UCLouvain
FUCaM MONS

LOUVAIN
School of Management

Louvain School of Management

Intégration des mesures de centralité sur graphe dans les systèmes de recommandation pour augmenter la nouveauté et la diversité de leurs recommandations.

Auteur : Louise Cotton
Promoteur : François Fouss
Année académique 2020-2021
Master [120] Ingénieur de gestion, à finalité spécialisée

Résumé

Les systèmes de recommandation sont de plus en plus présents sur internet. Ils permettent de filtrer les informations vues par un utilisateur en fonction de ses goûts et de ses intérêts. De nos jours, on attend plus seulement qu'un système de recommandation soit précis, mais qu'il propose des articles nouveaux et divers à l'utilisateur. Dans ce mémoire, nous étudierons *l'intégration des mesures de centralité sur graphe dans les systèmes de recommandation pour augmenter la nouveauté et la diversité de leurs recommandations.*

Pour ce faire, nous nous pencherons, tout d'abord, sur les études déjà effectuées dans ce domaine avec l'intégration des mesures sur graphe dans les systèmes de recommandation. Toutes ces études concluent que l'intégration des caractéristiques sur graphe améliore la performance des systèmes au niveau de leur précision.

Ensuite, nous implémenterons plusieurs systèmes de recommandations pour mener notre propre étude. Nous analyserons la performance de ces systèmes intégrant des mesures d'intermédiarité ou de proximité à différentes étapes. Cette performance est évaluée sur les critères de précision, de diversité et de nouveauté. Cette phase d'expérimentation se fera à l'aide du langage de programmation Python.

À la suite de cette étude, nous pouvons constater que l'intégration des mesures de centralité sur graphe peut avoir un impact sur la nouveauté et la diversité des recommandations. Cependant, cet impact n'est pas toujours positif. En effet, notre étude ne permet pas de conclure que l'intégration de ces mesures résulte à une augmentation de la nouveauté ou de la diversité dans les listes de recommandations. Nous avons aussi constaté que l'intégration de ces mesures avait aussi un impact sur la précision du système, celui-ci peut être positif ou négatif. Toutefois, dans cette étude, nous privilégions l'impact sur les mesures de nouveauté et de diversité à la précision.

Remerciements

Lors de la rédaction de ce mémoire, j'ai pu compter sur l'aide et le soutien de plusieurs personnes qui j'aimerais à présent remercier.

Tout d'abord, je remercie mon promoteur, Monsieur François Fouss, pour son encadrement et ses conseils tout au long de mon parcours académique. Il m'a accompagné et a su se rendre disponible pendant ce second quadrimestre pour m'orienter et répondre à mes questions.

De plus, je souhaiterais remercier les assistantes, Chloé Satinet et Elora Fernandes, pour leur aide pendant l'absence de Monsieur Fouss ce premier quadrimestre.

Ensuite, je remercie ma famille pour leur soutien sans failles tout au long de mon parcours scolaire et particulièrement durant ce master. Merci à ma maman, Marjorie Olivier, pour ses nombreuses relectures et ses conseils sur la rédaction de mon mémoire. Merci à mon papa, Jean Cotton, pour le temps consacré à la compréhension de mon mémoire et pour son aide technique. Et merci à ma sœur, Marie Cotton, pour ses encouragements tout au long de cette période.

Enfin, je remercie toutes les personnes qui ont de près ou de loin rendu cette rédaction plus facile ou agréable.

Table des matières

Listes des Tableaux

Liste des Figures

Liste des annexes

Introduction

Présentation de la question de recherche

À travers ce mémoire, nous allons nous intéresser à « *l'intégration des mesures de centralité sur graphe dans les systèmes de recommandation pour augmenter la nouveauté et la diversité de leurs recommandations* ».

Les systèmes de recommandation sont de plus demandés puisque le nombre de visiteurs et d'informations présents sur internet ne fait que croitre produisant ainsi une surcharge d'information (Isinkaye, Folajimi, et Ojokoh, 2015). Ils sont donc un outil pour interagir avec des espaces d'information larges et complexes et offrent une vue personnalisée de cet espace en priorisant des articles qui seront probablement plus appréciés de l'utilisateur (Burke, Felfernig, et Göker, 2011).

Les systèmes de recommandation sont bénéfiques aux entreprises comme aux utilisateurs. Ils améliorent les revenus du e-commerce et réduisent les coûts de transaction (Isinkaye et al., 2015). Ces derniers ont toujours été évalués sur leur précision (Bridge & Kaminskas, 2016), cependant la nouveauté et la diversité sont aussi des critères importants pour faire des recommandations satisfaisantes (Han & Yamana, 2017).

Dans ce mémoire, nous proposons d'utiliser des mesures de centralité sur graphe pour servir ces systèmes. Les mesures de centralité font référence à la position de certains nœuds dans la structure du graphe (Freeman, 1978). Elles résument l'investissement et la contribution à la cohésion du graphe d'un nœud (Borgatti & Everett,2005). Elles permettent donc d'identifier des nœuds qui sont centraux d'une manière ou d'une autre au graphe.

L'objet de ce mémoire vise donc à incorporer cette information provenant du domaine des graphes dans différents systèmes de recommandation afin d'évaluer l'impact de cette manipulation sur la nouveauté et la diversité de leurs recommandations.

Chapitre 1 : Revue de littérature

Introduction

La question de recherche de ce travail étant définie, nous allons maintenant approfondir certains points à partir de la littérature. Tout d'abord, nous nous attarderons sur la théorie des systèmes de recommandation en abordant les différents types existants, leur fonctionnement et particulièrement celui du filtrage collaboratif qui nous intéresse dans ce mémoire, et leurs évaluations. Nous terminerons ce point en présentant les bénéfices retirés de l'utilisation de ces systèmes par les entreprises ainsi que par les utilisateurs. Ensuite, nous aborderons la théorie des graphes en définissant un graphe ainsi que tous ses composants. Cela nous permettra de présenter différentes mesures de centralité construites sur graphe. Enfin, nous parcourons plusieurs études liant la théorie des graphes aux systèmes de recommandation.

Système de recommandation

Avant internet, les gens pouvaient trouver compliqué la recherche d'un livre à lire, d'un film à regarder ou d'un endroit à visiter. Ils prenaient généralement ce type de décisions en écoutant les commentaires donnés par leurs amis. Les systèmes de recommandation découlent de ce même concept (Khachane & Vaidya, 2017). L'importance grandissante d'internet comme moyen de transaction a donc servi de moteur au développement de leur technologie. Ils sont utilisés comme outil par le e-commerce afin de guider les utilisateurs dans leurs recherches en proposant des documents liés à leurs intérêts et préférences (Isinkaye et al., 2015). Ils servent aussi à la personnalisation en ligne (Jannach, Rook, et Zanker, 2019). C'est un moyen d'assister et d'augmenter le processus social qui utilise les recommandations des autres pour faire un choix quand il n'y a pas de connaissance personnelle suffisante (Isinkaye et al., 2015).

Les systèmes de recommandation sont donc un sous champ des systèmes de filtre (Antala, Dongare, Salunke, et Shah, 2017). Ils ont été développés pour contrer la surcharge d'information présente sur internet. Ce sont des systèmes de filtres qui se basent sur les préférences ou le comportement du consommateur. Ils ont la capacité de prédire si un utilisateur apprécierait un article ou pas en se basant sur le profil qui lui est associé (Isinkaye et al., 2015). Certains auteurs pensent que les systèmes de recommandation aident les consommateurs à découvrir de nouveaux produits et augmentent donc la diversité des ventes. D'autres sont d'avis qu'ils renforcent la visibilité et la vente des produits les plus populaires seulement. Certaines études confirment cette deuxième croyance (Fleder & Hosanagar, 2007). Ils sont bénéfiques aux utilisateurs et aux fournisseurs de service, car ils réduisent le cout de transaction lié à la recherche d'articles en ligne et augmentent les revenus du e-commerce (Isinkaye et al., 2015).

Les objectifs opérationnels des systèmes de recommandation sont de recommander des articles qui sont pertinents, nouveaux, surprenants et diversifiés pour l'utilisateur (Aggarwal, 2016). Le terme « Article » est généralement utilisé pour désigner ce que le système de recommandation propose à l'utilisateur (Antala et al., 2017). Le principe de base de ces systèmes est qu'il existe une dépendance significative entre les activités centrées sur les utilisateurs ou sur les articles (Aggarwal, 2016). La technique la plus éminente dans ce domaine est le filtrage collaboratif (Burke et al., 2011) qui utilise les interactions des articles et des utilisateurs. Une autre technique appelée « content-based recommender » utilise les attributs des utilisateurs et des articles tels que les profils textuels ou les mots-clés pertinents (Aggarwal, 2016). Cette technique recommande des articles similaires aux articles aimés précédemment en se basant sur

des caractéristiques spécifiques au contenu de l'article (Tourwé, 2012). Dans ce mémoire, nous travaillerons uniquement avec des systèmes de recommandation de filtrage collaboratif.

1.1. Filtrage Collaboratif

Cette technique utilise le pouvoir des notes données par les autres utilisateurs pour faire les recommandations (Aggarwal, 2016). En d'autres mots, elle se base sur l'hypothèse que les personnes qui étaient d'accord dans le passé ont de fortes chances pour être d'accord dans le futur (Antala et al., 2017), les notes non connues peuvent donc être attribuées à l'aide des notes connues. Deux types de méthodes de filtrage collaboratif sont couramment utilisées ; la méthode dite « memory-based » et la méthode dite « model-based » (Aggarwal, 2016). Dans la première méthode, le système utilise directement la matrice de notations des utilisateurs sur les articles pour effectuer ses recommandations, cette méthode est aussi appelée méthode basée sur le voisinage (Antala et al., 2017). Dans la deuxième méthode, l'apprentissage automatique et le « data mining » sont utilisés pour construire des modèles prédictifs (Aggarwal, 2016). Dans ce travail, nous travaillerons avec la méthode basée sur le voisinage. Cette méthode des plus proches voisins consiste à trouver un voisinage d'homologues avec des vues similaires (Burke et al., 2011), elle se concentre soit sur les corrélations entre les articles, soit sur les corrélations entre les utilisateurs (Aggarwal, 2016).

L'approche basée sur les utilisateurs évalue l'intérêt d'un utilisateur pour un article en prenant en considération les notes de ce même article par les autres utilisateurs, appelés ses voisins, qui ont les mêmes habitudes de cotation (Antala et al., 2017). L'idée de base est donc de déterminer les utilisateurs qui ont des intérêts similaires à l'utilisateur cible afin d'estimer les notes non connues de cet utilisateur en calculant la moyenne pondérée des notes de ce groupe d'utilisateurs (Aggarwal, 2016). L'approche basée sur les articles prédit les notes d'un article en se basant sur les notes des articles qui lui sont similaires (Antala et al., 2017). Afin de faire les prédictions des notes d'un article d'intérêt, la première étape est de déterminer les articles qui lui sont le plus similaires. Les notes de ces articles par l'utilisateur A sont utilisées pour prédire l'utilité de cet article d'intérêt pour l'utilisateur A (Aggerwal, 2016). Les similarités sont calculées soit avec les coefficients de corrélation soit avec les cosinus. Certaines études montrent que l'approche basée sur les articles délivre des résultats de meilleure qualité que l'approche basée sur les utilisateurs (Bell, Koren, et Volinsky, 2007).

Les principaux avantages de cette technique sont qu'elle est facile à implémenter et que les recommandations sont souvent faciles à expliquer (Aggarwal, 2016). De plus, elle ne demande pas de phase d'entrainement couteuse et sa stabilité. Cependant, cette méthode apporte aussi des désavantages. Tout d'abord, lors de l'arrivée de nouveaux utilisateurs ou de nouveaux articles, soit le système ne sait pas ce qu'il doit recommander soit il a une faible performance. Ce problème est connu sous le nom de « cold start ». De plus, cette technique nécessite une puissance de calcul conséquente puisqu'elle génère des recommandations pour des millions d'articles et d'utilisateurs. Enfin, seulement un petit sous-groupe d'articles est généralement noté par l'utilisateur ce qui peut entrainer une faible performance due à la faible densité des notes (Antala et al., 2017). Le système peut aussi parfois souffrir de surspécialisation quand il a tendance à recommander à un utilisateur des articles qui sont fortement similaires aux articles précédemment notés par celui-ci (Tourwé, 2012).

1.1.1. Mesures de similarité

La clé dans la technique des plus proches voisins est de trouver des utilisateurs similaires en utilisant la matrice de notations. Le cœur de cette technique de filtrage collaboratif est donc de calculer les similarités entre les utilisateurs ou les articles (Chandralekha, Sadasivam, et Saranya, 2016). Cela sera utilisé plus tard lors de la localisation des plus proches voisins. Dans l'approche basée sur les utilisateurs, la similarité est calculée entre les utilisateurs et ceux ayant des valeurs de similarité supérieures représenteront les voisins de l'utilisateur d'intérêt. Les notes de ceux-ci et leurs valeurs de similarité seront utilisées pour prédire la notation de l'utilisateur d'intérêt sur les articles qu'il n'a pas notés. Pour l'approche basée sur les articles, les étapes sont les mêmes, mais les similarités sont calculées cette fois-ci sur les articles. L'avantage de cette technique est que les similarités sont plus stables (Abdulgabber, Al-bashiri, Hujainah, et Romli, 2017). En effet, en utilisant la similarité sur les utilisateurs, celle-ci peut être biaisée puisque les utilisateurs peuvent avoir différentes échelles de notation. Un utilisateur peut être plus susceptible d'aimer la plupart des articles quand un autre peut être sujet à ne pas aimer la plupart (Aggarwal, 2016). Il est donc essentiel de sélectionner une fonction de similarité adaptée afin d'améliorer la précision du système (Abdulgabber et al., 2017).

Dans beaucoup de systèmes de recommandation, le coefficient de corrélation de Pearson (*PCC*) est appliqué pour mesurer la similarité entre les utilisateurs ou entre les articles (Chandralekha et al., 2016). Pour calculer la similarité entre un utilisateur u et un utilisateur v ($\mathrm{Sim}(u,v)$), la première étape est de calculer la moyenne des notations pour chaque utilisateur tel que ;

$$Moyenne\ des\ notations(u)\ =\ \mu_u = \frac{\sum_{i \in I_u} r_{ui}}{|I_u|} \qquad \forall u\ \in \{1 \ldots m\}$$

Où I_u représente l'ensemble des notations connues pour l'utilisateur *u* ;
r_{ui} est égal à la notation de l'utilisateur *u* pour l'article *i*.

Ensuite, le coefficient de corrélation de Pearson est défini comme suit ;

$$Sim(u,v) = Pearson(u,v) = \frac{\sum_{i \in I_u \cap I_v}(r_{ui} - \mu_u) \times (r_{vi} - \mu_v)}{\sqrt{\sum_{i \in I_u \cap I_v}(r_{ui} - \mu_u)^2} \times \sqrt{\sum_{i \in I_u \cap I_v}(r_{vi} - \mu_v)^2}}$$

Où I_u représente l'ensemble des notations connues pour l'utilisateur *u* ;
r_{ui} est égal à la notation de l'utilisateur *u* pour l'article *i* ;
I_v représente l'ensemble des notations connues pour l'utilisateur *v* ;
r_{vi} est égal à la notation de l'utilisateur *v* pour l'article *i*.

Puisque $I_u \cap I_v$ représente l'ensemble des articles qui ont été notés par l'utilisateur *u* et l'utilisateur *v*, le coefficient est calculé uniquement sur ces articles. La définition traditionnelle de Pearson exige que la moyenne μ soit calculée strictement sur ces mêmes articles. Cependant, il est commun de calculer μ une seule fois pour chaque utilisateur. Il est difficile de fournir un argument pour l'une ou l'autre approche, dans les cas extrêmes où les deux utilisateurs n'ont qu'un article en commun, la deuxième approche, la plus commune, prodiguera des résultats plus informatifs (Aggarwal, 2016). La valeur de similarité calculée à l'aide de ces formules sera entre -1 et 1. Plus sa valeur sera élevée, plus deux utilisateurs seront similaires (Chandralekha et al., 2016). Une autre mesure de similarité traditionnelle est le cosinus, la normalisation des scores des voisins communs. Cette mesure se calcule comme suit ;

$$Sim(u,v) = Cosinus(u,v) = \frac{\sum_{i \in I_u \cap I_v} r_{ui} \times r_{vi}}{\sqrt{\sum_{i \in I_u \cap I_v} {r_{ui}}^2} \times \sqrt{\sum_{i \in I_u \cap I_v} {r_{vi}}^2}}$$

Où I_u représente l'ensemble des notations connues pour l'utilisateur *u* ;
r_{ui} est égal à la notation de l'utilisateur *u* pour l'article *i* ;
I_v représente l'ensemble des notations connues pour l'utilisateur *v* ;
r_{vi} est égal à la notation de l'utilisateur *v* pour l'article *i*.

Le cosinus calcule l'amplitude de l'angle présent entre *u* et *v*, ceux-ci représentés sous forme de profil vectoriel. Plus l'angle est petit, plus les deux utilisateurs ou les deux articles sont considérés comme similaires puisqu'ils ont des avis similaires (Fouss, Saerens, et Shimbo, 2016). Comme précédemment, dans certaines applications, le facteur de normalisation du dénominateur peut se baser sur tous les articles de l'utilisateur plutôt que sur les notations mutuelles. Dans ce cas-là, la mesure se définit comme suit ;

$$Sim(u,v) = Cosinus(u,v) = \frac{\sum_{i \in I_u \cap I_v} r_{ui} \times r_{vi}}{\sqrt{\sum_{i \in I_u} {r_{ui}}^2} \times \sqrt{\sum_{i \in I_v} {r_{vi}}^2}}$$

Où I_u représente l'ensemble des notations connues pour l'utilisateur *u* ;
r_{ui} est égal à la notation de l'utilisateur *u* pour l'article *i* ;
I_v représente l'ensemble des notations connues pour l'utilisateur *v* ;
r_{vi} est égal à la notation de l'utilisateur *v* pour l'article *i.*

De manière générale, le coefficient de corrélation de Pearson est préférable au cosinus puisqu'il réduit le biais en centrant les notes sur la moyenne. Ce biais vient du fait que les utilisateurs montrent des degrés de générosité différents dans leur mode de notation (Aggarwal, 2016). Ces mesures ont l'avantage d'être faciles à calculer. Cependant, celles-ci reposent entièrement sur la matrice de notations, ainsi si deux utilisateurs n'ont aucun article en commun, la similarité sera de 0. De plus, ces mesures ne prennent pas en compte la proportion de notations communes (Abdulgabber et al., 2017).

1.2. Évaluation des systèmes de recommandation

Initialement, la plupart des systèmes de recommandation ont été évalués et classés selon leur pouvoir de prédiction c'est-à-dire leur capacité à prédire les choix de l'utilisateur de manière exacte. Cependant, à l'heure actuelle il est fortement répandu que même si la capacité de prédiction est essentielle, elle est insuffisante pour évaluer un bon système de recommandation (Gunawardana & Shani, 2015). La communauté prête une attention grandissante à la nouveauté et à la diversité comme qualités clés d'un système de recommandation (Castells & Vargas, 2011). En effet, dans beaucoup d'applications, le système de recommandation est utilisé pour plus qu'une anticipation exacte du goût des utilisateurs. Ceux-ci peuvent aussi avoir un intérêt pour la découverte de nouveaux articles ou pour explorer rapidement des articles divers (Gunawardana & Shani, 2015).

1.2.1. Mesures de précision

Les mesures d'évaluation du pouvoir de précision peuvent être regroupées en deux catégories ; celles évaluant la précision de prédiction et celles évaluant la précision de la liste de recommandations. La première catégorie concerne la capacité du système à prédire les notes tandis que la deuxième concerne sa capacité à proposer des articles pertinents dans une courte liste de recommandations (Burke et al., 2011).

Les mesures de précision statistiques comparent la note obtenue par le système avec la note donnée par l'utilisateur directement. L'erreur absolue moyenne (*MAE*) et l'erreur quadratique moyenne (*RMSE*) sont généralement utilisées à cette fin. La moyenne de l'erreur absolue mesure la déviation des recommandations des valeurs spécifiques d'un utilisateur (Isinkaye et al., 2015).

$$Erreur\ absolue\ moyenne\ =\ MAE\ =\ \frac{\sum_{i\in R}|p\ -\ r|}{|R|}$$

Où *p* représente la note obtenue par le système ;
r correspond à la note réelle donnée par l'utilisateur ;
R est égal au nombre de notations présentes dans l'ensemble test.

Plus petite est la *MAE*, plus grande est la précision. L'erreur quadratique moyenne mesure également la déviation entre les recommandations et la note réelle en pénalisant les larges erreurs (Aggarwal, 2016).

$$Erreur\ quadrique\ moyenne\ =\ RMSE\ =\ \sqrt{\frac{\sum_{i\in R}(p\ -\ r)^2}{|R|}}$$

Où *p* représente la note obtenue par le système ;
r est la note réelle donnée par l'utilisateur ;
R correspond au nombre de notations présentes dans l'ensemble test.

Comme l'erreur quadratique moyenne additionne le carré des erreurs, elle est affectée plus significativement par les larges erreurs. Quelques notes mal prédites peuvent gâcher la mesure *RMSE*. Cette mesure est plus appropriée lorsque la robustesse, la résistance du système aux fausses notes ou aux attaques, est très importante. D'un autre côté, la *MAE* est un meilleur reflet de la précision. Le principal problème avec la *RMSE* est qu'elle n'est pas le reflet de l'erreur moyenne, ce qui peut tendre à des résultats trompeurs (Aggarwal, 2016).

Les mesures de précision de la liste de recommandations ont pour but de quantifier l'utilité d'un système de recommandation pour un utilisateur. Le principe derrière ces mesures est que la liste de recommandations est courte comparée au nombre total d'articles disponibles et que les utilisateurs sont seulement intéressés par les premiers articles. L'utilité d'un système devrait donc être basée sur la pertinence des articles présents dans le début de la liste de recommandations. L'utilité d'un article dépend de sa position dans la liste et de sa note réelle. Un exemple de ce type de mesure est le gain cumulé actualisé (*DCG*) où le facteur d'actualisation d'un article *i* est défini par $log_2(v_i + 1)$, v_i représentant la position de l'article dans la liste (Aggarwal, 2016). La base du logarithme est un paramètre libre, cependant la base 2 est généralement utilisée pour garantir l'actualisation de toutes les positions (Gunawardana & Shani, 2015).

$$DCG = \sum_i \frac{r_{ui}}{\log_2(v_i + 1)}$$

Où r_{ui} est égal à la notation de l'utilisateur *u* pour l'article *i* ;

v_i correspond à la position de l'article *i* dans la liste de recommandations.

Le gain cumulatif actualisé normalisé (*nDCG*) est ensuite défini par le ratio du *DCG* sur sa valeur idéale (*IDCG*) (Aggarwal, 2016).

$$nDCG = \frac{DCG}{IDCG}$$

Une autre mesure appelée la « précision moyenne » (*MAP*), mesure la proportion d'articles pertinents présents dans la liste (Aggarwal, 2016). Pour procéder à une évaluation hors-ligne, nous possédons typiquement un ensemble de données comprenant un ensemble d'articles notés pour chaque utilisateur. Dans cet ensemble, nous sélectionnons certaines données qui seront alors cachées afin d'être prédites par le système de recommandation, ce sous-ensemble sera nommé « l'ensemble test ». Nous avons alors quatre résultats possibles pour ces données cachées et recommandées (Gunawardana & Shani, 2015) indiqués au Tableau 1.1.

	Recommandé	**Pas recommandé**
Noté	Vrais positifs (N_{vp})	Faux négatifs (N_{fn})
Pas noté	Faux positifs (N_{fp})	Vrais négatifs (N_{vn})

Tableau 1.1 - Résultats sur un ensemble test

Des mesures comme la précision et le « recall » interprètent la procédure de prédiction comme une opération binaire qui distingue les bons articles des mauvais articles. La précision est la fraction d'articles recommandés qui sont vraiment pertinents pour l'utilisateur quand le recall est la fraction d'articles pertinents qui font également partie de l'ensemble recommandé (Isinkaye et al., 2015). Ces mesures sont calculées comme suit ;

$$Précision = \frac{Articles\ correctement\ recommandés}{Nombre\ total\ d'articles\ recommandés} = \frac{\#N_{vp}}{\#N_{vp} + \#N_{fp}}$$

$$Recall = \frac{Articles\ correctement\ recommandés}{Nombre\ total\ d'articles\ pertinents} = \frac{\#N_{vp}}{\#N_{vp} + \#N_{fn}}$$

1.2.2. Au-delà de la précision

La nouveauté est une caractéristique souhaitée dans un système de recommandation puisque le but de celui-ci est fortement lié à la notion de découverte en proposant à l'utilisateur des articles pertinents qu'il n'aurait pas trouvés lui-même (Castells & Vargas, 2011).

La nouveauté d'un système de recommandation évalue donc sa probabilité de proposer des articles que l'utilisateur ne connaissait pas (Aggarwal, 2016). Un article nouveau est un article qui n'était pas connu préalablement par l'utilisateur (Bridge & Kaminskas, 2016). Sans retour direct de l'utilisateur, vérifier si un article est nouveau à un utilisateur est difficilement possible. Deux tendances peuvent être observées dans la littérature, les mesures peuvent être définies localement pour chaque utilisateur ou globalement pour l'ensemble des utilisateurs (Anonyme, 2020). Une méthode pour évaluer la nouveauté repose sur l'hypothèse qu'un article populaire a moins de chance d'être nouveau pour l'utilisateur (Gunawardana & Shani, 2015). La nouveauté d'un article est d'ailleurs typiquement définie par l'inverse de sa popularité, les articles moins populaires ont une plus grande probabilité d'être nouveaux pour l'utilisateur (Bridge & Kaminskas, 2016).

La popularité d'articles mesure la popularité moyenne des articles présents dans la liste de recommandations. Cette mesure prend typiquement la forme suivante ;

$$Popularité\ des\ articles = IP(R) = \frac{\sum_{i \in R} pop_i}{|R|}$$

Où pop_r indique la popularité de l'article i ;

R représente l'ensemble des articles présents dans la liste de recommandations de l'utilisateur.

Une autre approche pour calculer la nouveauté mesure l'indifférence d'un utilisateur par rapport aux articles, la distance d'un utilisateur permet de mesurer cette indifférence et se formule comme suit ;

$$Distance\ d'un\ utilisateur\ =\ DU(R, L_u)\ =\ \frac{\sum_{r_1 \in R} \sum_{r_2 \in L_u} dissim(r_1, r_2)}{|R| * |L_u|}$$

Où $L_u$représente l'ensemble des articles ayant eu une interaction avec l'utilisateur *u ;*

R est l'ensemble des articles présents dans la liste de recommandations de l'utilisateur *u* ;

$dissim(r_1, r_2)$ est égal à la dissimilarité entre l'article r_1 et l'article r_2.

L'idée derrière cette mesure est que si un article est dissimilaire des articles ayant des interactions fréquentes avec l'utilisateur, celui-ci est probablement inconnu à cet utilisateur (Han & Yamana, 2017).

La notion de diversité signifie qu'un ensemble d'articles proposés dans une même liste de recommandations doit être aussi diversifié que possible. Il existe une corrélation positive entre la diversité perçue dans la liste de recommandations et la précision perçue, et donc avec la satisfaction générale de l'utilisateur sur le système de recommandation (Bridge & Kaminskas, 2016). C'est donc une caractéristique importante pour un système de recommandation, de plus, si un utilisateur n'apprécie pas le premier choix, il y a de fortes chances qu'il n'apprécie aucun des articles recommandés si la liste n'est pas diversifiée. Présenter différentes sortes d'articles peut donc souvent augmenter la probabilité qu'un utilisateur sélectionne l'un d'entre eux (Aggarwal, 2016). La diversité est souvent définie comme l'opposé de la similarité (Gunawardana & Shani, 2015). Trois types de diversité se retrouvent dans la littérature ; la diversité simple, la diversité représentative du groupe d'articles et la diversité représentative de l'utilisateur. La diversité simple s'apparente à la similarité entre articles, cette définition se préoccupe uniquement de la dissimilarité entre des paires d'articles. La deuxième définition représente la diversité présente parmi les groupes ou catégories d'articles. La dernière définition calcule la proportion d'articles recommandés différents des préférences de l'utilisateur d'intérêt (Han & Yamana, 2017). Certains auteurs augmentent la diversité tout en gardant la perte de

précision la plus basse en proposant des articles moins populaires dans les premiers choix (Castells & Vargas, 2011).

La similarité calculée entre les articles utilisée pour l'évaluation peut être différente de celle utilisée pour le calcul de la liste de recommandations (Gunawardana & Shani, 2015). La diversité simple peut donc être mesurée à l'aide de la similitude de contenu entre des paires d'articles. La similitude moyenne de toutes les paires peut être déclarée comme diversité où des plus petites valeurs indiquent une meilleure hétérogénéité (Aggarwal, 2016).

La mesure de dissimilarité intraliste (*IDL*) est largement utilisée pour évaluer la diversité simple parmi les articles recommandés et se formule comme suit ;

$$Dissimilarité\ intra-liste\ =\ ILD(R)\ =\ \frac{\sum_{r_1 \in R} \sum_{r_2 \in R} dissim(r_1, r_2)}{|R| * |R|}$$

Où R représente l'ensemble des articles présents dans la liste de recommandations ;

$dissim(r_1, r_2)$ est une fonction de dissimilarité qui quantifie la différence entre deux articles.

Une étude réalisée par Willemsen et autre montre que les listes avec un degré élevé de diversification sont perçues comme plus diverses par les utilisateurs. L'attractivité perçue des recommandations augmente quant à elle entre une diversification faible et moyenne, mais pas avec une diversification élevée. Ce résultat suppose qu'après un certain degré de diversité, les utilisateurs n'apprécient plus une plus grande diversification (Bridge & Kaminskas, 2016).

1.3. Bénéfices d'un système de recommandation

Les systèmes de recommandation jouent un rôle majeur dans l'industrie d'aujourd'hui du e-commerce (Khachane & Vaidya, 2017). À l'ère d'internet, le plus gros problème d'un acheteur potentiel n'est pas la recherche d'information pour prendre une décision, mais de prendre une décision avec une énorme source d'information (Wang & Zhang, 2005). Certaines personnes avec un manque d'expérience peuvent prendre de mauvaises décisions si des recommandations ne sont pas disponibles. Grâce aux systèmes de recommandation, le visiteur du site a plus de chance d'acheter le produit et d'être satisfait (Kachane & Vaidya, 2017). Il a été démontré qu'ils augmentent la probabilité de correspondance entre un client et un article (Jannach et al., 2019). Au plus les clients sont satisfaits des systèmes de recommandations, au plus populaire devient le site internet (Kachane & Vaidya, 2017). Plusieurs études démontrent d'ailleurs que leur

utilisation augmente l'activité et le temps de connexion d'un utilisateur sur le site (Jannach et al., 2019).

Les systèmes de recommandation sont utilisés pour recommander une variété de produits tels que les films, la musique, les actualités, les livres et beaucoup d'autres (Kachane & Vaidya, 2015). Les systèmes de recommandation aident les vendeurs à promouvoir leurs ventes et ils facilitent le processus d'achat des consommateurs. Ils prennent le rôle des forces de vente virtuelles (Wang & Zhang, 2005). L'utilisation de systèmes de recommandation peut mener à une augmentation des ventes en inspirant plus d'achats à l'acheteur (Jannach et al., 2019).

Le vendeur peut également les utiliser pour analyser le comportement de ses clients, leurs besoins manifestes et leurs besoins potentiels. Du point de vue de l'acheteur, recevoir des recommandations pertinentes lors de leurs visites sur le site d'un marchand, leur permettent de réduire leur temps de recherche et leur énergie (Wang & Zhang, 2005).

Domaine des graphes

Un graphe est composé d'un ensemble de nœuds et d'arêtes reliant ces pairs de nœuds (Freeman, 1978). En général, les nœuds représentent des objets ou des entités et les arêtes attestent de l'existence d'un lien entre les deux entités reliées (par exemple : « est l'ami de »). Un graphe où la paire de nœuds (p_i, p_j) est distincte de la paire (p_j, p_i) est appelé un graphe dirigé. Si les liens n'ont pas de direction alors le graphe est dit non dirigé (Fouss et al., 2016). Quand deux points sont directement connectés par un lien, ceux-ci sont adjacents (Freeman, 1978) et sont donc voisins (Fouss et al., 2016). Le degré d'un nœud donné est mesuré par le nombre de nœuds adjacents à celui-ci (Freeman, 1978). Deux nœuds (p_i, p_j) sont joignables l'un à l'autre si et seulement s'il existe un chemin, une séquence d'un ou plusieurs nœuds, débutant au nœud p_i et terminant au nœud p_j, cela en passant éventuellement par des nœuds intermédiaires. Un chemin qui se termine et commence au même nœud est appelé un cycle. À chaque chemin est associée une distance égale au nombre de nœuds composant celui-ci, le chemin le plus court entre une paire de nœuds est dit « géodésique ». Un nœud présent sur le chemin géodésique d'une paire de nœuds est dit être entre ces deux nœuds (Freeman, 1978). Les graphes bipartis sont des graphes composés de deux différents types de nœuds, par exemple, des nœuds de type client et des nœuds de type article et il existera un lien entre un client *i* et un article *j* si le client a acheté cet article. Un graphe biparti est donc un graphe où les nœuds peuvent être divisés en deux ensembles disjoints *X* et *Y* tel que chaque lien relie un nœud de *X* à un nœud *Y* ou vice versa. Par conséquent, aucun lien ne relie deux nœuds de l'ensemble *X* ou de l'ensemble *Y* (Fouss et al., 2016).

Il y a plusieurs manières de représenter un graphe mathématiquement. Tout d'abord, un réseau peut être représenté par un nombre de nœuds accompagné d'une liste d'arêtes. Cependant, cette représentation est encombrante pour les développements mathématiques (Newman, 2010). Ensuite, la structure d'un graphe non dirigé peut également être saisie dans une matrice carrée de la taille du graphe. Cette matrice est appelée la matrice adjacente avec a_{ij} défini comme ci-dessous (Fouss et al., 2016). Cette matrice est symétrique et sa diagonale ne contient que des zéros si le réseau ne contient pas de lien d'un nœud vers lui-même (Newman, 2010).

$$a_{ij} = [A]_{ij} \triangleq \begin{cases} 1 & si\ i\ et\ j\ sont\ adjacents \\ 0 & autrement \end{cases}$$

N'importe quel réseau de communication peut être représenté par un graphe (Freeman, 1978). Dans le domaine des réseaux sociaux, les graphes peuvent être utilisés pour modéliser les liens entre les membres d'une communauté afin de trouver les proximités, les similarités ou les comportements communs de ses membres (Fouss et al., 2016). Ci-dessous, Figure 1.1, un exemple de graphe non dirigé contenant 10 nœuds et 18 liens. Dans la Figure 1.2, le plus court chemin entre le nœud 3 et le nœud 5 tandis que dans la Figure 1.3, nous pouvons observer les chemins géodésiques existants entre les nœuds 2 et 10. En effet, le chemin géodésique entre cette paire de nœuds peut passer par quatre nœuds différents, les nœuds 1, 3, 4, 7.

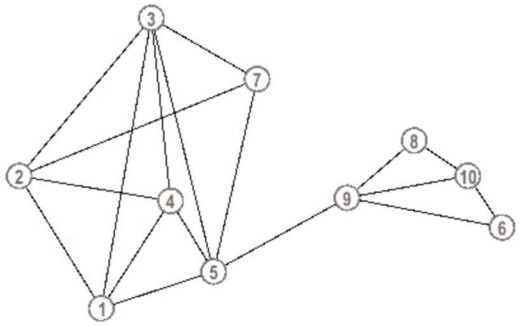

Figure 1.1 - Exemple de graphe non dirigé

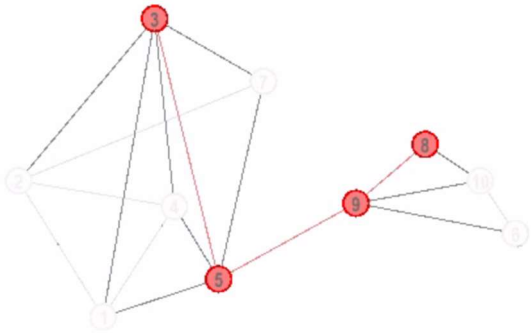

Figure 1.2 - Exemple de graphe avec chemin géodésique

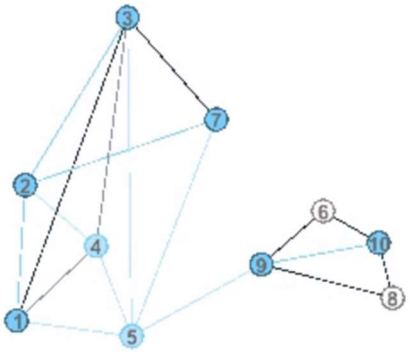

Figure 1.3 - Exemple de graphe avec chemins géodésiques

2.1. Mesures de centralité

Il n'y a pas de définition unanime sur ce qu'est exactement la centralité. Cependant, les mesures de centralité d'un nœud appartenant à un graphe découlent de l'intuition qu'un point positionné au centre d'une étoile est universellement supposé être structurellement plus central que tous les autres points. Cette localisation a le degré le plus élevé possible, est située sur le plus de chemins géodésiques et est la plus proche de tous les autres points du graphe (Freeman, 1977).

Le concept de centralité concerne la position des nœuds dans le réseau. Ce concept peut aussi être appliqué à la structure entière du réseau (Freeman, 1978). Les mesures de centralité appliquées à un nœud résument l'investissement et la contribution de ce nœud à la cohésion du réseau (Borgatti & Everett, 2005). Généralement, ces mesures de centralité sont calculées sur des graphes non dirigés ou sur des graphes dits « non directionnels », des graphes dirigés ignorant la direction des liens (Fouss et al., 2016).

L'utilisation de la centralité au domaine de la communication humaine a été introduite pour la première fois par Bavelas en 1948 qui a émis l'hypothèse d'une relation entre la centralité structurelle et l'influence dans les processus de groupe (Freeman, 1978). Cette hypothèse a été confirmée par de nombreuses études qui ont confirmé que la centralité est liée à l'efficience du groupe dans la résolution de problèmes, dans la perception de leadership et dans la satisfaction personnelle des participants (Freeman, 1978).

2.1.1. Degré

La façon la plus simple de conceptualiser la centralité d'un nœud est fonction du degré de ce dernier (Freeman, 1978). Le degré, comme déjà mentionné plus haut, est le nombre de nœuds qui lui sont adjacents (Fouss et al., 2016), et qui sont donc en contact direct avec celui-ci. Dans un graphe, chaque nœud peut être adjacent à maximum *n - 1* nœuds où *n* représente le nombre total de nœuds présents dans le graphe. Pour certains auteurs, cette définition est tellement intuitive que pour eux centralité signifie degré (Freeman, 1978). Dans les réseaux sociaux, par exemple, il semble raisonnable de supposer que les acteurs qui ont plus de connexions ont plus d'influence et d'accès à l'information que les acteurs avec moins de connexions (Newman, 2016). Cependant, cette mesure dépend fortement de la taille du réseau (Freeman, 1978).

$$Degré(i) = deg_i = \sum_j a_{ij}$$

Où a_{ij} est la valeur de la matrice adjacente aux indices ij.

Dans notre exemple, nous pouvons observer sur la Figure 1.4 que ce sont les nœuds 3 et 5 qui ont le degré le plus élevé avec une valeur de 5 tandis que les nœuds 8 et 6 ont le degré minimal d'une valeur de 2.

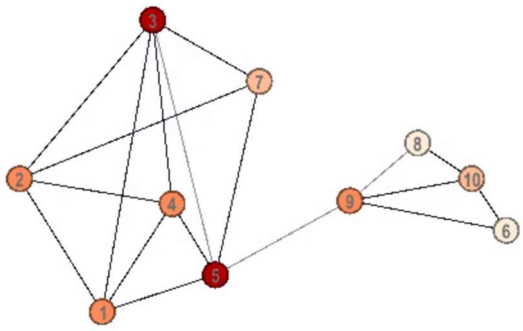

Figure 1.4 - Exemple de graphe

2.1.2. Intermédiarité

Selon cette mesure, un nœud est central au graphe dans la mesure où il se trouve sur le chemin le plus court de paires d'autres nœuds (Freeman, 1977). Autrement dit, l'intermédiarité d'un nœud est égale au nombre de fois qu'un nœud a besoin de ce nœud pour atteindre un autre nœud (Borgatti & Everett, 2005). Cette mesure quantifie l'importance d'un intermédiaire dans un réseau donné (Fouss et al., 2016). Une personne dans un groupe qui est située sur le chemin de communication le plus court reliant des paires d'autres personnes est dans une position centrale. Les autres membres du réseau sont dits « sensibles » à cette personne qui peut influencer le groupe en retenant ou en déformant l'information durant sa transmission (Bavelas, 1948). Ce nœud a donc un contrôle potentiel du flux d'informations dans le réseau (Freeman, 1977).

Dans un réseau social, par exemple, il pourrait y avoir des messages, des nouvelles ou des informations à faire passer d'une personne à l'autre. Faisons l'hypothèse qu'un message prend toujours le chemin le plus court. Les acteurs avec un indice d'intermédiarité supérieur sont ceux qui transmettent le plus de messages (Newman, 2016).

S'il n'y a qu'un seul chemin géodésique reliant n'importe quel nœud à n'importe quel autre, la mesure est égale au nombre de chemins géodésiques parcourant un nœud donné (Borgatti & Everett, 2005). Cependant, si pour une paire de nœuds donnée, il existe plusieurs chemins

géodésiques les reliant alors nous supposons qu'ils sont indifférents au chemin que prendra leur communication (Freeman, 1977). L'intermédiarité est mesurée comme suit ;

$$Intermédiarité(i) = bet_i \triangleq \sum_{\substack{j=1 \\ j \neq i}}^{n} \sum_{\substack{k=1 \\ k \neq i,j}}^{n} \frac{\eta(i \in P_{jk}^*)}{|P_{jk}^*|} = \sum_{\substack{j=1 \\ j \neq i}}^{n} \sum_{\substack{k=1 \\ k \neq i,j}}^{n} \frac{\sum_{\wp_{jk}^* \in P_{jk}^*} \delta(i \in \wp_{jk}^*)}{|P_{jk}^*|}$$

Où P_{jk}^* constitue l'ensemble de chemins le plus court reliant les nœuds j et k ;

$\wp_{jk}^*$ représente la proportion de chemins géodésiques entre le nœud j et le nœud k parcourant le nœud i (avec j différent de i et de k). On peut en déduire que $\wp_{jk}^* \in P_{jk}^*$; $\delta(i \in \wp_{jk}^*)$ est égale à 1 si le chemin le plus court passe par i, à 0 dans le cas contraire (Fouss et al., 2016).

Dans notre exemple, ce sont les nœuds 5 et 9 qui ont la valeur d'intermédiarité la plus élevée. Cela est observable sur la Figure 1.5 où en effet, ces deux nœuds relient deux parties du graphe. Ils appartiennent donc à beaucoup de chemins géodésiques. Contrairement aux nœuds 8 et 6 qui ont une intermédiarité nulle puisqu'ils ne se trouvent entre aucune autre paire de nœuds.

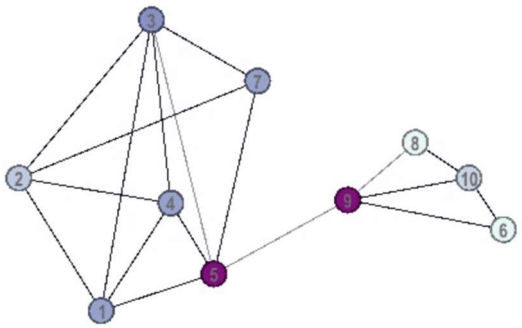

Figure 1.5 - Exemple de graphe

2.1.3. Proximité

Cette mesure calcule à quel point un nœud est proche de tous les autres nœuds du graphe (Freeman, 1977). C'est donc un indicateur de proximité entre un nœud donné et les autres nœuds dans un graphe non dirigé (Fouss et al., 2016). L'indépendance d'un nœud est déterminée par sa proximité vis-à-vis de tous les autres nœuds du graphe (Freeman, 1977). Un nœud avec un grand indice de proximité a un meilleur accès à l'information ou une directe influence sur les autres nœuds. Dans les réseaux sociaux, par exemple, les acteurs avec une distance moyenne aux autres acteurs plus petite atteignent plus rapidement la communauté avec leurs opinions que les acteurs avec une distance moyenne supérieure (Newman, 2016).

La centralité d'un nœud peut être mesurée en additionnant les distances géodésiques entre ce nœud et tous les autres nœuds du graphe (Sabidussi, 1966). Mais cette mesure représenterait plutôt une mesure de périphéricité puisqu'elle augmente quand les nœuds sont plus éloignés (Freeman, 1977). L'inverse de cette addition est une des manières les plus utilisées pour mesurer la proximité d'un nœud. Cependant, cette dernière dépend fortement de la taille du graphe et il est donc judicieux de la normaliser (Fouss et al., 2016). La proximité se mesure comme suit ;

$$Proximité(i) = cc_i = \frac{1}{\frac{1}{n-1}\sum_{j=1}^{n}\triangle_{ij}} = \frac{n-1}{\sum_{j=1}^{n}\triangle_{ij}}$$

Où $\triangle_{ij}$ est la distance du chemin le plus court, mais elle peut être remplacée par n'importe quelle autre mesure de dissimilarité ;

n est égal à la taille du graphe (Fouss et al., 2016).

Dans notre exemple, nous pouvons observer sur la Figure 1.6 que le nœud 5 possède la valeur de proximité la plus élevée puisque ce dernier est relié à chaque point avec des chemins d'une longueur de 1 ou 2. Par contre, les nœuds 8 et 6 ont la valeur de proximité minimale dans le graphe puisqu'ils doivent parcourir 4 nœuds pour atteindre les nœuds 2 et 3 par exemple.

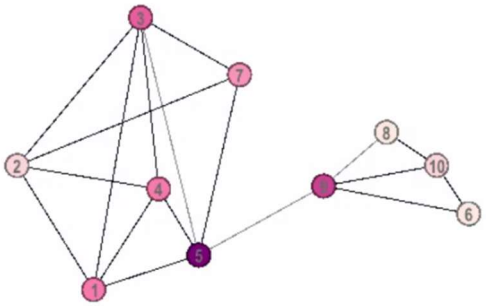

Figure 1.6 - Exemple de graphe

2.1.4. Excentricité

Cette métrique quantifie dans quelle mesure un nœud est distant des autres nœuds et fait référence à la périphéricité d'un nœud (Fouss et al., 2016). L'excentricité d'un nœud est mesurée par le plus long chemin géodésique entre ce nœud et n'importe quel autre nœud (Newman, 2016). Elle synthétise dans quelle mesure un nœud est loin du nœud le plus distant de lui dans le graphe. Cette mesure peut valoir entre 1 et n – 1 (Newman, 2016).

$$Excentricité(i) = ecc_i = \frac{1}{\max_j\{\triangle_{ij}\}}$$

Où $\triangle_{ij}$ est la distance du chemin le plus court (Fouss et al., 2016).

Dans notre exemple, la Figure 1.7 montre les points à la périphérie du graphe. Ce sont donc les nœuds 2, 6, 8 et 10 qui sont les plus périphériques puisqu'ils ont un chemin géodésique à la longueur maximale de 4 tandis que le nœud 5 a un chemin géodésique de longueur 2 au maximum.

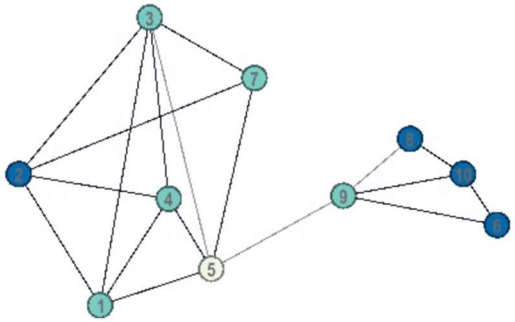

Figure 1.7 - Exemple de graphe

Centralité dans les systèmes de recommandation

Plusieurs études abordent déjà l'utilisation des mesures sur graphe dans le domaine des systèmes de recommandation. Certaines d'entre elles utilisent notamment les chaines de Markov afin de générer des mesures de dissimilarité, utilisées ensuite dans un système de recommandation collaboratif (Fouss, Pirotte, et Saerens, 2005). D'autres auteurs ont étudié l'impact de l'utilisation des réseaux sociaux dans un système de recommandation (Chatterjee & Davoudi, 2015). Deux autres études ont introduit les mesures de centralité dans un système de recommandation, la première pour la recherche d'articles périodiques (Imran, Khattak, Malik, Raza, et Waheed, 2019), l'autre sur des données publiques provenant de la base de données « MovieLens » (Frasincar & van Rossum, 2019).

3.1. Fouss F., Pirotte A. et Saerens M. (2005)

Cette étude présente une nouvelle approche pour quantifier la similarité entre les éléments d'une base de données. Cette approche exploite la structure en graphe de la base de données pour calculer trois similarités différentes. Celles-ci sont les similarités entre les personnes ce qui permet de les regrouper, les similarités entre les personnes et les articles afin de leur en suggérer et les similarités entre les personnes et les catégories d'articles pour les rattacher à la plus pertinente.

Cette approche utilise le trajet aléatoire des chaines de Markov qui décrivent la séquence de nœuds visités par un marcheur aléatoire se promenant dans le graphe. Le trajet aléatoire est une puissante technique utilisée pour analyser les données du réseau (Fouss et al., 2016). Dans ce cadre, on associe un état de la chaine de Markov à chaque nœud du graphe et on définit aussi une variable aléatoire *s(t)* qui représente l'état du modèle de Markov, ainsi si un marcheur aléatoire se trouve dans le nœud *i* au temps *t*, *s(t)* = *i*. Puis, le trajet aléatoire est défini par la probabilité de transition à une étape suivante :

$$p_{ij} = P(s(t+1) = j | s(t) = i) = \frac{a_{ij}}{a_{i.}}, \text{où } a_{i.} = \sum_{j=1}^{n} a_{ij}$$

Où pour chaque état ou nœud *s(t)* = *i*, on détermine la probabilité de passer dans un nœud adjacent *j*, *s(t+1)* = *j* qui est proportionnel au poids a_{ij}. La première propriété des chaines de Markov spécifie que la probabilité de transition dépend seulement de l'état présent et non de ceux passés. Dans cette étude, on suppose que le réseau est pondéré et non dirigé ainsi qu'entièrement connecté ce qui implique que la chaine de Markov est irréductible.

À partir de la définition des chaines de Markov, ou plus précisément de sa matrice de transition, plusieurs mesures peuvent être calculées. Dans ce travail, les auteurs s'intéressent à trois d'entre elles qui sont le temps moyen du premier passage, le temps de trajet moyen et le pseudo-inverse de la matrice laplacienne. La première mesure notée $m(k|i)$ quantifie le nombre moyen d'étapes qu'un marcheur aléatoire, partant d'un nœud i parcoure pour arriver dans le nœud k où $i \neq k$, pour la première fois. La deuxième mesure notée $n(i,j)$ mesure le nombre d'étapes nécessaires à un marcheur aléatoire, partant du nœud i où $i \neq j$, pour entrer dans le nœud j pour la première fois et pour retourner au nœud i, de telle sorte que $n(i,j) = m(j|i) + m(i|j)$. Le pseudo-inverse de la matrice laplacienne est noté $\mathbf{L}^+$. Le concept de pseudo-inverse généralise la notion de matrice inverse aux matrices qui ne sont pas inversibles. La matrice laplacienne est définie comme suit : $\mathbf{L} \triangleq \mathbf{D} - \mathbf{A}$ où $\mathbf{D}$ contient les degrés des nœuds sur sa diagonale (Fouss et al., 2016). $\mathbf{L}^+$ contient les produits intérieurs des vecteurs de nœuds, il peut donc être considéré comme une mesure de similarité entre les nœuds. Toutes ces mesures ont la particularité d'augmenter quand le nombre de chemins connectant deux éléments augmente ou quand la longueur de n'importe quel chemin décroit. Pour résumer, deux éléments sont considérés comme similaires s'ils ont de nombreux chemins les connectant.

Cette étude propose dix différentes méthodes de calcul des notes afin de les comparer. La première utilise l'algorithme de fréquence maximale qui recommande les films les plus populaires à tous les utilisateurs. La deuxième, le temps de trajet moyen qui permet de quantifier la dissimilarité entre les personnes et les articles. La troisième exploite l'analyse des composantes principales. Les deux suivantes utilisent le temps moyen du premier passage comme mesure de dissimilarité entre les personnes et les articles. La sixième se base sur le pseudo-inverse de la matrice laplacienne pour générer les similarités entre les utilisateurs et les articles. Les prochaines méthodes utilisent des approches répandues comme la méthode des plus proches voisins, le cosinus, l'indice de Katz et l'algorithme du chemin le plus court.

Les conclusions de cette recherche montrent que la technique utilisant le pseudo-inverse de la matrice laplacienne fournit les meilleurs résultats suivie de celle utilisant l'indice de Katz. Une des approches exploitant le temps de trajet moyen obtient également de bons résultats. Cette étude conclut donc que ces mesures peuvent être utilisées pour quantifier la similarité et apportent parfois de meilleurs résultats que les approches classiques. Cependant, le modèle de Markov a autant d'états que le nombre d'éléments présents dans la base de données, cette méthode ne s'adapte donc pas bien aux larges bases de données.

3.2. Chatterjee M. et Davoudi A., 2015

Cette recherche tente d'introduire une mesure de confiance entre les différents utilisateurs. En effet, elle part de l'hypothèse qu'un utilisateur fait davantage confiance à ses relations qu'aux autres et que cette confiance varie d'une relation à l'autre. De plus, l'importance de l'opinion d'une relation change au cours du temps. Il est important de souligner que la confiance n'est pas symétrique, en d'autres mots, un utilisateur *A* faisant confiance à un utilisateur *B* ne signifie pas que l'utilisateur *B* fait confiance à *A*. Les auteurs proposent d'utiliser un réseau social en association avec la matrice des notes afin de prédire les notes d'un produit. Le modèle proposé utilise donc les relations de confiance variant dans le temps pour calculer dans quelle mesure une relation est importante afin de pondérer les notes provenant de cette relation.

Pour modéliser l'importance d'une relation, les auteurs utilisent deux mesures de centralité ; le degré introduit ci-dessus (1.1.2.) et la centralité de vecteur propre. Cette dernière mesure l'importance d'un nœud en fonction de l'importance de ses voisins, autrement dit, l'importance d'un nœud du graphe augmente s'il est connecté à d'autres nœuds qui sont eux-mêmes importants (Newman, 2016). C'est une variante de l'algorithme du PageRank (Imran et al., 2019) et est défini comme suit ; $e_i = k_1^{-1} \sum_j a_{ij} e_j$ où k_1^{-1} représente la centralité de vecteur propre la plus élevée du graphe (Newman, 2016). Comme la confiance qui change au cours du temps, la centralité aussi. Pour le degré, les auteurs utilisent la matrice adjacente sans prêter attention à l'importance des relations. Pour la centralité de vecteur propre, elle est mise à jour en utilisant la matrice de confiance actuelle ainsi que la centralité de la période précédente.

Pour mettre en place leur système de recommandation, les auteurs utilisent donc un ensemble dynamique d'utilisateurs avec leurs relations sociales et un ensemble de produits. Ils génèrent donc une matrice de notes à partir des données de base et une matrice de confiance à partir de la matrice adjacente à un temps donné. En se basant sur les informations disponibles au temps *t*, la note d'un utilisateur *i* pour un article *j* au temps *t+1* est calculée comme suit :

$$R_{ij}(t+1) = \lambda \times R_{ij}^{C}(t) + (1-\lambda) \times R_{ij}^{NC}(t) \text{ où } R_{ij}^{C}(t) = \frac{\sum_{l \in N_i} I_{lj}(t) \times R_{lj}(t) \times C_l(t)}{\sum_{l \in N_i} I_{lj}(t) \times C_l(t)}$$

Où $R_{ij}^{C}(t)$ représente la moyenne pondérée des notes de l'article *j* par les relations de l'utilisateur *i* faisant partie de l'ensemble des relations N_i. L'indicateur I_{lj} est égal à 1 si l'utilisateur *l* a noté l'article *j*, 0 sinon et l'indice de centralité est représenté par C_l. $R_{ij}^{NC}(t)$ est

la moyenne des notes par les non-relations de ce même utilisateur. Le facteur social λ pondère l'impact des relations sociales, $0 \leq \lambda \leq 1$.

Dans leurs conclusions, les auteurs ont analysé l'impact du facteur social en faisant varier λ, ainsi un facteur social égal à zéro impliquerait que les contacts sociaux ne sont pas pris en compte tandis qu'avec un facteur social égal à un ne prend en compte que les connections des utilisateurs. Pour évaluer leur méthode, les auteurs ont utilisé la *MAE* (1.2.1.). Cette évaluation montre de meilleurs résultats quand la centralité de vecteur propre est utilisée pour modéliser l'impact social. L'impact social améliore la prédiction des notes jusqu'à un certain point, avec un $\lambda = 0{,}35$ dans cette étude-ci. L'utilisation du facteur de l'impact social conduit donc à une meilleure précision de prédiction. De plus, la méthode utilisée peut être utilisée sur de plus larges bases de données dues à sa complexité linéaire.

3.3. Imran M., Khattak H.A., Malik A.K., Raza B. et Waheed W. (2019)

Dans cette étude, les auteurs ont implémenté un système de recommandation hybride qui utilise les mesures de centralité dans le contexte de la recherche d'articles scientifiques. Les systèmes collaboratifs sont souvent les plus utilisés dans les systèmes de recommandation académiques. Ils recommandent un article en se basant sur la matrice d'article-citation qui montrent les préférences passées des utilisateurs, cette technique peut toutefois souffrir du « cold start » quand aucune donnée ou citation n'est disponible sur les nouveaux articles. Ce problème est souvent résolu avec l'utilisation de système de recommandation basé sur le contenu, mais cette technique ne saisit pas la sémantique des intérêts de l'utilisateur et ne peut pas gérer les ambiguïtés du langage naturel. Une autre méthode utilisée est l'analyse des co-citations, celle-ci suppose que les articles qui se citent soient étroitement liés, faisant d'eux des propositions utiles. Malheureusement, cette méthode ne prend en compte que les citations et non le contenu des articles, des articles cités par un autre, mais non utilisés pour son contenu peuvent donc être repris. Cette recherche propose donc un système de recommandation hybride utilisant un réseau de citations à plusieurs niveaux et un réseau d'auteurs. Les nœuds de ce réseau représentent les articles et les liens entre eux représentent les citations. Ce réseau est dirigé puisqu'une relation représente la citation d'un article vers un autre. L'analyse traditionnelle utilise un réseau à un niveau en examinant uniquement les documents directement liés au document d'intérêt. « Backward » est le nom utilisé pour identifier les relations dirigées vers le document tandis que « forward » est le nom utilisé pour les relations partant du document. Le nombre de niveaux d'un réseau équivaut à la somme des niveaux « backward » et « forward ». La Figure 1.8 montre

un exemple d'un réseau à six niveaux (Kim & Son, 2018). Les chercheurs utilisent un réseau à dix niveaux puisque c'est la taille recommandée comme raisonnable dans la littérature. L'importance de chaque article par rapport à un article d'intérêt est analysée grâce à quatre mesures de centralités ; l'intermédiarité (2.1.2.), la centralité de vecteur propre (3.2.), le degré (2.1.1.) et la proximité (2.1.3.).

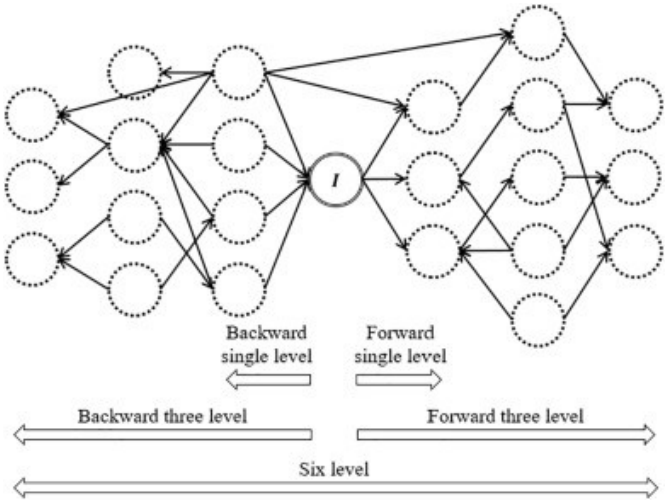

Figure 1.8 - Exemple d'un réseau à six niveaux

La méthodologie utilisée se résume comme suit ; premièrement, un réseau de citations des articles est généré pour l'article d'intérêt avec des relations de type « cite » et « cité par ». L'article d'intérêt a une relation dirigée vers les articles qu'il cite et inversement. Deuxièmement, un score de candidat est mesuré pour chaque article et les articles pertinents sont sélectionnés selon celui-ci. Les articles obtenant un score trop bas sont retirés du réseau. Troisièmement, les mesures de centralités sont calculées pour chaque article, afin d'évaluer leur importance. Celles-ci sont converties en grades. Quatrièmement, un grade moyen par article est calculé et les auteurs des articles les mieux classés sont extraits. Dans la cinquième étape, le réseau de collaboration de ces auteurs est généré où les nœuds représentent les auteurs et les relations entre eux représentent leur collaboration dans l'écriture d'un ou plusieurs articles. Un grade de collaboration est calculé pour chaque auteur grâce aux mesures de centralité comme pour les articles précédemment. Les auteurs les mieux classés sont sélectionnés. Enfin, les dix articles les mieux classés publiés par les auteurs les mieux classés sont recommandés à l'utilisateur, recommander plus d'articles pourrait embrouiller l'utilisateur.

Les mesures de centralité utilisées sont l'intermédiarité, la centralité de vecteur propre, le degré et la proximité. Pour le calcul du degré des articles, c'est le degré entrant qui est utilisé puisque ce graphe est dirigé. Ces mesures ont été définies plus haut, le degré entrant se calcule de la

même manière que le degré en ne prenant en compte que les relations dirigées vers l'article d'intérêt. Ces mesures sont générées pour le réseau des articles et des auteurs. Ainsi un article avec un score de degré entrant élevé est considéré comme plus influent et le degré d'un auteur fait référence à la fréquence de citation de celui-ci. Un article avec un grand score de proximité est vu comme central et important puisqu'il est lié à beaucoup d'autres articles et un auteur est considéré comme un auteur clé quand il a un score de proximité élevé. Un article ou un auteur possédant un indice d'intermédiarité élevé connecte des recherches provenant de deux sous-réseaux différents. Enfin, un auteur ou un article peut être vu comme pertinent quand il est cité par un article influent, et ayant donc une centralité de vecteur propre élevée.

Après l'expérimentation et l'évaluation de cette méthode comparée aux méthodes classiques. Les résultats montrent que l'approche proposée surpasse les autres méthodes existantes. En effet, la méthode proposée obtient un meilleur nDCG (1.2.1.) que les approches utilisées par Google Scholar et MSCN. Google Scholar recommande les articles en se basant seulement sur le nombre de citations, tandis que l'approche MSCN génère un réseau d'articles et recommande les articles les plus similaires à l'article d'intérêt. La méthode utilisée dans ce travail surmonte les limitations en utilisant une mesure additionnelle pour trouver les auteurs clés. Elle recommande des articles de haute qualité indépendamment du nombre de citations ou de la date de publication du papier d'intérêt.

3.4. Frasincar F. et Van Rossum B. (2019)

Dans ce travail, les auteurs ont étudié l'incorporation des caractéristiques de la théorie des graphes dans un système de recommandation se basant sur des données ouvertes. Ils proposent un système de recommandation hybride qui se base sur la centralité des nœuds. Derrière cette proposition, ils supposent que les articles faisant partie des nœuds les plus populaires dans le réseau saisissent probablement moins les préférences uniques d'un utilisateur et devraient donc moins contribuer au calcul de son score. Ils utilisent les mesures de centralité puisque ces dernières offrent différentes manières de mesurer la popularité d'un nœud, elles conviennent donc pour effectuer cette correction. Pour réaliser leurs expérimentations, ils ont utilisé la base de données MovieLens qui contient 1 000 209 notations sur 3 883 films par 6 040 utilisateurs. Après certaines modifications, les auteurs ont finalement expérimenté leur méthodologie sur une base de données comprenant 193 255 notations sur 3 196 films par 3 854 utilisateurs.

La méthodologie proposée requiert donc la construction d'un graphe, $G = (V, A)$ où V représente les nœuds et A les liens entre eux. Le graphe est composé de différents types de nœuds, dans ce cas, $V = I \cup U \cup E$, où I est l'ensemble des films, U l'ensemble des utilisateurs et E l'ensemble reprenant les acteurs, directeurs, sujets et toutes autres entités liées aux films. Le graphe est non dirigé puisqu'il s'intéresse aux associations entre les nœuds et non aux potentiels liens de causalité. Les liens se font uniquement entre un article et un utilisateur ou entre un article et une entité. Les auteurs introduisent la notion de méta-chemin qui spécifie le type de lien connectant les nœuds de deux types différents. L'ensemble des types de lien est noté T. Ils s'intéressent tout particulièrement aux méta-chemins ayant une longueur de 3 entre les articles, ainsi $P = V_1 \xrightarrow{t} V_2 \xrightarrow{t} V_3$ où $V_1, V_3 \in I$. Ces méta-chemins relient tous les films qui ont quelque chose en commun, par exemple, pour les films dirigés par le même réalisateur, ils fixent $V_2 \in E$ et $t = réalisateur$.

Pour chaque paire d'article – utilisateur, il est possible de construire un vecteur de caractéristique noté $x_{u,i} \in \mathbb{R}^{|T|}$ et défini comme suit ;

$$x_{u,i}(k) = \sum_{j \in I_u^+} \#path_{i,j}(k), \qquad k = 1, \ldots, |T|$$

Où I_u^+ est l'ensemble des articles liés à l'utilisateur u et $\#path_{i,j}(k)$ indique le nombre de méta-chemins $P(k)$ liant les nœuds i et j. L'intuition est que l'utilisateur u sera plus intéressé par un article i s'il existe plus de méta-chemins le liant à des articles qu'il a aimés précédemment, autrement dit, quand les éléments de ce vecteur ont une plus grande ampleur.

Ensuite, les chercheurs calculent le degré de chaque nœud. Celui-ci est calculé en considérant seulement les liens d'un certain type en fonction du méta-chemin d'intérêt. Ainsi, il est défini comme suit ;

$$C_D(v, P) = deg(v, P)$$

Où $deg(v, P)$ est le degré d'un nœud v dans un réseau où seulement les méta-chemins de type P sont autorisés. L'objectif est de pondérer chaque chemin entre un utilisateur et un article par l'inverse de la centralité de degré des nœuds sur ce chemin. Ainsi, les nœuds les plus populaires possédant un degré plus élevé auront un impact diminué sur ce chemin. Les auteurs se sont penchés sur deux procédures différentes pour ce faire. La première procédure normalise le vecteur de caractéristiques comme suit ;

$$x_{u,i}(k) = \sum_{j \in I_u^+} \#path_{i,j}^*(k), \qquad k = 1, \ldots, |T|$$

Où $$\#path_{i,j}^*(k) = \sum_{p \in P(k):i,s,j \in p} \frac{\bar{C}_D(i,p)}{C_D(i,p)} \times \frac{\bar{C}_D(s,p)}{C_D(s,p)} \times \frac{\bar{C}_D(j,p)}{C_D(j,p)}$$

Où $\bar{C}_D(i,p)$ est le degré C_D moyen sur tous les nœuds de type i. Les nœuds avec une grande centralité relative contribueront de manière moindre au vecteur de caractéristiques puisqu'ils sont multipliés par un facteur de pondération inférieur à 1. La deuxième procédure effectue une normalisation rendant le vecteur moins sensible aux données aberrantes, tout en diminuant l'impact des nœuds avec une grande centralité.

$$\#path_{i,j}^*(k) = \sum_{p \in P(k):i,s,j \in p} \frac{1}{3} \left(\frac{\bar{C}_D(i,p)}{C_D(i,p)} \times \frac{\bar{C}_D(s,p)}{C_D(s,p)} \times \frac{\bar{C}_D(j,p)}{C_D(j,p)} \right)$$

Pour les deux procédures, le vecteur de caractéristiques est converti en un score indiquant la pertinence attendue d'un article pour un utilisateur. Ils utilisent la technique de classification dite « Random Forest » pour entrainer la fonction de conversion. Les auteurs ont utilisé la métrique dite Recall (1.2.1.) pour évaluer leur système. La méthode standard utilise un graphe incluant les connexions entre les utilisateurs, les films et les informations sémantiques pertinentes. Elle inclut ensuite le nombre de connexions entre un utilisateur et un article dans un système de recommandation hybride afin de déterminer la probabilité que cet article soit recommandé à cet utilisateur. Les résultats de cette étude montrent que la première procédure ne surpasse pas la méthode standard, cependant la deuxième procédure performe aussi bien ou mieux que la méthode standard. De plus, les auteurs Adomavicius G. et Kwon Y. ont démontré que recommander des articles moins populaires augmente la diversité dans l'ensemble des articles recommandés. Cette méthodologie maintient donc la précision de prédiction tout en augmentant la diversité des articles recommandés puisqu'ils omettent les articles populaires.

3.5. Conclusion sur ces études

Toutes ces études utilisent les mesures sur graphe pour augmenter l'efficacité des systèmes de recommandation. Fouss, Pirotte et Saerens ont utilisé les chaines de Markov sur graphe afin d'en déduire un indice de similarité contrairement aux autres études qui utilisent toutes des mesures de centralité, en particulier la mesure du degré. Fouss, Pirotte et Saerens intègrent donc des mesures du domaine des graphes au niveau du calcul de l'indice de similarité dans leur

système de recommandation. Chatterjee et Davoudi souhaitent intégrer une notion de confiance à l'aide d'un réseau social. Ils utilisent le degré et la centralité de vecteur propre comme mesure de confiance entre les utilisateurs. Ces mesures seront utilisées au moment du calcul de la note prédite par le système de recommandation. Ces deux premières études réalisent donc un système de recommandation à filtrage collaboratif quand les deux suivantes utilisent des systèmes de recommandation hybrides. Ces systèmes utilisent plusieurs sources de données et combinent le pouvoir algorithmique de différents systèmes de recommandations (Aggarwal, 2016). Imran, Khattak, Malik, Raza et Waheed intègrent quatre mesures de centralité. Ces dernières leur permettent de calculer un score utilisé ensuite pour repérer les articles ou les auteurs par rapport à un article d'intérêt. C'est la seule étude qui crée un graphe de type dirigé. En effet, les trois autres études utilisent un graphe non dirigé pour calculer leurs mesures. Frasincar et van Rossum utilisent les mesures de centralité pour pénaliser les articles populaires afin de capturer au mieux les intérêts personnels d'un utilisateur. Ils utilisent uniquement le degré pour estimer cette popularité. Les articles dits populaires auront une probabilité réduite d'être choisis pour un utilisateur.

Toutes ces études intègrent donc des caractéristiques graphiques dans un système de recommandation. Elles utilisent toutes des mesures de précision pour évaluer leur système de recommandation. L'étude de Chatterjee et Davoudi est la seule à utiliser une mesure de précision de prédiction, la *MAE*. Tandis que les autres études utilisent des mesures de précision de la liste de recommandations. La conclusion de toutes ces études est que l'intégration des mesures sur graphe améliore la précision.

Conclusion

Dans ce chapitre, nous avons établi une revue littéraire en parcourant les points de vue et recherches de plusieurs auteurs. Nous avons ainsi découvert les systèmes de recommandation qui sont des systèmes de filtre d'information et leur fonctionnement. En particulier, celui du filtrage collaboratif qui se base sur les similarités existantes entre les utilisateurs ou les articles pour prédire des notations.

Ensuite, nous avons vu que la diversité et la nouveauté étaient des critères de plus en plus observés dans l'évaluation des systèmes de recommandations en plus de la précision. Nous avons présenté plusieurs métriques pour procéder à cette évaluation. Enfin, nous avons appris que les systèmes de recommandations augmentaient le nombre d'articles achetés et le temps passé sur un site de e-commerce. Ils réduisent le temps de recherche des utilisateurs.

À la suite de cela, nous avons abordé la théorie des graphes qui se composent de nœuds et de liens. Ceux-ci peuvent être utilisés pour représenter des réseaux de communication. Plusieurs mesures de centralités peuvent être calculées à partir d'un graphe, elles permettent d'identifier des nœuds qui ont une position centrale.

Enfin, nous avons parcouru plusieurs études intégrant des caractéristiques graphiques pour améliorer la performance des systèmes de recommandation. Nous avons vu que ces quatre études augmentent la précision des recommandations. Celles-ci intègrent des mesures sur graphe à différents moments de la construction des systèmes. Certains les utilisent directement dans le système comme mesure de similarité ou dans le calcul du score, d'autres les intègrent dans un système de recommandation hybride.

Chapitre 2 : Méthodologie

Introduction

Dans ce chapitre, nous allons présenter la méthodologie qui sera appliquée pour réaliser notre étude. L'objectif de ce travail est de mesurer l'impact sur la diversité et sur la nouveauté des recommandations d'un système de recommandation à la suite de l'intégration de mesures de centralité sur graphe.

À cette fin, nous utiliserons le langage de programmation Python ainsi que plusieurs librairies disponibles sur ce langage. Le logiciel Gephi, nous sera aussi d'une grande utilité. Nous expliquerons les outils utilisés dans la section prévue à cet effet.

Ensuite, nous présenterons la base de données qui sera utilisée dans ce travail. Cette base de données provient du site MovieLens et reprend les notes obtenues par plusieurs utilisateurs sur différents films.

À la suite de cela, nous expliquerons la construction de nos différents algorithmes. Tout d'abord, nous décrirons notre méthodologie pour développer nos deux systèmes de recommandation de référence, un basé sur les utilisateurs et un basé sur les articles. Ces derniers sont construits de manière classique en utilisant la mesure du cosinus pour la sélection du voisinage.

Ensuite, nous présenterons nos quatre systèmes de recommandation intégrant des mesures de centralité sur graphe. Deux d'entre eux sont basés sur les utilisateurs tandis que les autres sont basés sur les articles. Ils intègrent soit la mesure de proximité, soit la mesure d'intermédiarité. Ces mesures peuvent être intégrées au moment du calcul de similarité, de la sélection du voisinage ou du calcul du score d'intérêt.

Enfin, nous sélectionnerons les mesures d'évaluation qui seront utilisées pour mesurer la performance de nos différents systèmes. Nous utiliserons des mesures de précision des listes de recommandations, des mesures de nouveauté et des mesures de diversité.

1. Outils utilisés

Dans ce travail, nous utiliserons le langage de programmation Python pour réaliser toutes nos expérimentations. Afin d'utiliser ce langage, le logiciel PyCharm sera nécessaire. Ce logiciel est l'éditeur de code Python de l'entreprise JetBrain. Nous allons utiliser la version gratuite dite « Community » qui sert au développement pur en Python (JetBrain, 2021). Pour la bonne utilisation de celui-ci, cette version ainsi que la version Python 3.7 seront installées en local sur un ordinateur.

Lors de la création et l'exécution de nos algorithmes, nous ferons appel à plusieurs librairies disponibles avec le langage Python. Tout d'abord, la librairie SciPy sera utilisée pour créer des matrices dites « sparse », des matrices creuses en français. Ce type de matrice contient plus d'éléments égaux à zéro que d'éléments différents de zéro. Cette librairie nous permet donc d'utiliser moins de mémoire et d'économiser du temps de calcul (Gupta, 2021). Ensuite, pour le stockage de nos données pendant nos manipulations, nous utiliserons des fichiers de type JSON, JavaScript Object Notation. Ces fichiers sont sous un format léger d'échange de données, en effet, ce format est facilement lisible par les humains et facilement analysé par les machines (JSON, 2021). La librairie Json prenant en charge ce type de données nous sera donc d'une grande utilité. De plus, la base de données utilisée qui sera présentée ci-après est stockée en format CSV, la librairie « CSV » nous permettra d'extraire les données de notre fichier de manière optimale. Enfin, la dernière librairie utilisée sera scikit-learn, un outil simple et efficace pour l'apprentissage machine en Python (sklearn, 2021). Dans ce travail, le sous-module sklearn.metrics.pairwise nous permettra de calculer une mesure de similarité.

Afin de visualiser notre graphique, nous utiliserons le logiciel Gephi. Ce dernier est le leader pour la visualisation et l'exploration de tout type de graphes et de réseaux. Il permet de visualiser de grands graphes en 3D et en temps réel. C'est un logiciel open source gratuit d'utilisation (Gephi, 2021).

2. Base de données

Afin de réaliser nos expérimentations, nous avons besoin d'une base de données reprenant l'avis d'utilisateurs sur des articles. Pour ce faire, nous utiliserons deux bases de données libres « MovieLens » reprenant soit 610 utilisateurs et 9.724 films soit 943 utilisateurs et 1682 films. Ces bases de données sont composées de plusieurs fichiers dont un reprenant les notes attribuées à un film par un utilisateur, ce fichier est nommé « ratings.csv ». Il est stocké sous le format CSV, « Comma-separated values » qui présente les données tabulaires sous forme de valeurs séparées par des virgules (Wikipédia, 2021). Chaque ligne de ce fichier contient quatre éléments ; « userId » qui reprend le numéro d'identifiant de l'utilisateur, « movieId » reprenant celui du film, « rating » qui contient la note obtenue par cet utilisateur pour ce film. Ces notes peuvent être comprises entre 0,5 et 5 par pas de 0,5. Le dernier élément, « timestamp », contient une information temporelle sur la donnée en secondes, ainsi la valeur contenue dans cet élément équivaut au nombre de secondes écoulées depuis le 1er janvier 1970 à minuit. Par conséquent, sur la Figure 2.1, la première ligne du fichier nous indique que l'utilisateur avec l'identifiant 1 a regardé le film avec 1 comme identifiant et lui a donné la note de 4, le 30 juillet 2000.

```
userId,movieId,rating,timestamp
1,1,4.0,964982703
1,3,4.0,964981247
1,6,4.0,964982224
1,47,5.0,964983815
1,50,5.0,964982931
1,70,3.0,964982400
1,101,5.0,964980868
1,110,4.0,964982176
1,151,5.0,964984041
```

Figure 2.1 - Exemple de données

Ce fichier contient 100.832 notations avec en moyenne 165 notations par utilisateur. Cependant, 7.455 films n'atteignent pas le nombre de 10 notations. Dans ce travail, nous avons donc fait le choix de travailler uniquement avec les films possédants au moins 10 notations. Cela réduit notre première base de données à 2.269 films avec toujours 610 utilisateurs. La deuxième base de données contient 100.000 notations et nous travaillerons avec 1.152 films.

Ce fichier nous servira pour créer notre matrice de notations ainsi que notre matrice adjacente. Nous avons donc à notre disposition des notations de feedback explicite de la part des utilisateurs puisque ces derniers ont donné leur intérêt concernant un article de façon explicite (Isinkaye et al., 2015). Les notations de cette base de données sont basées sur des intervalles et

varient entre 0,5 et 5 (Aggarwal, 2016). Ces données nous permettront d'obtenir la matrice de notations reprenant les notes données par les différents utilisateurs pour les différents films. Une matrice de notations est parfois mentionnée sous le nom de matrice d'utilités, cependant les deux peuvent ne pas renvoyer à la même chose. En effet, il est possible de transformer les notations en des valeurs d'utilité en fonction de l'application. Nous disposons donc du choix entre trois formes différentes de matrice de notations, ces dernières sont présentées dans la Figure 2.2 :

1. La première version reprend les notes telles qu'elles ont été données par les utilisateurs. Les valeurs manquantes de la matrice seront remplacées par 0.
2. Les deux autres versions de la matrice de notations convertissent les données en valeur binaire et contiennent donc uniquement des 1 et des 0 :
    a. La première version notifie les films vus pour l'utilisateur c'est-à-dire qu'un film obtiendra la note de 1 à partir du moment où ce film a été noté par l'utilisateur. Dans le cas contraire, il aura la note de 0.
    b. La deuxième notifie les films appréciés par l'utilisateur. C'est-à-dire qu'un film ayant obtenu une note supérieure ou égale à 3 aura une valeur de 1 dans la matrice. Dans le cas contraire, il aura une note de 0.

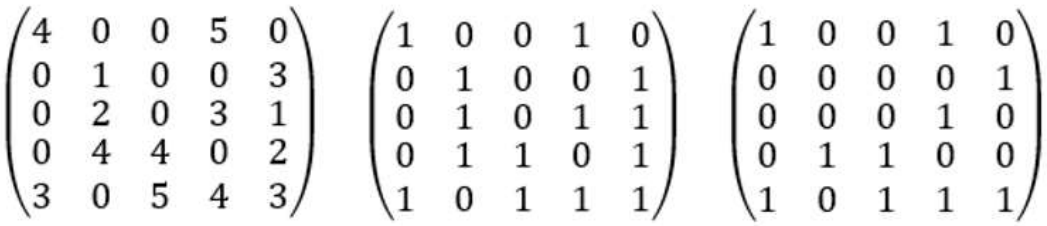

Figure 2.2(a) Figure 2.2(b) Figure 2.2(c)

Figure 2.2 - Matrices de notations

Selon Aggarwal (2016), utiliser une matrice binaire plutôt qu'une matrice avec les notations exactes présente l'avantage de réduire le biais dû à l'incorporation de 0 pour remplacer les valeurs manquantes. Ce biais est donc réduit avec l'utilisation d'une matrice binaire. De plus, dans notre système de recommandation, nous nous intéresserons davantage à la précision des listes de recommandations et non à la précision des prédictions des notations. Nous utiliserons donc une matrice binaire. Il reste à faire un choix entre les deux options ci-dessus ; la matrice reprenant les films vus par l'utilisateur (cf. Figure 2.2(b)) ou la matrice notifiant les films appréciés par l'utilisateur (cf. Figure 2.2(a)). L'utilisation de la matrice binaire reprenant les

films appréciés présente l'inconvénient d'ignorer les films vus et non appréciés par l'utilisateur. Dans ce cas, le système pourrait recommander des articles connus et non appréciés de l'utilisateur. Cela engendrerait une expérience négative pour ce dernier. Dans notre système, nous utiliserons donc la matrice de notations binaire reprenant les films vus (cf. Figure 2.2(b)) par les utilisateurs. Cette dernière nous servira également pour la création de notre matrice adjacente qui permettra la constitution de notre graphe. La matrice binaire est composée à 95% et 90% de 0.

3. Construction du graphe

L'objectif de ce travail est d'analyser l'impact de l'incorporation de mesures de centralité du domaine des graphes dans un système de recommandation. Dans cette partie, nous décrirons la construction du graphe utilisé.

Tout d'abord, au vu de la base de données utilisée qui reprend des films et des utilisateurs, nous construirons un graphe biparti, composé de deux types de nœuds distincts ; les films et les utilisateurs. Le graphe nommé $G = (V, A)$ où V représente les nœuds et A les liens entre eux. Le graphe est composé de différents types de nœuds, dans ce cas, $V = I \cup U$, où I est l'ensemble des films, U l'ensemble des utilisateurs. Les liens A sont non dirigés et se font uniquement entre un film et un utilisateur. Ensuite, comme mentionnée dans la partie précédente, notre matrice adjacente contiendra uniquement des valeurs binaires où la valeur de 1 signifiera l'existence d'un lien entre un film et un utilisateur. Un lien sera donc créé entre deux nœuds si et seulement si cet utilisateur a noté le film en question et cela indifféremment de la valeur de la note obtenue. La matrice construite à partir du fichier reprendra les utilisateurs et les films vus par ceux-ci (cf. Figure 2.2(b)). Cette matrice est appelée la matrice d'incidence, celle-ci est de forme rectangulaire de longueur n x p où n est le nombre de participants au réseau, dans notre cas le nombre d'utilisateurs, et p est le nombre de groupes, les films (Newman, 2016).

La matrice adjacente d'un graphe biparti prend la forme suivante ;

$$A = \begin{pmatrix} 0_{u,u} & W \\ W^T & 0_{i,i} \end{pmatrix}$$

Où W est la matrice binaire reprenant les films vus par les utilisateurs (cf. Figure 2.2(b)) ;
$0_{u,u}$ représente une matrice carrée de zéros avec pour longueur le nombre d'utilisateurs ;
$0_{i,i}$ représente une matrice carrée de zéros avec pour longueur le nombre de films ;

Dans ce cas, seulement la matrice W représente le graphe puisque le reste de la matrice adjacente peut être vu comme redondant. La matrice W est parfois appelée la « matrice biadjacente » (Fouss et al., 2016). Dans notre exemple, la matrice binaire (cf. Figure 2.2(b)) nous servira de matrice biadjacente. Cette matrice prendra alors la forme suivante ;

$$\begin{pmatrix} 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 1 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 1 & 1 & 1 \\ 1 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 1 & 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & 1 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 1 & 1 & 1 & 0 & 0 & 0 & 0 & 0 \end{pmatrix}$$

Figure 2.3 - Exemple de matrice adjacente d'un graphe biparti

À la suite de cela, nous pouvons construire le graphe biparti de notre exemple. Ce dernier est composé de nœuds de type film et de type utilisateur. Le premier type est représenté en vert sur la Figure 2.4 tandis que le deuxième type est en bleu.

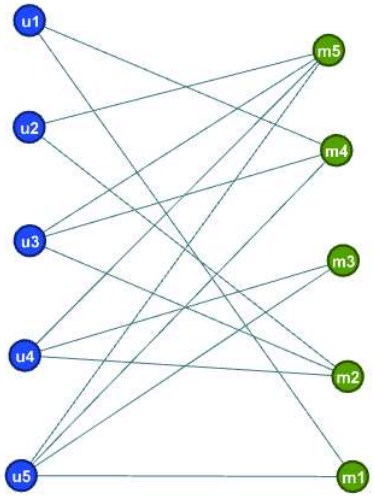

Figure 2.4 - Exemple de graphe biparti

3.1. Mesures de centralité

Dans la revue littéraire, nous avons introduit plusieurs mesures de centralité sur graphe telles que le degré, l'intermédiarité, la proximité, et l'excentricité. Dans ce travail, nous utiliserons les mesures d'intermédiarité et de proximité. Ces mesures peuvent en effet être utilisées pour donner de l'importance à un nœud. Elles ont d'ailleurs été introduites par Bavelas en 1948 pour expliquer la communication dans des groupes d'humains en hypothéquant une relation entre l'influence dans les groupes et la structure des graphes (Freeman, 1978). Les mesures de centralité sont importantes parce qu'elles indiquent quels nœuds ont une position critique dans

un réseau. Une position centrale est souvent synonyme d'une gouvernance remarquable, une bonne popularité ou d'une excellente réputation dans le réseau. Quand un acteur du réseau atteint une position plus centrale, ce dernier devient plus proche du centre du réseau et obtient plus de pouvoir et d'influence (Luo & Zhang, 2017)

La mesure d'intermédiarité peut être utilisée pour identifier les utilisateurs ou les films faisant office de bons intermédiaires. Ces nœuds sont positionnés de manière stratégique sur plusieurs chemins de communication liant d'autres nœuds (Freeman, 1978). Une personne dans cette position est donc centrale au graphe et les autres membres du réseau sont dits sensibles à celle-ci (Bavelas, 1948). Cohn et Marriott soulignaient que ces nœuds étaient centraux puisqu'ils liaient potentiellement le réseau ensemble en coordonnant les activités des autres nœuds (Freeman, 1977). Dans ce mémoire, nous partons de l'hypothèse qu'un nœud ayant un indice d'intermédiarité relie potentiellement plusieurs groupes de nœuds entre eux. D'une part, s'il s'agit d'utilisateurs, ces derniers relient donc des groupes d'utilisateurs ayant des goûts similaires. Privilégier ces utilisateurs intermédiaires pourrait donc ajouter plus de diversité dans les recommandations puisque par hypothèse, ils relient des groupes avec des goûts différents. Ils apporteraient donc de la diversité dans les films proposés. D'autre part, s'il s'agit d'articles, ceux-ci relient également des groupes d'articles ayant beaucoup d'interactions entre eux donc potentiellement similaires. Privilégier ces articles peut donc accroitre la diversité des listes de recommandations puisque par définition, ils sont positionnés entre plusieurs groupes d'articles.

Dans notre exemple, si nous calculons les valeurs d'intermédiarité de notre graphe biparti, nous pouvons observer sur la Figure 2.5 que les nœuds représentant l'utilisateur 1 et le film 5 obtiennent des valeurs supérieures. Ces deux nœuds sont donc de bons intermédiaires dans notre exemple et se trouvent entre multiples paires de nœuds.

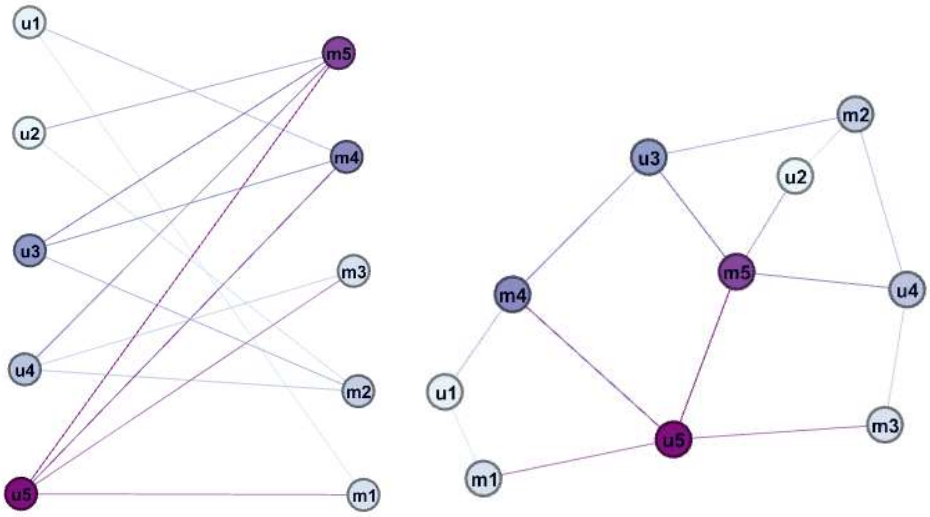

Figure 2.5 - Exemple d'intermédiarité

La mesure de proximité permet d'identifier les utilisateurs et les films relativement proches des autres. Dans la littérature, cette mesure est perçue comme une mesure d'indépendance dans la communication. En effet, des personnes dans cette position ont besoin de peu d'intermédiaires pour faire passer des messages (Hu, Li, Zhang et Ma, 2015). Freeman recommande d'utiliser la mesure de proximité pour estimer le niveau d'efficacité et de commodité (Luo & Zhang, 2017). D'une part, s'il s'agit des utilisateurs, ceux possédant un indice de proximité élevé sont donc plus proches des autres. Ils sont par hypothèse plus populaires puisqu'ils sont centraux au réseau. D'autre part, s'il s'agit des articles, ceux possédant un indice de proximité élevé sont plus proches des autres nœuds. Par hypothèse, ces articles sont plus connus des utilisateurs et donc plus populaires. Ceux-ci apporteraient donc moins de nouveauté aux listes de recommandations au vu de leur popularité. De plus, ils captureraient de manière moindre les préférences individuelles des utilisateurs.

Dans notre exemple, si nous calculons cette mesure de proximité sur tous nos nœuds de notre graphe biparti, nous observons sur la Figure 2.6 que les nœuds représentant l'utilisateur 5 et le film 5 obtiennent de nouveau les scores les plus élevés. Cependant, contrairement aux mesures d'intermédiarité, plusieurs nœuds obtiennent des scores égaux.

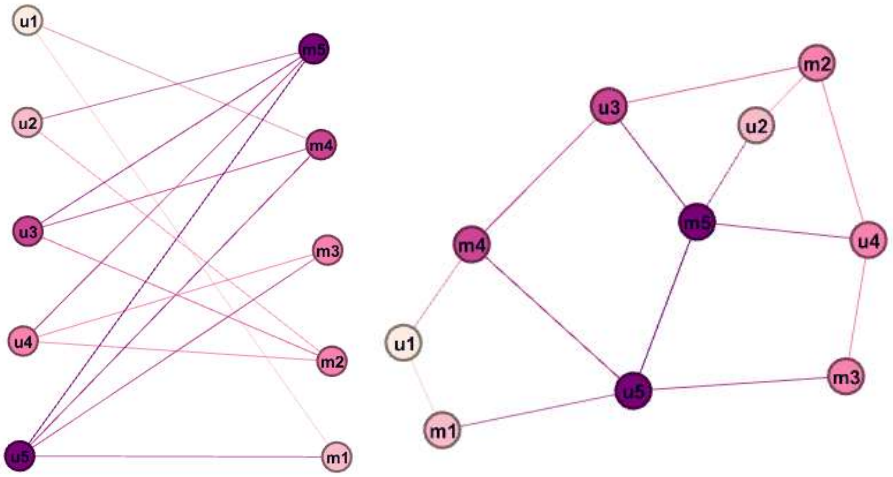

Figure 2.6 - Exemple de proximité

Dans la pratique de nos expérimentations, nous passerons par un algorithme pour calculer ces différentes mesures. La mesure d'intermédiarité sera calculée à l'aide de l'algorithme de Brandes (Annexe 1). Cet algorithme est une manière efficace de calculer le plus court chemin entre tous les nœuds du graphe. Il se base sur l'observation suivante ; pour un nœud fixe *i*, le nombre fois que les autres nœuds apparaissent sur le plus court chemin de *i* vers tous les autres nœuds peut être calculé de manière récursive (Fouss et al., 2016). Ce même algorithme sera utilisé pour le calcul de la proximité de tous les nœuds du graphe excepté que dans ce cas-ci, seulement un plus court chemin sera calculé entre tous les nœuds du graphe puisque seulement la longueur du chemin nous intéresse.

4. Systèmes de référence

Dans la revue littéraire, nous avons vu la méthode traditionnelle pour réaliser un système de recommandation à filtrage collaboratif. Nous avons aussi vu qu'il existait des systèmes basés sur les utilisateurs ainsi que des systèmes basés sur les articles. L'objectif de ce travail est de quantifier l'impact de l'incorporation des mesures de centralité sur graphe dans un système de recommandation et en particulier, son impact sur la diversité et la nouveauté des articles proposés à l'utilisateur. L'intérêt de cette section est de construire deux systèmes de recommandation à filtrage collaboratif afin de pouvoir comparer notre approche et la méthode traditionnelle.

4.1. Algorithme de filtrage collaboratif basé sur les utilisateurs

Comme explicité précédemment, un système de recommandation de filtrage collaboratif basé sur les utilisateurs calcule, à l'aide d'une mesure de similarité, les utilisateurs avec des intérêts similaires à un utilisateur cible. Dans la revue littéraire, nous avons défini deux mesures de similarité différentes, le coefficient de corrélation de Pearson et le cosinus. La particularité du coefficient de corrélation de Pearson par rapport au cosinus est qu'il normalise les notations sur leur moyenne. Par cette normalisation, il réduit le biais provenant des échelles de notations distinctes d'un utilisateur à un autre. Dans notre travail, nous travaillons avec une matrice de notations binaire, cette normalisation est donc inutile. Nous utiliserons donc le cosinus comme mesure de similarité.

La première étape est donc de calculer le cosinus entre les différents utilisateurs, pour rappel, cette mesure se calcule comme suit ;

$$Sim(u,v) = Cosinus(u,v) = \frac{\sum_{i \in I_u \cap I_v} r_{ui} \times r_{vi}}{\sqrt{\sum_{i \in I_u} {r_{ui}}^2} \times \sqrt{\sum_{i \in I_v} {r_{vi}}^2}}$$

Où I_u représente l'ensemble des notations connues pour l'utilisateur u ;
r_{ui} est égal à la notation de l'utilisateur u pour l'article i ;
I_v représente l'ensemble des notations connues pour l'utilisateur v ;
r_{vi} est égal à la notation de l'utilisateur v pour l'article i.

Ainsi, si nous reprenons notre matrice de notations (cf. Figure 2.2(b)), nous obtenons les mesures suivantes de similarité entre les utilisateurs ;

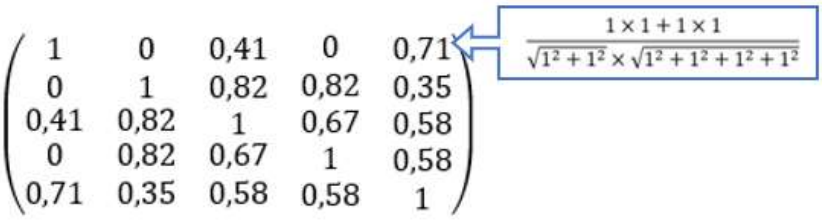

Figure 2.7 - Calcul du cosinus entre utilisateurs

La deuxième étape est de constituer l'ensemble des *k* plus proches voisins pour un utilisateur afin de lui faire des recommandations en fonction des notes de ses voisins. Dans notre exemple, si nous constituons un voisinage de 2 voisins, les deux plus proches voisins de l'utilisateur 1 seront les utilisateurs 3 et 5, ces derniers seront donc considérés comme les utilisateurs avec les intérêts les plus proches de notre utilisateur cible. Pour procéder aux recommandations de cet utilisateur 1, nous calculerons l'intérêt de celui-ci pour un article grâce aux notes obtenues pour ce même article par ses voisins. L'intérêt se calcule donc comme suit ;

$$Score(u,i) = \frac{\sum_{v \in N} r_{vi} \times w_{uv}}{\sum_{v \in N} w_{uv}}$$

Où *N* représente l'ensemble des voisins de l'utilisateur *u* ;

r_{vi} est égal à la note obtenue pour l'article *i* par l'utilisateur *v* ;

w_{uv} est égal à la similarité calculée précédemment entre l'utilisateur *u* et l'utilisateur *v*.

Dans notre exemple, si nous calculons les différents scores d'intérêt par nos cinq films pour les cinq utilisateurs, nous obtiendrons les scores suivants ;

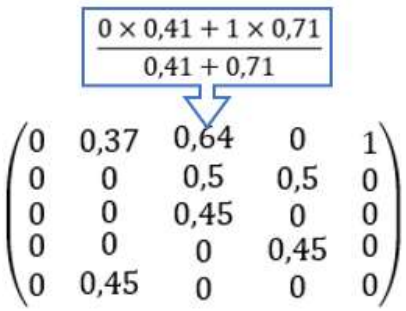

Figure 2.8 - Exemple de score d'intérêt

Pour effectuer nos recommandations, nous privilégions ainsi les articles ayant obtenu un score d'intérêt supérieur. Dans notre exemple, si nous devions recommander deux films à nos utilisateurs, nous sélectionnerions les deux films avec les plus hauts scores. Pour notre utilisateur 1 par exemple, nous lui recommanderions le film 5 suivi du film 3. En effet, le

premier a été vu par ses deux plus proches voisins et le deuxième a été vu par son plus proche voisin. Nous ne pourrions pas recommander les films 1 et 4 puisque ces deux derniers ont déjà été vus par notre utilisateur, ils obtiennent ainsi directement le score de 0. Enfin, le film 2 obtient un score moins élevé puisqu'il a été vu seulement par le deuxième voisin le plus proche.

Utilisateurs	Films vus
1	1, 4
2	2, 5
3	2, 4, 5
4	2, 3, 5
5	1, 3, 4, 5

Utilisateurs	Films recommandés
1	5, 3
2	3, 4
3	3, 1
4	4, 1
5	2

Figure 2.9 - Exemple de recommandations

4.2. Algorithme de filtrage collaboratif basé sur les articles

Un système de recommandation de filtrage collaboratif basé sur les articles calcule, à l'aide d'une mesure de similarité, les articles similaires à un article d'intérêt. Comme dans la section précédente, nous utiliserons le cosinus comme mesure de similarité. Les mêmes raisons nous poussent à choisir cette mesure contre le coefficient de corrélation de Pearson. De plus, pour la similarité entre articles, le biais d'échelle de cotation est faible puisque les articles sont cotés de la même manière par le même utilisateur. Ils ne souffrent donc pas d'échelles de cotation distinctes d'un article à l'autre. Pour construire ce système de recommandation, nous calculerons donc un indice de similarité entre les différents articles afin d'identifier les plus proches voisins d'un article d'intérêt. Le calcul du cosinus sur les articles est semblable à celui calculé sur les utilisateurs. Ainsi la formule de cette mesure pour les articles est la suivante ;

$$Sim(i,j) = Cosinus(i,j) = \frac{\sum_{u \in U} r_{ui} \times r_{uj}}{\sqrt{\sum_{u \in U} {r_{ui}}^2} \times \sqrt{\sum_{u \in U} {r_{uj}}^2}}$$

Où U représente l'ensemble des utilisateurs ;

r_{ui} est égal à la notation de l'utilisateur u pour l'article i ;

r_{uj} est égal à la notation de l'utilisateur v pour l'article i.

De ce fait, si nous reprenons notre matrice de notations (cf. Figure 2.2(b)), nous obtenons les mesures suivantes de similarités entre les articles ;

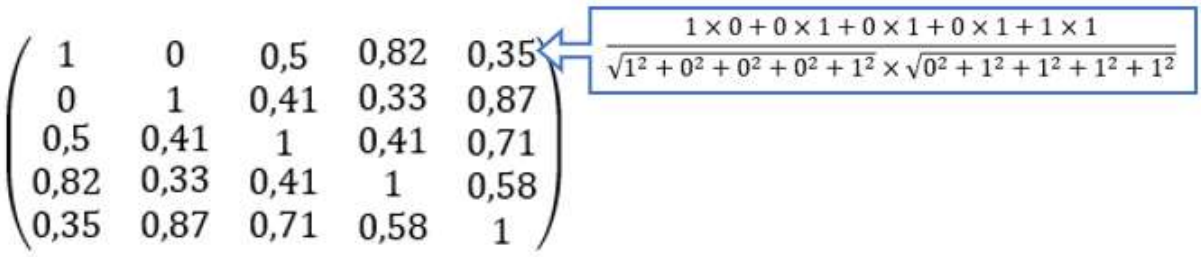

Figure 2.10 - Calcul de cosinus entre articles

La deuxième étape est de constituer l'ensemble des *k* plus proches voisins pour un article afin de faire des recommandations à un utilisateur en fonction des notes qu'il a attribuées à ses voisins. Dans notre exemple, si nous composons un voisinage de 2 voisins pour l'article 1, celui-ci sera composé de l'article 4 et de l'article 3. Pour chaque utilisateur, nous pouvons donc calculer un score d'intérêt pour un article en fonction des notes que cet utilisateur a données à ses voisins. Le score d'intérêt se calcule donc comme suit ;

$$Score(u,i) = \frac{\sum_{j \in N} r_{uj} \times w_{ij}}{\sum_{j \in N} w_{ij}}$$

Où *N* représente l'ensemble des voisins de l'article *i* ;

r_{uj} est égal à la note obtenue pour l'article *j* par l'utilisateur *u* ;

w_{ij} est égal à la similarité calculée précédemment entre l'article *i* et l'article *j*.

Dans notre exemple, si nous calculons les différents scores d'intérêt par nos cinq films pour les cinq utilisateurs, nous obtiendrons les scores suivants ;

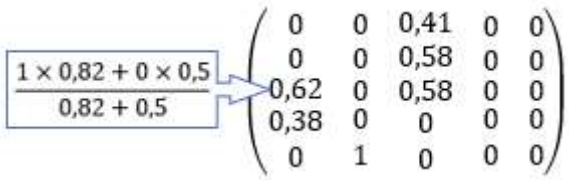

Figure 2.11 - Exemple de score d'intérêt

Pour effectuer nos recommandations, nous procéderons de la même manière que précédemment. Les articles ayant un score d'intérêt élevé par un utilisateur seront recommandés à ce dernier. Ainsi, dans notre exemple, si nous devions recommander deux articles à l'utilisateur 3, nous lui recommanderions l'article 1 suivi de l'article 3. Le premier article est fortement similaire à l'article 4 qui a déjà été vu par l'utilisateur 3. Le deuxième article, le film 3 est légèrement moins similaire au film 5, également visionné par l'utilisateur 3. Nous ne pouvons pas recommander les films 2, 4 et 5 puisqu'ils ont déjà été vus par l'utilisateur 3.

Films	Films similaires
1	3, 4
2	3, 5
3	1, 5
4	1, 5
5	2, 3

Utilisateurs	Films vus
1	1, 4
2	2, 5
3	2, 4, 5
4	2, 3, 5
5	1, 3, 4, 5

Utilisateurs	Films similaires
1	1, 3, 4, 5
2	2, 3, 5
3	1, 2, 3, 5
4	1, 2, 3, 5
5	1, 2, 3, 4

Utilisateurs	Films recommandés
1	3, 5
2	3, 1
3	1, 3
4	1, 4
5	2

Figure 2.12 - Exemple de recommandation

5. Intégration du domaine des graphes

L'objectif de ce travail étant d'intégrer des mesures de centralité sur graphe dans un système de recommandation, nous construirons plusieurs systèmes de recommandation à filtrage collaboratif intégrant différentes mesures de centralité à différents moments.

5.1. Algorithme de filtrage collaboratif basé sur les utilisateurs intégrant une mesure d'intermédiarité

Ce premier système de recommandation basé sur les utilisateurs intègre donc une mesure de centralité sur graphe, la mesure d'intermédiarité. Dans ce système, nous partons de l'hypothèse qu'un utilisateur ayant un indice d'intermédiarité (bet_i) élevé a potentiellement des goûts intermédiaires entre plusieurs groupes d'utilisateurs ayant des goûts similaires. Il serait ainsi positionné entre différents groupes d'utilisateurs et apporterait des recommandations plus diversifiées. Pour utiliser cette information dans notre système de recommandation, nous devons calculer une mesure d'intermédiarité pour chaque utilisateur. Cette mesure a déjà été définie et se calcule comme suit ;

$$Intermédiarité(i) = bet_i \triangleq \sum_{\substack{j=1 \\ j \neq i}}^{n} \sum_{\substack{k=1 \\ k \neq i,j}}^{n} \frac{\eta(i \in P_{jk}^*)}{|P_{jk}^*|} = \sum_{\substack{j=1 \\ j \neq i}}^{n} \sum_{\substack{k=1 \\ k \neq i,j}}^{n} \frac{\sum_{\wp_{jk}^* \in P_{jk}^*} \delta(i \in \wp_{jk}^*)}{|P_{jk}^*|}$$

Où P_{jk}^* constitue l'ensemble de chemins le plus court reliant les nœuds *j* et *k* ;

$\wp_{jk}^*$ représente la proportion de chemins géodésiques entre le nœud *j* et le nœud *k* parcourant le nœud *i* (avec *j* différent de *i* et de *k*). On peut en déduire que $\wp_{jk}^* \in P_{jk}^*$; $\delta(i \in \wp_{jk}^*)$ est égale à 1 si le chemin le plus court passe par *i*, à 0 dans le cas contraire (Fouss et al., 2016).

Cette mesure sera calculée comme explicité dans la section 3.1. Si nous reprenons notre exemple, nous obtenons les valeurs du Tableau 2.1 comme valeurs d'intermédiarité pour nos utilisateurs.

Utilisateurs	**Valeurs d'intermédiarité**
1	1,00
2	0,7
3	6,07
4	3,73
5	12,5

Tableau 2.1 - Exemple de valeurs d'intermédiarité sur les utilisateurs

Une fois cette mesure d'intermédiarité obtenue, nous calculerons une mesure de similarité entre les utilisateurs comme dans le système de recommandation basé sur les utilisateurs de référence. La mesure de similarité utilisée restera donc le cosinus. Cependant lors de la sélection du voisinage d'un utilisateur d'intérêt, nous prendrons en compte la mesure de similarité ainsi que la mesure d'intermédiarité afin que les utilisateurs avec une valeur d'intermédiarité plus élevée aient plus de chance d'être choisis dans un voisinage. Le poids w_{uv} sera donc défini comme suit ;

$$w_{uv} = Cosinus(u, v) * bet_v$$

Ainsi, dans notre exemple, nous observerons de nouveaux poids entre chaque paire d'utilisateurs :

$$\begin{pmatrix} 1 & 0 & 2,47 & 0 & 8,84 \\ 0 & 1 & 4,95 & 3,05 & 4,42 \\ 0,41 & 0,57 & 1 & 2,49 & 7,22 \\ 0 & 0,57 & 4,04 & 1 & 7,22 \\ 0,71 & 0,25 & 3,5 & 2,15 & 1 \end{pmatrix}$$

$0,71 \times 12,5$

Figure 2.13 - Exemple de poids par utilisateur

Sur la Figure 2.13, nous pouvons constater que la matrice des poids des utilisateurs n'est plus symétrique contrairement à la matrice de similarités. Cela est normal puisque la variable bet_v ne dépend que de *v* et non de la paire de nœuds. Le poids est donc impacté uniquement par la valeur d'intermédiarité de l'utilisateur *v.* De plus, la diagonale de notre matrice n'a pas été impactée par notre calcul puisqu'elle reflète la similarité entre un utilisateur et lui-même.

Dans notre exemple, nous pouvons observer que l'utilisateur 5, possédant la valeur d'intermédiarité la plus élevée, est ainsi privilégié dans tous les voisinages. Cela s'explique par

sa valeur d'intermédiarité deux fois plus élevée que la seconde plus grande valeur. Nous pouvons voir sur la Figure 2.14 les nouveaux voisinages obtenus avec cette modification.

Utilisateurs	Voisins
1	3, 5
2	3, 4
3	2, 4
4	2, 3
5	1, 3

Utilisateurs	Voisins
1	3, 5
2	3, 5
3	4, 5
4	3, 5
5	3, 4

Figure 2.14 - Comparaison des voisinages

À la suite de cela, l'intérêt d'un utilisateur pour un article sera calculé de la même manière que pour la méthode traditionnelle excepté que le poids des utilisateurs w_{uv} sera bien égal à la similarité multipliée par la valeur d'intermédiarité de l'utilisateur *v* ;

$$Score(u,i) = \frac{\sum_{v \in N} r_{vi} \times w_{uv}}{\sum_{v \in N} w_{uv}}$$

Où *N* représente l'ensemble des voisins de l'utilisateur *u* ;

r_{vi} est égal à la note obtenue pour l'article *i* par l'utilisateur *v*.

Dans notre exemple, les scores des articles seront maintenant égaux à la Figure 2.15. Nous pouvons donc constater que les scores d'intérêt sont différents des scores obtenus avec le système basé sur les utilisateurs de référence. Cela est expliqué par la composition différente des voisinages ainsi que par le poids supplémentaire donné aux utilisateurs intermédiaires.

$$\frac{0 \times 2{,}49 \; + \; 1 \times 7{,}22}{2{,}49 \; \times \; 7{,}22} \Rightarrow \begin{pmatrix} 0 & 0{,}22 & 0{,}78 & 0 & 1 \\ 0{,}47 & 0 & 0.47 & 1 & 0 \\ 0{,}74 & 0 & 1 & 0 & 0 \\ 0{,}64 & 0 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 & 0 \end{pmatrix}$$

Figure 2.15 - Exemple de scores d'intérêt en incorporant une mesure d'intermédiarité

Si nous devions réaliser une recommandation de deux films aux utilisateurs sur base des scores obtenus, les recommandations du deuxième tableau de la Figure 2.16 seront effectuées. Nous pouvons observer que peu de changement est perçu dans les listes de recommandations, cela peut être expliqué par le peu de films disponibles dans notre jeu de données. Toutefois, cet exemple sert d'illustration sur la méthodologie suivie et non d'expérimentation.

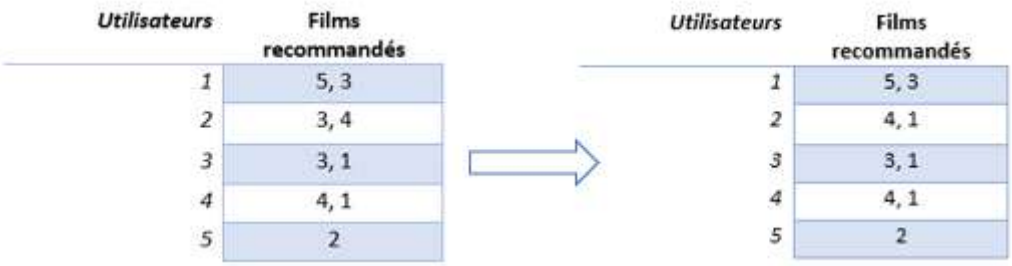

Figure 2.16 - Exemple de recommandations en incorporant une mesure d'intermédiarité

En conclusion, ce système de recommandation intègre la mesure d'intermédiarité dans la sélection du voisinage de l'utilisateur d'intérêt afin de favoriser les bons intermédiaires. Par hypothèse, cette manipulation a pour objectif de favoriser les utilisateurs avec des goûts à l'intermédiaire de différents groupes et ainsi d'apporter plus de diversité dans les articles recommandés.

5.2. Algorithme de filtrage collaboratif basé sur les utilisateurs intégrant une mesure de proximité

Ce deuxième système de recommandation sera également basé sur les utilisateurs et intégra quant à lui une autre mesure de centralité sur graphe, la mesure de proximité. Nous partons de l'hypothèse qu'un nœud possédant un indice de proximité (cc_i) élevé est plus populaire. En effet, la proximité calcule dans quelle mesure un nœud est proche de tous les autres nœuds du graphe. Cette dernière se calcule comme suit ;

$$Proximité(i) = cc_i = \frac{1}{\frac{1}{n-1}\sum_{j=1}^{n}\triangle_{ij}} = \frac{n-1}{\sum_{j=1}^{n}\triangle_{ij}}$$

Où $\triangle_{ij}$ est la distance du chemin le plus court, mais elle peut être remplacée par n'importe quelle autre mesure de dissimilarité ;
n est égal à la taille du graphe.

Cette mesure sur graphe sera calculée uniquement sur les articles comme explicitée dans la section 3.1. Ainsi, les articles possédant une valeur de proximité élevée seront vus comme des articles populaires. Dans notre exemple, les différents articles obtiennent les valeurs de proximité du Tableau 2.2. Nous pouvons ainsi constater que le film 5 est le plus populaire.

Films	Valeurs de proximité
1	0,43
2	0,47
3	0,47
4	0,53
5	0,60

Tableau 2.2 - Exemple de valeurs de proximité sur les articles

Pour ce système de recommandation, cette mesure de proximité sera utilisée au moment du calcul de la similarité. Par hypothèse, nous estimons qu'un article populaire capture de manière moindre les préférences individuelles d'un utilisateur. Nous souhaitons donc calculer notre mesure de similarité en pénalisant les articles populaires. Ainsi, les plus proches voisins d'un utilisateur d'intérêt auraient des goûts spécifiques communs avec celui-ci. Le cosinus entre les utilisateurs se calculera donc comme suit ;

$$Sim(u,v) = Cosinus(u,v) = \frac{\sum_{i \in I_u \cap I_v} (r_{ui}/cc_i{}^2) \times (r_{vi}/cc_i{}^2)}{\sqrt{\sum_{i \in I_u} (r_{ui}/cc_i{}^2)^2} \times \sqrt{\sum_{i \in I_v} (r_{vi}/cc_i{}^2)^2}}$$

Où I_u représente l'ensemble des notations connues pour l'utilisateur u ;

r_{ui} est égal à la notation de l'utilisateur u pour l'article i ;

I_v représente l'ensemble des notations connues pour l'utilisateur v ;

r_{vi} est égal à la notation de l'utilisateur v pour l'article i.

Dans notre exemple, nous obtenons de cette manière les similarités entre utilisateurs de la Figure 2.17.

$$\begin{pmatrix} 1 & 0 & 0{,}31 & 0 & 0{,}78 \\ 0 & 1 & 0{,}83 & 0{,}76 & 0{,}17 \\ 0{,}31 & 0{,}83 & 1 & 0{,}63 & 0{,}38 \\ 0 & 0{,}76 & 0{,}63 & 1 & 0{,}48 \\ 0{,}78 & 0{,}17 & 0{,}38 & 0{,}48 & 1 \end{pmatrix} \leftarrow \frac{(1/0{,}43^2 \times 1/0{,}43^2) + (1/0{,}53^2 \times 1/0{,}53^2)}{\sqrt{1/0{,}43^2{}^2 + 1/0{,}53^2{}^2} \times \sqrt{1/0{,}43^2{}^2 + 1/0{,}47^2{}^2 + 1/0{,}53^2{}^2 + 1/0{,}6^2{}^2}}$$

Figure 2.17 - Exemple de similarité en intégrant une mesure de proximité

La construction du voisinage d'un utilisateur cible se fera en fonction de cette nouvelle mesure de similarité. Ainsi, les utilisateurs possédant un indice de similarité élevé avec notre utilisateur

d'intérêt constitueront son voisinage de k voisins. Ce voisinage nous permettra de calculer l'intérêt d'un utilisateur pour un article avec la même formule que pour le système de référence.

Dans notre exemple, nous obtenons donc les scores d'intérêt de la Figure 2.18. Les données de notre exemple étant restreintes à cinq utilisateurs et cinq films, notre manipulation n'a pas beaucoup d'impact sur le voisinage des utilisateurs ou sur les listes de recommandations. Cet exemple sert toutefois d'illustration.

$$\begin{pmatrix} 1 & 0{,}27 & 0{,}73 & 0 & 1 \\ 0 & 0 & 0{,}49 & 0{,}51 & 0 \\ 0 & 0 & 0{,}44 & 0 & 0 \\ 0 & 0 & 0 & 0{,}45 & 0 \\ 0 & 0.36 & 0 & 0 & 0 \end{pmatrix}$$

Figure 2.18 - Exemple de scores avec la nouvelle similarité

Contrairement au système de recommandation basé sur les utilisateurs de référence, celui-ci favorise les utilisateurs ayant des goûts spécifiques en commun en délaissant les goûts populaires. Cela aura pour impact de sélectionner dans le voisinage des personnes appréciant de mêmes films moins populaires.

5.3. Algorithme de filtrage collaboratif basé sur les articles intégrant une mesure d'intermédiarité

Ce troisième système de recommandation sera quant à lui basé sur les articles. Il intègrera la mesure d'intermédiarité comme mesure de centralité sur graphe. Cependant, contrairement au système de recommandation décrit au point 5.1, cette mesure d'intermédiarité sera calculée sur les articles, et non sur les utilisateurs. Toutefois, cette mesure se calcule toujours de la manière suivante ;

$$Intermédiarité(i) = bet_i \triangleq \sum_{\substack{j=1 \\ j \neq i}}^{n} \sum_{\substack{k=1 \\ k \neq i,j}}^{n} \frac{\eta(i \in P_{jk}^*)}{|P_{jk}^*|} = \sum_{\substack{j=1 \\ j \neq i}}^{n} \sum_{\substack{k=1 \\ k \neq i,j}}^{n} \frac{\sum_{\wp_{jk}^* \in P_{jk}^*} \delta(i \in \wp_{jk}^*)}{|P_{jk}^*|}$$

Où P_{jk}^* constitue l'ensemble de chemins le plus court reliant les nœuds j et k ;

$\wp_{jk}^*$ représente la proportion de chemins géodésiques entre le nœud j et le nœud k parcourant le nœud i (avec j différent de i et de k). On peut en déduire que $\wp_{jk}^* \in P_{jk}^*$;

$\delta(i \in \wp_{jk}^*)$ est égale à 1 si le chemin le plus court passe par i, à 0 dans le cas contraire (Fouss et al., 2016).

Selon notre hypothèse, un article possédant un indice d'intermédiarité (bet_i) est considéré comme un article plus divers puisqu'il se situe entre différents groupes d'articles. Ce système vise donc à favoriser ces articles dans les listes de recommandations afin d'apporter plus de diversité dans celles-ci. Dans notre exemple, nous obtenons les valeurs présentées dans le Tableau 2.3 comme mesure d'intermédiarité pour nos articles. Nous pouvons ainsi constater que le film 5 reste le plus central au graphe avec la plus grande valeur d'intermédiarité.

Films	**Valeurs d'intermédiarité**
1	1,92
2	3,08
3	1,95
4	7,03
5	10,02

Tableau 2.3 - Exemple de valeurs d'intermédiarité sur articles

Dans ce système de recommandation, la mesure de centralité sera intégrée au moment du calcul d'intérêt d'un utilisateur pour un article. Le calcul de similarité entre les articles reste donc inchangé par rapport au système de recommandation de référence. Le cosinus entre les articles sera calculé de la même manière.

La sélection des *k* plus proches voisins d'un article d'intérêt se basera sur ces valeurs de similarités obtenues et restera donc inchangée par rapport au système de référence. Notre système diverge de celui-ci, pour le calcul du score d'intérêt. Dans notre système, nous souhaitons privilégier la présence des articles possédant un indice d'intermédiarité élevé dans les recommandations. Cela a pour objectif de favoriser les articles vus comme plus divers. L'intérêt d'un utilisateur pour un article sera calculé comme suit ;

$$Score(u,i) = \frac{\sum_{j \in N} r_{uj} \times w_{ij} \times bet_i}{\sum_{j \in N} w_{ij} \times bet_i}$$

Où *N* représente l'ensemble des voisins de l'article *i* ;

r_{uj} est égal à la note obtenue pour l'article *j* par l'utilisateur *u* ;

w_{ij} est égal à la similarité calculée précédemment entre l'article *i* et l'article *j*.

Dans notre exemple, si nous recalculons nos scores d'intérêts selon cette formule, nous obtenons les résultats suivants (cf. Figure 2.19). En comparaison avec les résultats obtenus en

point 4.2 avec le système de recommandation basé sur les articles de référence (cf. Figure 2.11), nous pouvons constater que désormais l'article 3 obtient un meilleur score que l'article 1 pour l'utilisateur 1. En effet, l'article 3 possède un indice d'intermédiarité supérieur à l'article 1, il sera donc privilégié par notre système.

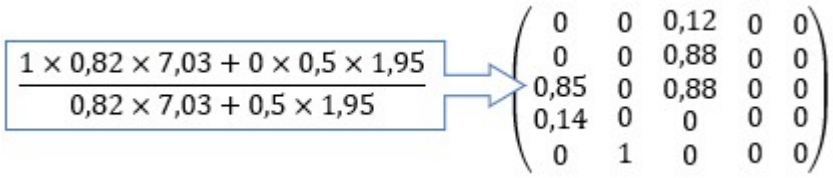

Figure 2.19 - Exemple de score en intégrant une mesure de proximité

En conclusion, ce troisième système de recommandation vise à favoriser la présence des bons intermédiaires dans les listes de recommandation. Contrairement aux deux systèmes précédents, l'intégration des mesures de centralité n'impacte pas la sélection du voisinage d'un article. En effet, nous estimons que le voisinage des articles ne devrait pas être impacté par cette mesure. Pour nous, un utilisateur restera intéressé par des articles qui sont similaires aux articles vus précédemment. L'objectif ici reste donc de sélectionner ces articles qui sont potentiellement pertinents pour l'utilisateur sans incorporer de biais. Les articles plus diversifiés faisant partie de cet ensemble d'articles seront ensuite privilégiés pour apparaitre dans la liste de recommandations de l'utilisateur.

5.4. Algorithme de filtrage collaboratif basé sur les articles intégrant une mesure de proximité

Ce dernier système de recommandation se base également sur les articles et prend en compte la mesure de proximité dans ses recommandations. Ce système partira également de l'hypothèse qu'un article possédant un indice de proximité (cc_i) élevé est considéré comme un article populaire. En effet, la proximité calcule dans quelle mesure un nœud est proche de tous les autres nœuds du graphe, celui-ci est donc plus probablement connu des utilisateurs. L'indice de proximité sera donc calculé de la même manière que pour le système décrit au point 5.2, sur chaque article et de la manière suivante ;

$$Proximité(i) = cc_i = \frac{1}{\frac{1}{n-1}\sum_{j=1}^{n}\triangle_{ij}} = \frac{n-1}{\sum_{j=1}^{n}\triangle_{ij}}$$

Où $\triangle_{ij}$ est la distance du chemin le plus court, mais elle peut être remplacée par n'importe quelle autre mesure de dissimilarité ;

n est égal à la taille du graphe.

Dans notre exemple, nous obtenons donc les mêmes valeurs qu'en point 5.2 avec une proximité supérieure pour le film 5 (cf. Tableau 2.2).

La première étape pour réaliser un système de recommandation à filtrage collaboratif est le calcul d'une mesure de similarité pour composer le voisinage. Dans notre cas, nous procéderons de la même manière que pour le système de référence. La mesure de similarité choisie reste donc le cosinus et elle sera calculée sur les articles.

La sélection des *k* plus proches voisins se fera donc grâce à cette mesure et rassemblera les articles les plus similaires à un article d'intérêt. À la suite de cela, nous calculerons les scores d'intérêt d'un utilisateur pour un article. L'objectif de ce système est d'utiliser la mesure de proximité afin de pénaliser les films populaires. Ceci a pour but de favoriser la présence d'articles moins populaires dans les listes de recommandations, ceux-ci apporteraient plus de nouveauté. Le score d'intérêt d'un utilisateur pour un film sera donc calculé comme suit ;

$$Score(u,i) = \frac{\sum_{j \in N}(r_{uj} \times w_{ij}/cc_i)}{\sum_{j \in N}(w_{ij}/cc_i)}$$

Où *N* représente l'ensemble des voisins de l'article *i* ;

r_{uj} est égal à la note obtenue pour l'article *j* par l'utilisateur *u* ;

w_{ij} est égal à la similarité calculée précédemment entre l'article *i* et l'article *j*.

Dans notre exemple, nous pouvons observer les différents scores obtenus avec notre manipulation sur la Figure 2.20. Nous constatons que contrairement au système de recommandation précédent, l'article 1 obtient cette fois-ci un score plus élevé que l'article 3 pour l'utilisateur 1. Le film 1 a notamment une valeur de proximité légèrement inférieure à celle de l'article 3.

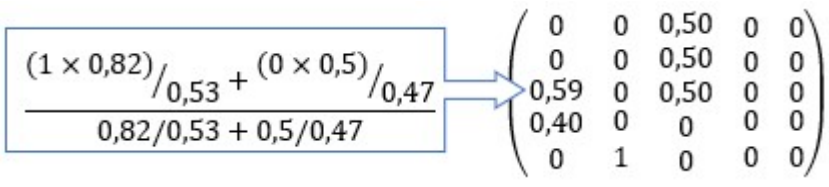

Figure 2.20 - Exemple de scores en intégrant une mesure de proximité

En conclusion, ce système de recommandation basé sur les articles pénalise les articles vus populaires afin qu'ils apparaissent moins dans les listes de recommandations. Les articles

populaires sont identifiés grâce à leur indice de proximité. Cela a pour but d'apporter plus de nouveauté dans les listes avec des articles moins connus des utilisateurs. Comme pour le système de recommandation précédent, notre manipulation n'impacte pas le voisinage d'un article. Cela est expliqué de la même manière que précédemment, pour nous, un utilisateur a de l'intérêt pour les articles qui sont similaires à ceux qu'il a déjà regardés. Cette manipulation a donc pour objectif de favoriser les articles moins connus dans le haut de la liste.

6. Métriques d'évaluation

Pour servir la question de recherche de ce travail, les différents systèmes de recommandation que nous avons développés à la section précédente devront être évalués. Cette évaluation se fera sur plusieurs critères. Dans la revue littéraire, nous avons défini différentes mesures d'évaluation calculant les performances d'un système de recommandation selon différents objectifs.

Tout d'abord, les mesures de précision regroupant les mesures de précision de prédiction et les mesures de précision de la liste de recommandations. Ces mesures servent à mesurer l'exactitude des recommandations ou des listes de recommandations. Ainsi, elles évaluent d'une part les scores obtenus avec les scores réels, et d'autre part, la proportion d'articles pertinents dans les listes de recommandations. Pour se faire, notre base de données sera divisée en dix parties afin de pouvoir procéder à une validation croisée. La validation croisée est une solution pour contrer le problème de surapprentissage d'un modèle par rapport à ses données. En effet, il est très facile de construire un modèle qui s'adapte parfaitement à ses données, mais qui est incapable d'être généralisé correctement sur un nouveau jeu de données (Berrar D., 2018). Pour contrer ce phénomène, la validation croisée fut employée en créant deux échantillons de données provenant de la même population. De ce fait, un échantillon était utilisé pour la calibration du modèle tandis que l'autre servait à son évaluation (Browne M., 2000). Dans ce travail, nous utiliserons la validation croisée k-fold qui partitionne les données disponibles en k sous-ensembles distincts de même taille. Cette séparation est faite de manière aléatoire. Le modèle est ensuite entrainé sur $k-1$ sous-ensembles rassemblés pour former l'ensemble d'entrainement, et sera évalué sur le sous-ensemble restant, l'ensemble test. Cette procédure est observée jusqu'à ce que chaque sous-ensemble fasse office d'ensemble test, la performance du modèle sera alors la moyenne des k évaluations (Berrar D., 2018). Nous procéderons à une validation croisée 10-fold pour ce mémoire.

Dans ce mémoire, nous tentons d'apporter plus de nouveauté et de diversité dans les listes de recommandations proposées aux utilisateurs. Nous devons donc évaluer nos différents systèmes de recommandation sur les critères de nouveauté et de diversité. Dans la revue littéraire, nous avons d'ailleurs vu que ces deux critères étaient de plus en plus recherchés dans les systèmes de recommandation. Nous cherchons donc à améliorer ces mesures de nouveauté et de diversité tout en conservant une précision satisfaisante.

6.1. Mesures de précision

Dans ce travail, nous nous intéresserons uniquement aux mesures de précision des listes de recommandations. Puisque nous avons fait le choix d'utiliser une matrice binaire comme matrice de notations, nous avons de ce fait renoncé à produire des prédictions de notations d'un utilisateur pour un film. Pour rappel, nous utilisons une matrice binaire afin de réduire le bruit occasionné par la présence de zéros pour spécifier les valeurs non connues dans la matrice de notations d'origine. De plus, l'objectif de nouveauté et de diversité concerne également les listes de recommandations. En effet, l'objectif est d'évaluer la proportion d'articles nouveaux et diversifiés présents dans les listes de recommandations. La performance liée à la précision reflètera donc également une proportion d'articles, dans ce cas, pertinents dans les listes de recommandations.

Dans ce travail, nous utiliserons trois mesures de précision différentes ; le nDCG, le Recall et la Précision. Tout d'abord, le nDCG, le gain cumulatif actualisé normalisé, a été défini comme suit dans la revue de littérature ;

$$nDCG = \frac{DCG}{IDCG}$$

Où

$$DCG = \sum_i \frac{r_{ui}}{\log_2(v_i + 1)}$$

Où r_{ui} est égal à la notation de l'utilisateur *u* pour l'article *i* ;
v_i correspond à la position de l'article *i* dans la liste de recommandations.

Cependant, puisque nous utilisons une matrice binaire, nous n'avons pas accès aux notations réelles des utilisateurs. Le gain cumulatif actualisé (DCG) sera donc calculé comme suit ;

$$DCG = \sum_i \frac{2^{rel_i} - 1}{log_2(v_i + 1)}$$

Où rel_i est un binaire qui vaut 1 si l'utilisateur a noté l'article *i* dans l'ensemble test, 0 sinon ;
v_i correspond à la position de l'article *i* dans la liste de recommandations (Croft, Metzler et Strohman, 2010).

Plus cette mesure est élevée, au plus les articles pertinents sont positionnés dans le haut la liste. Un article est pertinent à un utilisateur si ce dernier l'a noté dans l'ensemble test. Selon cette mesure, une liste de recommandations idéale présenterait en premier lieu tous les articles pertinents à un utilisateur.

La deuxième mesure, le Recall, est fortement liée à la mesure précédente puisqu'elle mesure la proportion d'articles pertinents présents dans la liste de recommandations et se calcule donc comme ceci ;

$$Recall = \frac{Articles\ correctement\ recommandés}{Nombre\ total\ d'articles\ pertinents}$$

Contrairement au nDCG, cette mesure calcule simplement la proportion d'articles pertinents dans la liste sans prendre en compte leur position dans celle-ci.

La dernière mesure, la Précision, calcule la proportion d'articles pertinents d'une liste de recommandations. Contrairement à la méthode suivante qui calcule la proportion des articles pertinents se retrouvant bien dans la liste de recommandations, celle-ci calcule la proportion d'articles pertinents de cette liste. Elle se calcule donc comme suit ;

$$Précision = \frac{Articles\ correctement\ recommandés}{Nombre\ total\ d'articles\ recommandés}$$

Dans notre exemple, si nous divisons notre échantillon de départ en deux sous-ensembles, l'ensemble d'entrainement et l'ensemble de test, nous obtenons les deux matrices suivantes ;

$$\begin{pmatrix} 1 & 0 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 & 1 \\ 0 & 1 & 0 & 1 & 1 \\ 0 & 1 & 0 & 0 & 1 \\ 0 & 0 & 1 & 1 & 1 \end{pmatrix}$$

Figure 2.21- Exemple de matrice d'entrainement

$$\begin{pmatrix} 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 \end{pmatrix}$$

Figure 2.22 - Exemple de matrice test

Avec cette division, nous pouvons évaluer notre système de recommandation. Pour illustrer les différentes mesures présentées dans cette sous-section, nous allons évaluer notre système de recommandation basé sur les utilisateurs de référence. Nous obtenons de ce fait les recommandations suivantes ;

Utilisateurs	**Films recommandés**
1	5, 2
2	4, 1
3	1, 3
4	4, 1
5	2, 1

Figure 2.23 - Recommandations

Nous pouvons donc évaluer les recommandations de la Figure 2.23. Les mesures d'évaluation présentées au Tableau 2.4 sont obtenues grâce aux différentes formules décrites dans cette sous-section. Nous pouvons constater que nous obtenons un nDCG faible puisqu'une seule des deux données présentes dans l'ensemble de test se retrouve dans les listes de recommandations. En effet, dans une liste idéale, on recommanderait le film 3 à l'utilisateur 4. De plus, on recommanderait le film 1 à l'utilisateur 5 en premier lieu. Ensuite, le Recall est bien égal à 0,5 puisqu'un seul des deux films notés dans l'ensemble test est présent dans les listes de recommandations. Pour finir, la Précision est égale à 0,1 puisqu'un seul des films recommandés est présent dans l'ensemble test.

Mesures	**Calculs**	**Résultats**
nDCG	$\frac{\frac{2^0-1}{log_2(1+1)}+\frac{2^0-1}{log_2(2+1)}+\frac{2^0-1}{log_2(1+1)}+\frac{2^1-1}{log_2(2+1)}}{\frac{2^1-1}{log_2(1+1)}+\frac{2^1-1}{log_2(2+1)}+\frac{2^1-1}{log_2(1+1)}+\frac{2^1-1}{log_2(2+1)}}$	0,126
Recall	$\frac{1}{2}$	0,5
Précision	$\frac{1}{10}$	0,1

Tableau 2.4 - Mesures d'évaluation de précision

6.2. Mesure de nouveauté et de diversité

Les notions de nouveauté et de diversité sont de plus en plus prises en compte dans les critères d'évaluation des systèmes de recommandation. La nouveauté d'un système de recommandation fait référence à sa capacité à proposer des articles inconnus de l'utilisateur. Cette mesure est

souvent calculée avec l'inverse de la popularité, en effet, un article populaire a plus de chance d'être connu de l'utilisateur qu'un article moins populaire. La nouveauté peut donc être calculée grâce à la popularité des articles qui se traduit comme suit ;

$$Popularité\ des\ articles\ =\ IP(R) = \frac{\sum_{i \in R} pop_i}{|R|}$$

Où pop_r indique la popularité de l'article *i* ;
R représente l'ensemble des articles présents dans la liste de recommandations de l'utilisateur.

Cette mesure calcule donc la popularité moyenne des articles présents dans une liste de recommandations. Une deuxième approche pour calculer la nouveauté consiste à mesurer l'indifférence d'un utilisateur par rapport à un article. Cette mesure calcule la dissimilarité existante entre les articles présents dans la liste et les articles ayant eu une interaction avec l'utilisateur. La distance d'un utilisateur fait référence à cette mesure et se calcule comme suit ;

$$Distance\ d'un\ utilisateur\ =\ DU(R, L_u)\ =\ \frac{\sum_{r_1 \in R} \sum_{r_2 \in L_u} dissim(r_1, r_2)}{|R| * |L_u|}$$

Où $L_u$représente l'ensemble des articles ayant eu une interaction avec l'utilisateur *u ;*
R est l'ensemble des articles présents dans la liste de recommandations de l'utilisateur *u* ;
$dissim(r_1, r_2)$ est égal à la dissimilarité entre l'article r_1 et l'article r_2.

La notion de diversité dans l'évaluation implique que les articles présents dans une liste de recommandations soient les plus diversifiés possible. Dans ce travail, nous utiliserons la dissimilarité intra-liste pour mesurer la diversité simple dans la liste de recommandations. Cette mesure se calcule comme suit ;

$$Dissimilarité\ intra - liste\ =\ ILD(R)\ =\ \frac{\sum_{r_1 \in R} \sum_{r_2 \in R} dissim(r_1, r_2)}{|R| * |R|}$$

Où *R* représente l'ensemble des articles présents dans la liste de recommandations ;
$dissim(r_1, r_2)$ est une fonction de dissimilarité qui quantifie la différence entre deux articles.

L'idée derrière cette dernière est que les articles sont probablement plus diversifiés s'ils sont moins similaires. La mesure de dissimilarité utilisée sera l'opposé du cosinus donc 1 diminué de la similarité.

Dans notre exemple, si nous évaluons les mêmes résultats obtenus qu'à la Figure 2.23, nous obtenons les mesures d'évaluation du Tableau 2.5.

Mesures	**Calculs**	**Résultats**
IP	$\frac{\frac{0,47+0,6}{2}+\frac{0,47+0,53}{2}+\frac{0,47+0,43}{2}+\frac{0,53+0,43}{2}+\frac{0,47+0,43}{2}}{5}$	0,483
DU	$\frac{\frac{1+0,67+1+0,43}{2\times 2}+\frac{1+0,5+0,67+0,43}{2\times 2}+\frac{1+1+0,43+0,43+0,5+1}{2\times 3}+\frac{0,67+1+0,43+1}{2\times 2}+\frac{1+0,67+0,14+1+0,43+1}{2\times 3}}{5}$	0,727
ILD	$\frac{\frac{0,14+0,14}{2\times 2}+\frac{0,43+0,43}{2\times 2}+\frac{1+1}{2\times 2}+\frac{0,43+0,43}{2\times 2}+\frac{1+1}{2\times 2}}{5}$	0,3

Tableau 2.5 - Mesures d'évaluation de nouveauté et de diversité

Conclusion

Pour conclure ce chapitre, nous allons résumer les étapes et les hypothèses qui le composent. Tout d'abord, nous avons présenté la base de données qui sera utilisée pour effectuer nos expérimentations. Celle-ci provient de MovieLens et reprend les notations de différents utilisateurs sur différents films. Ces notations nous permettront de construire notre matrice de notations, cette dernière sera binaire et spécifiera les films vus par l'utilisateur. Ainsi, elle indiquera une notation de 1 si l'utilisateur a vu le film, de 0 dans le cas contraire.

Grâce à cette matrice de notation binaire, nous constituerons notre matrice adjacente qui permettra de construire notre graphe. À la suite de cela, les mesures d'intermédiarité et de proximité seront calculées afin d'identifier les nœuds centraux au graphe.

Alors, nous construirons 6 algorithmes. Deux d'entre eux serviront d'algorithme de référence afin de mesurer l'impact sur la performance de l'intégration des mesures de centralités. Les quatre autres constitueront le cœur de notre expérimentation puisque ceux-ci intègrent des mesures sur graphe à différentes étapes. Ainsi, les deux premiers de ces systèmes se basent sur les utilisateurs. Le premier intègre une mesure d'intermédiarité lors du choix des voisins d'un utilisateur. De ce fait, les utilisateurs avec un indice d'intermédiarité élevé seront favorisés dans les voisinages. Le deuxième système intègre une mesure de proximité lors du calcul de similarité entre les utilisateurs. Les articles avec un indice de proximité élevé seront pénalisés au cours du calcul du cosinus entre deux utilisateurs.

Les deux derniers systèmes de recommandation sont basés sur les articles. Ceux-ci intègrent une mesure de centralité au calcul du score d'intérêt d'un utilisateur pour un article. Le premier système privilégie les articles avec un indice d'intermédiarité élevé tandis que le deuxième pénalise les articles avec un indice de proximité. De ce fait, ces articles seront plus ou moins présents dans les listes de recommandations.

Enfin, pour chaque système de recommandation, des mesures de performance seront calculées. La performance d'un système sera évaluée sur la précision des listes de recommandations, la nouveauté et la diversité des articles présents dans ces listes.

Chapitre 3 : Analyse des résultats

Introduction

Dans cette partie nous allons parcourir les résultats obtenus par nos différents algorithmes. Tout d'abord, nous passerons en revue les différentes décisions prises lors de cette expérimentation.

Ensuite, nous entrerons dans le cœur de notre étude avec la présentation des résultats obtenus par nos algorithmes sur les différentes mesures d'évaluation. Nous tenterons ainsi de répondre à notre question de recherche en analysant si l'intégration de nos mesures sur graphe a un impact sur les performances de nouveauté ou de diversité.

1. Prise de décisions et expérimentations

Afin d'appliquer notre méthodologie précédemment présentée, nous avons eu recours à un langage de programmation, Python. Celui-ci nous a permis de coder les différents algorithmes qui ont servi à cette étude. Toutes nos méthodes sont disponibles en annexe 2.

Nous avons donc élaboré un code permettant d'exploiter nos différentes bases de données en extrayant les différentes notations sous forme de matrice de notations binaires et de matrice adjacente pour la construction de notre graphe. Ce code permettait également d'initialiser des fichiers reprenant les nœuds et les liens afin de visualiser le graphe sur Gephi. Ensuite, deux méthodes permettent de mesurer les mesures d'intermédiarité et de proximité bien que celles-ci soient également disponibles sur le logiciel Gephi. À la suite de ça, notre matrice de notations est divisée en 10 sous-ensembles afin de procéder à une évaluation croisée pour chaque algorithme. Chaque algorithme présenté dans le chapitre méthodologie est ainsi entrainé et testé 10 fois afin de mesurer sa performance. Les systèmes de recommandations recommandaient pour chaque utilisateur 10, 20 ou 50 articles. Enfin, plusieurs méthodes permettent de calculer les différentes mesures d'évaluation définies précédemment.

Concernant les décisions prises pendant cette phase d'expérimentations, nous avons tout d'abord, choisi de composer un voisinage de 20 voisins par utilisateur. Ce choix se base sur une étude qui conclut que la performance d'un système de recommandation sur une base de données provenant de MovieLens performe mieux sur un voisinage de 5, 10 ou 15 voisins. Puisque cette étude expérimente des voisinages de 15, 30, 45 et 60 voisins, nous avons décidé de prendre un nombre de voisins peu élevé (Bahadorpour, Nadimi-Shahrahl et Neyslani, 2017). Ensuite, pour des raisons de temps de calcul, les mesures de centralité sur graphe sont calculées une fois pour chaque base de données. C'est-à-dire que ces mesures ne font pas partie du procédé de validation croisée. Nous estimons que cela n'impactera que très peu la précision de nos systèmes puisque l'idée sous-jacente de ces mesures est d'identifier les nœuds faisant office de bons intermédiaires et proches des autres nœuds du graphe. Ces mesures peuvent donc rester les mêmes pour les 10 phases d'entrainement.

2. Présentation des résultats

2.1. Algorithme de filtrage collaboratif basé sur les utilisateurs intégrant une mesure d'intermédiarité

Ce premier système de recommandation intègre une mesure d'intermédiarité lors de la sélection des voisins de l'utilisateur ainsi que dans le calcul du score d'intérêt. Ceci a pour objectif de favoriser les voisins qui ont un indice d'intermédiarité élevé. Par hypothèse, ceux-ci auraient des goûts plus diversifiés et amèneraient ainsi plus de diversité dans les listes de recommandations.

Tout d'abord, la performance de notre système de recommandation diffère avec le nombre d'articles proposés dans les listes (cf. Tableau 3.1 et Tableau 3.2). Nous constatons que notre système obtient des résultats supérieurs avec l'augmentation du nombre d'articles dans la liste. Cependant la mesure de Précision quant à elle diminue avec l'augmentation du nombre d'articles, ceci est logique puisque la Précision a pour dénominateur ce même nombre. Notons que la popularité diminue aussi, mais cela est positif puisque le but est d'avoir des articles moins populaires.

	10 recommandations	**20 recommandations**	**50 recommandations**
nDCG	0,050	0,069	0,099
Recall	0,051	0,087	0,164
Précision	0,068	0,058	0,043
Popularité moyenne	0,476	0,475	0,472
Distance d'un utilisateur	0,693	0,698	0,707
Dissimilarité intra-liste	0,484	0,527	0,575

Tableau 3.1 - Résultats de l'intégration de l'intermédiarité sur la base de données 1

	10 recommandations	**20 recommandations**	**50 recommandations**
nDCG	0,164	0,206	0,269
Recall	0,132	0,210	0,363
Précision	0,138	0,109	0,075
Popularité moyenne	0,469	0,461	0,451
Distance d'un utilisateur	0,670	0,684	0,706
Dissimilarité intra-liste	0,510	0,579	0,654

Tableau 3.2 - Résultats de l'intégration de l'intermédiarité sur la base de données 2

Ensuite, si nous comparons les performances de notre système avec celles du système basé sur les utilisateurs de référence (cf. Tableau 3.3), nous constatons que nous perdons beaucoup en précision dans les listes de recommandations. Cela signifie que moins d'articles présents dans l'ensemble test se retrouvent dans ces listes. Concernant les mesures de nouveauté, les résultats sont plutôt mitigés. En effet, la popularité moyenne des articles augmente pour les deux bases de données représentant donc une perte de nouveauté tandis que la distance d'un utilisateur augmente, cela se traduisant par un gain de nouveauté. Enfin, la dissimilarité intra-liste diminue pour notre première base de données et augmente pour la deuxième.

	Base de données 1		**Base de données 2**	
	Système de référence	*Système intégrant une mesure d'intermédiarité*	*Système de référence*	*Système intégrant une mesure d'intermédiarité*
nDCG	0,225	0,073	0,293	0,213
Recall	0,208	0,101	0,314	0,235
Précision	0,124	0,057	0,147	0,107
Popularité moyenne	0,472	0,474	0,453	0,460
Distance d'un utilisateur	0,685	0,700	0,668	0,686
Dissimilarité intra-liste	0,553	0,530	0,561	0,581

Tableau 3.3 – Comparaison du système au système de référence

Afin de valider nos résultats, nous avons réalisé une comparaison de moyennes à l'aide d'un test de Student avec pour hypothèse nulle des moyennes égales sur nos deux échantillons. Le premier échantillon provient de nos dix sous-ensembles test réalisés pour notre validation croisée et le deuxième provient de ces mêmes sous-ensembles, mais provenant de notre système intégrant notre mesure de centralité. Nous constatons que toutes les moyennes sont statistiquement différentes puisque nous obtenons des p-value inférieures à notre seuil de significativité de 10%. Dans ce cas, nous pouvons rejeter l'hypothèse nulle (cf. Tableau 3.4).

	Base de données 1	**Base de données 2**
nDCG	1,257E-03	2,906E-27
Recall	2,417E-10	5,340E-19

Précision	1,534E-17	1,110E-14
Popularité moyenne	7,135E-22	1,639E-26
Distance d'un utilisateur	7,320E-28	4,521E-20
Dissimilarité intra-liste	1,145E-29	1,312E-23

Tableau 3.4 - P-value provenant de la comparaison de moyenne grâce à un test de Student

Si nous observons la distribution de notre mesure d'intermédiarité, nous constatons que cette dernière est dispersée sur un grand intervalle (cf. Tableau 3.5). Cette intégration a donc un gros impact sur la création des voisinages. En effet, notre système ne conserve que 15% et 1% des voisins du système de référence pour nos deux bases de données, de plus, notre mesure n'étant pas normalisée et s'étendant sur un grand intervalle, cette dernière substitue l'impact du cosinus. En effet, le voisinage d'un utilisateur lui est en moyenne similaire à 18%, contre 31% pour notre système de référence. Nous avons donc choisi de catégoriser nos utilisateurs selon notre mesure d'intermédiarité, ainsi un utilisateur obtenant un score d'intermédiarité inférieur au premier quartile sera considéré comme mauvais intermédiaire et obtiendra un score égal à 1. À l'inverse, un utilisateur obtenant un score supérieur au dernier quartile obtiendra un score de 4.

	Base de données 1	**Bases de données 2**
Intervalles	[7,686 ; 464.972,97]	[4,656 ; 51.318,299]
1er quartile	70,584	73,856
2ème quartile	216,269	275,382
3ème quartile	760,864	1305,302

Tableau 3.5 - Dispersion de la mesure d'intermédiarité

À la suite de cette manipulation, nous constatons que la précision a augmenté par rapport à l'intégration précédente. Cependant elle reste inférieure au système de recommandation de référence (cf. Tableau 3.6). Au niveau de la nouveauté, nous obtenons les mêmes conclusions avec une popularité moyenne en baisse, mais une distance de l'utilisateur qui diminue. Nous ne pouvons donc pas conclure que notre système apporte plus de nouveauté. Enfin, la diversité qui était l'objectif premier de notre système est en diminution. Notre intégration ne permet donc

pas d'observer plus de diversité dans les articles proposés. Notons toutefois que les moyennes de la dissimilarité intra-liste et de la distance utilisateur ne sont statistiquement pas différentes aux moyennes du système de référence pour la première base de données.

	Base de données 1		**Base de données 2**	
	Système de référence	*Système intégrant une mesure d'intermédiarité*	*Système de référence*	*Système intégrant une mesure d'intermédiarité*
nDCG	0,225	0,192	0,293	0,250
Recall	0,208	0,184	0,314	0,272
Précision	0,124	0,109	0,147	0,127
Popularité moyenne	0,472	0,430	0,453	0,413
Distance d'un utilisateur	0,685	0,619	0,668	0,606
Dissimilarité intra-liste	0,553	0,495	0,561	0,502

Tableau 3.6 - Comparaison du système au système de référence avec une catégorisation de la mesure d'intermédiarité

2.2. Algorithme de filtrage collaboratif basé sur les utilisateurs intégrant une mesure de proximité

Rappelons que ce système de recommandation intègre une mesure de proximité lors du calcul de la mesure de similarité entre les utilisateurs. L'objectif étant de calculer cette mesure en pénalisant les articles obtenant un indice de proximité élevé puisque ceux-ci sont vus comme populaires et capturent donc de manière moindre les préférences individuelles d'un utilisateur.

Tout d'abord, la performance de notre système évolue avec le nombre d'articles proposés dans la liste de recommandations (cf. Tableau 3.7 et Tableau 3.8). Les mesures de performance s'améliorent toutes avec le nombre de recommandations excepté pour la précision, ce qui s'explique puisque le dénominateur équivaut au nombre de films recommandés par le système.

	10 recommandations	**20 recommandations**	**50 recommandations**
nDCG	0,180	0,220	0,275
Recall	0,123	0,189	0,313
Précision	0,164	0,126	0,083
Popularité moyenne	0,476	0,473	0,468

Distance d'un utilisateur	0,674	0,682	0,698
Dissimilarité intra-liste	0,493	0,550	0,617

Tableau 3.7 - Résultats de l'intégration de proximité sur la base de données 1

	10 recommandations	**20 recommandations**	**50 recommandations**
nDCG	0,235	0,290	0,364
Recall	0,190	0,291	0,470
Précision	0,197	0,151	0,098
Popularité moyenne	0,460	0,454	0,443
Distance d'un utilisateur	0,650	0,664	0,691
Dissimilarité intra-liste	0,493	0,557	0,638

Tableau 3.8 - Résultats de l'intégration de proximité sur la base de données 2

Ensuite, nous pouvons comparer notre système de recommandation avec notre système basé sur les utilisateurs de référence qui n'intègre pas de mesures de centralité sur graphe (cf. Tableau 3.9). Nous constatons que notre intégration a un très faible impact sur notre performance puisque les mesures diffèrent de centièmes. Cependant, nous remarquons que notre système obtient de meilleurs résultats sur la précision avec un Recall et une Précision plus élevée. En effet, notre système semble apporter plus de précision que le système de référence. Concernant la nouveauté, notre système propose en moyenne des résultats moins populaires et plus distants de l'utilisateur. Ce qui représente un gain de nouveauté, l'objectif principal de notre système. Enfin, la diversité intra-liste est également plus élevée.

	Base de données 1		**Base de données 2**	
	Système de référence	*Système intégrant une mesure de proximité*	*Système de référence*	*Système intégrant une mesure de proximité*
nDCG	0,2254	0,2249	0,293	0,296
Recall	0,208	0,209	0,314	0,317
Précision	0,1242	0,1243	0,147	0,149
Popularité moyenne	0,4724	0,4722	0,4527	0,4525
Distance d'un utilisateur	0,6849	0,6847	0,6684	0,6686
Dissimilarité intra-liste	0,5527	0,5534	0,561	0,562

Tableau 3.9 - Comparaison du système au système de référence

Au vu de l'impact faible de notre intégration sur nos résultats, nous avons effectué un test de Student avec comme hypothèse nulle que les moyennes de nos deux échantillons provenant des dix sous-ensembles de test, obtenus soit avec notre système de référence soit avec notre système intégrant la mesure de centralité. Nous observons que concernant les mesures de précision, notre p-value est supérieure à notre seuil de significativité de 10% pour notre première base de données et très proche pour la seconde. De ce fait, nous ne pouvons pas rejeter l'hypothèse nulle, nos moyennes ne sont donc pas statistiquement différentes. Toutefois, pour les mesures de nouveauté et de diversité, nous observons une p-value inférieure à ce seuil, l'hypothèse nulle peut donc être rejetée (cf. Tableau 3.10).

	Base de données 1	**Base de données 2**
nDCG	3,305E-01	5,393E-03
Recall	1,072E-01	2,612E-02
Précision	7,791E-01	9,029E-02
Popularité moyenne	2,787E-08	6,725E-09
Distance d'un utilisateur	6,667E-06	1,225E-08
Dissimilarité intra-liste	7,470E-04	1,083E-03

Tableau 3.10 - P-value provenant de la comparaison de moyenne grâce à un test de Student

Pour évaluer l'impact de notre intégration dans le voisinage d'un utilisateur, nous avons comparé le pourcentage de voisins communs aux deux systèmes. Nous constatons que les deux systèmes partagent 90% de leurs voisinages pour la première base de données et 88% pour la deuxième. De ce fait, nous avons pris la décision d'intégrer notre mesure de proximité d'une autre manière. Pour nos deux bases de données, notre mesure est distribuée sur un intervalle disponible sur le Tableau 3.11, nous pouvons diviser cette dispersion en quatre quartiles afin de catégoriser nos articles. Ainsi, un article possédant un indice inférieur au premier quartile sera considéré comme peu central et obtiendra un score de 1, à l'inverse un article possédant un score supérieur au troisième quartile sera fortement central au graphe et recevra un score de 4.

	Base de données 1	Bases de données 2
Intervalles	[0,303 ; 0,605]	[0,301 ; 0,528]
1er quartile	0,392	0,393
2ème quartile	0,435	0,405
3ème quartile	0,451	0,421

Tableau 3.11 - Dispersion de la mesure de proximité

À la suite de cette modification, nous constatons une baisse au niveau de notre précision par rapport à notre première intégration (cf. Tableau 3.12). Celle-ci est maintenant inférieure aux mesures du système de référence. Concernant les mesures de nouveauté, nous observons une popularité plus faible que celle de référence et que l'intégration précédente, mais une distance de l'utilisateur plus faible pour notre deuxième base de données. Nous pouvons toutefois supposer que notre système apporte plus nouveauté. La diversité augmente légèrement pour notre première base de données tandis qu'elle diminue pour notre deuxième. Par ailleurs, toutes nos moyennes sont statistiquement différentes selon notre test de Student.

	Base de données 1		Base de données 2	
	Système de référence	*Système intégrant une mesure de proximité*	*Système de référence*	*Système intégrant une mesure de proximité*
nDCG	0,225	0,187	0,293	0,222
Recall	0,208	0,181	0,314	0,242
Précision	0,124	0,107	0,147	0,113
Popularité moyenne	0,4724	0,4722	0,453	0,415
Distance d'un utilisateur	0,685	0,698	0,668	0,622
Dissimilarité intra-liste	0,553	0,563	0,561	0,522

Tableau 3.12 - Comparaison du système au système de référence avec une catégorisation de la mesure de proximité

2.3. Algorithme de filtrage collaboratif basé sur les articles intégrant une mesure d'intermédiarité

Pour rappel, ce système de recommandation basé sur les articles intègre une mesure d'intermédiarité lors du calcul du score d'intérêt d'un article pour un utilisateur. De ce fait, ce système favorise les articles avec une plus grande valeur d'intermédiarité. Ces articles ont donc plus de probabilités d'être présents dans les listes de recommandations.

Tout d'abord, si nous évaluons la performance du système de recommandation, nous constatons que toutes les mesures de performance augmentent avec le nombre de recommandations excepté pour la mesure de précision et les mesures de nouveauté qui traduisent une perte de nouveauté dans le cas de la deuxième base de données (cf. Tableau 3.13 et Tableau 3.14).

	10 recommandations	**20 recommandations**	**50 recommandations**
nDCG	0,060	0,083	0,118
Recall	0,045	0,081	0,156
Précision	0,059	0,054	0,041
Popularité moyenne	0,459	0,457	0,452
Distance d'un utilisateur	0,751	0,757	0,779
Dissimilarité intra-liste	0,611	0,667	0,738

Tableau 3.13 - Résultats de l'intégration de l'intermédiarité sur la base de données 1

	10 recommandations	**20 recommandations**	**50 recommandations**
nDCG	0,110	0,158	0,243
Recall	0,105	0,187	0,379
Précision	0,109	0,097	0,079
Popularité moyenne	0,426	0,428	0,431
Distance d'un utilisateur	0,737	0,7324	0,7317
Dissimilarité intra-liste	0,653	0,690	0,721

Tableau 3.14 - Résultats de l'intégration de l'intermédiarité sur la base de données 2

Ensuite, comparons la performance de notre système de recommandation au système basé sur les articles de référence (cf. Tableau 3.15). Celui-ci obtient de moins bons résultats sur la précision avec un nDCG, un Recall et une Précision inférieurs aux résultats obtenus par le système de référence. Concernant la nouveauté des articles proposés, nous observons que notre système intégrant une mesure de centralité obtient une valeur de popularité inférieure à celui de référence. Ce qui signifie que les articles présents dans la liste de recommandations sont en

moyenne moins populaires. De plus, la distance des utilisateurs augmente également. Enfin, concernant la diversité, notre système obtient une meilleure mesure de dissimilarité intra-liste ce qui suggère des listes de recommandations plus diversifiées.

	Base de données 1		**Base de données 2**	
	Système de référence	*Système intégrant une mesure d'intermédiarité*	*Système de référence*	*Système intégrant une mesure d'intermédiarité*
nDCG	0,229	0,087	0,256	0,170
Recall	0,225	0,094	0,304	0,224
Précision	0,133	0,052	0,137	0,095
Popularité moyenne	0,462	0,456	0,434	0,428
Distance d'un utilisateur	0,701	0,762	0,696	0,734
Dissimilarité intra-liste	0,615	0,672	0,630	0,688

Tableau 3.15 - Comparaison du système au système de référence

Afin d'évaluer si nos résultats sont significatifs, nous avons procédé à un test de Student sur nos échantillons provenant de notre évaluation croisée. Nous effectuons donc une comparaison de moyenne d'échantillons pairés. Si nous regardons les p-value obtenues (cf. Tableau 3.16), nous constatons qu'elles sont toutes inférieures à 0,10, notre seuil de significativité. Cela signifie que nous pouvons rejeter l'hypothèse nulle qui suppose que les moyennes de nos échantillons sont égales. Nos moyennes sont donc statistiquement différentes.

	Base de données 1	**Base de données 2**
nDCG	1,289E-26	2,032E-35
Recall	1,296E-16	4,697E-23
Précision	2,107E-16	1,784E-13
Popularité moyenne	2,456E-23	2,691E-15
Distance d'un utilisateur	1,560E-37	4,535E-17

Dissimilarité intra-liste	9,793E-27	1,507E-18

Tableau 3.16 - P-value provenant de la comparaison de moyenne grâce à un test de Student

Nous avons constaté au point 2.1 que notre mesure d'intermédiarité était dispersée sur un grand intervalle (cf. Tableau 3.5). Comme précédemment, nous avons choisi de catégoriser nos nœuds qui sont dans ce cas-ci des articles. Ainsi, un article obtenant un score d'intermédiarité inférieur au premier quartile sera considéré comme mauvais intermédiaire et obtiendra un score égal à 1. À l'inverse, un utilisateur obtenant un score supérieur au dernier quartile obtiendra un score de 4.

À la suite de cette modification, nous pouvons réévaluer la performance de notre système en comparaison au système basé sur les articles de référence (cf. Tableau 3.17). Nous constatons de ce fait que la précision de notre système s'est ainsi bien améliorée même si elle reste inférieure au système de référence. Cependant, les mesures de diversité restent supérieures au système de recommandation de référence. De plus, toutes nos moyennes sont statistiquement différentes du système de référence selon un test de Student. Notre intégration a donc l'impact souhaité en réalisant un compromis entre la précision et la diversité des articles proposés à l'utilisateur. La nouveauté reste quant à elle légèrement supérieure au système de référence. Toutefois, le but premier de ce système était d'apporter plus de diversité grâce aux mesures d'intermédiarité.

	Base de données 1		**Base de données 2**	
	Système de référence	*Système intégrant une mesure d'intermédiarité*	*Système de référence*	*Système intégrant une mesure d'intermédiarité*
nDCG	0,229	0,221	0,256	0,245
Recall	0,225	0,219	0,304	0,294
Précision	0,133	0,129	0,137	0,132
Popularité moyenne	0,463	0,459	0,440	0,436
Distance d'un utilisateur	0,701	0,707	0,696	0,703
Dissimilarité intra-liste	0,615	0,630	0,630	0,641

Tableau 3.17 - Comparaison du système au système de référence avec une catégorisation de la mesure d'intermédiarité

2.4. Algorithme de filtrage collaboratif basé sur les articles intégrant une mesure de proximité

Ce système de recommandation intègre une mesure de proximité au moment du calcul du score d'intérêt d'un utilisateur pour un film. L'objectif est de pénaliser les articles possédant un indice de proximité élevé afin qu'ils aient moins de probabilités d'être présents dans les listes de recommandations. Les articles avec un score de proximité élevé sont vus comme populaires, ils apportent donc moins de nouveauté aux utilisateurs.

Tout d'abord, si nous analysons les résultats obtenus par notre système, nous observons qu'il obtient de meilleurs résultats en augmentant le nombre de recommandations, puisque toutes les mesures d'évaluation s'améliorent à l'exception de la mesure de Précision qui diminue (cf. Tableau 3.18 et Tableau 3.19).

	10 recommandations	**20 recommandations**	**50 recommandations**
nDCG	0,181	0,225	0,284
Recall	0,130	0,205	0,341
Précision	0,173	0,136	0,091
Popularité moyenne	0,464	0,463	0,460
Distance d'un utilisateur	0,691	0,697	0,713
Dissimilarité intra-liste	0,561	0,612	0,670

Tableau 3.18 - Résultats de l'intégration de la proximité sur la base de données 1

	10 recommandations	**20 recommandations**	**50 recommandations**
nDCG	0,191	0,250	0,332
Recall	0,168	0,275	0,476
Précision	0,175	0,143	0,099
Popularité moyenne	0,441	0,440	0,436
Distance d'un utilisateur	0,685	0,692	0,708
Dissimilarité intra-liste	0,576	0,626	0,682

Tableau 3.19 - Résultats de l'intégration de la proximité sur la base de données 2

Ensuite, ce système obtient des résultats légèrement supérieurs au système de recommandation basé sur les articles de référence (cf. Tableau 3.20). En effet, les mesures du nDCG, du Recall et de Précision ont augmenté d'un centième. Concernant les mesures de nouveauté et de diversité, notre système obtient des résultats inférieurs d'un centième également. Ceci signifie que l'impact de notre intégration est faible, cependant la mesure de proximité ne semble pas

améliorer la nouveauté des articles présents dans la liste. Toutefois, elle permet d'augmenter la précision.

	Base de données 1		**Base de données 2**	
	Système de référence	*Système intégrant une mesure de proximité*	*Système de référence*	*Système intégrant une mesure de proximité*
nDCG	0,229	0,230	0,256	0,258
Recall	0,2246	0,2252	0,304	0,307
Précision	0,1328	0,1333	0,137	0,139
Popularité moyenne	0,4626	0,4627	0,43920	0,43924
Distance d'un utilisateur	0,701	0,700	0,696	0,695
Dissimilarité intra-liste	0,615	0,614	0,630	0,629

Tableau 3.20 - Comparaison du système au système de référence

Toujours pour évaluer nos résultats statistiquement, nous avons réalisé une comparaison de moyenne sur nos échantillons provenant de notre validation croisée (cf. Tableau 3.21). De ce fait, nous constatons que les moyennes de toutes nos mesures sont significativement différentes puisque notre p-value est inférieur à notre seuil de significativité de 0,10. Nous rejetons donc l'hypothèse nulle qui suppose que les moyennes de nos deux échantillons soient égales.

	Base de données 1	**Base de données 2**
nDCG	1,422E-05	7,871E-11
Recall	5,476E-04	1,025E-10
Précision	7,163E-04	1,696E-06
Popularité moyenne	2,106E-13	2,256E-06
Distance d'un utilisateur	1,529E-18	7,102E-18
Dissimilarité intra-liste	2,987E-15	1,018E-17

Tableau 3.21 - P-value provenant de la comparaison de moyenne grâce à un test de Student

Comme au point 2.2, nous avons pris la décision d'augmenter l'impact de notre mesure de proximité en catégorisant nos différents articles. De ce fait, un article possédant un indice inférieur au premier quartile sera considéré comme peu central et obtiendra un score de 1, à l'inverse un article possédant un score supérieur au troisième quartile sera très central au graphe et recevra un score de 4 (cf. Tableau 3.11).

À la suite de cette modification, nous avons réévalué la performance de notre système. Nous constatons que cette catégorisation ne permet toujours pas un impact significatif de l'intégration de notre mesure. En effet, la précision reste très proche de notre système de recommandation de référence. D'ailleurs, si nous réalisons une comparaison de moyenne grâce à un test de Student, nous constatons que nous ne pouvons pas rejeter l'hypothèse nulle, nos moyennes sur nos mesures de précision ne sont donc statistiquement pas différentes pour la première base de données. Cependant, les moyennes effectuées sur les mesures de diversité et de nouveauté sont quant à elle statistiquement différentes. L'intégration d'une mesure de proximité au calcul du score d'intérêt d'un article pour un utilisateur ne permettrait donc pas d'augmenter la nouveauté puisque la popularité moyenne est plus élevée que le système de référence tandis que la distance d'un utilisateur est plus faible. La diversité des articles présents dans la liste a aussi diminué (cf. Tableau 3.22).

	Base de données 1		**Base de données 2**	
	Système de référence	*Système intégrant une mesure de proximité*	*Système de référence*	*Système intégrant une mesure de proximité*
nDCG	0,229	0,208	0,256	0,259
Recall	0,225	0,204	0,304	0,307
Précision	0,133	0,121	0,137	0,139
Popularité moyenne	0,462	0,421	0,439	0,441
Distance d'un utilisateur	0,701	0,635	0,696	0,693
Dissimilarité intra-liste	0,615	0,556	0,630	0,623

Tableau 3.22 - Comparaison du système au système de référence avec une catégorisation de la mesure de proximité

Conclusion

Dans ce chapitre, nous avons présenté et analysé les résultats obtenus grâce à nos algorithmes. Nous constatons que l'intégration des mesures de centralité sur graphe a eu un impact sur les performances de nos systèmes. Cependant, cet impact ne se traduit pas toujours par une hausse de nouveauté ou de diversité dans nos listes de recommandations.

En effet, plusieurs de nos systèmes de recommandation ne permettent pas de conclure à un apport en nouveauté ou diversité. Cela est le cas pour le système basé sur les utilisateurs intégrant une mesure d'intermédiarité et pour le système basé sur les articles intégrant une mesure de proximité qui diminuent la diversité et n'améliorent pas la nouveauté. Cependant, le deuxième système améliore toutefois la précision du système. Dans le cas contraire, les deux autres systèmes permettent une amélioration de la nouveauté ou de la diversité.

Nous avons également vu dans ce chapitre les deux types d'intégration de la mesure de centralité soit avec la donnée brute soit en attribuant un score en fonction de la position de cette mesure dans sa dispersion.

Conclusion

Dans ce mémoire, nous avons tenté de mesurer l'impact de l'intégration des mesures de centralité sur graphe dans des systèmes de recommandation de filtrage collaboratif, cette intégration permettrait par hypothèse d'avoir plus de nouveauté ou de diversité dans leurs recommandations. Nous avons tout d'abord introduit notre question de recherche en parcourant la revue littérature. Nous avons d'ailleurs découvert que plusieurs études avaient déjà été réalisées en intégrant des mesures sur graphe dans des systèmes de recommandation. Toutes ces études concluent par ailleurs à une amélioration de la précision suite à l'intégration.

Ensuite, nous avons présenté notre méthodologie qui constituait en la mise en place de plusieurs systèmes de recommandation à filtrage collaboratif intégrant des mesures de centralité à différents moments. Par la même occasion, nous avons choisi les différentes mesures qui nous ont permis de mesurer la performance de nos systèmes sur les critères de la précision, de la nouveauté et de la diversité.

Enfin, nous avons présenté et analysé les résultats obtenus par nos différents systèmes. Tout d'abord, nous avons intégré une mesure de centralité sur graphe dans un système basé sur les utilisateurs. Nous avons, dans un premier temps, intégré la mesure d'intermédiarité des utilisateurs au moment du choix des voisins d'un utilisateur et du calcul du score d'intérêt. Ce système n'apporte pas plus de diversité dans les listes de recommandations et diminue la précision. Cependant, ce système est plus performant quand la mesure d'intermédiarité est intégrée sous forme de catégorie.

Dans un deuxième temps, nous avons intégré une mesure de proximité lors du calcul de la mesure de similarité entre les utilisateurs. Ce système apporte plus de précision que le système de référence lorsque la mesure n'est pas catégorisée. L'objectif de cette intégration est atteint puisqu'elle permet d'avoir plus de nouveauté. De plus, nous constatons une augmentation au niveau de la diversité dans les listes de recommandations (cf. Figure 4.1).

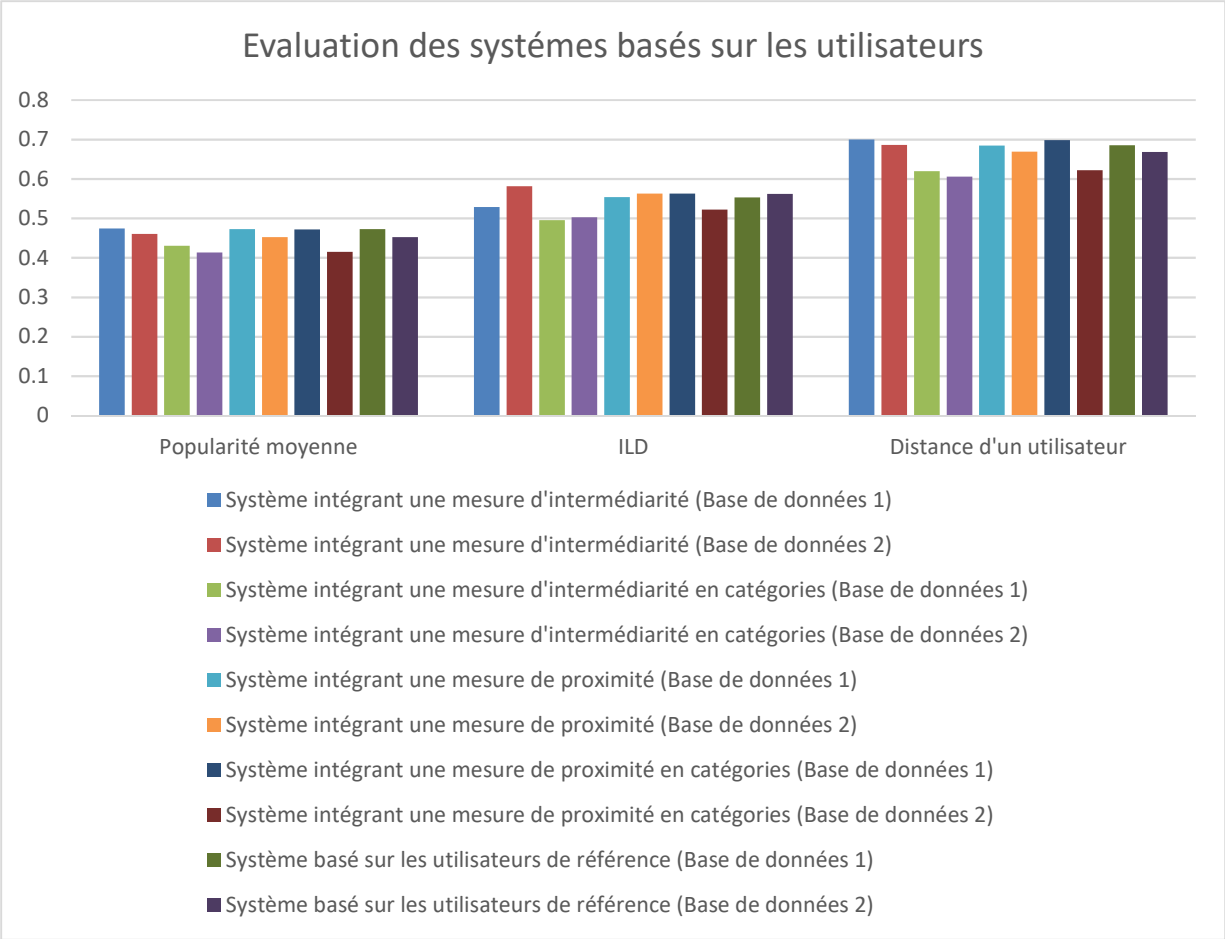

Figure 4.1 - Évaluation des systèmes basés sur les utilisateurs

Ensuite, nous avons intégré ces mêmes mesures à un système de recommandation basé sur les articles. Nous avons, dans un premier temps, intégré une mesure d'intermédiarité au moment du calcul du score d'intérêt. Ce système a eu comme résultat de diminuer la précision tout en augmentant les mesures de nouveauté et de diversité dans les listes de recommandations. Ce système est plus performant quand la mesure est catégorisée puisque dans le cas opposé, la performance sur la précision est fort basse.

Dans un deuxième temps, nous avons intégré une mesure de proximité au niveau du calcul du score d'intérêt. Cette intégration ne permet pas d'obtenir plus de nouveauté ou de diversité, on observe même l'effet inverse. Cependant, elle améliore la précision, mais l'objectif qui était d'augmenter la nouveauté n'est pas atteint (cf. Figure 4.2).

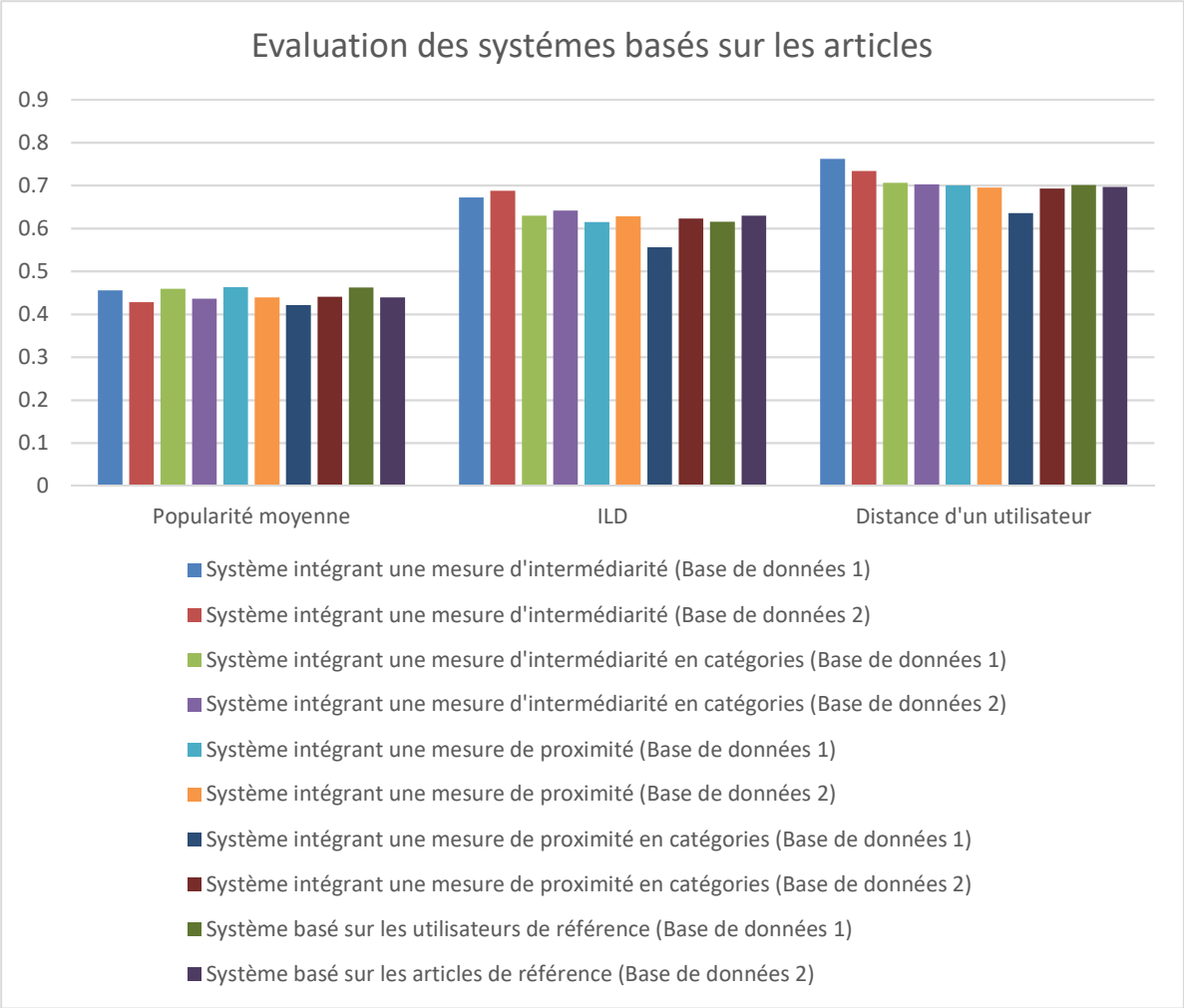

Figure 4.2 - Évaluation des systèmes basés sur les utilisateurs

En conclusion, l'intégration de mesures de centralité sur graphe a un impact sur les performances de nos systèmes de recommandation puisqu'une grande majorité de nos comparaisons de moyennes concluent à une différence statistique. De plus, deux de nos systèmes de recommandations atteignent notre objectif avec une amélioration de la nouveauté dans le cas du système de recommandation basé sur les utilisateurs intégrant une mesure de proximité ou une amélioration de la diversité dans le cas du système de recommandation basé sur les articles intégrant une mesure d'intermédiarité. L'amélioration de la performance générale d'un système de recommandation est positive à une entreprise puisque l'utilisation de ces systèmes permet d'augmenter le chiffre d'affaires des sites de e-commerce. Ainsi, augmenter la satisfaction des utilisateurs en leur proposant des articles nouveaux et divers peut améliorer l'efficacité de ces systèmes en augmentant le nombre d'articles transférés dans le panier d'achats et en diminuant le temps de recherche d'un utilisateur.

Limites et perspectives de recherche

Bien que nous ayons réalisé cette étude le plus sérieusement possible, nous pouvons identifier plusieurs limites à ce travail.

Tout d'abord, l'intégration des mesures de centralité sur graphe aurait pu se faire de bien d'autres manières. En effet, ces mesures auraient pu faire l'objet de différentes modifications avant leur intégration ou être appliquées à l'aide d'une fonction différente. Dans ce travail, nous nous sommes contentés d'intégrer ces mesures en majorité de façon linéaire avec une simple multiplication ou division par cette mesure. Il peut être intéressant de poursuivre cette recherche en modifiant l'intégration de ces mesures afin de mesurer leur impact. Les mesures de centralité pourraient également faire partie du procédé de validation croisée en les recalculant pour chaque sous-ensemble.

Ensuite, nous avons vu dans la revue littéraire qu'il existait plusieurs mesures de centralité telles que le degré, la proximité, l'intermédiarité et l'excentricité. Dans ce travail, nous n'utilisons que deux d'entre elles dans des systèmes distincts. Des recherches plus poussées pourraient expérimenter l'intégration de plusieurs mesures dans un même système ou l'utilisation de nouvelles mesures.

De plus, dans ce travail nous utilisons deux bases de données du même type. En effet, nos deux bases de données proviennent de MovieLens et caractérisent donc des notations effectuées sur des films. Il serait intéressant de réaliser la même étude sur une base de données différente afin de comparer nos résultats.

Enfin, nous avons dû faire un choix concernant les mesures d'évaluation utilisées notamment au niveau de la nouveauté et de la diversité. En effet, il existe plusieurs manières de mesurer ces métriques et ce choix influence nos résultats. Par exemple, nous avions deux mesures pour la nouveauté et nous avons constaté que ces dernières n'allaient pas toujours dans le même sens.

Bibliographie

Articles périodiques

Abdulgabber, M.A., Al-bashiri, H., Hujainah, F. & Romli, A. (2017). Collaborative Filtering Similarity Measures: Revisiting. *Proceedings of the International Conference on Advances in Image Processing*, 195-200. doi: 10.1145/3133264.3133299

Anonyme (2020). *A closer-to-reality framework for comparing relevant dimensions of recommender systems, with application to novelty*.

Antala, K., Dongare, S., Salunke, A. & Shah, K. (2017). Recommender Systems: An Overview of different approaches to recommendations. *2017 International Conference on Innovations in Information, Embedded and Communication Systems (ICIIECS),* 1-4. doi: 10.1109/ICIIECS.2017.8276172

Bahadorpour, A.M., Neysiani B.S. & Nadimi Shahraki C.M. (2017). Determining Optimal Number of Neighbors in Item-based kNN Collaborative Filtering Algorithm for Learning Preferences of New Users. *Journal of Telecommunication, Electronic and Computer Engineering 9(3)*, 163-167.

Bavelas, A. (1948). A mathematical model for group structures. *Human Organization, 7(3*), 16-30. doi: 10.17730/humo.7.3.f4033344851gl053

Bell, R., Koren, Y. & Volinsky, C. (2007). Modeling Relationships at Multiple Scales to Improve Accuracy of Large Recommender Systems. *Proceedings of the 13th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining - KDD '07*, 95. doi: 10.1145/1281192.1281206

Berrar, D. (2018). Cross-Validation. *Encyclopedia of Bioinformatics and Computational Biology, 1*, 542-545. doi: 10.1016/B978-0-12-809633-8.20349-X

Borgatti, S.P. & Everett, M.G. (2005). A Graph-theoretic perspective on centrality. *Social Networks 28*(4), 466-484. doi: 10.1016/j.socnet.2005.11.005

Bridge, D. & Kaminskas, M. (2016). Diversity, Serendipity, Novelty, and Coverage : A Survey and Empirical Analysis of Beyond-Accuracy Objectives in Recommender Systems. *ACM Transactions on Interactive Intelligent Systems, 7*(1), 1-42. doi: 10.1145/2926720

Browne, M.W. (2000). Cross-Validation Methods. *Journal of Mathematical Psychology, 44*, 108-132. doi: 10.1006/jmps.1999.1279

Burke, R., Felfernig, A. & Göker, M.H. (2011). Recommender Systems: An Overview. *AI Magazine, 32*(3), 13-18. doi: 10.1609/aimag.v32i3.2361

Castells, P. & Vargas, S. (2011). Rank and Relevance in Novelty and Diversity Metric for Recommender Systems. *Proceedings of the Fifth ACM Conference on Recommender Systems - RecSys '11*, 109. doi: 10.1145/2043932.2043955

Chandralekha, M., Sadasivam, G.S. & Saranya, K.G. (2016). Performance Comparison of Different Similarity Measures for Collaborative Filtering Technique. *Indian Journal of Science and Technology, 9*(29), 1-8. doi: 10.17485/ijst/2016/v9i29/91060

Chatterjee, M. & Davoudi, A. (2015). Product Rating Prediction Using Centrality Measures in Social Networks. *2015 36th IEEE Sarnoff Symposium*, 94-98. doi: 10.1109/SARNOF.2015.7324650

Fleder, D. & Hosanagar, K. (2007). Recommender Systems and their Impact on Sales Diversity. *Proceedings of the 8th ACM Conference on Electronic Commerce - EC '07*, 192. doi: 10.1145/1250910.1250939

Fouss, F., Pirotte, A. & Saerens, M. (2005). A Novel Way of Computing Similarities between Nodes of a Graph, with Application to Collaborative Recommendation. *The 2005 IEEE/WIC/ACM International Conference on Web Intelligence (WI'05),* 550-556. doi: 10.1109/WI.2005.9

Frasincar, F. & van Rossum, B. (2019). Augmenting LOD-Based Recommender Systems Using Graph Centrality Measures. In M. Bakaev, F. Frasincar, & I.-Y. Ko (Éds.), *Web Engineering* (Vol. 11496, p. 19-31). Springer International Publishing. doi: 10.1007/978-3-030-19274-7_2

Freeman, L.C. (1977). A Set of Measures of Centrality Based on Betweenness. *Sociometry, 40*(1), 35-41. doi:10.2307/3033543

Freeman, L.C. (1978). Centrality in Social Networks Conceptual Clarification. *Social Networks*, 1(3), 215-239. doi: 10.1016/0378-8733(78)90021-7

Gunawardana, A. & Shani, G. (2005). Evaluating Recommendation Systems. In F. Ricci, L. Rokach, B. Shapira, & P. B. Kantor (Éds.), Recommender Systems Handbook (p. 257-297). Springer US. doi: 10.1007/978-0-387-85820-3_8

Han, J. & Yamana, H. (2017). A Survey on Recommendation Methods Beyond Accuracy. *IEICE Transactions on Information and Systems, E100.D*(12), 2931-2944. doi: 10.1587/transinf.2017EDR0003

Hu, R.J., Li, Q., Ma, W.C. & Zhang, G.Y. (2015). Centrality Measures in Directed Fuzzy Social. *Fuzzy Information and Engineering, 7*(1), 115-128. doi: 10.1016/j.fiae.2015.03.008

Isinkaye, F.O., Folajimi, Y.O. & Ojokoh, B.A. (2015). Recommendation systems: Principles, methods and evaluation. *Egyptian Informatics Journal, 16*(3), 261-273. doi: 10.1016/j.eij.2015.06.005

Imran, M., Khattak, H.A., Malik, A.K., Raza, B. & Waheed, W. (2019). A Hybrid Approach Toward Research Paper Recommendation Using Centrality Measures and Author Ranking. *IEEE Access, 7*, 33145-33158. doi: 10.1109/ACCESS.2019.2900520

Jannach, D., Rook, L. & Zanker, M. (2019). Measuring the Impact of Online Personalisation: Past, Present and Future. *International Journal of Human-Computer Studies*. doi: 10.1016/j.ijhcs.2019.06.006

Khachane, A.R. & Vaidya N. (2017). Recommender Systems-The need of the Ecommerce Era. *IEEE 2017 International Conference on Computing Methodologies and Communication,* 100-114.

Kim, S.B. & Son, J. (2018). Academic paper recommender system using multiplevel simultaneous citation networks. *Decision Support Sustems., 105*, 24-33. doi: 10.1016/j.dss.2017.10.011

Luo, Y. & Zhang J. (2017). Degree Centrality, Betweenness Centrality, and Closeness Centrality in Social Network. *Proceedings of the 2017 2nd International Conference on Modelling, Simulation and Applied Mathematics*, 1951-6851. doi: 10.2991/msam-17.2017.68

Sabidussi, G. (1966). The centrality index of a graph. *Psychometrika, 31*, 581-603. doi: 10.1007/BF02289527

Wang, H. & Zhang, X. (2005). Study on Recommender Systems for Business-To-Business Electronic Commerce. *Communications of the IIMA, 5*(4), 53-62.

Articles de presse

Tourwé, T. (2012). *Recommender Systems: An Overview of the State-of-the-art*. Sirris. En ligne sur le site web Sirris, https://www.sirris.be/blog/recommender-systems-overview-state-art, consulté le 05/11/2020.

Gupta, A. (2021). *Sparse Matrix and its representations | Set 1 (Using Arrays and Linked Lists)*. En ligne sur le site web GeeksforGeeks, https://www.geeksforgeeks.org/sparse-matrix-representation/ consulté le 06/05/2021.

Livres

Aggarwal, C.C. (2016). *Recommender Systems: The Textbook*. Suisse, Gewerbestrasse: Springer International Publishing.

Croft, B., Metzler, D., & Strohman, T. (2010). *Search Engines: Information Retrieval in Practice*. Londres: Addison Wesley.

Faust, K. & Wasserman, S. (1994). *Social Network Analysis*. Cambridge: Cambridge University Press.

Fouss, F., Saerens, M. & Shimbo, M. (2016). *Algorithms and Models for Network Data and Link Analysis*. Cambridge: Cambridge University Press.

Newman M.E.J. (2010). *Networks: An Introduction*. Oxford: Oxford University Press.

Sites internet

Gephi (2021). The Open Graph Viz Platform. En ligne sur https://gephi.org/, consulté le 6 mai 2021.

JetBrain (2021). Essential tools for software developers and teams. En ligne https://www.jetbrains.com/, consulté le 23 avril 2021.

JSON (2021). Présentation de JSON. En ligne sur https://www.json.org/json-fr.html, consulté le 6 mai 2021.

Sklearn (2021). Scikit-learn. En ligne sur le site https://sklearn.org, consulté le 6 mai 2021.

Wikipédia (2021). Comma-separated values. En ligne sur le site https://fr.wikipedia.org/wiki/Comma-separated_values, consulté le 23 avril 2021.

UNIVERSITÉ CATHOLIQUE DE LOUVAIN
Louvain School of Management
Chaussée de Binche 151, 7000 Mons, Belgique | **www.uclouvain.be/lsm**