



REPUBLIQUE ALGERIENNE DEMOCRATIQUE ET POPULAIRE
MINISTERE DE L'ENSEIGNEMENT SUPERIEUR ET DE LA RECHERCHE SCIENTIFIQUE

UNIVERSITE IBN KHALDOUN - TIARET

MEMOIRE

Présenté à :

FACULTÉ DES MATHÉMATIQUES ET DE L'INFORMATIQUE
DÉPARTEMENT D'INFORMATIQUE

Pour l'obtention du diplôme de :

MASTER

Spécialité : Génie Logiciel

Par :

BESTANI Imen
BENADJROUDA Nouredine

Sur le thème

Développement des Systèmes de Recommandation de Santé à l'aide des Réseaux Convolutifs Graphiques

Soutenu publiquement le 10 / 07 / 2023 à Tiaret devant le jury composé de :

Dr. MESTFAOUI Sid Ahmed Mokhtar	MCA Université Tiaret	Président
Dr. BOUDAA Boudjema	MCA Université Tiaret	Encadrant
Dr. KHARROUBI Sahraoui	MCA Université Tiaret	Examineur

2022-2023

Résumé

Les systèmes de recommandation fournissent des suggestions utiles d'articles (produits ou services) aux utilisateurs dans le cadre de leurs processus de prise de décision. L'efficacité des systèmes de recommandation est désormais clairement confirmée dans divers domaines d'application (par exemple, YouTube, Amazon, Facebook, ResearchGate). Dans la littérature, de nombreux travaux de recherche ont abordé l'application des recommandations dans le domaine de la santé, ce que l'on appelle les systèmes de recommandation de santé (HRS en anglais). Les HRS constituent une alternative novatrice lorsqu'il s'agit de fournir des informations pour aider les médecins dans le diagnostic/traitement des maladies, ainsi que d'aider les patients avec des recommandations sur la manière de maintenir leur bien-être. Cependant, les approches de développement existantes dans la littérature se limitent à l'utilisation des modèles traditionnels qui manquent de précision et d'efficacité, et qui sont essentielles dans le domaine des soins de santé. Ce travail présente un modèle de conception pour les systèmes de recommandation de santé en utilisant les réseaux de neurones graphiques (GNN) via son architecture de réseaux convolutifs graphiques (GCN). Dans ce modèle, la couche de convolution est gérée par un algorithme simplifié et efficace appelé LightGCN. Les méthodes basées sur GCN font partie des approches de pointe dans les systèmes de recommandation, et LightGCN a prouvé sa supériorité en termes de précision des recommandations. De manière inédite et efficace, le présent travail a réussi d'appliquer les réseaux convolutifs graphiques via le modèle LightGCN dans le développement des systèmes de recommandation de santé.

Mots-clés : Système de recommandation, Santé, Système de recommandation de santé, Réseau convolutif graphique, LightGCN.

Abstract

Recommendation systems provide valuable suggestions for articles (products or services) to users as part of their decision-making processes. The effectiveness of recommendation systems has been clearly demonstrated in various application domains, such as YouTube, Amazon, Facebook, and ResearchGate. In the literature, numerous research studies have addressed the application of recommendations in the healthcare domain, known as Health Recommender Systems (HRS). HRS offer an innovative alternative for providing information to assist physicians in disease diagnosis/treatment and helping patients with recommendations on maintaining their well-being. However, existing development approaches in the literature are limited to the use of traditional models that lack accuracy and efficiency, which are essential in the healthcare field. This work presents a design model for health recommender systems using Graph Neural Networks (GNNs) through its Graph Convolutional Networks (GCN) architecture. In this model, the convolution layer is handled by a simplified and efficient algorithm called LightGCN. GCN-based methods are at the forefront of recommendation systems approaches, and LightGCN has proven its superiority in terms of recommendation accuracy. Uniquely and efficiently, this study successfully applies Graph Convolutional Networks through the LightGCN model in the development of health recommender systems.

Key-words : Recommender System, Healthcare, Health Recommender System, Graph Convolutional Network, LightGCN

Remerciements

Nous tenons tout d'abord à remercier le bon Dieu de nous avoir donné la force et le courage de mener ce projet à terme, ainsi que de nous avoir permis de surmonter toutes les difficultés. Nous avons ressenti sa guidance chaque jour, et c'est grâce à lui que nous avons pu réussir nos études. Nous continuerons à lui faire confiance pour notre avenir.

*Nous souhaitons exprimer nos sincères remerciements à notre encadrant de mémoire, **M. Boudaa Boudjema**. Nous tenons également à lui témoigner notre gratitude pour sa patience et son soutien précieux, qui nous ont aidés à mener notre travail à bien. Ses conseils avisés et son assistance tout au long de cette période ont été d'une grande aide.*

Nous souhaitons également remercier les membres du jury et nos professeurs pour la qualité de leur enseignement tout au long de ces cinq années passées au département d'informatique. Leur contribution a été essentielle à notre formation.

Enfin, nous tenons à exprimer notre reconnaissance sincère et profonde envers nos familles, qui nous ont toujours soutenu et nous ont donné les moyens de poursuivre nos aspirations. Leur confiance en nous a été une source d'inspiration constante. Nous souhaitons également remercier tous ceux qui ont contribué, de près ou de loin, à la réalisation de ce mémoire.

Dédicaces

*Je tiens tout d'abord à exprimer ma gratitude envers Allah, qui a facilité mon parcours jusqu'à la réalisation de ce modeste travail, et sans lui je n'aurais rien pu accomplir. Je souhaite dédier ce travail et exprimer ma profonde reconnaissance à mes parents car ils m'ont inculqué un esprit de persévérance et ont toujours été présents pour me pousser et me motiver dans mes études, à mes chères sœurs et mon cher frère, à tous mes amis et toute personne m'ayant aidé de près ou de loin, à Mon collègue : **Benadjrouda Noureddine**.*

BESTANI Imen

Avant tout, je remercie Allah le Tout-Puissant de m'avoir donné le courage et la patience nécessaires pour accomplir ce travail malgré toutes les difficultés rencontrées.

Je dédie humblement ce travail :

À tous ceux qui m'ont soutenu, aidé ou motivé dans cette vie, à tous ceux qui ont cru en moi à chaque instant, à mes parents, amis et famille.

À mes chers parents, il n'y a pas assez de mots pour vous dire à quel point je vous suis reconnaissant. Merci pour tout ce que vous avez fait pour moi.

*À mes cousins et ma grande famille, à mes chers amis, à tous ceux que j'aime et qui m'aiment, à tous mes collègues d'études, à mon binôme **Bestani Imen**.*

BENADJROUDA Noureddine

Table de matière

Introduction Générale.....	2
1. Contexte.....	2
2. Problème.....	2
3. Approche	2
4. L'organisation du mémoire	2
Chapitre 1 : Les systèmes de recommandation de santé	4
1.1 Introduction	4
1.2 Les systèmes de recommandation	4
1.2.1 Définitions	4
1.2.2 Exemples	5
1.2.3 Classification des systèmes de recommandation	5
1.2.3.1 La classification classique	5
1.2.3.2 La Classification de [Su et al, 2009].....	5
1.2.3.3 La classification de Burke et La classification de [Rao et al, 2008].....	6
1.2.4 Les approches de systèmes de recommandation	6
1.2.4.1 Approches basées sur le contenu	7
1.2.4.2 Approches basées sur le filtrage collaboratif.....	8
1.2.4.3 Approches basées sur le filtrage hybride	10
1.2.5 Évaluation des systèmes de recommandation	10
1.2.6 Les défis des systèmes de recommandation	12
1.2.6.1 Démarrage à froid	12
1.2.6.2 La difficulté d'obtenir suffisamment de données de qualité.....	12
1.2.6.3 La diversité des goûts et des préférences	12
1.2.6.4 Parcimonie (Sparsity)	12
1.2.6.5 La rareté des données (scarcity)	13
1.3 Systèmes de recommandation de santé	13
1.3.1 Définition.....	13
1.3.2 Les scénarios de système de recommandation de santé	13
1.3.2.1 Recommandation alimentaire	13
1.3.2.2 Recommandation de médicaments	14
1.3.2.3 Prédiction de l'état de santé	14
1.3.2.4 Recommandation d'activité physique.....	14
1.3.2.5 Recommandations des professionnels de santé.....	14
1.3.3 Architecture du système de recommandation de santé.....	15
1.3.4 Les défis de la recherche sur les systèmes de recommandation de santé	16
1.3.4.1 Défis éthiques	16
1.3.4.2 Données de santé de qualité.....	16
1.3.4.3 La prolifération des normes de santé	17
1.3.4.4 Fiabilité.....	17
1.3.4.5 Précision	17
1.3.4.6 Le désaccord des patients et de leurs proches	17
1.3.4.7 Autorisation et accès aux dossiers médicaux électroniques	17

Table de matière

1.3.4.8	Convaincre les gens	17
1.3.4.9	Les aspects légaux	18
1.3.5	Un cadre de travail pour les systèmes de recommandation de santé.....	18
1.4	Conclusion.....	19
Chapitre 2 : Les réseaux convolutifs graphiques		21
2.1	Introduction	21
2.2	Fondement de l'IA, du ML et du DL.....	21
2.2.1	Définition de l'IA	21
2.2.2	Historique	21
2.2.3	Définition de l'apprentissage automatique	22
2.2.4	Types d'apprentissage automatique supervisé, non supervisé, renforcement	23
2.2.4.1	L'apprentissage automatique supervisé	23
2.2.4.2	L'apprentissage automatique non supervisé.....	24
2.2.4.3	L'apprentissage par renforcement	24
2.2.5	Les avantages de l'utilisation du Deep Learning.....	25
2.2.6	L'Apprentissage Profond	25
2.2.6.1	Définition.....	25
2.2.6.2	Évolution de l'apprentissage profond.....	26
2.3	Théorie des graphes.....	27
2.3.1	Définition des graphes	27
2.3.2	Matrice d'adjacence.....	28
2.3.3	Types de graphe orienté, non orienté, biparti	28
2.3.3.1	Graphe orienté	28
2.3.3.2	Graphe non orienté	29
2.3.3.3	Graphe biparti	29
2.4	Réseaux de neurones graphiques.....	30
2.4.1	Définition.....	30
2.4.2	Avantages et inconvénients des GNN.....	30
2.4.2.1	Avantages.....	31
2.4.2.2	Inconvénients	31
2.4.3	L'apport des réseaux de neurones graphiques pour les systèmes de recommandation	31
2.4.4	Fonctionnement des réseaux de neurones graphiques	32
2.4.4.1	Construction du graphe.....	32
2.4.4.2	Initialisation des embeddings	32
2.4.4.3	La technique de message passing	32
2.4.4.4	La mise à jour des embeddings.....	33
2.4.4.5	Les embeddings de nœud.....	33
2.4.4.6	La répétition du message passing	33
2.4.4.7	L'invariance par permutation	33
2.4.5	Architectures des réseaux de neurones graphiques	34
2.4.5.1	Réseaux convolutifs graphiques (GCN)	34
2.4.5.2	Réseaux d'attention graphiques (GAT)	34
2.4.5.3	Réseaux de neurones à propagation de messages (MPNN).....	34
2.4.6	Les différents domaines pour l'application de GNN.....	34

Table de matière

2.4.6.1	Recommandation de produits	34
2.4.6.2	Classification d'images et de vidéos	34
2.4.6.3	Prédiction de la structure des molécules.....	34
2.4.6.4	Analyse de sentiments	34
2.4.6.5	Détection de fraude.....	34
2.4.7	Les problèmes résolus par les GNN	35
2.4.7.1	Classification des nœuds.....	35
2.4.7.2	Classification des graphes.....	35
2.4.7.3	Visualisation des graphes.....	35
2.4.7.4	Prédiction de liens.....	35
2.4.7.5	Regroupement de graphes.....	35
2.4.8	L'importance de GNN.....	36
2.5	Réseaux convolutifs graphiques (GCN).....	36
2.5.1	Définition de GCN.....	36
2.5.2	Diffusion de l'information dans les graphes avec GCN	37
2.5.2.1	Initialisation	37
2.5.2.2	Agrégation de voisinage	37
2.5.2.3	Mise à jour	37
2.5.2.4	Répétition.....	37
2.5.2.5	Utilisation des activations finales	38
2.5.3	Les catégories de réseaux Graphiques convolutifs.....	38
2.5.3.1	Les réseaux convolutifs à graphes spatiaux.....	38
2.5.3.2	Les réseaux convolutifs à graphes spectraux.....	39
2.5.4	Différence entre Réseaux de neurones convolutifs et réseaux Graphiques convolutifs	39
2.6	Conclusion.....	40
Chapitre 3 : Les systèmes de recommandation de santé basés sur LightGCN		42
3.1	Introduction	42
3.2	Conception de GCN pour HRS basé sur FC	42
3.2.1	Pourquoi les réseaux convolutifs graphiques pour les HRS	42
3.2.1.1	Traitement des données non euclidiennes.....	42
3.2.1.2	Connectivité d'ordre élevé	42
3.2.1.3	Incorporation de plusieurs comportements.....	43
3.2.2	Le modèle proposé pour les systèmes de recommandation de santé.....	43
3.3	Outils de mise en œuvre	45
3.3.1	Python.....	45
3.3.2	Colaboratory	46
3.3.3	Bibliothèques	46
3.3.3.1	PyTorch Geometric	46
3.3.3.2	Matplotlib	46
3.3.3.3	Pandas	47
3.3.3.4	Numpy	47
3.3.3.5	Scikit-learn.....	47
3.3.4	Environnement d'exécution.....	48
3.4	Expérimentation	48

Table de matière

3.5 Résultats et discussion.....	53
3.5.1 Résultats.....	53
3.5.2 Comparaison.....	55
3.5.3 Discussion.....	55
3.6 Conclusion.....	57
Conclusion Générale	59
Bibliographie.....	61
Annexes.....	67

Liste des Figures

Figure 1.1 : Classification principale des systèmes de recommandations.	6
Figure 1.2: Classification des systèmes de recommandation.	7
Figure 1.3 : Recommandation basé sur le contenu.	8
Figure 1.4 : Recommandation basé sur le filtrage collaboratif.	8
Figure 1.5: Scénarios de système de recommandation de santé.	15
Figure 1.6: Architecture du système de recommandation de santé (HRS).	16
Figure 1.7: Les trois volets du cadre pour l'élaboration d'un système de recommandation de santé.	19
Figure 1.8 : Les étapes incorporées comme une extension des étapes typiques.	19
Figure 2. 1 : Schéma de l'intelligence artificielle.	21
Figure 2. 2: Les trois principaux types d'apprentissage automatique.	23
Figure 2. 3: L'apprentissage automatique supervisé.	23
Figure 2. 4: L'apprentissage automatique non supervisé.	24
Figure 2. 5: L'apprentissage par Renforcement.	25
Figure 2. 6: Un neurone biologique en comparaison avec un neurone artificiel (un perceptron).	27
Figure 2. 7: Exemple d'un graphe.	27
Figure 2. 8: Exemple d'un graphe avec sa matrice d'adjacence.	28
Figure 2. 9: Exemple d'un graphe orienté.	28
Figure 2. 10: Exemple d'un graphe non orienté.	29
Figure 2. 11: Exemple d'un graphe biparti.	29
Figure 2. 12: Les réseaux de neurones graphiques.	30
Figure 2. 13: Données non euclidiennes sous forme de graphe.	33
Figure 2. 14: Fonctionnement des réseaux neuronaux graphiques.	33
Figure 2. 15: Les problèmes résolus par les GNN.	35
Figure 2. 16: Les réseaux convolutifs graphiques.	37
Figure 2. 17: Flux de travail d'un système de recommandation basé sur l'apprentissage en profondeur utilisant des données structurées sous forme de graphe.	38
Figure 3. 1: Modèle GCN pour les systèmes de recommandation de santé basés sur la FC.	43
Figure 3. 2: Logo Python.	45
Figure 3. 3: Logo Colab.	46
Figure 3. 4: Logo Pytorch Geometric.	46
Figure 3. 5: Logo Matplotlib.	47
Figure 3. 6: Logo Pandas.	47
Figure 3. 7: Logo NumPy.	47
Figure 3. 8: Logo Scikit-Learn.	48
Figure 3. 9: Dataset MovieLens.	48
Figure 3. 10: Dataset UCI ML Drug Review.	49
Figure 3. 11: Importation des données.	49
Figure 3. 12: Suppression des données inutiles.	49
Figure 3. 13: Fonction du chargement des nœuds.	50

Liste des Figures

Figure 3. 14: Fonction du chargement des arrêtes.....	50
Figure 3. 15: Code pour la répartition entraînement-test.	51
Figure 3. 16: Fonction de message et d'agrégation.	52
Figure 3. 17: Fonction de précision.....	52
Figure 3. 18: Fonction de rappel.	52
Figure 3. 19: Le graphe d'erreur pour l'ensemble des données MovieLens.....	53
Figure 3. 20 : Le graphe de précision et de rappel pour l'ensemble des données MovieLens.....	54
Figure 3. 21: Le graphe d'erreur pour l'ensemble des données UCI ML Drug Review.	54
Figure 3. 22 : Le graphe de précision et de rappel pour l'ensemble des données UCI ML Drug Review.....	55

Listes des Tables

Table 1. 1: Matrice de confusion	11
Table 3. 1 : Notations et descriptions.	44
Table 3. 2: Comparaison des résultats obtenus avec des méthodes de références.	55

Annexes

Annexe 1 : Attestation de participation à la conférence NTIC 2022	67
Annexe 2 : Graph Convolutional Networks for Designing Collaborative Filtering-Based Health Recommender Systems" notre aarticle publié lors d'une conférence de l'IEEE	68

Introduction Générale

Introduction Générale

1. Contexte

De nos jours, les systèmes de recommandation (RS) jouent un rôle prépondérant dans notre vie sociale et professionnelle en fournissant des suggestions utiles dans divers domaines tels que le commerce électronique, le tourisme et les loisirs. Ils sont efficaces dans des applications populaires comme YouTube, Amazon, Facebook et ResearchGate. Récemment, les applications de recommandation se sont également développées dans le domaine de la santé, donnant ainsi naissance aux systèmes de recommandation de santé (HRS). Les HRS visent à aider les individus à vivre une vie plus saine et meilleure en fournissant des recommandations dans des domaines tels que le diagnostic de maladies, la prescription de traitements, l'alimentation saine et la pratique du sport. Ces recommandations sont basées sur des données de santé individualisées, telles que celles contenues dans un dossier de santé personnel (PHR).

2. Problème

Malgré la diversité des cas d'utilisation des systèmes de recommandation de santé, la précision et la sécurité contre les recommandations préjudiciables demeurent une préoccupation majeure pour les parties prenantes telles que les patients, les médecins et les professionnels de la santé. Une recommandation insuffisamment précise pourrait avoir des conséquences néfastes sur la santé des patients.

3. Approche

Pour relever ce défi, ce travail propose un nouveau modèle de conception pour les HRS basé sur le filtrage collaboratif et l'utilisation des réseaux convolutifs graphiques (GCN). Les GCN sont l'architecture la plus couramment utilisée dans les réseaux neuronaux graphiques (GNN) pour les applications réelles. Ils permettent de modéliser des données non ordonnées et irrégulières grâce à une représentation basée sur des graphiques composés de nœuds et de liens. Dans ce travail, une version simplifiée des GCN, appelée LightGCN, est utilisée pour offrir une alternative moins complexe en termes de calcul par rapport au GCN traditionnel. Il convient de noter que l'utilisation de LightGCN dans les HRS est une innovation, car cela n'avait pas été fait auparavant.

4. L'organisation du mémoire

Ce travail sera organisé en trois chapitres principaux.

- Chapitre1 : Portera sur les systèmes de recommandation de santé, en présentant leurs caractéristiques et leurs applications.
- Chapitre2 : Sera consacré aux réseaux convolutifs graphiques, en expliquant leur fonctionnement et leur utilisation dans les systèmes de recommandation.
- Chapitre3 : Présentera le modèle de conception proposé basé sur LightGCN dans les systèmes de recommandation de santé.

Les systèmes de recommandation de santé

Chapitre 1 : Les systèmes de recommandation de santé

1.1 Introduction

Avec la grande quantité d'informations disponibles sur Internet aujourd'hui, il est devenu essentiel de concevoir des mécanismes permettant aux utilisateurs d'accéder le plus rapidement possible au contenu qui les intéresse. En effet, leurs besoins sont difficiles à appréhender, d'une part parce qu'ils ne sont pas clairement formulés, et d'autre part parce qu'ils évoluent constamment. Les systèmes de recommandation apportent une solution à ce problème. Ces outils puissants sont conçus pour filtrer et personnaliser les informations pour chaque utilisateur. En conséquence, ils sont devenus indispensables dans une grande variété d'industries, d'entreprises, de services financiers, de musique et de radio en ligne, de télévision et de vidéo, de publications en ligne et d'innombrables autres. Dans ce chapitre, nous allons présenter les fondements théoriques de notre travail. Nous commençons par donner quelques définitions d'un système de recommandation, ses différentes classifications, ses limites et les techniques adoptées par les systèmes de recommandation. Ensuite, nous présentons les systèmes de recommandation de santé, leurs scénarios, leurs défis et l'architecture d'un système de recommandation de santé. Enfin, nous terminons le chapitre par une conclusion.

1.2 Les systèmes de recommandation

1.2.1 Définitions

Dans l'art antérieur, plusieurs définitions ont été données pour les systèmes de recommandation. Nous donnons quelques-unes dans cette section.

Saimadhu(*Introduction to Recommendation Engine - Dataaspirant*, 2015) a défini un système de recommandation comme une boîte noire qui analyse un ensemble spécifique d'utilisateurs et présente des items que l'utilisateur pourrait aimer. Le terme "item" fait référence à des éléments qui peuvent avoir des formes très différentes : documents texte, images, vidéos, lieux, produits commerciaux, etc.

Burke(Burke, 2002) a défini un système de recommandation comme un système capable de fournir des recommandations personnalisées pour guider les utilisateurs afin de trouver des ressources intéressantes et utiles dans un grand espace de données.

Dans une définition plus simple, un système de recommandation est une forme spécifique de filtrage d'informations basée sur les préférences et le comportement d'un utilisateur afin de lui montrer des items susceptibles de le satisfaire.

Les systèmes de recommandation prédisent généralement les préférences des utilisateurs pour recommander des articles à acheter. Un système de recommandation peut simplement être un système qui vous recommande des articles similaires à ceux que vous avez achetés dans le passé, ou il peut s'agir de vidéos que d'autres utilisateurs similaires ont regardées, etc. Les systèmes de recommandation se concentrent sur la prédiction basée sur des mesures de similarité. De nombreuses techniques ont été utilisées pour les tâches de recommandation, y compris les systèmes de recommandation collaboratifs, les systèmes de recommandation basés sur le contenu, les systèmes de recommandation basés sur les connaissances et bien d'autre.

1.2.2 Exemples

De nombreux systèmes de recommandation ont été développés dans différents domaines d'application. Nous pouvons en citer :

- Les systèmes de recommandation de films, tels que : Movielens(O'Connor & Herlocker, 1999) et Eachmovie (Breese & Kadie, 1999).
- Les systèmes de recommandation de livres (Bookcrossing5).
- Les systèmes de recommandation de musique (LastFM6).
- Les systèmes de recommandation de blagues(Goldberg et al., 1992).
- Les systèmes de recommandations introduits sur des sites e-commerce (Amazon).
- Les systèmes de recommandation de restaurants(Burke, 2002) .
- Les systèmes de recommandation intégrés aux Extranets documentaires (L'Extranet documentaire du Crédit Agricole).
- Les systèmes de recommandations intégrés aux moteurs de recherche (Le moteur de recherche d'AOL).
- Les systèmes de recommandations implémentés sur des sites de recrutement (Job-Finder).
- Les systèmes de recommandations de citations bibliographiques (Bhagavatula et al., 2018) (Cosley et al., 2002).

1.2.3 Classification des systèmes de recommandation

Durant ces dernières années, plusieurs classifications sont apparues. Celles-ci se basent sur différents facteurs pour catégoriser un système de recommandation notamment : la connaissance des goûts de l'utilisateur, son positionnement par rapport aux autres et la connaissance des items à recommander. (Voir la figure1.1) :

1.2.3.1 La classification classique

Cette classification est reconnue par trois types de filtrage.

- Un filtrage collaboratif (CF en anglais).
- Un filtrage basé sur le contenu (CBF en anglais).
- Le filtrage hybride.

1.2.3.2 La Classification de [Su et al, 2009]

Elle est utilisée dans les systèmes de collaboration. Ils proposent une sous classification qui comprend les techniques hybrides, les classer dans les méthodes de collaboration hybrides. [Su et al, 2009] classe filtrage collaboratif en deux catégories :

- Approches CF à base de mémoire : pour K-plus proches voisins.
- Approches CF basées sur un modèle englobant une variété de techniques telles que : clustering, les réseaux bayésiens, factorisation de matrices, les processus de décision de Markov.

1.2.3.3 La classification de Burke et La classification de [Rao et al, 2008]

Burke et Rao ont proposé une classification des systèmes de recommandation basée sur le contexte, tandis que Talwar a proposé une classification basée sur le type de données utilisées par les systèmes de recommandation.

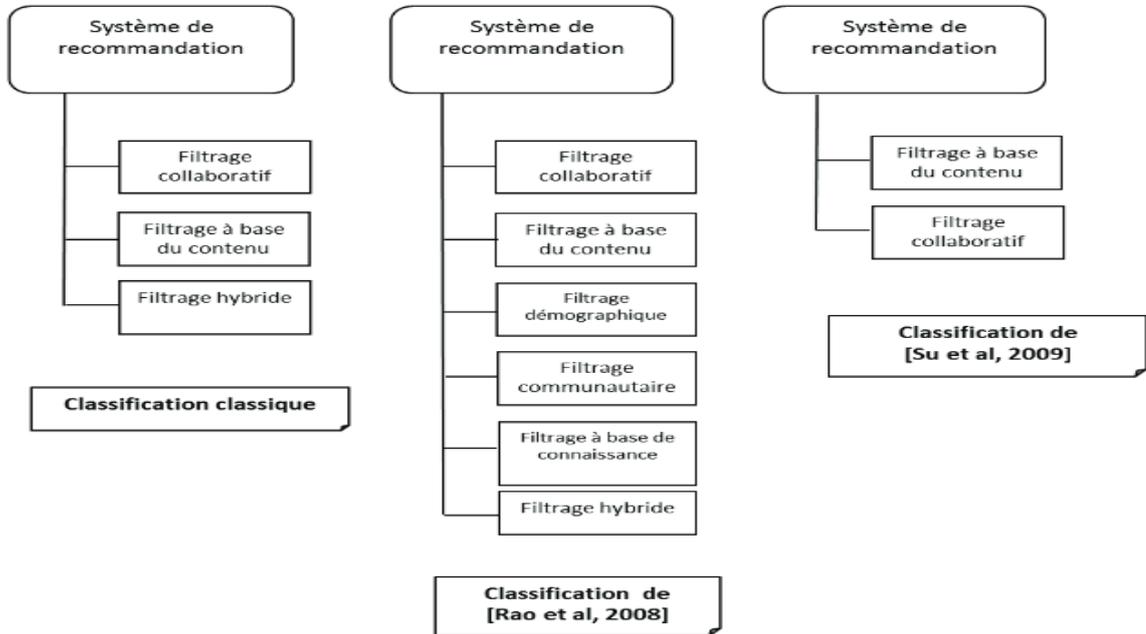


Figure 1.1 : Classification principale des systèmes de recommandations.

1.2.4 Les approches de systèmes 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 largement utilisée est basée sur trois types : le filtrage basé sur le contenu, le filtrage collaboratif et le filtrage hybride (Adomavicius & Tuzhilin, 2005).

Le but ici est de s'appuyer sur la classification la plus connue sur laquelle s'appuient nos recherches. Nous présentons ci-dessous les approches basées sur le contenu et le filtrage collaboratif, suivis des approches hybrides (Fig. 1.2). Nous détaillerons la technique de filtrage collaboratif que nous utiliserons dans notre travail.

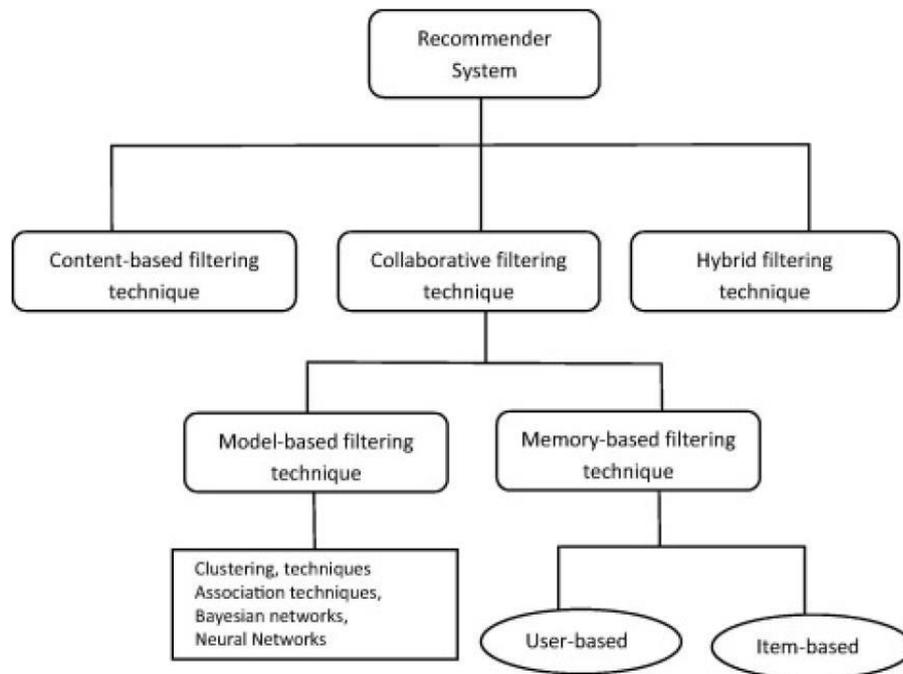


Figure1.2: Classification des systèmes de recommandation.

1.2.4.1 Approches basées sur le contenu

Un système de recommandation basé contenu est un système de recherche qui utilise des descriptions similaires entre les items à recommander à l'utilisateur actuel et les items qu'il a apprécié dans le passé, par exemple dans une recommandation des films le système basé contenu prend en considération les points communs comme : acteurs spécifiques, réalisateurs, genres de film, etc. (Adomavicius & Tuzhilin, 2005). Un avantage des systèmes basés contenu est qu'ils permettent d'associer des documents à des profils d'utilisateurs. Notamment grâce à l'utilisation de techniques d'indexation et d'intelligence artificielle. L'utilisateur est indépendant des autres, même s'il est le seul utilisateur du système, il peut obtenir des recommandations pour éviter certaines limitations et inconvénients des systèmes basés sur le contenu et collaboratifs (Belloui, 2008). Les approches basées sur le contenu sont particulièrement utiles lorsque les données sont limitées ou lorsque les informations sur l'utilisateur sont peu fiables ou non disponibles. Cependant, elles ont tendance à ne pas être aussi performantes que d'autres méthodes d'apprentissage automatique telles que les approches collaboratives.

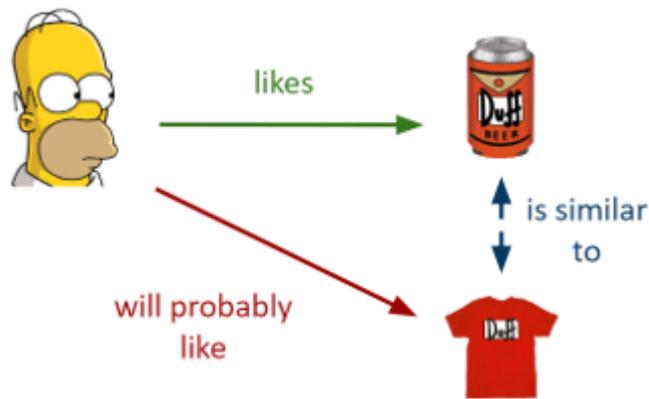


Figure 1.3 : Recommandation basée sur le contenu.

1.2.4.2 Approches basées sur le filtrage collaboratif

Les approches basées sur le filtrage collaboratif sont une méthode d'apprentissage automatique utilisée pour effectuer des recommandations personnalisées en se basant sur les comportements et les préférences des utilisateurs. L'idée principale derrière cette approche est que si deux utilisateurs ont des comportements ou des préférences similaires, ils auront probablement des goûts similaires pour d'autres produits ou services (MAATALLAH, 2016).

Ce type de filtrage recueille des informations sur les utilisateurs en leur demandant d'évaluer les items et en faisant des recommandations basées sur les éléments les mieux notés par les utilisateurs ayant des goûts similaires. Les méthodes FC font des recommandations basées sur les évaluations d'items d'un ensemble d'utilisateurs (voisins) dont les profils d'évaluation sont les plus similaires à ceux de l'utilisateur cible. Ces méthodes supposent que si les utilisateurs ont la même préférence pour un ensemble d'items, ils sont susceptibles d'avoir la même préférence pour un autre ensemble d'items qu'ils n'ont pas encore évalués (Garnine, 2020). Les méthodes de filtrage collaboratif peuvent être divisées en deux grandes catégories : basées sur la mémoire et basées sur les modèles (Figure 1.1).

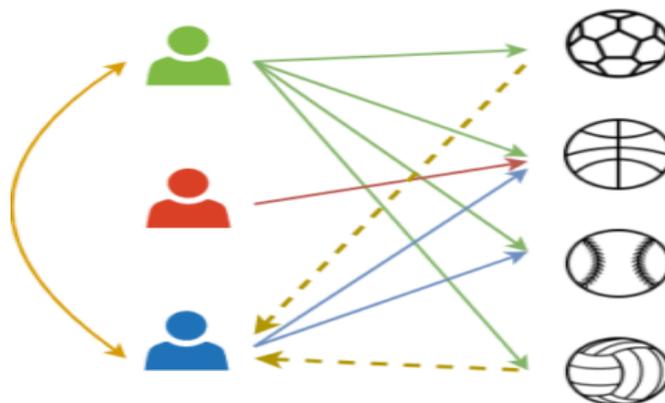


Figure 1.4 : Recommandation basée sur le filtrage collaboratif.

1.Filtrage basé mémoire

Dans la collaboration basée sur la mémoire, les évaluations des utilisateurs des items stockés dans le système sont directement utilisées pour prédire les évaluations des items non évalués(Garnine, 2020).

Le filtrage basé mémoire est relativement simple à mettre en œuvre et peut fournir des résultats de recommandation rapides et précis pour des ensembles de données de taille moyenne. Ce type de filtrage passe par deux étapes importantes : le calcul de similarité et le calcul de prédiction de score.

- La similarité : le calcul de la similarité se fait soit sur l'ensemble des utilisateurs (Resnick & Varian, 1997) soit sur l'ensemble des items (Sarwar et al., 2001) en utilisant des mesures de similarité et de distance, les plus utilisées dans la littérature sont la similarité de cosinus, ou la corrélation de Pearson.
- La prédiction : il s'agit de prédire la note des items non notés par les utilisateurs en se basant sur les notes données par ses voisins.

Filtrage collaboratif basé utilisateurs

Le filtrage collaboratif basé sur les utilisateurs a été introduit pour la première fois dans le système GroupLens (Resnick & Varian, 1997), son principe de fonctionnement est très simple : déterminer les utilisateurs qui sont similaires à l'utilisateur courant, puis calculer une valeur de prédiction pour chaque item candidat à la recommandation en analysant les notes que les voisins de l'utilisateur courant ont exprimées sur cet item. L'approche basée sur les utilisateurs commence par la création d'une matrice de similarité qui compare les utilisateurs en fonction de leurs comportements ou de leurs préférences pour les produits ou les services. Cette matrice de similarité est utilisée pour identifier les utilisateurs ayant des comportements ou des préférences similaires, et pour recommander des produits ou des services que les utilisateurs similaires ont appréciés.

Filtrage collaboratif basé items

Le filtrage collaboratif à base d'items a été introduit par (Sarwar et al., 2001). Son principe de fonctionnement est le suivant : pour l'item i candidat à la recommandation, on détermine les voisins les plus proches (les items similaires) en calculant sa similarité avec les autres items disponibles et on calcule ensuite la prédiction de la note de l'utilisateur courant u pour l'item i à partir des notes que u a attribué aux voisins de i . Le filtrage collaboratif basé sur les items peut être efficace dans des situations où les caractéristiques des produits ou des services sont bien définies et faciles à quantifier, et où il y a suffisamment de données sur les comportements des utilisateurs.

2.Filtrage basé sur un modèle

Le filtrage collaboratif basé sur un modèle est une technique de recommandation de produits ou de contenu en ligne qui utilise des algorithmes pour prédire les préférences d'un utilisateur en fonction des données de ses interactions précédentes avec le système.

Cette méthode consiste à créer un modèle de préférence pour chaque utilisateur en analysant les données d'autres utilisateurs ayant des préférences similaires. Le modèle est ensuite utilisé pour prédire les évaluations ou les préférences des produits ou du contenu que l'utilisateur n'a pas encore évalué.

L'avantage du filtrage collaboratif basé sur un modèle est la possibilité de recommander des produits ou du contenu aux utilisateurs ayant des préférences similaires, même si ces préférences sont peu communes ou peu connues. Cependant, cette approche présente également des inconvénients, tels que le besoin de données de qualité pour construire un modèle précis, et le risque de biais de recommandation si les données d'entrée sont biaisées (Oufaida & Nouali, 2008).

1.2.4.3 Approches basées sur le filtrage hybride

Les systèmes de filtrage hybrides combinent le système de filtrage collaboratif avec d'autres techniques de recommandation (généralement avec des systèmes basés sur le contenu) pour faire des prédictions ou des recommandations pour éviter certaines limitations et inconvénients des systèmes basés sur le contenu et collaboratifs (Rao & Talwar, 2008). En général, le filtrage hybride utilise certaines méthodes pour combiner les ensembles de recommandations telles que la pondération, la cascade, la commutation, afin de produire des recommandations finales pour les utilisateurs (Burke, 2002). Le filtrage hybride est souvent utilisé pour améliorer la qualité des recommandations dans des domaines tels que le commerce électronique, la publicité en ligne etc.

1.2.5 Évaluation des systèmes de recommandation

Évaluer un système de recommandation permet de mesurer ses performances vis-à-vis de ses objectifs. De ce fait, le choix des mesures à utiliser diffère selon les objectifs fixés. La base initiale est partagée en deux groupes de données. Le premier va servir de base d'apprentissage et le deuxième va servir de base de test. On prend chaque individu u de la base de test, on essaie de prédire sa note pour l'item i à partir des notes des individus de la base d'apprentissage et de ses notes disponibles pour ses autres items (ZIANI, 2018).

Parmi les mesures typiques la métrique prédictive RMSE, et les métriques d'aide à la décision (la précision et le rappel). La métrique prédictive calcule la précision des prédictions par rapport à l'évaluation réelle effectuée par l'utilisateur, par contre les métriques d'aide à la décision n'évaluent pas la qualité de la prédiction, mais la pertinence des recommandations.

➤ **La mesure de l'erreur absolue moyenne (EAM ou MAE pour Mean Absolute Error en anglais)** qui est la mesure la plus employée dans les systèmes de recommandation, elle estime la moyenne de la différence absolue entre les évaluations et les prédictions (Garnine, 2020). Le système de recommandation collaboratif est jugé performant quand la valeur de MAE est petite. Cette mesure est donnée par l'équation suivante :

$$\text{MAE} = \frac{\sum_{(u,i) \in k} |r_{u,i} - \hat{r}_{u,i}|}{|k|} \quad (1.1)$$

$r_{u,i}$ Est la vraie note donnée par u à i .

$\hat{r}_{u,i}$ Est la note prédite par le SR.

K est l'ensemble des couples (utilisateur, item) pour lesquels la confrontation est effectuée.

➤ **RMSE (Root Mean squared error)**

$$\text{RMSE} = \sqrt{\frac{\sum_{(u,i) \in k} (r_{u,i} - \hat{r}_{u,i})^2}{|k|}} \quad (1.2)$$

➤ **Précision et Rappel** : Lorsqu'on cherche à prédire si un utilisateur est intéressé ou non par un item, quatre possibilités sont offertes par la matrice de confusion (MAATALLAH, 2016).

Table 1. 1: Matrice de confusion

Item	Pertinent	Non pertinent
Recommandé	Vrai Positif (vp)	Faux Positif (fp)
Non recommandé	Faux Négatif (fn)	Vrai Négatif (vn)

La précision : est le pourcentage d'articles pertinents qui sont correctement recommandés sur tous les articles recommandés. Il est calculé en utilisant l'expression suivante :

$$\text{Précision} = \frac{VP}{VP+FP} \quad (1.3)$$

Le rappel (recall) : est une mesure d'exhaustivité qui détermine la proportion des items pertinents recommandés parmi tous les items pertinents.

Un système de recommandation parfait doit avoir un rappel près de la valeur 1. Il est calculé par la formule ci-dessous :

$$\text{Rappel} = \frac{VP}{VP+FN} \quad (1.4)$$

Un système de recommandation parfait doit avoir une précision et un rappel près de la valeur 1, mais ces deux exigences sont souvent contradictoires et une très forte précision ne peut être obtenue qu'au prix d'un rappel faible et vice-versa.

1.2.6 Les défis des systèmes de recommandation

Les systèmes de recommandation présentent un outil très important avec beaucoup d'avantages. Cependant, ces systèmes présentent un certain nombre de limites, on va citer dans ce qui suit quelques problèmes de ces systèmes.

1.2.6.1 Démarrage à froid

C'est un problème qui se pose avec l'arrivée d'un nouvel élément dans le système de recommandation soit un utilisateur soit un item. Le système doit le gérer comme un cas spécial, le problème de démarrage à froid c'est qu'on n'a pas assez d'informations pour émettre des recommandations (Rashid et al., 2002).

1.2.6.2 La difficulté d'obtenir suffisamment de données de qualité

Les systèmes de recommandation nécessitent de grandes quantités de données pour fonctionner de manière optimale, mais il peut être difficile d'obtenir des données de qualité sur les préférences des utilisateurs et les caractéristiques des produits ou des services recommandés.

1.2.6.3 La diversité des goûts et des préférences

Les goûts et les préférences des utilisateurs sont très variés et peuvent évoluer au fil du temps. Il est donc difficile pour les systèmes de recommandation de fournir des recommandations personnalisées et précises pour chaque utilisateur.

1.2.6.4 Parcimonie (Sparsity)

Est un problème courant dans les systèmes de recommandation car les données d'entrée ont souvent des matrices creuses, c'est-à-dire que la plupart des utilisateurs n'ont noté ou acheté qu'un petit sous-ensemble d'articles dans la base de données. Cette caractéristique peut rendre difficile la prédiction des préférences des utilisateurs pour de nouveaux articles ou produits, car il y a peu d'informations disponibles sur les préférences des utilisateurs. De ce fait, la matrice des notes est une matrice creuse avec un taux de valeurs manquantes pouvant atteindre 95% du total des valeurs (Papagelis et al., 2005). Par exemple, sur Amazon, si les utilisateurs actifs peuvent avoir acheté 1% des articles et que le montant total des articles est d'environ 2 millions de livres, cela signifie qu'il n'y a que 20 000 livres évalués. Par conséquent, une telle rareté dans les évaluations dégrade la sélection

précise des voisins dans l'étape de formation du quartier et conduit à de mauvais résultats de recommandation (Adomavicius & Tuzhilin, 2005) (Linden et al., 2003).

1.2.6.5 La rareté des données (scarcity)

Est un problème important dans les systèmes de recommandation, Lorsque les données sont rares dans les systèmes de recommandation, cela peut affecter la précision et la qualité des recommandations, car le modèle peut manquer d'informations pour apprendre les préférences et les comportements des utilisateurs.

1.3 Systèmes de recommandation de santé

1.3.1 Définition

Un système de recommandation de santé, ou en Anglais Health Recommender System (HRS), est considéré comme une spécialisation d'un système de recommandation visant à aider, à améliorer les décisions liées aux soins de santé (T. N. T. Tran et al., 2021). Ces systèmes sont souvent utilisés pour aider les gens à surveiller leur santé, à prévenir les maladies et à améliorer leur qualité de vie et ils recommandent des informations de soins de santé aux professionnels de santé et aux patients.

HRS doit analyser l'état du patient et faire des recommandations, diagnostic de la maladie, médicaments, alimentation personnalisée, activités physiques ou d'autres services de santé (Trang Tran et al., 2018).

Les systèmes de recommandation de santé peuvent utiliser des algorithmes d'apprentissage automatique pour analyser les données médicales d'un individu et fournir des recommandations basées sur ses antécédents médicaux, son âge, son sexe et d'autres facteurs pertinents. Ils peuvent également utiliser la technologie de traitement du langage naturel pour comprendre les symptômes qu'une personne décrit et fournir des conseils appropriés.

Il convient de noter que le système de recommandation de santé ne doit pas être utilisé comme un substitut à un traitement médical professionnel, mais comme un outil supplémentaire pour aider les personnes à prendre soin de leur santé.

1.3.2 Les scénarios de système de recommandation de santé

Les systèmes de recommandation de santé (HRSs) proposent aux utilisateurs différents types de recommandations qui contribuent à améliorer leur bien-être.

Ces systèmes aident également les professionnels de la santé à prendre des décisions plus précises axées sur le patient. Les services de santé fournis par un HRS peuvent prendre plusieurs scénarios et formulaires de recommandation.

La figure 1.5 affiche cinq scénarios de soins de santé tels que mentionnés dans (T. N. T. Tran et al., 2021), à savoir :

1.3.2.1 Recommandation alimentaire

La recommandation alimentaire est une application courante des systèmes de recommandation. Ces systèmes peuvent aider les gens à prendre des décisions alimentaires saines pour diminuer le risque de maladies chroniques, à trouver des recettes adaptées à leurs préférences et à leur régime alimentaire. La recommandation alimentaire (Trang Tran et al., 2018) est devenue un standard dans les domaines de la santé et du bien-être.

1.3.2.2 Recommandation de médicaments

Les erreurs de médication pouvant menacer la vie des patients. Cela peut être causé par des expériences/connaissances limitées des médecins sur les médicaments et les maladies, et l'information croissante sur les médicaments disponibles, ce qui a entraîné des difficultés pour découvrir les médicaments pertinents et les interactions médicament-maladie. À cet égard, les systèmes de recommandation de médicaments sont un élément essentiel pour aider les médecins et les pharmaciens à recommander des médicaments appropriés en fonction des besoins spécifiques des patients.

1.3.2.3 Prédiction de l'état de santé

Afin de faire face à la croissance des maladies dans le monde (cas COVID-19) et leur impact négatif sur les sociétés et les états, le sujet de la prédiction des risques a attiré de plus en plus d'attention au cours des dernières années, et des recherches intensives liées sont déployées (Etemadi et al., 2023). Pour éviter ces maladies, HRS a été développé dans le but de détecter les symptômes de la maladie le plus tôt possible. De plus, ils peuvent aider les médecins à prédire l'état de santé d'un patient, à développer un traitement précis pour les patients.

1.3.2.4 Recommandation d'activité physique

La recommandation d'activité physique est une application courante des systèmes de recommandation dans le domaine de la santé et du bien-être. Les activités sont également dans les résultats attendus de HRS (Ferretto et al., 2020). Ils aident à garder les patients en bonne santé et à prévenir d'autres complications de santé.

1.3.2.5 Recommandations des professionnels de santé

Avec la surcharge des informations médicales disponibles, les patients auraient des difficultés à rechercher des médecins et des professionnels de la santé appropriés possédant la meilleure expertise pour résoudre leurs problèmes de santé. Ces systèmes fournissent des recommandations de santé personnalisées de la part de professionnels de santé qualifiés, tels que des médecins, des diététiciens et des coachs de santé. Cet écart soulève un sujet ouvert sur matchmaking patient-médecin, dans lequel peut être abordé par développement des HRS.



Figure 1.5: Scénarios de système de recommandation de santé.

1.3.3 Architecture du système de recommandation de santé

Les données de santé sont devenues une source importante de mégadonnées en raison de l'augmentation rapide de la quantité de données produites, de la vitesse à laquelle elles sont générées et de la variété des sources. La véracité est également une caractéristique importante car des données incorrectes peuvent entraîner des erreurs de diagnostic et de traitement. Les systèmes de recommandation sont de plus en plus utilisés pour aider à gérer les données de santé en filtrant les informations pertinentes et en fournissant des recommandations personnalisées pour les patients. Il y a un besoin pour un nouveau HRS qui peut améliorer le système de soins de santé et gérer les patients souffrant de différentes maladies à la fois (Sahoo et al., 2019), Ces systèmes permettent aux professionnels de la santé de coordonner les soins pour les patients atteints de multiples affections chroniques, en fournissant un point centralisé pour suivre les soins et en identifiant les domaines où les soins peuvent être améliorés.

L'architecture HRS est illustrée à la figure 1.6

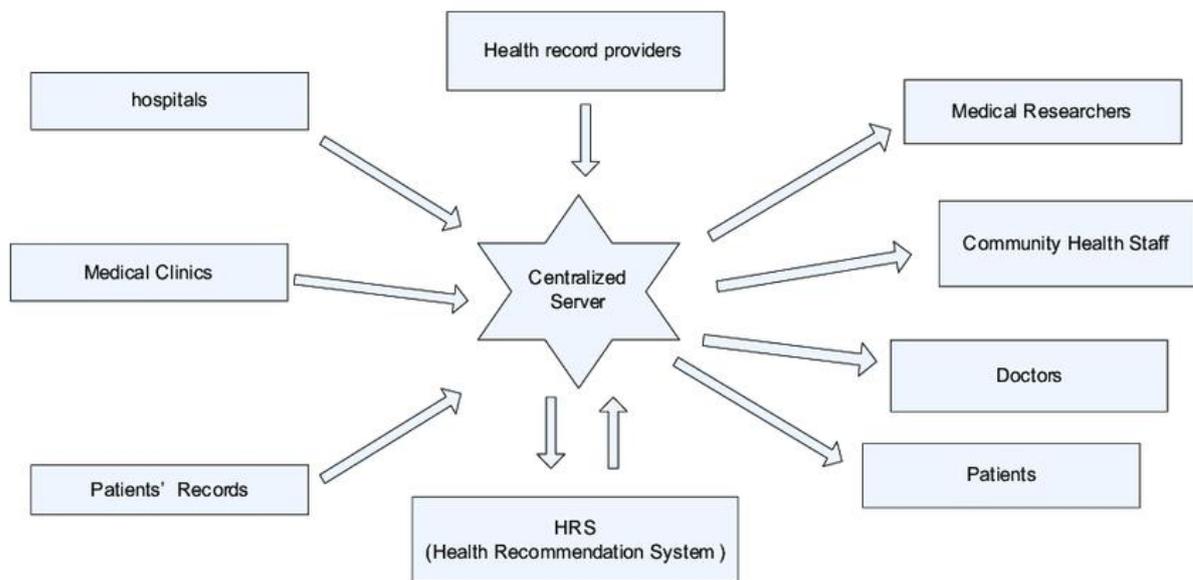


Figure 1.6: Architecture du système de recommandation de santé (HRS).

1.3.4 Les défis de la recherche sur les systèmes de recommandation de santé

Cette partie fait référence à des problèmes importants qui n'ont pas été pris en compte et étudiés dans les HRS, ce qui entraîne l'existence de certains problèmes de recherche ouverts qui peuvent être discutés ici dans cette partie.

1.3.4.1 Défis éthiques

Avec l'utilisation rapide d'Internet, une grande quantité de données est partagée. Ainsi, la vie privée devient un défi important pour tout HRS (Himgauri & Apashabi, 2015). Les données de santé sont souvent sensibles et doivent être protégées pour garantir la vie privée des patients. Les systèmes de recommandation de santé doivent être conçus pour respecter la confidentialité et la sécurité des données de santé. Des défis éthiques tels que la confidentialité, les dispositions sur la vie privée, le contrôle de l'accès aux informations des patients, l'anonymisation des informations des patients, la commercialisation, la gouvernance et la propriété des données des patients sont les principales préoccupations qui entravent le partage d'informations opérationnelles et efficaces dans les systèmes de santé entre les prestataires de soins et les patients. Les utilisateurs estiment qu'ils peuvent faire davantage confiance pour partager leurs données s'ils sont assurés que la confidentialité sera préservée.

1.3.4.2 Données de santé de qualité

Les systèmes de recommandation de santé nécessitent des données de santé précises et de haute qualité pour fournir des recommandations utiles et précises. Cependant, les données de santé peuvent souvent être incomplètes, inexactes ou obsolètes, ce qui peut entraîner des erreurs de recommandation.

1.3.4.3 La prolifération des normes de santé

Sont des spécifications qui permettent aux différents systèmes de santé, plates-formes, technologies et outils de coopérer les uns avec les autres. Toutefois, les organisations de santé utilisent souvent différentes normes, ce qui peut entraîner des variations dans les médicaments, les codes et les titres des rapports de cas, des examens et des maladies d'un hôpital à l'autre (Olaronke et al., 2013). De plus, l'absence d'un standard commun dans les systèmes de santé peut entraver l'interopérabilité entre les systèmes hétérogènes. Par conséquent, la normalisation des codes médicaux dans différents systèmes de santé reste un défi majeur.

1.3.4.4 Fiabilité

Le RS basé sur la santé est un système de prise de décision qui recommande des informations de santé adaptées tant aux patients qu'aux professionnels de la santé. Pour être utile aux patients, il est impératif que le HRS soit fiable et digne de confiance.

1.3.4.5 Précision

L'aspect le plus critique des résultats dans le domaine de la santé est l'exactitude. Étant donné qu'il est crucial pour la santé du patient, des résultats insuffisamment précis en raison d'un système défectueux peuvent agir comme une source de stress pour les patients et entraîner des conséquences graves. La précision des résultats revêt une importance cruciale dans le domaine médical et devrait être aussi proche que possible des conditions réelles. Par conséquent, il serait possible d'obtenir des prédictions et des recommandations précises si une quantité importante de données était accessible (Meingast et al., 2006).

1.3.4.6 Le désaccord des patients et de leurs proches

Lorsque des personnes malades consultent leur médecin, celui-ci recommande le meilleur traitement disponible, qui peut être coûteux, ce qui peut entraîner des désaccords entre les patients et leurs proches. Des situations similaires peuvent se produire dans les systèmes de santé intelligents, car ils recommandent les meilleures solutions pour des problèmes qui pourraient être plus coûteuses. Le filtrage des coûts de traitement peut être une bonne solution à ce problème, qui nécessite une plus grande attention (Breslin et al., 2005).

1.3.4.7 Autorisation et accès aux dossiers médicaux électroniques

Les établissements de santé doivent avoir des droits d'accès aux profils et aux antécédents médicaux de leurs patients pour des raisons financières, cliniques et médico-légales. D'autre part, en ayant des droits d'accès, les patients peuvent télécharger leurs dossiers médicaux et consulter leurs médecins pour plus de recommandations (Mandl et al., 2001).

1.3.4.8 Convaincre les gens

Il est difficile de convaincre les patients de connaître les vrais problèmes et les solutions associées en appliquant une RS parce qu'ils ne peuvent pas faire confiance ou ne sont pas conscients de l'applicabilité des systèmes de santé (Dixon-Woods et al., 2012).

1.3.4.9 Les aspects légaux

Compte tenu de l'utilisation généralisée de l'intelligence artificielle, des directives juridiques essentielles (plutôt que purement éthiques) pour l'innovation et l'application de l'intelligence artificielle en médecine sont nécessaires. L'IA médicale, d'autre part, peut avoir un impact significatif sur l'intégrité physique et mentale d'une personne, et tout dysfonctionnement pourrait avoir des conséquences graves, en faisant de ce domaine de l'IA un domaine très pertinent en termes de droits légaux. Non seulement l'ensemble des données de formation doit être correctement inspecté pour détecter les biais, mais les performances continues de l'IA doivent également être surveillées en permanence pour détecter les biais. Lorsque l'IA médicale est adaptée à des sous-groupes de patients qui n'ont pas été suffisamment représentés dans l'ensemble de données, l'efficacité des résultats doit être évaluée. Afin d'éviter les inconvénients, la diversité des patients doit toujours être prise en compte dans la programmation ou l'application (Stöger et al., 2021).

1.3.5 Un cadre de travail pour les systèmes de recommandation de santé

Le développement d'un système de recommandation de santé réussi requiert l'intégration de critères supplémentaires et de procédures spécifiques pour garantir son efficacité. Le domaine de la santé et des systèmes de recommandation médicale est complexe et rencontre plusieurs défis. Tout d'abord, il n'y a pas de définition claire des tâches pour les systèmes de recommandation de santé, car le but de ces systèmes dépend de l'élément recommandé (Calero Valdez et al., 2016). Par exemple, un système de recommandation typique dans un appareil mobile pourrait suggérer des activités physiques adaptées à la situation actuelle de l'utilisateur pour améliorer sa santé. Les patients souffrant d'arthrite et d'obésité pourraient bénéficier davantage des recommandations en matière d'activité physique qui n'exercent aucune pression supplémentaire sur les articulations enflammées, comme faire une promenade agréable par temps ensoleillé. En revanche, un autre exemple de recommandation de santé pourrait être un système qui propose différentes formes de thérapies anticancéreuses pour le médecin et le patient. Ce système pourrait prendre en compte les propriétés du patient telles que d'autres maladies, des médicaments supplémentaires, les exigences professionnelles et la situation familiale pour recommander des thérapies optimales et alternatives. Il pourrait également afficher la durée, l'expérience et les effets secondaires possibles de la multithérapie pour aider à accroître le contrôle des patients sur leur situation. Dans les deux scénarios, les algorithmes sous-jacents peuvent être dérivés de la recherche sur les systèmes de recommandation, mais les objectifs sont radicalement différents, ce qui nécessite des exigences différentes pour le système de recommandation. En somme, pour créer un système de recommandation de santé efficace, des critères et des procédures supplémentaires doivent être intégrés pour garantir son succès.

Afin d'aider à comprendre l'espace de conception de tels systèmes de recommandation, trois étapes de conception supplémentaires sont proposées (voir Figure 1.7) lors de la conception d'un système de recommandation de santé. Chaque étape propose des questions d'orientation ou des méthodes et procédures pour enrichir l'image contextuelle du scénario d'utilisation (voir Figure 1.8).

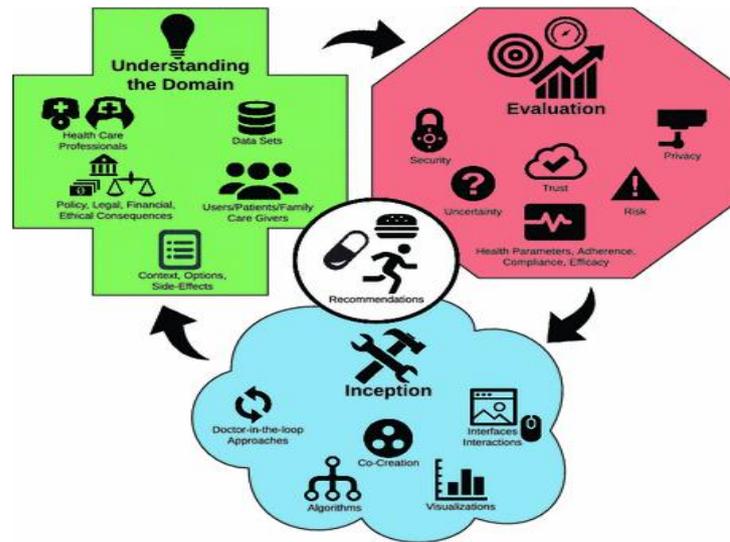


Figure 1.7: Les trois volets du cadre pour l'élaboration d'un système de recommandation de santé.

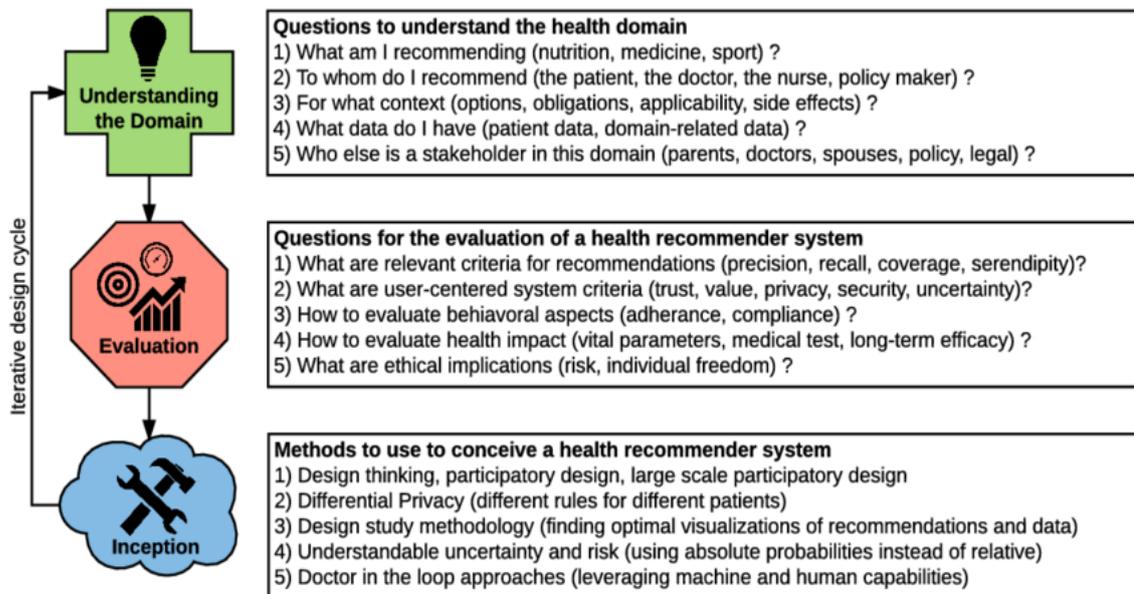


Figure 1.8 : Les étapes incorporées comme une extension des étapes typiques.

1.4 Conclusion

Ce chapitre a donné une présentation assez complète sur les systèmes de recommandation. Dans une première partie, il a commencé par définir les systèmes de recommandation en donnant leurs types, et leurs défis et comment peuvent être évalués. Ensuite et particulièrement dans une seconde partie, nous avons vu aussi les systèmes de recommandation de santé, leurs scénarios et leurs défis spécifiques tout en présentant un cadre de travail pour ce genre de système. Dans le chapitre suivant nous allons présenter les fondements de l'apprentissage automatique via les réseaux de neurones convolutifs graphiques.

Chapitre 2 : Les réseaux convolutifs graphiques

Chapitre 2 : Les réseaux convolutifs graphiques

2.1 Introduction

Les réseaux convolutifs graphiques, ou en Anglais Graph Convolutional Networks (GCN), sont devenus une approche populaire pour l'analyse de données de structure non euclidienne, tels que les réseaux sociaux, les graphes de connaissances et les données génomiques. Les GCN exploitent les propriétés topologiques des graphes pour extraire des informations significatives et offrent une solution prometteuse pour les tâches de classification et de prédiction. Cependant, leur fonctionnement diffère de celui des réseaux de neurones classiques, nécessitant une compréhension approfondie de la théorie des graphes et des techniques d'apprentissage profond. Ce chapitre vise à fournir une introduction complète aux réseaux de neurones à convolution graphiques, en explorant les bases de la théorie des graphes et des réseaux de neurones, ainsi que les différents types d'apprentissage automatique. Nous aborderons également les architectures des GCN, leur fonctionnement, ainsi que les domaines d'application où ils peuvent offrir une valeur ajoutée. Ce chapitre est destiné aux professionnels de l'intelligence artificielle, aux scientifiques de données, aux chercheurs en informatique et à toute personne souhaitant approfondir sa compréhension des réseaux de neurones à convolution graphiques et leur potentiel pour l'analyse de données non structurées.

2.2 Fondement de l'IA, du ML et du DL

2.2.1 Définition de l'IA

L'intelligence artificielle (IA) est un domaine scientifique et d'ingénierie qui se concentre sur la création de machines intelligentes capables de réfléchir par elles-mêmes et de prendre des décisions autonomes. L'objectif est de reproduire le comportement et l'intelligence humaine (McCarthy, 2007).

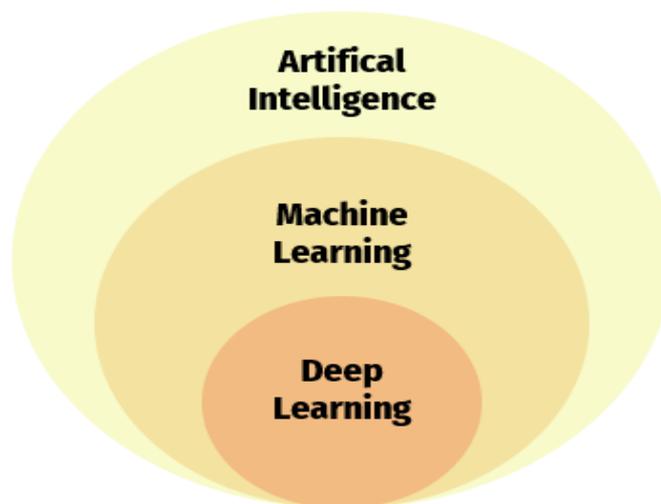


Figure 2. 1 : Schéma de l'intelligence artificielle.

2.2.2 Historique

Les prémices de l'intelligence artificielle remontent à l'Antiquité, où les philosophes classiques ont cherché à systématiser la pensée humaine (Haenlein & Kaplan, 2019). Les premiers romans de

science-fiction, tels que "Frankenstein" de Mary Shelley en 1818 et "R.U. R" de Karel Čapek en 1920, ont également contribué à son émergence.

Cependant, les recherches concrètes sur l'IA ont véritablement débuté après la Seconde Guerre mondiale, lorsque des scientifiques ont commencé à travailler sur des machines intelligentes. En 1947, Alan Turing a prononcé la première conférence sur l'IA, où il a préconisé l'utilisation de programmes informatiques plutôt que la construction de machines spécifiques. En 1950, il a publié un article intitulé "Computing Machinery and Intelligence", dans lequel il a proposé le célèbre "test de Turing". Ce test visait à déterminer si une machine pouvait être considérée comme intelligente ou non.

Toutefois, de nombreux experts en IA considèrent aujourd'hui que le test de Turing n'est pas une mesure adéquate de l'intelligence artificielle. Ils le jugent comme un test limité, car il ne se concentre que sur le comportement observable (Haenlein & Kaplan, 2019). Malgré cela, aucun autre substitut n'a encore été développé de manière satisfaisante, ce qui fait que le test d'imitation de Turing reste le premier critère pour évaluer l'IA.

2.2.3 Définition de l'apprentissage automatique

L'apprentissage automatique (Machine Learning ou ML) est une branche de l'intelligence artificielle qui repose sur un ensemble d'algorithmes, généralement appelés "modèles" (Nichols et al., 2018). Ces modèles sont conçus pour construire des applications intelligentes capables d'apprendre à partir de l'expérience, de faire des prédictions intelligentes et de s'améliorer au fil du temps en se basant sur un ensemble de données (*What Is Deep Learning?*, 2020). Le choix de l'algorithme dépend du type de données que nous utilisons pour former le modèle.

La construction d'une application de ML comprend quatre étapes : la sélection et la préparation d'un ensemble de données, le choix d'un algorithme approprié pour traiter ces données, l'entraînement de l'algorithme pour créer le modèle, et enfin l'utilisation et l'amélioration continue du modèle (*What Is Deep Learning?*, 2020).

Voici une liste de certains des algorithmes de ML couramment utilisés, qui peuvent être appliqués à presque tous types de problèmes de données : Régression linéaire, Régression logistique, Arbre de décision, SVM, Naive Bayes, kNN, K-Means, Random Forest, etc. (Ray, 2017).

2.2.4 Types d'apprentissage automatique supervisé, non supervisé, renforcement

Il existe plusieurs types d'apprentissage automatique. Les trois principaux types sont :

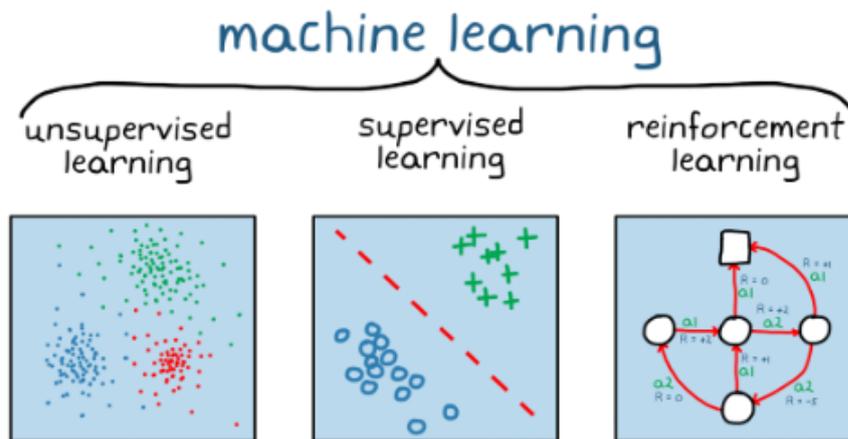


Figure 2. 2: Les trois principaux types d'apprentissage automatique.

2.2.4.1 L'apprentissage automatique supervisé

L'apprentissage supervisé est une méthode qui utilise des données contenant des attributs supplémentaires que nous cherchons à prédire. Il peut prendre la forme d'un problème de classification, lorsqu'il s'agit de classer des échantillons dans deux ou plusieurs catégories et d'apprendre à prédire la classe des données non étiquetées. Par exemple, la reconnaissance de chiffres manuscrits est un problème de classification dans lequel nous cherchons à attribuer à chaque vecteur d'entrée une catégorie discrète et finie. L'autre type de problème rencontré dans l'apprentissage supervisé est la régression, qui consiste à prédire une ou plusieurs variables continues en fonction des données d'entrée. Par exemple, prédire la longueur d'un saumon en fonction de son âge et de son poids est un exemple de problème de régression (Soni, 2020).

Supervised Learning

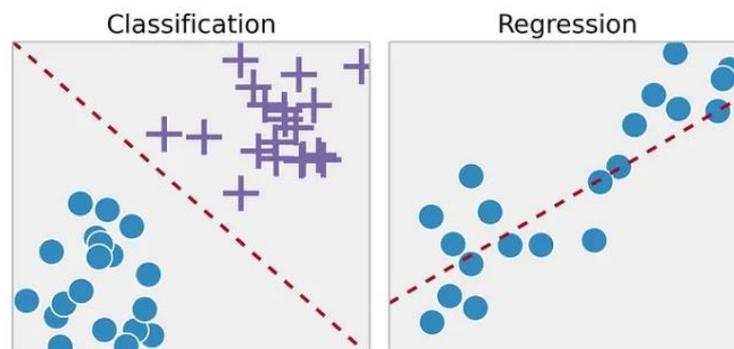


Figure 2. 3: L'apprentissage automatique supervisé.

2.2.4.2 L'apprentissage automatique non supervisé

L'apprentissage automatique non supervisé est une approche qui utilise des données d'entrée sans avoir de valeurs cibles correspondantes. Son objectif est de découvrir des groupes d'exemples similaires dans les données, ce qui est appelé clustering, ou de déterminer la distribution des données dans l'espace d'entrée, ce qui est appelé estimation de densité, ou encore de projeter les données d'un espace de grande dimension vers deux ou trois dimensions dans le but de les visualiser. Lorsqu'on aborde l'apprentissage supervisé, il y a deux considérations principales : la complexité du modèle et le compromis biais-variance. La complexité du modèle fait référence à la complexité de la fonction que l'on cherche à apprendre, similaire au degré d'un polynôme. Le niveau de complexité approprié du modèle est généralement déterminé par la nature des données d'entraînement. Le compromis biais-variance concerne la capacité de généralisation du modèle. Dans tout modèle, il existe un équilibre entre le biais, qui représente l'erreur constante, et la variance, qui représente l'amplitude de l'erreur pouvant varier entre différents ensembles d'entraînement (Soni, 2020).

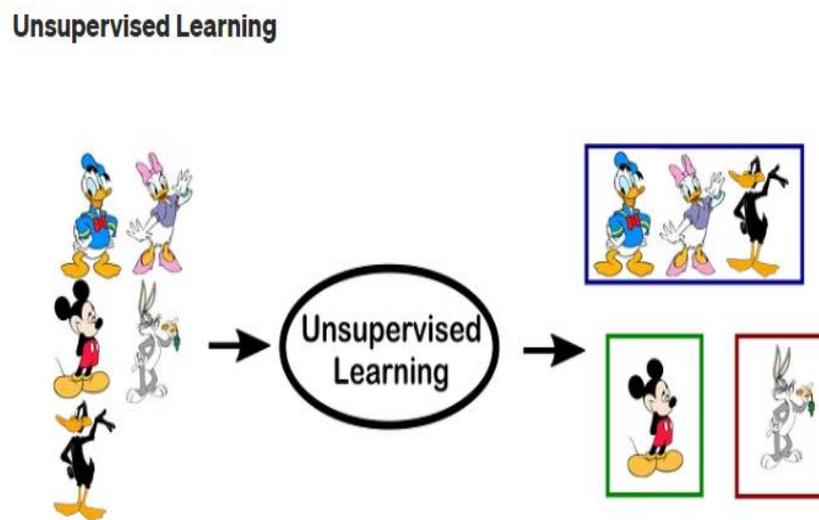


Figure 2. 4: L'apprentissage automatique non supervisé.

2.2.4.3 L'apprentissage par renforcement

L'apprentissage par renforcement est un processus dans lequel l'association entre les actions et les situations est apprise afin de maximiser une récompense quantitative. Dans ce type d'apprentissage, l'apprenant n'est pas informé des actions à entreprendre, mais doit les découvrir en essayant différentes approches. Ce qui rend les situations les plus intéressantes, c'est que les actions peuvent avoir un impact à la fois sur les récompenses immédiates et sur les situations futures, ce qui peut influencer les récompenses à plus long terme. Deux caractéristiques clés de l'apprentissage par renforcement sont l'exploration par essai-erreur et la prise en compte des récompenses à long terme (Bouzy, 2010).

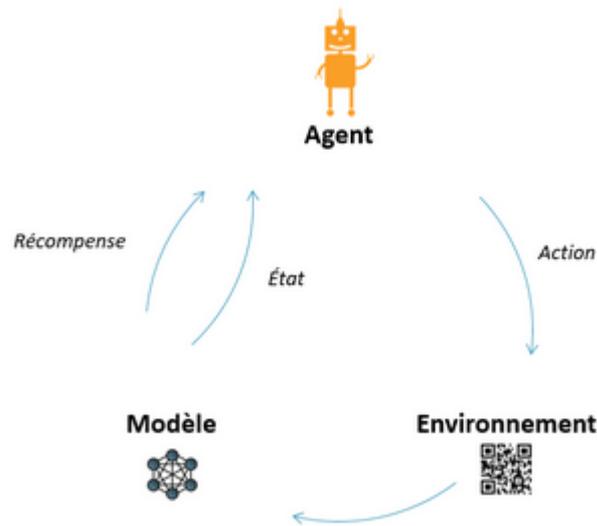


Figure 2. 5: L'apprentissage par renforcement.

2.2.5 Les avantages de l'utilisation du Deep Learning

L'apprentissage profond présente la capacité d'extraire, de traiter et d'apprendre des représentations à partir de données non modifiées et non étiquetées, dans leur forme originale. Cette approche se distingue de manière significative de l'apprentissage machine traditionnel, qui nécessite souvent une sélection minutieuse des caractéristiques des données et une expertise en ingénierie pour les représenter. L'un des avantages clés du Deep Learning par rapport à l'apprentissage machine réside dans sa capacité à comprendre automatiquement les caractéristiques des données brutes, sans nécessiter d'extraction manuelle comme c'est le cas dans l'apprentissage machine. Cela permet de gagner du temps et des efforts, notamment lorsqu'il s'agit de traiter de grandes quantités des données (Nitin & Krishna, 2018). De plus, les résultats obtenus grâce à l'apprentissage profond sont généralement supérieurs en termes de précision et de sophistication par rapport à ceux de l'apprentissage machine traditionnel (Setiowati et al., 2017).

2.2.6 L'Apprentissage Profond

2.2.6.1 Définition

L'apprentissage profond, également connu sous le nom de Deep Learning (DL), est une forme d'intelligence artificielle qui tire son origine de l'apprentissage machine. Dans cette approche, les machines peuvent apprendre de manière autonome, sans supervision ni intervention humaine directe. Elles parviennent à réaliser avec précision des tâches complexes en simulant le fonctionnement du cerveau humain grâce à des réseaux neuronaux artificiels (Deluzarche, 2021).

2.2.6.2 Évolution de l'apprentissage profond

Les premiers algorithmes d'apprentissage que nous connaissons aujourd'hui ont été influencés par le fonctionnement de l'apprentissage biologique, certains d'entre eux cherchant à imiter le cerveau Humain. Le Deep Learning (DL) est une forme d'intelligence artificielle qui repose sur des réseaux neuronaux artificiels (ANN), qui sont la version artificielle des neurones biologiques.

Les recherches sur les ANN remontent aux débuts de l'IA. En 1943, le neuroscientifique Warren McCulloch et le logicien Walter Pitts ont publié un article intitulé "A logical calculus of the ideas immanent in nervous activity". Dans cet article, ils ont proposé le premier modèle de neurone artificiel capable de réaliser des opérations arithmétiques et logiques (McCulloch & Pitts, 1990).

L'objectif de l'apprentissage profond est de reproduire le raisonnement intelligent du cerveau humain en utilisant les neurones comme base. En inversant le mécanisme d'apprentissage du cerveau, il est possible de créer un modèle intelligent (Goodfellow et al., 2016).

Il est donc pertinent d'examiner à la fois le neurone biologique et le neurone artificiel, de comparer leur fonctionnement et d'expliquer les similitudes entre ces deux types de neurones.

1. Neurone Biologique

Un neurone biologique est une cellule vivante qui se compose de trois composantes principales : le corps cellulaire (soma), les dendrites et l'axone (y compris l'axone terminal ou synapse). Le corps cellulaire (soma) peut avoir différentes formes (pyramidale, sphérique, étoilée) et est entouré d'une membrane qui sépare l'intérieur du neurone de l'extérieur. Le neurone possède des prolongements appelés dendrites (entrées) qui reçoivent des signaux provenant d'autres cellules (neurones). L'axone est le seul prolongement du neurone qui transmet le signal à d'autres neurones (sortie) (Stoianov, 2001). Les dendrites et les axones établissent des connexions entre les cellules, où l'axone reçoit l'impulsion électrique du soma et la transmet aux dendrites des neurones connectés par le biais de synapses (terminaisons axonales). Lorsqu'un signal est reçu, le neurone peut être activé ou non (opération booléenne) en fonction de la puissance du signal. S'il est activé, le signal est transmis le long du cerveau, transmettant ainsi le message et déclenchant le processus d'apprentissage (Nedjar, 2020). Voir la figure 2.6 pour plus de détails.

2. Neurone Artificiel

Un neurone artificiel est une représentation mathématique et informatique d'un neurone biologique, capable de traiter les entrées reçues à partir d'un ensemble de connexions caractérisées par des poids synaptiques. Ces poids déterminent l'importance de chaque connexion (Tucci, 2018). L'entrée reçue est traitée à l'intérieur d'un nœud (neurone) et transmise en tant que sortie au nœud suivant. Le nœud reçoit le signal d'entrée et, en tant que fonction de seuil linéaire simple, le transmet à un autre nœud uniquement si le signal est supérieur au seuil limite du nœud (Robins, 2020). Le signal résultant est étiqueté avec un poids qui indique l'amplification ($w > 0$) ou la diminution ($w < 0$) du signal. Ce processus entraîne la mise à jour des poids synaptiques et permet ainsi l'apprentissage (Nedjar, 2020). En 1958, Frank Rosenblatt a mis en œuvre les premiers modèles (Goodfellow et al., 2016) dans son article intitulé "A Probabilistic Model for Information Storage and Organization in the Brain" (Un modèle probabiliste pour le stockage et l'organisation de l'information dans le cerveau). Cela a conduit à un exemple plus flexible appelé le perceptron, capable de mettre à jour automatiquement les poids de classification d'une entrée (Rosenblatt, 1958).

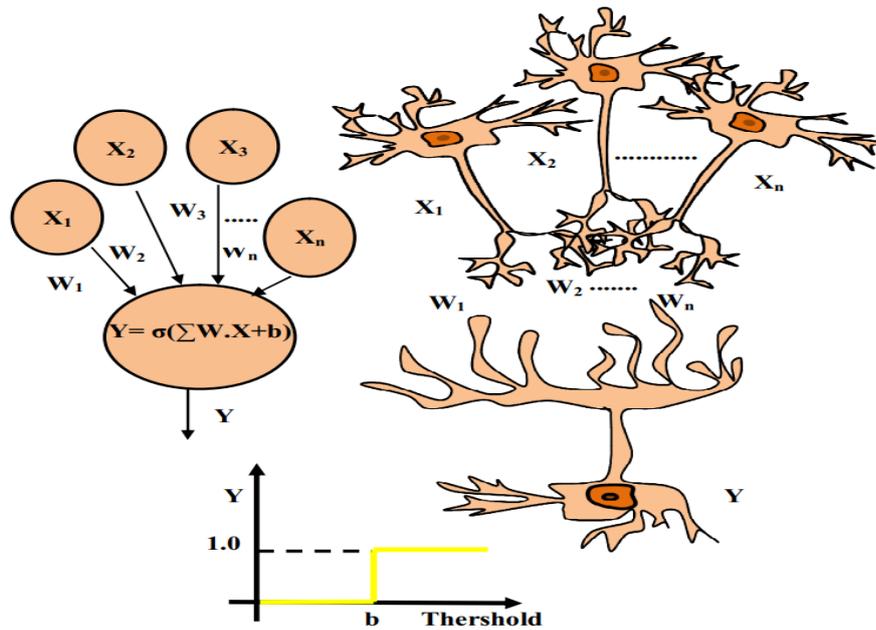


Figure 2. 6: Un neurone biologique en comparaison avec un neurone artificiel (un perceptron).

2.3 Théorie des graphes

2.3.1 Définition des graphes

Un graphe est une représentation visuelle d'un ensemble d'objets, également appelés nœuds ou sommets, qui sont reliés les uns aux autres par des liens, appelés arêtes ou liaisons. Les graphes sont couramment utilisés en informatique pour représenter les relations entre différentes entités, telles que des réseaux sociaux, des circuits électriques, des systèmes de transport routier, et bien d'autres.

Un graphe G est défini comme un couple $G = (V, E)$ composé d'un ensemble de sommets V (vertices en anglais) et d'un ensemble d'arêtes E (edges) reliant certaines paires de sommets de V . Formellement, $E \subseteq V \times V$. Nous considérons le cas de graphes finis, c'est-à-dire ayant un nombre fini de sommets et d'arêtes et notons $n = |V|$ et $m = |E|$ (Canu, 2017).

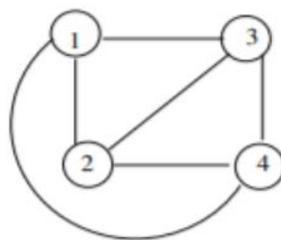


Figure 2. 7: Exemple d'un graphe.

2.3.2 Matrice d'adjacence

La matrice d'adjacence A est une matrice carrée $n \times n$. Pour un graphe non orienté, la matrice d'adjacence est symétrique où $A_{i,j} = 1$ si i et j sont voisins, 0 sinon (Garnine, 2020).

$$A_{i,j} = \{1, \text{si } i \text{ et } j \text{ sont voisins } 0, \text{sinon}\}$$

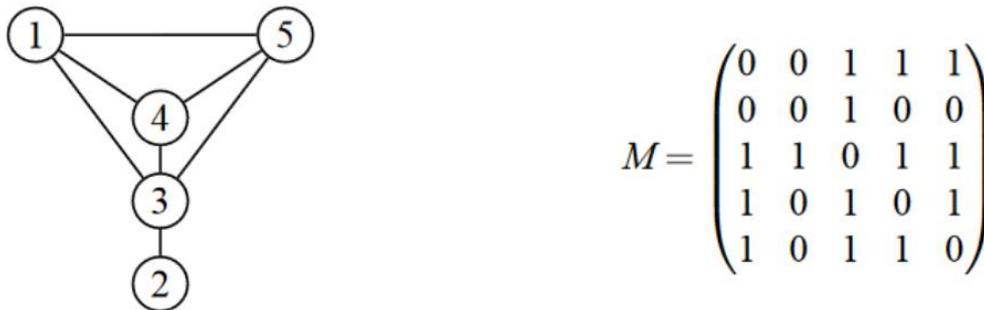


Figure 2. 8: Exemple d'un graphe avec sa matrice d'adjacence.

2.3.3 Types de graphe orienté, non orienté, biparti

2.3.3.1 Graphe orienté

Un graphe orienté est un type spécifique de représentation graphique qui est défini par un ensemble fini de sommets et un ensemble fini d'arcs. Chaque arc est représenté par une paire ordonnée de sommets, indiquant une direction spécifique. Chaque arc a une extrémité initiale et une extrémité terminale, qui sont tous deux des sommets du graphe. Ainsi, pour chaque arc e , il existe une fonction $I(e)$ qui associe l'extrémité initiale de l'arc, et une fonction $T(e)$ qui associe l'extrémité terminale de l'arc (Belaouedj, 2021).

En d'autres termes, un graphe orienté peut être représenté par un quadruplet (X, E, I, T) , où X est l'ensemble des sommets, E est l'ensemble des arcs, I est la fonction qui associe chaque arc à son extrémité initiale, et T est la fonction qui associe chaque arc à son extrémité terminale. Lorsqu'il n'y a pas d'ambiguïté, on peut simplement noter le graphe comme $G(X, E)$.

L'ordre du graphe orienté correspond au nombre de sommets dans l'ensemble X , noté $|X|$. La taille du graphe correspond au nombre d'arcs dans l'ensemble E , noté $|E| = m$ (Seo et al., 2022).

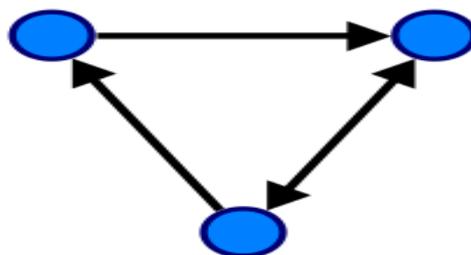


Figure 2. 9: Exemple d'un graphe orienté.

2.3.3.2 Graphe non orienté

Un graphe non orienté est un ensemble d'objets appelés sommets (ou nœuds), reliés par des objets appelés arêtes. Les sommets représentent les entités du graphe, tandis que les arêtes représentent les relations entre ces entités. Les arêtes sont des paires de sommets non ordonnées, c'est-à-dire qu'elles ne possèdent pas de direction. Ainsi, si un sommet x est relié à un sommet y par une arête, on peut dire que x et y sont voisins ou adjacents. Le graphe est donc défini par le couple $G = (X, E)$, où X est l'ensemble des sommets et E est l'ensemble des arêtes, tel que chaque arête soit une paire non ordonnée d'éléments de X . Le nombre de sommets dans le graphe est appelé son ordre et est noté $|X|$, tandis que le nombre d'arêtes est appelé sa taille et est noté $|E|$ (Belaouedj, 2021) .

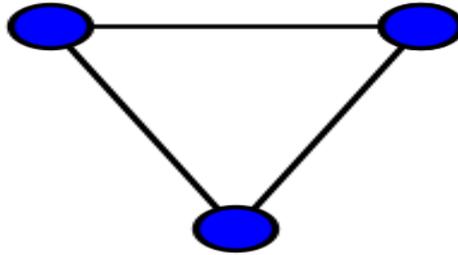


Figure 2. 10: Exemple d'un graphe non orienté.

2.3.3.3 Graphe biparti

Un graphe biparti, également appelé bigraphe, est un réseau dans lequel les nœuds sont divisés en deux ensembles distincts, notés U et V . Les seules connexions présentes dans le graphe sont celles qui relient un nœud de l'ensemble U à un nœud de l'ensemble V . Il n'existe aucun lien direct entre deux nœuds appartenant à U ou entre deux nœuds appartenant à V (Garnine, 2020). Un graphe biparti est également caractérisé par le fait qu'il ne possède pas de boucles, car une boucle connecterait deux sommets dans la même classe.

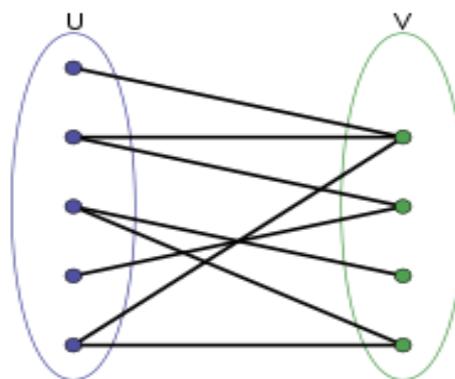


Figure 2. 11: Exemple d'un graphe biparti.

2.4 Réseaux de neurones graphiques

2.4.1 Définition

Un réseau de neurones graphiques, connu sous le nom de Graph Neural Networks (GNN) en anglais, est une classe de réseaux de neurones artificiels (ANN) utilisée pour traiter des données représentées sous forme de graphes. Les GNN sont des architectures neuronales puissantes qui permettent d'apprendre des représentations vectorielles pour les nœuds et les graphes. Ils sont particulièrement efficaces pour résoudre une grande variété de problèmes d'apprentissage automatique en exploitant la structure des données graphiques.

Les GNN ont trouvé des applications dans de nombreux domaines, tels que la vision par ordinateur, la reconnaissance de la parole, la recommandation de contenu, la prédiction de liens, et bien d'autres. Par exemple, en vision par ordinateur, les GNN peuvent analyser la structure spatiale des objets dans une image représentée sous forme de graphe. Dans le domaine de la recommandation de contenu, les GNN sont utilisés pour modéliser les relations entre les utilisateurs et les articles, améliorant ainsi la précision des systèmes de recommandation (Seo et al., 2022).



Figure 2. 12: Les réseaux de neurones graphiques.

2.4.2 Avantages et inconvénients des GNN

Les réseaux de neurones en graphique ont ouvert de nouvelles perspectives qui étaient auparavant limitées avec les réseaux de neurones classiques. Par le passé, il était difficile de traiter efficacement les données en graphe, car les relations entre les données n'étaient pas prises en compte de manière adéquate. Cependant, grâce aux GNN, il est désormais possible de considérer les arêtes avec la même précision que les nœuds. Néanmoins, certaines problématiques liées aux réseaux de neurones persistent avec les GNN, notamment celle de la boîte noire, qui demeure largement non résolue. Nous ne comprenons toujours pas complètement comment un réseau de neurones, en particulier en graphique, effectue sa sélection finale, car il est presque impossible de suivre les processus internes du système depuis l'extérieur (*Graph Neural Network*, 2020).

. Voici quelques avantages et inconvénients des GNN :

2.4.2.1 Avantages

1. **Traitement efficace de données structurées** : les GNN sont très efficaces pour traiter des données structurées, telles que des graphes, des réseaux sociaux, des systèmes de recommandation, etc. Ils peuvent modéliser des relations complexes entre les nœuds et les arêtes d'un graphe, ce qui est difficile à faire avec d'autres modèles.
2. **Robustesse** : les GNN ont une grande capacité à généraliser à de nouveaux graphes qui ont des propriétés similaires aux graphes utilisés lors de l'apprentissage. Cela rend les GNN très robustes et capables de traiter des données très variées.
3. **Interprétabilité** : Les GNN permettent une interprétation plus facile de leurs résultats que les modèles de boîte noire, car ils permettent de visualiser les relations entre les nœuds et les arêtes.

2.4.2.2 Inconvénients

1. **Complexité computationnelle** : Les GNN peuvent être très coûteux en termes de temps et de puissance de calcul, en particulier pour des graphes de grande taille.
2. **Risque de sur-apprentissage** : Les GNN peuvent être sujets au sur-apprentissage, en particulier lorsqu'ils sont utilisés avec des graphes de petite taille.
3. **Besoin de données de grande qualité** : Les GNN ont besoin de données de grande qualité pour fonctionner correctement. Les graphes mal structurés ou bruyants peuvent entraîner des résultats incohérents.

2.4.3 L'apport des réseaux de neurones graphiques pour les systèmes de recommandation

Ces dernières années, les systèmes de recommandation ont évolué rapidement, passant des approches traditionnelles basées sur la factorisation à des modèles avancés utilisant des réseaux de neurones profonds. Parmi ces modèles, ceux basés sur les GNN sont devenus très performants dans de nombreux aspects de la recommandation, couvrant différents scénarios, étapes, objectifs et applications. Le succès de ces systèmes s'explique principalement par trois facteurs : la prise en compte des données structurelles, la connectivité du haut niveau et la disponibilité d'un signal de supervision.

Les plateformes en ligne collectent une grande variété de données, telles que les interactions des utilisateurs (évaluations, clics, achats, etc.), les profils d'utilisateurs (sexe, âge, revenu, etc.), les attributs des articles (marque, catégorie, prix, etc.), et bien d'autres encore. Cependant, les approches traditionnelles de recommandation ne peuvent pas efficacement exploiter ces multiples sources de données, car elles se concentrent généralement sur une ou quelques sources de données spécifiques, ce qui peut entraîner des performances sous-optimales car de nombreuses informations sont ignorées. Les GNN fournissent une méthode unifiée pour utiliser toutes les données disponibles en représentant l'ensemble des données sous forme de nœuds et de bords dans un graphe. Les modèles GNN ont également une forte capacité à apprendre des représentations de haute qualité pour les utilisateurs, les éléments et d'autres fonctionnalités, qui sont essentielles pour améliorer la performance de la recommandation (CHEN et al., 2023).

La précision des recommandations repose sur la capacité à capturer la similitude entre les utilisateurs et les éléments, qui est reflétée dans l'espace d'incorporation appris. Les modèles traditionnels de recommandation ne capturent que de manière implicite l'effet du filtrage collaboratif,

qui est connu pour être crucial pour la précision des recommandations. En revanche, les modèles basés sur GNN peuvent efficacement capturer la connectivité d'ordre élevé, et l'effet de filtrage collaboratif peut être naturellement exprimé en utilisant les voisins à sauts multiples dans le graphe. Cela permet d'améliorer significativement la performance de la recommandation (CHEN et al., 2023).

Les signaux de supervision sont souvent peu fréquents dans les données collectées, ce qui peut poser un problème pour les systèmes de recommandation. Cependant, les modèles basés sur GNN peuvent atténuer ce problème en exploitant des signaux semi-supervisés. Par exemple, dans une plateforme de commerce électronique, les comportements d'achat sont souvent moins fréquents que les autres comportements tels que la recherche ou l'ajout au panier. Les modèles basés sur GNN peuvent utiliser efficacement des signaux semi-supervisés sur le graphe pour incorporer plusieurs comportements non ciblés et améliorer la performance de la recommandation. En outre, des signaux auto-supervisés peuvent être utilisés en concevant des tâches auxiliaires sur le graphe pour améliorer encore davantage la précision de la recommandation.

2.4.4 Fonctionnement des réseaux de neurones graphiques

Il est possible de diviser les approches qui appliquent l'apprentissage profond aux graphes en deux catégories distinctes. La première catégorie comprend les Graph Neural Networks (GNN), qui ont pour objectif de permettre aux techniques d'apprentissage profond de traiter des données de type graphe en entrée. La deuxième catégorie comprend les méthodes de plongement de graphes ou graph embedding, qui projettent les informations du graphe dans un espace vectoriel de faible dimension compatible avec les algorithmes d'apprentissage profond classiques.

Les Graph Neural Networks (GNN) utilisent des vecteurs caractéristiques, connus sous le nom d'embeddings, pour encoder les informations relatives à un graphe, comme sa structure et ses relations. Ces embeddings permettent de créer une représentation numérique du graphe, qui peut être utilisée pour entraîner des algorithmes de Machine Learning. Dans un graphe, il y a deux types d'embeddings : les embeddings de nœud (Node embeddings) qui représentent les propriétés de chaque nœud, et les embeddings d'arêtes qui modélisent les propriétés des relations entre les nœuds.

Les étapes du fonctionnement des GNN sont (*Graph Neural Networks (GNN): qu'est-ce que c'est ?*, 2023) (Rishit, 2022):

2.4.4.1 Construction du graphe

Le graphe est construit à partir des données disponibles, en créant des nœuds et des arêtes pour représenter les éléments et leurs relations respectives.

2.4.4.2 Initialisation des embeddings

Chaque nœud est initialement représenté par un vecteur caractéristique appelé embedding. Les embeddings peuvent être initialisés aléatoirement ou à partir de pré-entraînements.

2.4.4.3 La technique de message passing

Cette technique permet de traiter l'information à travers un échange d'information entre les nœuds d'un graphe et leurs voisins. Le but est de modifier l'état de chaque nœud en utilisant les informations locales de ses voisins et des arêtes.

2.4.4.4 La mise à jour des embeddings

Une fonction non linéaire est utilisée pour que chaque nœud destinataire puisse combiner tous les messages reçus avec son embedding initial et les embeddings d'arêtes correspondants, de manière à mettre à jour son propre embedding.

2.4.4.5 Les embeddings de nœud

Contiennent des informations sur les propriétés de chaque nœud et sont transmis via la technique de "message passing". Chaque nœud envoie un message contenant son embedding à ses voisins, fournissant des informations sur lui-même et les propriétés spécifiques de l'arête.

2.4.4.6 La répétition du message passing

On répète plusieurs fois la technique de message passing pour que chaque nœud puisse intégrer progressivement l'information de ses voisins et ainsi capturer de plus en plus d'informations sur la structure du graphe.

2.4.4.7 L'invariance par permutation

Les GNN présentent une invariance par permutation, c'est-à-dire que les fonctions de message passing sont insensibles à l'ordre des nœuds voisins, ce qui permet aux GNN de considérer les interactions entre les nœuds indépendamment de l'ordre des nœuds et d'obtenir des résultats cohérents.

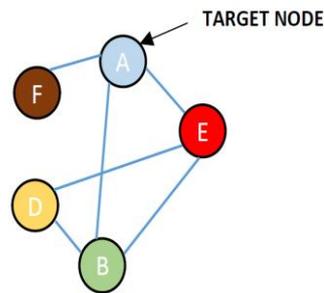


Figure 2. 13: Données non euclidiennes sous forme de graphe.

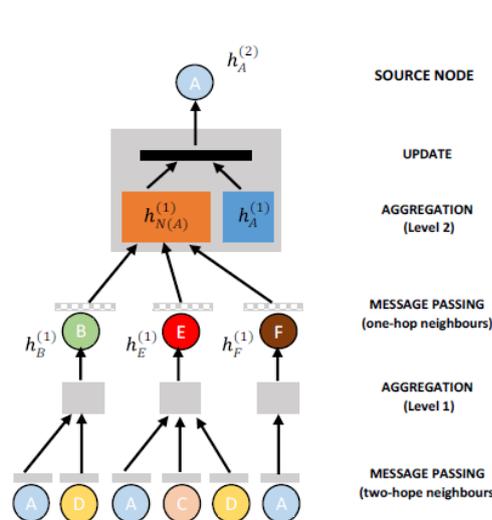


Figure 2. 14: Fonctionnement des réseaux neuronaux graphiques.

2.4.5 Architectures des réseaux de neurones graphiques

Il existe de nombreuses architectures de GNN (Graph Neural Networks) qui sont utilisées pour des tâches variées, et les choix sont faits en fonction de la tâche spécifique à résoudre. Voici quelques exemples d'architectures couramment utilisées (*Graph Neural Networks (GNN): qu'est-ce que c'est ?*, 2023) (Rishit, 2022):

2.4.5.1 Réseaux convolutifs graphiques (GCN)

Les GCN appliquent une couche de convolution sur les embeddings des nœuds et des arêtes pour agréger l'information des nœuds voisins.

2.4.5.2 Réseaux d'attention graphiques (GAT)

Les GAT utilisent l'attention pour pondérer l'importance des nœuds voisins lors de la propagation de l'information.

2.4.5.3 Réseaux de neurones à propagation de messages (MPNN)

Les MPNN utilisent le message passing pour mettre à jour les états des nœuds et des arêtes en utilisant des fonctions de mise à jour définies par l'utilisateur.

2.4.6 Les différents domaines pour l'application de GNN

Les réseaux de neurones graphiques (GNN) offrent une multitude d'applications potentielles et peuvent être utilisés dans divers domaines pour résoudre des problèmes complexes impliquant des données structurées. Voici quelques exemples d'applications (*Que sont les réseaux de neurones graphiques et comment fonctionnent-ils ?*, 2022) (*Graph Neural Network (GNN)*, 2023):

2.4.6.1 Recommandation de produits

Les GNN peuvent être utilisés pour recommander des produits en analysant les interactions entre les utilisateurs et les produits dans un réseau social ou une plateforme de commerce en ligne.

2.4.6.2 Classification d'images et de vidéos

Les GNN peuvent être utilisés pour classifier des images et des vidéos en prenant en compte les relations spatiales et temporelles entre les objets présents dans ces médias.

2.4.6.3 Prédiction de la structure des molécules

Les GNN peuvent être utilisés pour prédire la structure des molécules en se basant sur la structure de leur graphe moléculaire, permettant ainsi des avancées dans le domaine de la chimie et de la pharmacologie.

2.4.6.4 Analyse de sentiments

Les GNN peuvent être utilisés pour l'analyse de sentiments en prenant en compte les relations entre les mots dans un texte, ce qui permet de mieux comprendre les opinions et les sentiments exprimés.

2.4.6.5 Détection de fraude

Les GNN peuvent être utilisés pour détecter des fraudes en analysant les motifs de comportement anormal dans un réseau de transactions financières, aidant ainsi à prévenir les activités frauduleuses.

Il convient de noter que ces exemples ne sont pas exhaustifs et que les GNN peuvent être appliqués à de nombreux autres domaines et problèmes, démontrant leur polyvalence et leur potentiel dans le traitement des données structurées.

2.4.7 Les problèmes résolus par les GNN

Les problèmes résolus par les GNN peuvent être classés dans les catégories suivantes (Menzli, 2023):

2.4.7.1 Classification des nœuds

Dans cette tâche, il s'agit de déterminer l'étiquetage des échantillons (représentés par des nœuds) en examinant les étiquettes de leurs voisins. Habituellement, ces problèmes sont entraînés de manière semi-supervisée, avec seulement une partie du graphe étant étiquetée.

2.4.7.2 Classification des graphes

Dans cette tâche, il s'agit de classifier l'ensemble du graphe en différentes catégories. C'est comme la classification d'images, mais dans le domaine des graphes. Les applications de la classification des graphes sont nombreuses, allant de la détermination de savoir si une protéine est une enzyme ou non en bio-informatique, à la catégorisation de documents en traitement du langage naturel (NLP) ou à l'analyse des réseaux sociaux.

2.4.7.3 Visualisation des graphes

Il s'agit d'un domaine des mathématiques et de l'informatique, à l'intersection de la théorie des graphes géométriques et de la visualisation de l'information. Il s'intéresse à la représentation visuelle des graphes qui révèle les structures et les anomalies présentes dans les données, et aide l'utilisateur à comprendre les graphes.

2.4.7.4 Prédiction de liens

Ici, l'algorithme doit comprendre la relation entre les entités dans les graphes et essayer de prédire s'il y a une connexion entre deux entités. C'est essentiel dans les réseaux sociaux pour déduire les interactions sociales ou suggérer des amis potentiels aux utilisateurs. Cela a également été utilisé dans les problèmes de systèmes de recommandation et dans la prédiction d'associations criminelles.

2.4.7.5 Regroupement de graphes

Cela fait référence à la classification des données sous forme de graphes. Il existe deux formes distinctes de regroupement effectuées sur les données de graphe. Le regroupement des sommets cherche à regrouper les nœuds du graphe en groupes de régions densément connectées en se basant soit sur les poids des arêtes, soit sur les distances entre les arêtes. La deuxième forme de regroupement de graphes traite les graphes comme des objets à regrouper et regroupe ces objets en fonction de leur similarité.

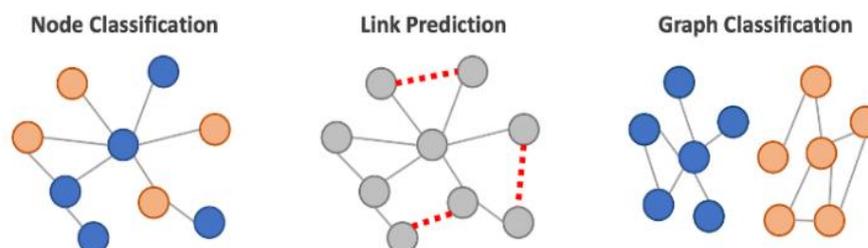


Figure 2. 15: Les problèmes résolus par les GNN.

2.4.8 L'importance de GNN

Les Graph Neural Networks (GNN) sont des outils puissants pour résoudre des problèmes complexes en exploitant les graphes, qui sont des structures de données complexes et non structurées. Leur fonctionnement repose sur la manipulation des nœuds et des arêtes du graphe à l'aide d'architectures telles que les Graph Convolutional Networks (GCN) et les Graph Attention Networks (GAT), qui permettent de créer des représentations adaptées à la structure et aux informations du graphe. Les GNN peuvent être appliqués dans divers domaines tels que la vision par ordinateur, la prédiction de liens, la recommandation et le traitement du langage naturel (NLP). Avec leur potentiel prometteur, leur utilisation devrait continuer à croître pour résoudre de nombreux autres problèmes complexes. De plus, étant donné que les GNN sont en constante évolution, de nouvelles architectures pourraient émerger à l'avenir pour repousser les limites actuelles. Les GNN ont été introduits pour la première fois dans un article publié en 2009 par des chercheurs italiens. Deux chercheurs d'Amsterdam ont également démontré la puissance des GNN avec une autre variante appelée Graph Convolutional Network (GCN).

2.5 Réseaux convolutifs graphiques (GCN)

2.5.1 Définition de GCN

Les réseaux convolutifs graphiques (GCN) sont une classe de modèles de réseaux de neurones convolutifs conçus pour traiter des données structurées sous forme de graphes. Les GCN sont des modèles de propagation de l'information sur des graphes qui utilisent une opération de convolution spéciale pour agréger les informations des nœuds voisins dans le graphe. Cette opération de convolution est adaptée à la structure du graphe et permet d'exploiter les relations locales entre les nœuds (Kipf & Welling, 2017).

Les réseaux convolutifs graphiques sont un type de réseau de neurones qui sont conçus pour travailler avec des données structurées sous forme de graphes. Dans un graphe, les nœuds représentent des entités telles que des utilisateurs dans un réseau social, des molécules dans une structure chimique, ou des mots dans un document. Les arêtes représentent les relations entre ces entités, telles que les amitiés entre les utilisateurs, les liens chimiques entre les molécules, ou les co-occurrences de mots dans un document. Les GCN utilisent une convolution définie sur le graphe pour apprendre des représentations de nœuds qui peuvent être utilisées pour des tâches telles que la classification, la segmentation et la prédiction de liens. Cette convolution sur le graphe est définie de manière similaire à la convolution en traitement d'image, mais avec une matrice de convolution qui est apprise par le réseau. Cette matrice de convolution permet au réseau d'apprendre des motifs locaux dans le graphe qui peuvent être utilisés pour prédire les étiquettes ou les propriétés des nœuds. Les GCN utilisent également la propagation de messages pour mettre à jour les représentations des nœuds en fonction de leurs voisins dans le graphe. Ce processus est répété pour chaque couche du réseau pour obtenir des représentations de plus en plus complexes des nœuds. La propagation de messages permet également de prendre en compte l'ensemble du graphe lors de la mise à jour des représentations de chaque nœud, ce qui peut améliorer les performances du réseau pour des tâches telles que la classification (S. Zhang et al., 2019).

Les GCN peuvent être entraînés à l'aide de techniques d'apprentissage supervisé ou non supervisé pour apprendre des représentations de nœuds de haute qualité. Les représentations apprises peuvent ensuite être utilisées pour résoudre une variété de tâches, telles que la recommandation, la prédiction de liens, la segmentation sémantique et la classification.

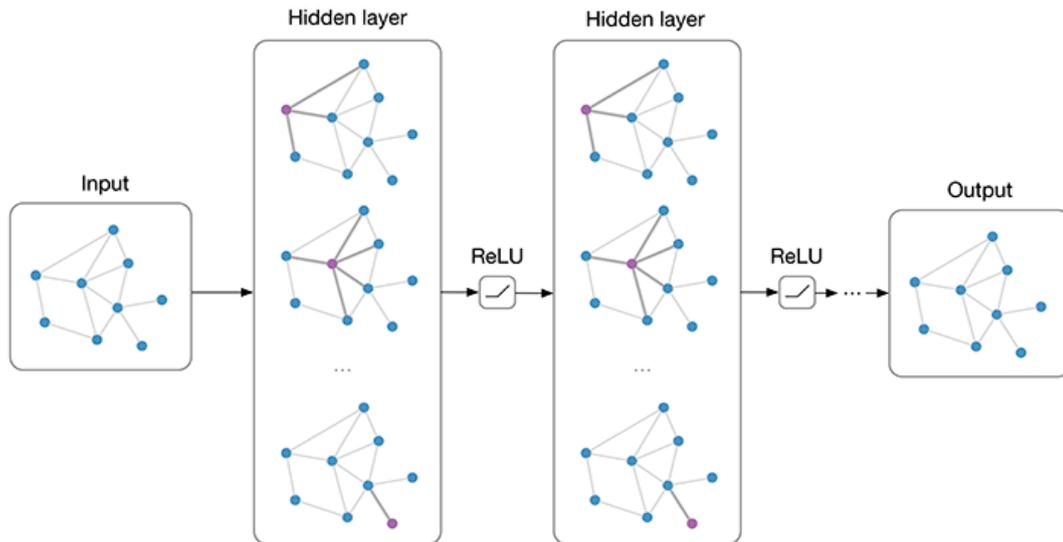


Figure 2. 16: Les réseaux convolutifs graphiques.

2.5.2 Diffusion de l'information dans les graphes avec GCN

Les Graph Convolutional Networks (GCNs) sont des réseaux de neurones profonds qui apprennent des représentations de nœuds et de graphes en utilisant des convolutions de graphes, similaires aux convolutions dans les réseaux de neurones convolutifs pour les données structurées telles que les images.

Les GCNs apprennent à représenter les nœuds et les graphes en agrégeant les informations de leurs voisins et en utilisant des transformations linéaires et non linéaires pour générer de nouvelles représentations à chaque couche. Ces représentations peuvent ensuite être utilisées pour résoudre diverses tâches d'apprentissage supervisées et non supervisées sur les données de graphe.

Les étapes de la diffusion sont (WelcomeAIOverlords, 2020) :

2.5.2.1 Initialisation

Chaque nœud du graphe est représenté par un vecteur initial (par exemple, un vecteur aléatoire ou un vecteur pré-entraîné).

2.5.2.2 Agrégation de voisinage

À chaque couche du GCN, les nœuds agrègent les informations de leurs voisins directs en utilisant une matrice de pondération. Cela permet de prendre en compte les relations locales entre les nœuds du graphe.

2.5.2.3 Mise à jour

Les activations agrégées sont ensuite utilisées pour mettre à jour les activations des nœuds. Cette mise à jour peut être effectuée en utilisant une fonction d'activation non linéaire, telle que la fonction ReLU.

2.5.2.4 Répétition

Les étapes d'agrégation et de mise à jour sont répétées sur plusieurs couches de GCN pour permettre la propagation de l'information à travers le graphe. À chaque itération, les activations des

nœuds se propagent aux nœuds voisins, permettant ainsi une diffusion de l'information dans l'ensemble du graphe.

2.5.2.5 Utilisation des activations finales

Une fois que toutes les itérations sont terminées, les activations finales des nœuds peuvent être utilisées pour effectuer différentes tâches sur le graphe, telles que la classification des nœuds, la prédiction des liens ou la recommandation.

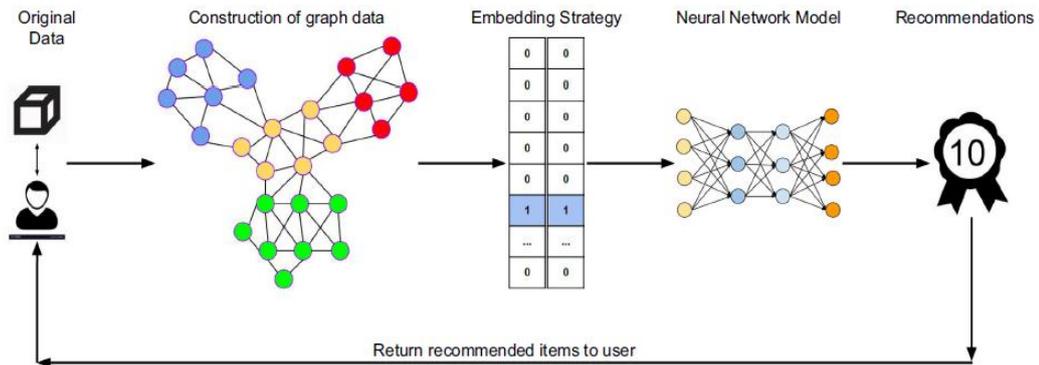


Figure 2. 17: Flux de travail d'un système de recommandation basé sur l'apprentissage en profondeur utilisant des données structurées sous forme de graphe (D. Tran et al., 2021).

2.5.3 Les catégories de réseaux Graphiques convolutifs

Les Graph Convolutional Networks (GCN) peuvent être classés en deux principales variantes les GCN spatiaux (spatial GCN) et les GCN spectraux (spectral GCN). Chacune de ces variantes utilise une approche différente pour effectuer la convolution sur les données structurées en graphe.

2.5.3.1 Les réseaux convolutifs à graphes spatiaux

Les réseaux convolutifs à graphes spatiaux utilise une approche basée sur la propagation d'informations locales dans le graphe. L'idée principale est de combiner les caractéristiques des nœuds voisins pour mettre à jour les caractéristiques du nœud cible. Les étapes principales du GCN spatial sont les suivantes (Bruna et al., 2014) :

Agrégation des caractéristiques des nœuds voisins : Les caractéristiques des nœuds voisins sont agrégées en utilisant des opérations telles que la somme, la moyenne ou la concaténation.

Transformation linéaire : Une transformation linéaire est appliquée à la représentation agrégée pour mettre à jour les caractéristiques du nœud cible.

Fonction d'activation : Une fonction d'activation non linéaire est appliquée pour introduire de la non-linéarité dans le modèle.

Les réseaux convolutifs à graphes spatiaux est efficace pour capturer les relations locales entre les nœuds voisins dans le graphe. Il est principalement utilisé dans des tâches telles que la classification de nœuds et la prédiction de liens dans les réseaux sociaux, où les informations locales jouent un rôle important.

2.5.3.2 Les réseaux convolutifs à graphes spectraux

Le GCN spectral utilise une approche basée sur la transformation de Fourier sur le graphe. Il se base sur la décomposition spectrale de la matrice d'adjacence du graphe pour effectuer la convolution. Les étapes principales du GCN spectral sont les suivantes (Rishit, 2022) (Bruna et al., 2014) :

Calcul des valeurs propres et des vecteurs propres : La matrice d'adjacence du graphe est utilisée pour calculer les valeurs propres et les vecteurs propres.

Transformation spectrale : Les caractéristiques des nœuds sont transformées dans le domaine spectral en utilisant les vecteurs propres.

Filtre spectral : Un filtre est appliqué dans le domaine spectral pour filtrer les caractéristiques transformées.

Transformation inverse : La transformation inverse est effectuée pour obtenir les caractéristiques mises à jour dans le domaine original.

Les réseaux convolutifs à graphes spectraux est capable de capturer les structures globales et les motifs du graphe en utilisant les informations fréquentielles. Il est souvent utilisé dans des tâches telles que la classification de graphe et la recommandation de produits, où les relations globales dans le graphe sont importantes.

Donc, Les réseaux convolutifs à graphes spatiaux se concentrent sur la propagation d'informations locales, tandis que Les réseaux convolutifs à graphes spectraux exploitent les propriétés spectrales du graphe pour la convolution. Chacune de ces variantes à ses avantages et ses applications spécifiques, et il est important de choisir la bonne approche en fonction de la tâche et des caractéristiques du graphe.

Les CNN (Convolutional Neural Networks) et les GCN (Graph Convolutional Networks) sont tous deux des architectures de réseaux neuronaux utilisées pour l'apprentissage automatique. Cependant, ils sont conçus pour traiter des types de données différents.

2.5.4 Différence entre Réseaux de neurones convolutifs et réseaux Graphiques convolutifs

Les CNN sont principalement utilisés pour l'analyse de données structurées sous forme de grilles, telles que des images. Ils sont très efficaces pour extraire des caractéristiques locales et spatiales en utilisant des opérations de convolution et de pooling. Les CNN sont capables de capturer des motifs locaux dans des régions voisines et de les combiner pour former des caractéristiques de niveau supérieur. Ils sont largement utilisés dans des tâches telles que la classification d'images, la détection d'objets et la reconnaissance faciale.

En revanche, les GCN sont spécialement conçus pour traiter des données structurées sous forme de graphes, tels que des réseaux sociaux, des données biologiques et des recommandations. Les graphes sont des structures complexes qui représentent des relations entre des entités. Les GCN utilisent des opérations de convolution sur les graphes pour agréger l'information des voisins d'un nœud et mettre à jour les caractéristiques de ce nœud en conséquence. Ils permettent une propagation de l'information efficace à travers les graphes et peuvent capturer des motifs locaux et globaux.

Les GCN sont utilisés pour des tâches telles que la classification de nœuds, la prédiction de liens et la recommandation personnalisée.

Donc, la principale différence entre les CNN et les GCN réside dans les types de données qu'ils traitent. Les CNN sont adaptés aux données structurées sous forme de grilles, tandis que les GCN sont adaptés aux données structurées sous forme de graphes.

2.6 Conclusion

Les réseaux de neurones à convolution graphiques (GCN) ont émergé comme une approche prometteuse pour l'analyse de données non structurées. Ils exploitent les propriétés topologiques des graphes pour extraire des informations significatives et offrent une solution efficace pour les tâches de classification et de prédiction. Cependant, leur utilisation nécessite une compréhension approfondie de la théorie des graphes et des techniques d'apprentissage profond. Le prochain chapitre abordera l'implémentation de LightGCN, un modèle de GCN efficace pour la recommandation de produits, en se concentrant sur les détails du codage et de la mise en œuvre.

Chapitre 3 : Les systèmes de recommandation de santé basés sur LightGCN

Chapitre 3 : Les systèmes de recommandation de santé basés sur LightGCN

3.1 Introduction

Dans ce chapitre, nous abordons les systèmes de recommandation de santé basés sur LightGCN, une méthode innovante pour fournir des recommandations personnalisées dans le domaine de la santé, en se concentrant sur les détails du codage et de la mise en œuvre. De plus, nous explorerons les performances de LightGCN par rapport aux autres modèles de recommandation populaires. L'objectif de ce chapitre est de présenter une approche complète pour la conception, l'implémentation et l'évaluation des systèmes de recommandation de santé basés sur LightGCN. En comprenant les détails techniques et les performances de notre modèle, nous pourrions mieux apprécier son potentiel pour améliorer la précision et la personnalisation des recommandations dans le domaine de la santé.

3.2 Conception de GCN pour HRS basé sur FC

Dans cette section, nous présentons notre modèle de réseaux convolutifs graphiques (GCN) pour la conception des systèmes de recommandation de santé (HRS) et nous mettons en évidence les principaux avantages de l'utilisation de GCN.

3.2.1 Pourquoi les réseaux convolutifs graphiques pour les HRS

Les réseaux neuronaux graphiques commencent progressivement à jouer un rôle de plus en plus important dans l'apprentissage des représentations. En particulier, les réseaux convolutifs graphiques (GCN) présentent une capacité remarquable à traiter des données non ordonnées et irrégulières telles que les signaux cérébraux, Selon (Wu et al., 2022) et (Gao et al., 2021), nous pouvons résumer les principales raisons de choisir GCN pour les systèmes de recommandation de santé comme suit :

3.2.1.1 Traitement des données non euclidiennes

Les systèmes de recommandation traditionnels basés sur les modèles d'apprentissage profond (comme les CNN ou les RNN) ne sont pas capables de tirer parti des données non euclidiennes (par exemple, les réseaux sociaux, les molécules et la propagation des maladies). Les GCN en tant qu'architecture GNN permettent de surmonter cette limitation en représentant ces données sous forme des graphes et en effectuant des prédictions utiles.

3.2.1.2 Connectivité d'ordre élevé

Les approches classiques de filtrage collaboratif se limitent à une connectivité de premier ordre, où seules les interactions directes entre les utilisateurs et les éléments sont prises en compte. Cependant, les mêmes éléments sont également interagis par d'autres utilisateurs voisins qui sont pertinents pour l'utilisateur dans le contexte du filtrage collaboratif. Les GCN, grâce à leur connectivité d'ordre élevé, peuvent capturer facilement ces interactions complexes entre les utilisateurs et les éléments partagés (Wang et al., 2019). Dans les systèmes de recommandation de santé utilisant les GCN, l'effet de filtrage collaboratif peut être exprimé par des voisins à plusieurs niveaux sur le graphe (par exemple, des médicaments et des régimes similaires entre les patients) et est intégré dans les représentations apprises grâce à la propagation et à l'agrégation des embeddings (convolutions). Cela conduit à une amélioration des performances et de la précision des recommandations en matière de santé.

3.2.1.3 Incorporation de plusieurs comportements

Les interactions utilisateur-élément impliquent généralement plusieurs comportements de l'utilisateur (par exemple, pratiquer un sport, prendre un médicament, suivre un régime, etc.). Généralement, les systèmes de recommandation traitent un seul comportement cible présentant une faible densité de données. Les GCN peuvent incorporer efficacement plusieurs comportements et signaux de collaboration sur les graphes en utilisant des couleurs et des arêtes pondérées. Cela peut permettre d'améliorer les performances des systèmes de recommandation de santé.

3.2.2 Le modèle proposé pour les systèmes de recommandation de santé

Pour la construction de systèmes de recommandation de santé, de nombreuses techniques de recommandation de base peuvent être utilisées (T. N. T. Tran et al., 2021) (Etemadi et al., 2023), telles que le filtrage collaboratif, le filtrage basé sur le contenu, la recommandation basée sur les connaissances et la recommandation hybride, qui combinent les techniques de recommandation mentionnées précédemment. Dans notre cas, la technique du filtrage collaboratif (CF) est adoptée au sein de réseaux de convolution neuronaux hétérogènes (deux types de nœuds : utilisateurs et articles) (Li et al., 2022). Selon (T. N. T. Tran et al., 2021), l'utilisation du CF dans les SRH peut être interprétée comme suit: "les patients qui partagent des conditions de santé/profils de maladie similaires auraient alors des services de santé/traitements similaires" et ainsi de suite.

Dans la littérature, très peu de travaux de recherche ont utilisé les GCN pour le développement de systèmes de recommandation de santé, dans lesquels une couche de convolution graphique standard (H. Zhang et al., 2022) a été utilisée. Ce travail se concentre sur l'utilisation d'une couche GCN simplifiée et efficace telle qu'évoluée dans (He et al., 2020). La figure 5 représente un modèle d'architecture pour développer des systèmes de recommandation de santé en utilisant LightGCN. Ce modèle peut être défini comme un réseau neuronal profond avec deux ou plusieurs couches cachées visant à améliorer la précision des prédictions à partir des graphes d'entrée.

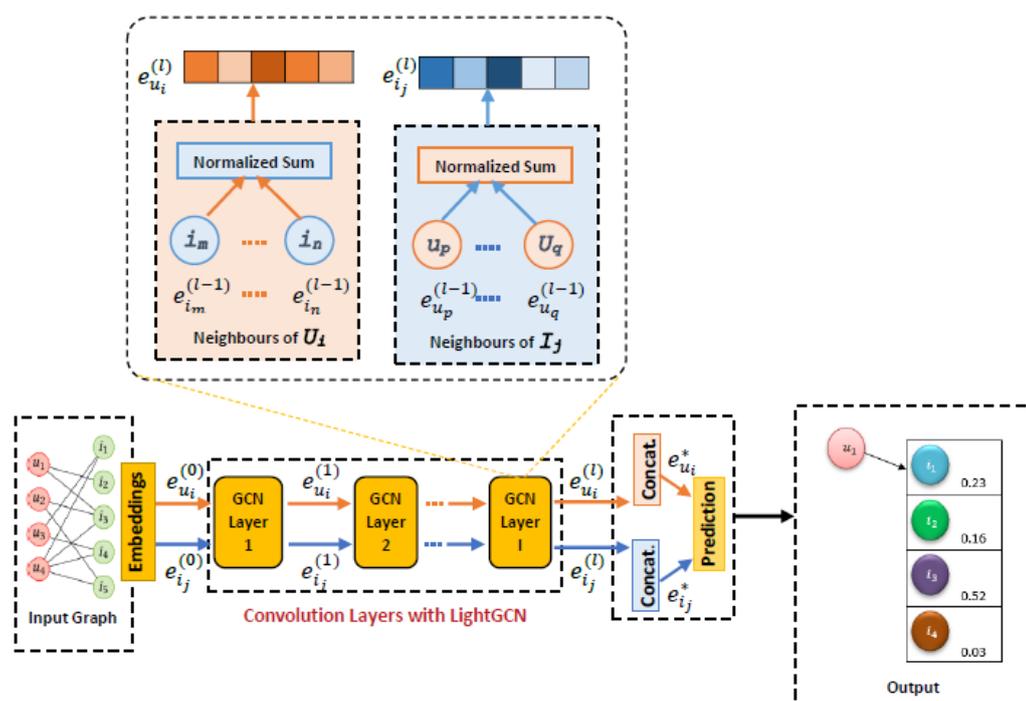


Figure 3. 1: Modèle GCN pour les systèmes de recommandation de santé basés sur la FC.

Pour comprendre le modèle proposé, la Table 3.1 récapitule les notations utilisées et leurs descriptions.

Table 3. 1 : Notations et descriptions.

Notation	Description
$e_u^{(0)}$	Embedding de l'utilisateur u à la couche 0
$e_i^{(0)}$	Embedding de l'item i à la couche 0
$e_u^{(k)}$	Embedding raffiné de l'utilisateur u après k couches de propagation
$e_i^{(k)}$	Embedding raffiné de l'item i après k couches de propagation
N_u	L'ensemble des items avec lesquels l'utilisateur u interagit
N_i	L'ensemble des utilisateurs qui interagissent avec l'item i
α_k	L'importance de l'embedding de la -ème couche dans la constitution de l'ensemble final d'embeddings est définie comme $1/(k + 1)$
$ N_u $	La somme des embeddings normalisés des voisins de l'utilisateur u
k	Nombre de couches

Les GCN sont principalement basés sur l'apprentissage de la représentation des nœuds en lissant les caractéristiques sur le graphe. Cela est réalisé en agrégeant de manière itérative les caractéristiques voisines en tant que nouvelle représentation d'un nœud cible (c'est-à-dire une opération de convolution de graphe). Une telle agrégation de voisinage peut être formalisée comme suit :

$$e_u^{(k+1)} = \text{AGGREGATE}(e_u^k, \{e_i^k : i \in N_u\})$$

Dans LightGCN (He et al., 2020), nous distinguons deux opérations, à savoir :

- **La convolution graphique** : dans LightGCN utilise une fonction d'agrégation basée sur une somme pondérée. L'opération de convolution graphique, également appelée règle de propagation, est définie de la manière suivante :

$$e_u^{(k+1)} = \sum_{i \in N_u} \frac{1}{\sqrt{|N_u|} \sqrt{|N_i|}} e_i^{(k)}$$

$$e_i^{(k+1)} = \sum_{u \in N_i} \frac{1}{\sqrt{|N_i|} \sqrt{|N_u|}} e_u^{(k)}$$

- **Concaténation et prédiction** : Au début, les embeddings à la couche 0 pour tous les utilisateurs et les items sont donnés pour l'entraînement, notés $e_u^{(0)}$ et $e_i^{(0)}$ respectivement. Ensuite, les embeddings aux couches supérieures peuvent être calculés via l'opération de convolution graphique définie dans les deux équations précédentes. Après k couches de convolution graphique, l'embedding final d'un utilisateur (d'un item) est calculé en concaténant les embeddings obtenus à chaque couche :

$$e_u = \sum_{k=0}^k \alpha_k e_u^{(k)}$$

$$e_i = \sum_{k=0}^k \alpha_k e_i^{(k)}$$

Dans ce modèle, la prédiction est définie comme le produit scalaire des embeddings finaux de l'utilisateur et de l'item :

$$\widehat{y}_{ui} = e_u^T e_i$$

3.3 Outils de mise en œuvre

Cette section est consacrée à un aperçu du langage de programmation, des bibliothèques, des outils et du matériel utilisés dans l'ensemble pour la mise en œuvre.

3.3.1 Python

Python est un langage de programmation orienté objet interprété inventé vers 1991 dans le but de mettre l'accent sur la lisibilité du code. Il prend en charge les modules et les packages, ce qui encourage la modularité du programme et la réutilisation du code. Python est le langage idéal pour mettre en œuvre rapidement une idée abstraite, rassembler différentes bibliothèques pour réaliser quelque chose de nouveau, traiter et extraire des données depuis Internet (Chockz21, 2021).



Figure 3. 2: Logo Python.

3.3.2 Colaboratory

Colaboratory¹, connu sous le nom de "Colab" pour faire court, est un produit de Google Research. Colab permet à quiconque d'écrire et d'exécuter du code Python arbitraire via un navigateur, et il est particulièrement adapté à l'apprentissage automatique, à l'analyse de données et à l'éducation. Plus précisément, Colab est un service de cahier Jupyter hébergé qui peut être utilisé sans aucune configuration préalable, offrant ainsi un accès gratuit à des ressources informatiques, y compris des GPU (*Google Colab*).



Figure 3. 3: Logo Colab.

3.3.3 Bibliothèques

3.3.3.1 PyTorch Geometric

PyG² est une bibliothèque construite sur PyTorch pour faciliter la création et l'entraînement de réseaux neuronaux graphiques (GNN), y compris le type GCN, pour une large gamme d'applications liées aux données structurées. Elle comprend différentes méthodes pour l'apprentissage en profondeur sur les graphes et autres structures irrégulières. Grâce à cette bibliothèque, nous pouvons facilement représenter les différentes données nécessaires pour construire des systèmes de recommandation basés sur le filtrage collaboratif efficace, en utilisant des réseaux de convolution graphique (*Graphcore intègre Pytorch Geometric à sa pile logicielle*).



Figure 3. 4: Logo Pytorch Geometric.

3.3.3.2 Matplotlib

Matplotlib est une bibliothèque Python open source utilisée pour la visualisation de données. Elle offre une large gamme de graphiques différents (*Matplotlib : Tout savoir sur la bibliothèque Python de Dataviz*).

¹ <https://colab.research.google.com/?hl=fr>

² <https://www.pyg.org/>



Figure 3. 5: Logo Matplotlib.

3.3.3.3 Pandas

Pandas est une autre bibliothèque open-source pour Python utilisée pour l'analyse de données. Elle est particulièrement utile pour l'importation de données. Sa classe Data Frame est une méthode excellente pour représenter des données tabulaires, ce qui facilite le prétraitement, la modification ou la découpe des données (rédac, 2022).



Figure 3. 6: Logo Pandas.

3.3.3.4 Numpy

Créé en 2005 par Travis Oliphant, est une bibliothèque Python utilisée pour manipuler des tableaux. Elle propose un objet tableau qui peut être jusqu'à 50 fois plus rapide que les listes Python classiques (*Introduction à NumPy — Cours Python*).



Figure 3. 7: Logo NumPy.

3.3.3.5 Scikit-learn

Scikit-learn est une bibliothèque logicielle open-source qui offre une variété d'outils pour l'analyse de données. Elle met à disposition des classes dédiées à la préparation des données et à la résolution de problèmes de machine Learning tels que la classification, la régression, le regroupement (clustering) et la réduction de dimensionnalité. Cette bibliothèque s'appuie sur les fonctionnalités des bibliothèques NumPy et Matplotlib (*scikit-learn: machine learning in Python*).



Figure 3. 8: Logo Scikit-Learn.

3.3.4 Environnement d'exécution

Pour l'implémentation, nous avons utilisé une machine Windows équipée d'un processeur Intel i5 de 6ème génération, de 8 Go de RAM et d'un SSD de 256 Go.

3.4 Expérimentation

Les expériences ont été réalisées avec un modèle LightGCN sur deux ensembles de données. Nous avons évalué le système de recommandation proposé en calculant les deux métriques présentées dans le chapitre 1 (section 1.2.5).

3.4.1 Ensembles de données

- Le premier ensemble de données est la base de données des films MovieLens³ du groupe GroupLens Research Center pour évaluer les performances du système de recommandation que nous avons proposé. La base de données MovieLens est une ressource largement utilisée par la communauté scientifique pour tester les algorithmes de recommandation. Elle est téléchargeable depuis le site MovieLens et présente l'avantage d'être construite à partir de notes réelles, ce qui en fait un bon support de validation. La base de données MovieLens se compose de 100 000 notes attribuées par 943 utilisateurs à 1 682 films, sur une échelle de 1 à 5 (R). Chaque utilisateur a noté au moins 20 films, ce qui entraîne un taux de 93,70% de notes manquantes dans la matrice (Falih, 2018). Pour notre étude, nous avons divisé cette base de données en une base d'apprentissage contenant 80% des données et une base de test contenant 20% des données.

	userId	movieId	rating	timestamp
0	1	1	4.0	964982703
1	1	3	4.0	964981247
2	1	6	4.0	964982224
3	1	47	5.0	964983815
4	1	50	5.0	964982931

Figure 3. 9: Dataset MovieLens.

³ <https://grouplens.org/datasets/movielens/latest/>

- Le deuxième ensemble de données UCI ML Drug Review⁴ fournit des avis de patients sur des médicaments spécifiques ainsi que des conditions connexes et un système d'évaluation des patients 10 étoiles reflétant la satisfaction globale des patients. Les données ont été obtenues en explorant des sites de revues pharmaceutiques en ligne. Ces données ont été publiées dans une étude sur l'analyse des sentiments de l'expérience de la drogue sur plusieurs facettes. Le dataset UCI ML Drug Review se compose de 200 000 interactions, 885 maladies et 3436 médicaments.

	uniqueID	drugName	condition	review	rating	date	usefulCount
0	206461	Valsartan	Left Ventricular Dysfunction	It has no side effect, I take it in combinatio...	9	20-May-12	27
1	95260	Guanfacine	ADHD	My son is halfway through his fourth week of l...	8	27-Apr-10	192
2	92703	Lybrel	Birth Control	I used to take another oral contraceptive, whi...	5	14-Dec-09	17
3	138000	Ortho Evra	Birth Control	This is my first time using any form of birth ...	8	3-Nov-15	10
4	35696	Buprenorphine / naloxone	Opiate Dependence	Suboxone has completely turned my life around....	9	27-Nov-16	37

Figure 3. 10: Dataset UCI ML Drug Review.

3.4.2 Détails d'implémentation

Dans cette section, nous décrirons le processus de développement de notre modèle et le traitement des données.

3.4.2.1 Préparation des données

Nous importons d'abord les données depuis le stockage cloud de Google Drive.

```

from google.colab import drive
drive.mount('/content/drive')
df = pd.read_csv('/content/drive/MyDrive/data/Drug.csv')

```

Figure 3. 11: Importation des données.

Après avoir chargé les données, dans le cadre du processus de nettoyage des données, nous procédons à la suppression des colonnes qui ne sont pas utilisées.

```

df=pd.read_csv('/content/drive/MyDrive/data/Drug.csv')
df = df.drop('Date', axis=1)
df.to_csv('/content/drive/MyDrive/data/Drug.csv', index=False)

```

Figure 3. 12: Suppression des données inutiles.

Nous avons créé deux fonctions pour charger les nœuds des maladies et des médicaments, ainsi que les arêtes (liens) entre les maladies et les médicaments.

⁴ <https://www.kaggle.com/datasets/jessicali9530/kuc-hackathon-winter-2018>

La fonction `load_node_csv` est utilisée pour charger les informations sur les nœuds (maladies et médicaments) à partir d'un fichier CSV. Elle prend en argument le chemin vers le fichier CSV et le nom de la colonne d'index. Une fois le fichier CSV chargé à l'aide de la bibliothèque pandas, la fonction crée un mapping (une correspondance) entre les indices des lignes dans le DataFrame et les identifiants des nœuds (utilisateurs ou films). Ce mapping est stocké dans un dictionnaire où chaque clé représente l'index de la ligne dans le DataFrame et la valeur correspondante est l'identifiant du nœud correspondant.

```
def load_node_csv(path, index_col):
    df = pd.read_csv(path, index_col=index_col)
    mapping = {index: i for i, index in enumerate(df.index.unique())}
    return mapping
```

Figure 3. 13: Fonction du chargement des nœuds.

La fonction `load_edge_csv` est utilisée pour charger les arêtes (liens) entre les utilisateurs et les films à partir d'un fichier CSV. Elle prend en argument le chemin vers le fichier CSV, les noms de colonnes pour les maladies (`src_index_col`), les médicaments (`dst_index_col`), et l'interaction entre les maladies et les médicaments (`link_index_col`). Elle nécessite également les mappings (`src_mapping` et `dst_mapping`) créés précédemment pour faire correspondre les indices des lignes dans le DataFrame aux identifiants des nœuds (maladies et médicaments). La fonction lit le fichier CSV en utilisant pandas et extrait les colonnes correspondantes aux utilisateurs, aux films et à l'interaction. Un seuil de notation (`rating_threshold`) est utilisé pour déterminer la positivité de l'arête. Par défaut, une notation supérieure ou égale à 4 est considérée comme positive. Ensuite, la fonction crée les arêtes en utilisant les mappings pour obtenir les identifiants de nœuds correspondants aux maladies et aux médicaments. Les arêtes sont représentées par une matrice 2xN de tenseurs torch, où N est le nombre d'arêtes. Chaque arête est représentée par une paire de nœuds (maladie, médicament).

L'ensemble des arêtes est retourné sous forme d'un tenseur torch, stocké dans la variable `edge_index`. Ces arêtes sont essentielles pour représenter les interactions entre les maladies et les médicaments, ce qui permet de construire des graphes appropriés pour les algorithmes de recommandation.

```
def load_edge_csv(path, src_index_col, src_mapping, dst_index_col, dst_mapping, link_index_col, rating_threshold=3.5):
    df = pd.read_csv(path)
    edge_index = None
    src = [src_mapping[index] for index in df[src_index_col]]
    dst = [dst_mapping[index] for index in df[dst_index_col]]
    edge_attr = torch.from_numpy(df[link_index_col].values).view(-1, 1).to(torch.long) >= rating_threshold
    edge_index = [[], []]
    for i in range(edge_attr.shape[0]):
        if edge_attr[i]:
            edge_index[0].append(src[i])
            edge_index[1].append(dst[i])
    return torch.tensor(edge_index)
```

Figure 3. 14: Fonction du chargement des arrêtes.

3.4.2.2 Répartition entraînement-test

Nous divisons les données chargées en ensembles d'apprentissage et de test, Le code fourni est utilisé pour diviser les arêtes du graphe en ensembles d'entraînement, de validation et de test, en utilisant une répartition de 80/10/10.

Tout d'abord, les dimensions du graphe sont déterminées : le nombre des maladies (`num_ds`), le nombre de médicaments (`num_ms`) et le nombre total d'interactions (`num_interactions`) représentées par les arêtes. Ensuite, une liste `all_indices` est créée pour stocker les indices de toutes les interactions.

```
num_ds, num_ms = len(disease_mapping), len(medicine_mapping)
num_interactions = edge_index.shape[1]
all_indices = [i for i in range(num_interactions)]

train_indices, test_indices = train_test_split(
    all_indices, test_size=0.2, random_state=1)
val_indices, test_indices = train_test_split(
    test_indices, test_size=0.5, random_state=1)

train_edge_index = edge_index[:, train_indices]
val_edge_index = edge_index[:, val_indices]
test_edge_index = edge_index[:, test_indices]
```

Figure 3. 15: Code pour la répartition entraînement-test.

La première division est effectuée entre l'ensemble d'apprentissage (`train_indices`) et l'ensemble de test (`test_indices`) en utilisant la fonction `train_test_split` de la bibliothèque `scikit-learn`. Ici, les `all_indices` sont divisés en utilisant une taille de test de 20% (`test_size=0.2`) et une valeur de graine aléatoire de 1 (`random_state=1`). Ainsi, 80% des indices sont attribués à l'ensemble d'apprentissage et 20% aux données de test. Ensuite, la division est répétée entre l'ensemble de test (`test_indices`) et l'ensemble de validation (`val_indices`) de manière similaire, en utilisant une taille de test de 50% (`test_size=0.5`) des indices restants de l'ensemble de test. Cela permet d'obtenir un ensemble de validation et un ensemble de test, chacun représentant 10% des indices totaux. Finalement, les arêtes correspondantes sont extraites pour chaque ensemble en utilisant les indices appropriés : `train_edge_index`, `val_edge_index` et `test_edge_index`.

Cette division permet d'avoir des ensembles de données indépendants pour l'entraînement, la validation et le test, ce qui est essentiel pour évaluer la performance du modèle de recommandation.

3.4.2.3 Construction du modèle LightGCN

Le modèle est défini en tant que classe `LightGCN` qui hérite de la classe `MessagePassing` de `PyTorch Geometric`. Il prend en compte le nombre d'utilisateurs (`num_users`), le nombre d'articles (`num_items`), la dimensionnalité des embeddings (`embedding_dim`), le nombre de couches de propagation de messages (`K`), et l'option d'ajouter des boucles d'auto-rétroaction (`add_self_loops`).

Dans la méthode `__init__`, les embeddings initiaux des utilisateurs et des articles sont créés en utilisant des couches d'embedding de `PyTorch`. Ces embeddings sont initialisés avec des valeurs aléatoires à l'aide de la méthode `nn.init.normal_()`.

La méthode `forward` réalise la propagation avant du modèle. Elle prend en entrée une matrice d'adjacence au format `SparseTensor` représentant les relations entre les utilisateurs et les articles. Le modèle calcule une matrice d'adjacence normalisée symétriquement (\tilde{A}) à partir de cette matrice d'adjacence, puis effectue la propagation de messages sur `K` couches en utilisant la méthode `propagate`. Les embeddings à chaque couche de propagation sont stockés dans une liste `embs`. Ensuite, les embeddings finaux sont calculés en prenant la moyenne des embeddings de toutes les couches. La méthode `message` renvoie simplement les embeddings des articles adjacents dans le graphe. La méthode `message_and_aggregate` effectue la multiplication de la matrice d'adjacence normalisée (\tilde{A}) par les embeddings des articles, afin de calculer le message agrégé à envoyer aux utilisateurs.

Le modèle est instancié avec `model = LightGCN(num_ds, num_ms)`, où `num_ds` est le nombre de maladies et `num_ms` est le nombre des médicaments dans le jeu de données.

```
def message(self, x_j: Tensor) -> Tensor:
    return x_j

def message_and_aggregate(self, adj_t: SparseTensor, x: Tensor) -> Tensor:
    # computes  $\tilde{A} @ x$ 
    return matmul(adj_t, x)
```

Figure 3. 16: Fonction de message et d'agrégation.

Le modèle LightGCN est une architecture de recommandation qui utilise une propagation de messages sur un graphe bipartite pour apprendre les embeddings des utilisateurs et des articles. Il permet de capturer les relations entre les utilisateurs et les articles pour générer des recommandations précises.

3.4.3 Métriques d'évaluation

Les deux métriques présentées dans le chapitre 1 (section 1.2.5).

Nous avons créé une fonction qui calcule le rappel (recall) et la précision (precision) à un certain niveau K. La fonction RecallPrecision_ATk prend trois paramètres : groundTruth (vérité terrain), r (résultats de recommandation) et k (niveau K).

La fonction calcule d'abord le nombre de prédictions correctes pour chaque utilisateur en effectuant une somme sur la dimension -1 (les articles recommandés). Elle calcule ensuite le nombre d'articles aimés par chaque utilisateur dans l'ensemble de test.

3.4.3.1 Précision

La précision est obtenue en prenant la moyenne des prédictions correctes, puis en divisant par K.

```
precision = torch.mean(num_correct_pred) / k
```

Figure 3. 17: Fonction de précision.

3.4.3.2 Rappel (Recall)

Le rappel est obtenu en prenant la moyenne des prédictions correctes divisées par le nombre d'articles aimés par chaque utilisateur.

```
recall = torch.mean(num_correct_pred / user_num_liked)
```

Figure 3. 18: Fonction de rappel.

La fonction renvoie le rappel et la précision au niveau K sous forme de tuple. Les tenseurs utilisés ici sont issus de la bibliothèque PyTorch, et la méthode .item() est utilisée pour obtenir la valeur numérique réelle du tenseur.

3.4.4 Les lignes de base

3.4.4.1 BPR

BPR (Bayesian Personnalisé Ranking) : Le BPR est une approche de classement par paires. Il échantillonne les commentaires négatifs pour chaque commentaire positif, c'est-à-dire que pour chaque interaction de l'utilisateur, il échantillonne un élément qui n'a aucune interaction avec cet élément utilisateur.

3.4.4.2 WARP

La perte WARP (Weighted Approximate-Rank Pairwise loss) est également une approche par paires et est similaire au BPR. Il considère la variation des préférences prédites entre les items positifs et les échantillons éléments négatifs pour un utilisateur.

3.5 Résultats et discussion

Dans cette section, nous allons présenter les résultats obtenus à partir de l'évaluation de notre modèle, mettant en évidence les performances et les améliorations significatives qu'il a démontrées lors de l'analyse des ensembles de données.

3.5.1 Résultats

Nous commençons par les graphes d'erreur de notre modèle créé pour les deux ensembles de données qui sont illustré dans les figures ci-dessous :

- **Résultat du premier ensemble de données**

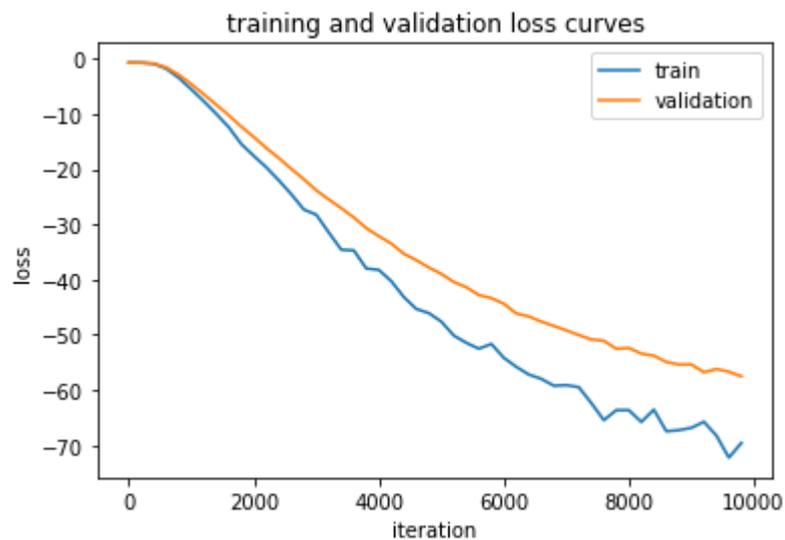


Figure 3. 19: Le graphe d'erreur pour l'ensemble des données MovieLens.

D'après la figure 3.19, nous constatons que l'erreur de notre modèle lors de l'entraînement et de test diminue à chaque itération, Cela signifie que les prédictions du modèle deviennent de plus en plus précises et se rapprochent des valeurs réelles.

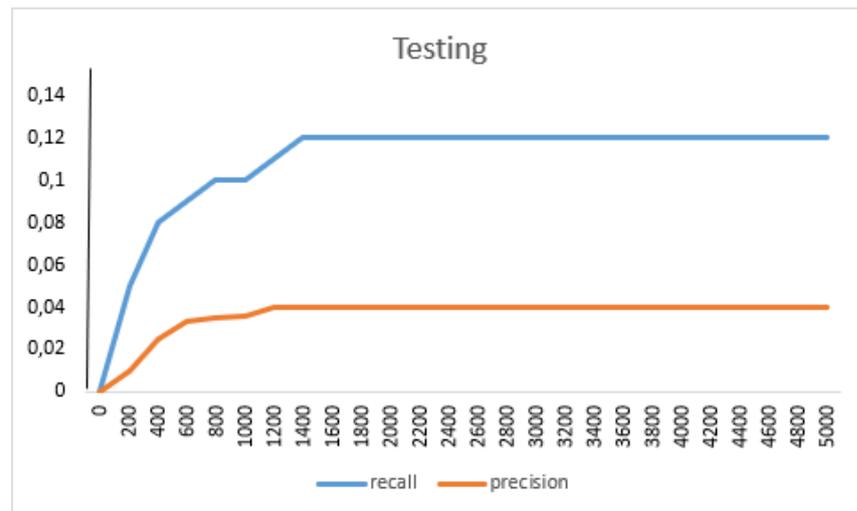


Figure 3. 20 : Le graphe de précision et de rappel pour l'ensemble des données MovieLens.

D'après la figure 3.20, nous constatons que la précision et le rappel de notre modèle augmente à chaque itération, Cela signifie que les prédictions du modèle deviennent de plus en plus précises.

- **Résultat du deuxième ensemble de données**

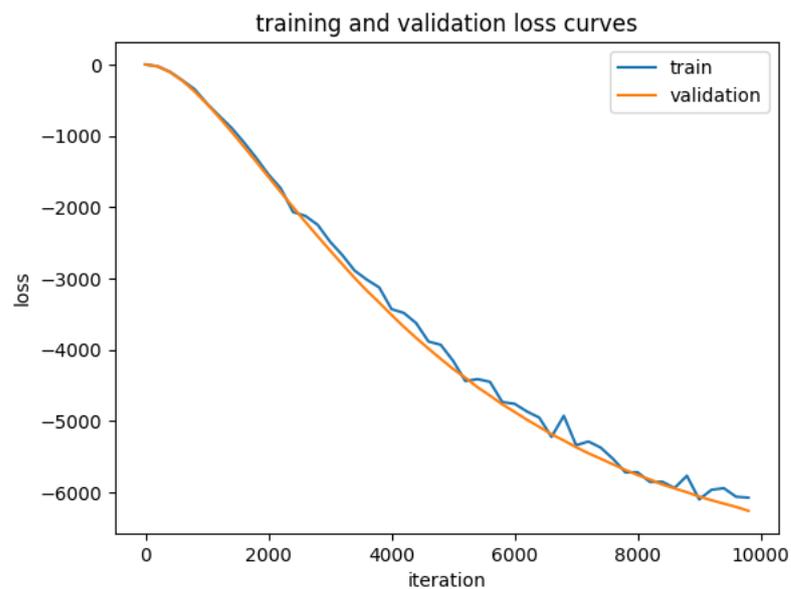


Figure 3. 21: Le graphe d'erreur pour l'ensemble des données UCI ML Drug Review.

D'après la figure 3.21, nous constatons que l'erreur de notre modèle lors de l'entraînement et de test diminue à chaque itération ce qui suggère que le modèle est en train de s'améliorer.

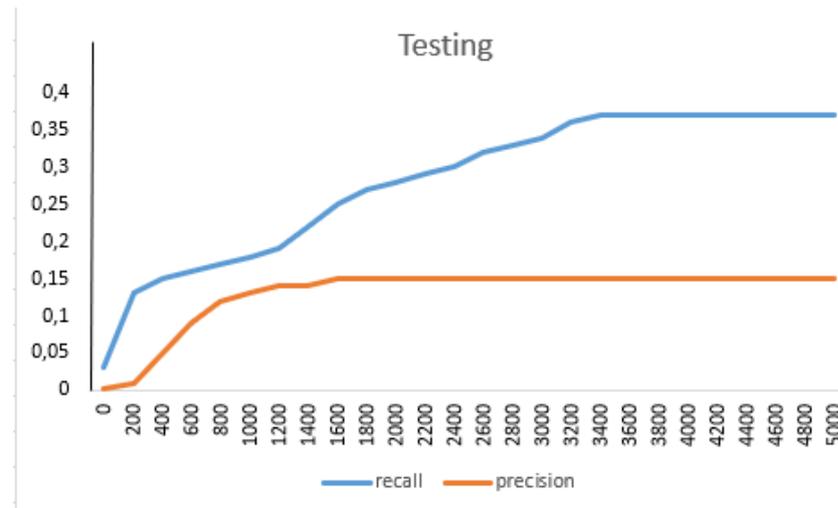


Figure 3.22 : Le graphe de précision et de rappel pour l'ensemble des données UCI ML Drug Review.

D'après la figure 3.22, il est clair que la précision et le rappel du modèle augmentent à chaque itération, ce qui suggère une amélioration progressive de la performance du modèle.

3.5.2 Comparaison

Pour évaluer les performances de notre modèle, nous utilisons des métriques et comparons les résultats entre les baselines et notre modèle LightGCN.

Les métriques et les références utilisées pour les jeux de données MovieLens et UCI ML Drug Review sont les suivantes :

Table 3.2: Comparaison des résultats obtenus avec des méthodes de références.

	MoviLens		Drug		DrugDataset10		DrugDataset20	
	Recall@10	Précision@10	Recall@10	Précision@10	Recall@10	Précision@10	Recall@10	Précision@10
LightGCN	0.12543	0.04629	0.37968	0.15769	0.66777	0.16	0.50117	0.15
Bpr	0.12393	0.03699	0.34203	0.15634	0.66908	0.16875	0.44694	0.18148
Warp	0.11233	0.08013	0.37004	0.15316	0.64325	0.15862	0.49406	0.12955

En examinant les métriques, notre modèle LightGCN affiche les prédictions les plus précises. Ce résultat s'applique aux deux ensembles de données.

3.5.3 Discussion

La rareté des ensembles de données de santé a été un défi majeur tout au long de notre recherche, les données de santé sont souvent protégées en raison de leur nature sensible, ce qui limite leur disponibilité pour les chercheurs. De plus, les ensembles de données de santé de grande taille et de haute qualité sont rares, car ils nécessitent une collecte minutieuse, une anonymisation appropriée et une documentation précise. Après une longue et ardue recherche nous avons trouvé un ensemble de données contenant 200 000 lignes, mais nous avons constaté que les lignes étaient répétées, ce qui n'était pas utile pour notre cas spécifique. Nous avons donc décidé de nous concentrer sur un ensemble de données qui présente de nombreuses interactions entre les maladies et les médicaments, plutôt que

des répétitions. Dans ce contexte, nous avons sélectionné les 10 médicaments (les 20 médicaments) les plus interagissant avec différentes maladies, ce qui a conduit à de bons résultats. Cette approche ciblée nous a permis de réduire la complexité de l'ensemble de données tout en conservant les interactions pertinentes pour notre travail avec LightGCN. Cette réussite est encourageante car nous avons pu identifier les médicaments les plus pertinents en termes d'interactions avec différentes maladies. Parfois, il est nécessaire de faire des choix et de se concentrer sur des éléments spécifiques pour obtenir des résultats significatifs. En sélectionnant ces 10 médicaments clés, nous avons pu réduire la complexité de notre ensemble de données tout en préservant les interactions pertinentes pour notre travail avec LightGCN. Cela met en évidence l'importance de la préparation des données et de la sélection judicieuse des caractéristiques dans les projets de modélisation. Au lieu de travailler avec un ensemble de données volumineux mais répétitif, nous avons adopté une approche ciblée pour extraire les informations les plus pertinentes pour notre tâche spécifique. Cette stratégie s'est avérée efficace pour optimiser les performances de notre modèle et obtenir des résultats significatifs. Il est important de noter que l'exploration de différentes approches et la recherche de datasets plus complets et riches en interactions peuvent également être bénéfiques à long terme. Cependant, dans notre situation actuelle, la sélection des 10 médicaments les plus interagissant semble être une solution adaptée pour obtenir de bons résultats.

Travailler avec LightGCN dans le domaine de la santé nécessite une sélection et une analyse minutieuses du dataset. Il est préférable de faire appel à des experts du domaine ou d'acquérir un dataset spécifique, car les datasets de santé appropriée peuvent être rares et difficiles à trouver. En raison des contraintes de confidentialité et de réglementation entourant les données de santé, il peut être complexe d'accéder à des datasets complets et riches en informations dans ce domaine. De plus, les datasets existants peuvent ne pas contenir suffisamment d'interactions entre les utilisateurs (par exemple, les patients) et les items (par exemple, les médicaments ou les maladies) pour répondre aux exigences de LightGCN. Pour remédier à cette situation, il est recommandé de collaborer avec des experts en santé ou des chercheurs du domaine. Leur expertise peut aider à identifier des sources de données appropriées ou à analyser et sélectionner les interactions les plus pertinentes pour votre modèle. Ils peuvent également apporter des connaissances approfondies sur les relations entre les maladies, les médicaments et d'autres éléments du domaine de la santé. Si la recherche d'un dataset adapté s'avère extrêmement difficile, il peut être envisagé d'acheter un dataset auprès de fournisseurs spécialisés. Il existe des entreprises et des organisations qui collectent et proposent des datasets de santé pour des utilisations spécifiques, moyennant des frais. Cela peut être une option à considérer si cela correspond à vos besoins et à vos ressources.

Dans le cadre de notre recherche sur le développement d'un système de recommandation basé sur le filtrage collaboratif, nous visons à optimiser les performances de LightGCN en utilisant des ensembles de données pertinents. Pour atteindre cet objectif, plusieurs aspects clés doivent être pris en considération. Le filtrage collaboratif repose sur l'idée que les recommandations sont basées sur les comportements passés des utilisateurs, ainsi que sur les similitudes entre les items. Par conséquent, il est crucial que notre ensemble de données contienne des médicaments (items) qui présentent de multiples relations avec les maladies (les utilisateurs). Cette approche nécessite que chaque médicament soit associé à un nombre considérable d'interactions avec d'autres médicaments, étant donné que l'interaction commune entre deux médicaments correspond à la même maladie entre ces deux médicaments.

En outre, il est préférable que chaque médicament présente un grand nombre d'interactions, ce qui se traduit par un nombre élevé de maladies associées. Cette caractéristique renforce les liens entre les médicaments. Par exemple, lorsque deux médicaments partagent un lien avec la même maladie, cela indique une recommandation particulièrement solide entre ces médicaments.

En appliquant ces critères et en considérant ces mesures, nous obtenons un ensemble de données avec une densité significative, ce qui signifie qu'il existe de nombreuses relations entre les médicaments. Cette densité élevée favorise de meilleures performances et permet d'obtenir des

résultats améliorés pour l'algorithme LightGCN. Le modèle est capable de capturer les similitudes subtiles entre les médicaments en se basant sur les interactions passées entre les utilisateurs et les items. Il est donc essentiel de s'assurer que notre ensemble de données est riche en relations entre les médicaments, ce qui nous permettra de fournir des recommandations plus précises et pertinentes pour les utilisateurs du système de recommandation basé sur le filtrage collaboratif.

En conclusion, pour travailler avec LightGCN dans le domaine de la santé, il est crucial de choisir et d'analyser attentivement le dataset. La participation d'experts du domaine et l'achat d'un dataset spécialisé peuvent être des solutions pour surmonter les difficultés liées à la disponibilité limitée de datasets de santé appropriée.

3.6 Conclusion

Nous avons mené une étude approfondie sur l'utilisation de LightGCN dans le contexte d'un système de recommandation de santé. Nos résultats confirment que LightGCN est particulièrement puissant et performant lorsque l'ensemble de données est densément connecté, avec de nombreuses relations entre les items. Son architecture et son approche de filtrage collaboratif exploitent pleinement ces interactions pour générer des recommandations précises et pertinentes. Cependant, il est crucial de noter que la qualité et la densité des interactions dans l'ensemble de données sont essentielles pour garantir une performance optimale de LightGCN. Si le dataset ne répond pas à ces critères, il est recommandé d'explorer d'autres modèles de recommandation adaptés aux données disponibles.

Conclusion Générale

Conclusion Générale

➤ *Conclusion*

Cette étude souligne l'importance croissante des systèmes de recommandation de santé dans divers services de santé. Elle met en évidence l'importance cruciale de la qualité et de la densité des interactions dans l'ensemble de données lors de l'utilisation de LightGCN pour un système de recommandation. Les systèmes de recommandation jouent un rôle essentiel en fournissant un soutien précieux aux patients et aux professionnels de la santé dans leurs décisions relatives au bien-être. L'étude se concentre spécifiquement sur l'efficacité des réseaux convolutifs graphiques (GCN) pour améliorer les recommandations de soins de santé en exploitant la structure des données relationnelles. Une partie du travail présenté dans ce mémoire a été publié dans un proceedings indexé IEEE Xplore lors de la conférence NTIC'2022 (voir annexes 1 et 2 à la fin de ce document).

➤ *Perspectives*

L'intégration des modèles de réseaux convolutifs graphiques, tels que LightGCN, dans les systèmes de recommandation de santé ouvre de nouvelles perspectives passionnantes. Voici quelques perspectives clés pour l'avenir :

1. Amélioration continue de la précision : La recherche et le développement devraient se concentrer sur l'amélioration continue de la précision des systèmes de recommandation de santé. Cela peut inclure l'exploration de techniques d'apprentissage plus avancées, l'incorporation de nouvelles sources de données pertinentes et l'optimisation des modèles existants.
2. Intégration transparente dans les pratiques cliniques : Pour maximiser l'impact des systèmes de recommandation de santé, il est crucial de les intégrer de manière transparente dans les pratiques cliniques. Cela nécessite une collaboration étroite entre les développeurs de systèmes, les professionnels de la santé et les patients pour garantir une utilisation efficace et une adoption généralisée.
3. Les tests réalisés en utilisant des ensembles de données présentant différentes caractéristiques fournissent des informations précieuses pour améliorer les modèles de recommandation de santé. Ces tests permettent d'évaluer les performances, d'identifier les contextes spécifiques auxquels les modèles peuvent être adaptés, de comparer les différentes approches et d'améliorer la capacité de généralisation. En combinant ces résultats avec la collecte de données supplémentaires, nous pourrions développer des modèles de recommandation plus solides et plus adaptés aux défis réels de la santé.
4. L'utilisation de modèles d'apprentissage profond avancés, tels que le modèle d'hypergraphe, présente de grandes opportunités pour améliorer les recommandations de soins de santé en prenant en compte la complexité des relations entre les entités de santé. Cependant, des recherches supplémentaires sont nécessaires pour résoudre les défis liés à l'interprétabilité, à l'adaptation aux données hétérogènes et à l'intégration pratique dans les systèmes de santé existants. En investissant dans ces domaines, nous pourrions exploiter pleinement le potentiel des modèles d'apprentissage profond avancés et améliorer la qualité des soins de santé.

Bibliographie

Bibliographie

- Adomavicius, G., & Tuzhilin, A. (2005). Toward the next generation of recommender systems : A survey of the state-of-the-art and possible extensions. *IEEE Transactions on Knowledge and Data Engineering*, 17(6), 734-749. <https://doi.org/10.1109/TKDE.2005.99>
- Belaouedj, M. (2021). *Les Problèmes De Cheminement Dans Un Réseau*. <https://theses-algerie.com/2387092578489336/memoire-de-master/universite-abdelhamid-ibn-badis-mostaganem/les-problemes-de-cheminement-dans-un-reseau>
- Belloui, A. (2008). *L usage des concepts du web sémantique dans le filtrage d information collaboratif—PDF Free Download*. <https://docplayer.fr/1209796-L-usage-des-concepts-du-web-semantique-dans-le-filtrage-d-information-collaboratif.html>
- Bhagavatula, C., Feldman, S., Power, R., & Ammar, W. (2018). Content-Based Citation Recommendation. *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)*, 238-251. <https://doi.org/10.18653/v1/N18-1022>
- Bouzy, B. (2010). *Apprentissage par renforcement (3)article de revue*.
- Breese, J. S., & Kadie, D. H. C. (1999). *Empirical Analysis of Predictive Algorithms for Collaborative Filtering*.
- Breslin, J. M., MacRae, S. K., Bell, J., Singer, P. A., & the University of Toronto Joint Centre for Bioethics Clinical Ethics Group. (2005). Top 10 health care ethics challenges facing the public : Views of Toronto bioethicists. *BMC Medical Ethics*, 6(1), 5. <https://doi.org/10.1186/1472-6939-6-5>
- Bruna, J., Zaremba, W., Szlam, A., & LeCun, Y. (2014). *Spectral Networks and Locally Connected Networks on Graphs* (arXiv:1312.6203). arXiv. <http://arxiv.org/abs/1312.6203>
- 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>
- Calero Valdez, A., Ziefle, M., Verbert, K., Felfernig, A., & Holzinger, A. (2016). Recommender Systems for Health Informatics : State-of-the-Art and Future Perspectives. In A. Holzinger (Éd.), *Machine Learning for Health Informatics* (Vol. 9605, p. 391-414). Springer International Publishing. https://doi.org/10.1007/978-3-319-50478-0_20
- Canu, M. (2017). *Détection de communautés orientée sommet pour des réseaux mobiles opportunistes sociaux* [Phdthesis, Université Pierre et Marie Curie - Paris VI]. <https://theses.hal.science/tel-01745380>
- CHEN, G., YU, Z., NIAN, L., YINFENG, L., YINGRONG, Q., JINGHUA, P., YUHAN, Q., JIANXIN, C., DEPENG, J., XIANGNAN, H., & YONG, L. (2023). [2109.12843] *A Survey of Graph Neural Networks for Recommender Systems : Challenges, Methods, and Directions*. <https://arxiv.org/abs/2109.12843>
- Chockz21. (2021, juillet 7). AI VS ML VS DL-Let's Understand The Difference. *Analytics Vidhya*. <https://www.analyticsvidhya.com/blog/2021/07/ai-vs-ml-vs-dl-lets-understand-the-difference/>
- Cosley, D., Lawrence, S., & Pennock, D. (2002). *REFeree : An open framework for practical testing of recommender systems using ResearchIndex*.

Bibliographie

- Deluzarche, C. (2021). *Définition | Deep Learning—Apprentissage profond | Futura Tech*. Futura. <https://www.futura-sciences.com/tech/definitions/intelligence-artificielle-deep-learning-17262/>
- Dixon-Woods, M., McNicol, S., & Martin, G. (2012). Ten challenges in improving quality in healthcare : Lessons from the Health Foundation's programme evaluations and relevant literature. *BMJ Quality & Safety*, 21(10), 876-884. <https://doi.org/10.1136/bmjqs-2011-000760>
- Etemadi, M., Bazzaz Abkenar, S., Ahmadzadeh, A., Haghi Kashani, M., Asghari, P., Akbari, M., & Mahdipour, E. (2023). A systematic review of healthcare recommender systems : Open issues, challenges, and techniques. *Expert Systems with Applications*, 213, 118823. <https://doi.org/10.1016/j.eswa.2022.118823>
- Falih, I. (2018). *Attributed Network Clustering : Application to recommender systems* [These de doctorat, Sorbonne Paris Cité]. <https://www.theses.fr/2018USPCD011>
- Ferretto, L. R., Bellei, E. A., Biduski, D., Bin, L. C. P., Moro, M. M., Cervi, C. R., & De Marchi, A. C. B. (2020). A Physical Activity Recommender System for Patients With Arterial Hypertension. *IEEE Access*, 8, 61656-61664. <https://doi.org/10.1109/ACCESS.2020.2983564>
- Gao, C., Zheng, Y., Li, N., Li, Y., Qin, Y., Piao, J., Quan, Y., Chang, J., Jin, D., He, X., & Li, Y. (2021). *Graph Neural Networks for Recommender Systems : Challenges, Methods, and Directions*.
- Garnine, N. (2020). *Système de recommandation basé sur la détection de communautés*.
- Goldberg, D., Nichols, D., Oki, B. M., & Terry, D. (1992). Using collaborative filtering to weave an information tapestry. *Communications of the ACM*, 35(12), 61-70. <https://doi.org/10.1145/138859.138867>
- Goodfellow, I., Bengio, Y., & Courville, A. (2016). *Deep Learning*. The MIT Press.
- Google Colab. (s. d.). Consulté 25 juin 2023, à l'adresse <https://research.google.com/colaboratory/faq.html?hl=fr>
- Graph Neural Network (GNN). (2023). SEON. <https://seon.io/resources/dictionary/graph-neural-network-gnn/>
- Graph Neural Network : La prochaine étape du Deep learning. (2020, mars 16). IONOS Digital Guide. <https://www.ionos.fr/digitalguide/web-marketing/search-engine-marketing/graph-neural-network/>
- Graph Neural Networks (GNN) : Qu'est-ce que c'est ? (2023). <https://datascientest.com/graph-neural-networks-tout-savoir>
- Graphcore intègre Pytorch Geometric à sa pile logicielle. (s. d.). ActuaIA. Consulté 25 juin 2023, à l'adresse <https://www.actuaia.com/actualite/graphcore-integre-pytorch-geometric-a-sa-pile-logicielle/>
- Haenlein, M., & Kaplan, A. (2019). A Brief History of Artificial Intelligence : On the Past, Present, and Future of Artificial Intelligence. *California Management Review*, 61, 000812561986492. <https://doi.org/10.1177/0008125619864925>
- He, X., Deng, K., Wang, X., Li, Y., Zhang, Y., & Wang, M. (2020). *LightGCN : Simplifying and Powering Graph Convolution Network for Recommendation* (arXiv:2002.02126). arXiv. <https://doi.org/10.48550/arXiv.2002.02126>

Bibliographie

Himgauri, A., & Apashabi, P. (2015). Recommender System Challenges and Methodologies in Social Network : Survey. *International Journal of Science and Research (IJSR)*, 4(11), 286-289. <https://doi.org/10.21275/v4i11.NOV151061>

Introduction à NumPy—Cours Python. (s. d.). Consulté 25 juin 2023, à l'adresse <https://courspython.com/apprendre-numpy.html>

Introduction to Recommendation engine—Dataaspirant. (2015, janvier 24). <https://dataaspirant.com/recommendation-engine-part-1/>

Kipf, T. N., & Welling, M. (2017). *Semi-Supervised Classification with Graph Convolutional Networks* (arXiv:1609.02907). arXiv. <https://doi.org/10.48550/arXiv.1609.02907>

Li, W., Ni, L., Wang, J., & Wang, C. (2022). Collaborative representation learning for nodes and relations via heterogeneous graph neural network. *Knowledge-Based Systems*, 255, 109673. <https://doi.org/10.1016/j.knsys.2022.109673>

Linden, G., Smith, B., & York, J. (2003). Amazon.com recommendations : Item-to-item collaborative filtering. *IEEE Internet Computing*, 7(1), 76-80. <https://doi.org/10.1109/MIC.2003.1167344>

MAATALLAH, M. (2016). *Une Technique Hybride pour les Systèmes de Recommandation.*

Mandl, K. D., Szolovits, P., & Kohane, I. S. (2001). Public standards and patients' control : How to keep electronic medical records accessible but private. *BMJ : British Medical Journal*, 322(7281), 283-287.

Matplotlib : Tout savoir sur la bibliothèque Python de Dataviz. (s. d.). Consulté 25 juin 2023, à l'adresse <https://datascientest.com/matplotlib-tout-savoir>

McCarthy, J. (2007). *WHAT IS ARTIFICIAL INTELLIGENCE?*

Mcculloch, W. S., & Pitts, W. (1990). *A LOGICAL CALCULUS OF THE IDEAS IMMANENT IN NERVOUS ACTIVITY.*

Meingast, M., Roosta, T., & Sastry, S. (2006). Security and privacy issues with health care information technology. *Conference Proceedings: ... Annual International Conference of the IEEE Engineering in Medicine and Biology Society. IEEE Engineering in Medicine and Biology Society. Annual Conference, 2006*, 5453-5458. <https://doi.org/10.1109/IEMBS.2006.260060>

Menzli, A. (2023). *Graph Neural Network and Some of GNN Applications : Everything You Need to Know.* <https://neptune.ai/blog/graph-neural-network-and-some-of-gnn-applications>

Nedjar, I. (2020). *Medical images indexation and annotation* [Thesis, 11-04-2021]. <http://dspace.univ-tlemcen.dz/handle/112/16332>

Nichols, J. A., Herbert Chan, H. W., & Baker, M. A. B. (2018). Machine learning : Applications of artificial intelligence to imaging and diagnosis. *Biophysical Reviews*, 11(1), 111-118. <https://doi.org/10.1007/s12551-018-0449-9>

Nitin, K. C., & Krishna, S. (2018). (PDF) *A Review on Conventional Machine Learning vs Deep Learning.* https://www.researchgate.net/publication/332429270_A_Review_on_Conventional_Machine_Learning_vs_Deep_Learning

O'Connor, M., & Herlocker, J. (1999). *Clustering Items for Collaborative Filtering.*

Bibliographie

- Olaronke, I., Soriyan, A., Gambo, I., & Olaleke, J. (2013). Interoperability in Healthcare : Benefits, Challenges and Resolutions. *International Journal of Innovation and Applied Studies*, 3, 2028-9324.
- Oufaida, H., & Nouali, O. (2008). Le filtrage collaboratif et le web 2.0. Etat de l'art. *Document numérique*, 11(1-2), 13-35.
- Papagelis, M., Plexousakis, D., & Kutsuras, T. (2005). *Alleviating the Sparsity Problem of Collaborative Filtering Using Trust Inferences*. 3477, 224-239. https://doi.org/10.1007/11429760_16
- Que sont les réseaux de neurones graphiques et comment fonctionnent-ils ? - Intelligence des données Platon*. (2022, mars 1). <https://zephyrnet.com/fr/que-sont-les-r%C3%A9seaux-de-neurones-graphiques-et-comment-fonctionnent-ils/>
- Rao, K., & Talwar, V. (2008). Application Domain and Functional Classification of Recommender Systems—A Survey. *DESIDOC Journal of Library & Information Technology*, 28. <https://doi.org/10.14429/djlit.28.3.174>
- Rashid, A. M., Albert, I., Cosley, D., Lam, S. K., McNee, S. M., Konstan, J. A., & Riedl, J. (2002). *Getting to know you : 2002 International Conference on intelligent User Interfaces (IUI 02)*. 127-134. <http://www.scopus.com/inward/record.url?scp=0036384150&partnerID=8YFLogxK>
- Ray, S. (2017, septembre 8). Top 10 Machine Learning Algorithms (with Python and R Codes). *Analytics Vidhya*. <https://www.analyticsvidhya.com/blog/2017/09/common-machine-learning-algorithms/>
- rédac, T. (2022, janvier 10). Pandas : La bibliothèque Python dédiée à la Data Science. *Formation Data Science | DataScientest.com*. <https://datascientest.com/pandas-python-data-science>
- Resnick, P., & Varian, H. R. (1997). Recommender systems. *Communications of the ACM*, 40(3), 56-58. <https://doi.org/10.1145/245108.245121>
- Rishit, D. (2022). *What Are Graph Neural Networks ? How GNNs Work, Explained with Examples*. <https://www.freecodecamp.org/news/graph-neural-networks-explained-with-examples/>
- Robins, n. (2020, mai 21). *The Difference Between Artificial Intelligence, Machine Learning and Deep Learning*. <https://community.intel.com/t5/Blogs/Tech-Innovation/Artificial-Intelligence-AI/The-Difference-Between-Artificial-Intelligence-Machine-Learning/post/1335666>
- Rosenblatt, F. (1958). The perceptron : A probabilistic model for information storage and organization in the brain. *Psychological Review*, 65(6), 386-408. <https://doi.org/10.1037/h0042519>
- Sahoo, A. K., Pradhan, C., Barik, R. K., & Dubey, H. (2019). DeepReco : Deep Learning Based Health Recommender System Using Collaborative Filtering. *Computation*, 7(2), 25. <https://doi.org/10.3390/computation7020025>
- Sarwar, B., Karypis, G., Konstan, J., & Riedl, J. (2001). Item-based collaborative filtering recommendation algorithms. *Proceedings of the 10th International Conference on World Wide Web*, 285-295. <https://doi.org/10.1145/371920.372071>
- Scikit-learn : Machine learning in Python—Scikit-learn 1.2.2 documentation*. (s. d.). Consulté 25 juin 2023, à l'adresse <https://scikit-learn.org/stable/>

Bibliographie

- Seo, C., Jeong, K.-J., Lim, S., & Shin, W.-Y. (2022). *SiReN : Sign-Aware Recommendation Using Graph Neural Networks* (arXiv:2108.08735). arXiv. <https://doi.org/10.48550/arXiv.2108.08735>
- Setiowati, S., Zulfanahri, Franita, E. L., & Ardiyanto, I. (2017). A review of optimization method in face recognition : Comparison deep learning and non-deep learning methods. *2017 9th International Conference on Information Technology and Electrical Engineering (ICITEE)*, 1-6. <https://doi.org/10.1109/ICITEED.2017.8250484>
- Soni, D. (2020, juillet 21). *Supervised vs. Unsupervised Learning*. Medium. <https://towardsdatascience.com/supervised-vs-unsupervised-learning-14f68e32ea8d>
- Stöger, K., Schneeberger, D., & Holzinger, A. (2021). Medical artificial intelligence : The European legal perspective. *Communications of the ACM*, 64, 34-36. <https://doi.org/10.1145/3458652>
- Stoianov, I. (2001). *Connectionist lexical processing*.
- Tran, D., Sheng, Q., Zhang, W. E., Aljubairy, A., Zaib, M., Hamad, S., Tran, N., & Nguyen, K. (2021). HeteGraph : Graph learning in recommender systems via graph convolutional networks. *Neural Computing and Applications*, 35. <https://doi.org/10.1007/s00521-020-05667-z>
- Tran, T. N. T., Felfernig, A., Trattner, C., & Holzinger, A. (2021). Recommender systems in the healthcare domain : State-of-the-art and research issues. *Journal of Intelligent Information Systems*, 57(1), 171-201. <https://doi.org/10.1007/s10844-020-00633-6>
- Trang Tran, T. N., Atas, M., Felfernig, A., & Stettinger, M. (2018). An overview of recommender systems in the healthy food domain. *Journal of Intelligent Information Systems*, 50(3), 501-526. <https://doi.org/10.1007/s10844-017-0469-0>
- Tucci, L. (2018). *What is artificial neuron? | Definition from TechTarget*. CIO. <https://www.techtarget.com/searchcio/definition/artificial-neuron>
- Wang, X., He, X., Wang, M., Feng, F., & Chua, T.-S. (2019). Neural Graph Collaborative Filtering. *Proceedings of the 42nd International ACM SIGIR Conference on Research and Development in Information Retrieval*, 165-174. <https://doi.org/10.1145/3331184.3331267>
- WelcomeAIOverlords (Réalisateur). (2020). *Graph Convolutional Networks (GCNs) made simple*. <https://www.youtube.com/watch?v=2KRAOZIULzw>
- What is Deep Learning? | IBM*. (2020). <https://www.ibm.com/topics/deep-learning>
- Wu, S., Sun, F., Zhang, W., Xie, X., & Cui, B. (2022). *Graph Neural Networks in Recommender Systems : A Survey* (arXiv:2011.02260). arXiv. <https://doi.org/10.48550/arXiv.2011.02260>
- Zhang, H., Lu, G., Zhan, M., & Zhang, B. (2022). Semi-Supervised Classification of Graph Convolutional Networks with Laplacian Rank Constraints. *Neural Processing Letters*, 54(4), 2645-2656. <https://doi.org/10.1007/s11063-020-10404-7>
- Zhang, S., Tong, H., Xu, J., & Maciejewski, R. (2019). Graph convolutional networks : A comprehensive review. *Computational Social Networks*, 6(1), 11. <https://doi.org/10.1186/s40649-019-0069-y>
- ZIANI, A. (2018). *La recommandation via l'analyse d'opinions*.

Annexes

Annexes

Annexe 1 : Attestation de participation à la conférence NTIC 2022



*International Conference on
New Technologies of Information and
Communication*



Certificate of Participation

AWARDED TO

*Boudjemaa Boudaa, Imen Bestani and
Noureddine Benadjrouda*

*In recognition of outstanding as communicants during the Second
IEEE International Conference on New Technologies of
Information and Communication, at A. Boussouf University
Center of Mila on December 21-22, 2022.*

GENERAL CHAIR
DR. NARDJES BOUCHEMAL

THE DIRECTOR
مدير معهد الطور و التكنولوجيا
الأستاذ، كمال سامي

Annexe 2 : Graph Convolutional Networks for Designing Collaborative Filtering-Based Health Recommender Systems" notre article publié lors d'une conférence de l'IEEE

Graph Convolutional Networks for Designing Collaborative Filtering-B...

https://ieeexplore.ieee.org/document/10100590

IEEE.org IEEE Xplore IEEE SA IEEE Spectrum More Sites [Subscribe](#) [Subscribe](#) [Cart](#) [Create Account](#) [Pers Sign](#)

 [Browse](#) [My Settings](#) [Help](#) [Institutional Sign In](#)

[Institutional Sign In](#)

All  [ADVANCED SEARCH](#)

Conferences > 2022 2nd International Confer... 

Graph Convolutional Networks for Designing Collaborative Filtering-Based Health Recommender Systems

Publisher: IEEE [Cite This](#) [PDF](#)

Boudjemaa Boudaa ; Imen Bestani ; Noureddine Benadjrouda [All Authors](#) 

1
Full Text View

Abstract

Document Sections

- I. Introduction
- II. Preliminaries
- III. GCN Design for CF-Based HRS
- IV. Implementation Opportunities
- V. Related Work

Show Full Outline 

- Authors
- Figures
- References
- Keywords
- Metrics
- More Like This
- Footnotes

Abstract: Recommender systems provide useful item suggestions (products or services) to users as part of their decision-making processes. The effectiveness of recommender systems i... [View more](#)

Metadata

Abstract: Recommender systems provide useful item suggestions (products or services) to users as part of their decision-making processes. The effectiveness of recommender systems is now clearly confirmed in various fields of application (e.g., YouTube, Amazon, Facebook, ResearchGate). In the literature, many research works have addressed the application of recommendations in the field of health in what are called health recommender systems (HRS). HRS is an innovative alternative when it comes to providing information to help doctors in the diagnosis/treatment of diseases, as well as helping patients with recommendations on how to maintain their well-being. However, the proposed development approaches in this field are limited to traditional models that lack the accuracy and effectiveness, which are vital in healthcare. This paper presents a design model for collaborative filtering-based health recommender systems using graph neural networks (GNN) via its promising Graph Convolutional Network (GCN) architecture. In this model, the convolution layer works with a simplified and efficient GCN algorithm named LightGCN. GCN-based methods are among the new cutting-edge approaches in recommender systems, and LightGCN has proven its superiority in recommendation accuracy.

Published in: 2022 2nd International Conference on New Technologies of Information and Communication (NTIC)

Date of Conference: 21-22 December 2022 **INSPEC Accession Number:** 22963495

Date Added to IEEE Xplore: 21 April 2023 **DOI:** 10.1109/NTIC55069.2022.10100590

ISBN Information: **Publisher:** IEEE

Conference Location: Mila, Algeria