Pertinence des Mots-Clés Les résultats montrent que les méthodes Baseline et TF-IDF capturent des mots-clés plus alignés avec les fréquences de discussion dans les commentaires. En revanche, TextRank semble moins performant, probablement en raison de la nature structurelle de son approche, qui pourrait nécessiter une révision ou une adaptation pour mieux saisir les mots-clés contextuellement importants.

Choix de Méthode Si la fréquence d'apparition est un indicateur clé de la pertinence, Baseline et TF-IDF sont préférables pour notre contexte spécifique. Cependant, si la diversité des mots-clés et la couverture de sujets uniques sont plus critiques, il peut être bénéfique d'examiner plus en détail pourquoi TextRank est moins performant et comment ses résultats pourraient être complémentaires plutôt que directement comparables.

Suggestions d’Améliorations Nous avons proposé des ajustements ou des combinaisons de méthodes pour améliorer l’extraction de mots-clés et la génération d’explications.

Cette première sous-partie établit donc une base solide pour comprendre les mécanismes techniques et algorithmiques mis en œuvre dans notre système de recommandation. Elle pose également les fondations pour la deuxième sous-partie, qui se concentrera sur l’analyse des résultats obtenus à partir de ces méthodologies et leur impact sur l’explicabilité des recommandations.

4.3 Discussion

4.3.1 Techniques d’extraction de mots-clés envisagées

Nous avons envisagé plusieurs techniques d’extraction de mots-clés, notamment les algorithmes **LDA** (*Latent Dirichlet Allocation*), **YAKE** (*Yet Another Keyword Extractor*), **RAKE** (*Rapid Automatic Keyword Extractor*) et **KeyBERT**. LDA et YAKE, bien que prometteurs, n’ont pas été testés dans notre étude en raison de la nature organisée des mots-clés extraits par sujet. Mesurer le taux d’apparition des mots-clés aurait été moins significatif dans ce contexte. En revanche, RAKE et KeyBERT ont été testés, mais ils n’ont pas démontré une efficacité satisfaisante dans l’extraction de mots-clés à partir de commentaires. Cela pourrait être dû à la nature spécifique des commentaires, qui peuvent contenir un langage informel, des abréviations ou des phrases incomplètes, ce qui rend leur traitement plus complexe pour ces algorithmes.

4.3.2 Utilisation potentielle de ChatGPT

Une autre piste intéressante à explorer est l’utilisation de ChatGPT pour générer des explications à partir des commentaires les plus pertinents. En analysant les réponses générées par ChatGPT, nous pourrions obtenir des informations supplémentaires sur les sujets discutés dans les commentaires, ainsi que sur les questions ou préoccupations des utilisateurs. Cette approche pourrait compléter l’analyse des mots-clés en fournissant des insights supplémentaires sur le contenu des commentaires et en identifiant les tendances ou les points saillants.

4.3.3 Analyse de la qualité des mots-clés extraits

Enfin, il serait pertinent de mener une analyse approfondie de la qualité des mots-clés extraits par les différentes techniques utilisées. Cette analyse pourrait comprendre une évaluation qualitative, où les mots-clés extraits sont comparés au contenu des commentaires pour évaluer leur pertinence et leur exhaustivité. De plus, une évaluation quantitative pourrait être réalisée en mesurant la précision et le rappel des mots-clés extraits par rapport aux mots-clés de référence, le cas échéant. Cela nous permettrait de mieux comprendre les forces et les faiblesses de chaque méthode d’extraction de mots-clés et d’identifier des pistes d’amélioration pour de futures études.

5 Conclusion

Dans cette étude, nous avons examiné l’explicabilité des systèmes de recommandation en utilisant des méthodes basées sur le filtrage collaboratif et des techniques d’extraction de mots-clés à partir des commentaires des utilisateurs. L’analyse des données a révélé que, bien que le filtrage collaboratif convienne à notre base de données, les méthodes de recommandation basées sur le contenu sont limitées par la nature des informations disponibles. Parmi les algorithmes testés, kNNWithZScore a démontré des performances prometteuses en termes de précision des recommandations.

Nous avons également comparé diverses techniques d’extraction de mots-clés pour améliorer l’explicabilité des recommandations. Les méthodes Baseline et TF-IDF ont montré une meilleure pertinence par rapport à TextRank. Cependant, des améliorations et ajustements sont nécessaires pour optimiser ces techniques, notamment en explorant l’utilisation de modèles comme ChatGPT pour générer des explications plus riches et contextuelles.

En résumé, bien que nous ayons identifié des méthodes efficaces pour améliorer l’explicabilité des systèmes de recommandation, des défis subsistent, notamment en ce qui concerne l’intégration

de techniques avancées et l'adaptation à des données spécifiques. Les perspectives futures incluent l'exploration de nouvelles méthodes et l'application de ces techniques à d'autres domaines pour renforcer la transparence et la confiance des utilisateurs dans les systèmes de recommandation.

References

- [KS16] Balraj Kumar¹ and Neeraj Sharma. Approaches, issues and challenges in recommender systems: A systematic review. *Indian Journal of Science and Technology*, Vol 9(47), DOI: 10.17485/ijst/2016/v9i47/94892, December 2016.