



HAL
open science

Enabling Diversity of Content Sources in Online Social Networks

Jonathan Colin

► **To cite this version:**

Jonathan Colin. Enabling Diversity of Content Sources in Online Social Networks. Social and Information Networks [cs.SI]. Université Paris-Saclay, 2025. English. ⟨NNT : 2025UPASG100⟩. ⟨tel-05523131⟩

HAL Id: tel-05523131

<https://theses.hal.science/tel-05523131v1>

Submitted on 23 Feb 2026

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.



HAL Authorization

Diversité des Sources de Contenu dans les Réseaux Sociaux

*Enabling Diversity of Content Sources in Online Social
Networks*

Thèse de doctorat de l'université Paris-Saclay

École doctorale n°580, sciences et technologies de l'information et de la
communication (STIC)

Spécialité de doctorat : Informatique

Graduate School : Informatique et sciences du numérique

Réfèrent : Faculté des sciences d'Orsay

Thèse préparée au Laboratoire interdisciplinaire des sciences du numérique (Université
Paris-Saclay, CNRS),
sous la direction de **Silviu MANIU**, Professeur

Thèse soutenue à Paris-Saclay, le 11 décembre 2025, par

Jonathan COLIN

Composition du jury

Membres du jury avec voix délibérative

| | |
|--|---------------------------|
| Johanne COHEN Directrice de recherche, Laboratoire Interdisciplinaire des Science du Numérique (LISN) | Présidente |
| Angela BONIFATI Professeure, Université Claude Bernard Lyon 1 | Rapporteur & Examinatrice |
| Cedric DU MOUZA Professeur, Conservatoire National des Arts et Metiers (CNAM) | Rapporteur & Examineur |
| Oana BALALAU Chercheur, Institut National de Recherche en Sciences et Technologies du Numérique (INRIA) | Examinatrice |
| Noha IBRAHIM Maitresse de Conference, Université Grenoble Alpes | Examinatrice |

Titre : Diversité des Sources de Contenu dans les Réseaux Sociaux

Mots clés : Web données, réseaux sociaux, optimisation dans les graphes, fouille de données

Résumé : Des événements récents ont montré à quel point le Web social est devenu important dans notre vie quotidienne. En effet, la plupart des informations ne se propagent plus par des points d'information uniques et centralisés, tels que les journaux télévisés ou papiers. Elle se propage plutôt par des moyens décentralisés, à travers les vastes réseaux sociaux. La nature décentralisée du Web signifie que les informations de toutes sortes peuvent se propager instantanément et sans filtrage, ce qui entraîne à la fois des effets positifs, comme lorsque des foules s'organisent spontanément pour atteindre des objectifs communs, et des effets négatifs, comme la diffusion de fausses informations qui peuvent être transmises comme étant valides par des consommateurs par ailleurs bien intentionnés.

Comment une fausse information peut-elle passer pour valide? Cela peut se produire lorsque les consommateurs d'informations pensent qu'ils disposent de toutes les connaissances en raison de leur voisinage social et que les informations diffusées sont représentatives du monde. Le concept est bien connu dans la littérature : on parle de bulles de filtres ou de chambres d'écho, où les consommateurs d'informations sont coincés dans un environnement qui renforce leur propre point de vue et ne le remet pas en question. Ce problème est encore aggravé par les algorithmes de recommandation commerciaux qui se concentrent essentiellement sur les articles que l'on a déjà consommés ou qui sont consommés par le voisinage social immédiat; en tant que tels, ils ne se concentrent pas sur la recommandation d'articles vraiment divers provenant de sources différentes. Cela est d'autant plus évident lorsque le réseau social est éclaté en plusieurs communautés organisées autour d'intérêts similaires, voire en communautés idéologiques qui ne communiquent pas entre elles.

Dans ce contexte, la recommandation de nouveaux contenus aux utilisateurs devrait faire plus que renforcer le comportement

passé. Notre thèse ici est qu'elle devrait également être un outil permettant d'exposer les utilisateurs à de nouveaux contenus, et donc à des points de vue multiples. D'une certaine manière, Internet devrait être considéré comme une place publique mondiale, où le contenu (y compris les produits, les idées et l'information) est échangé librement et, surtout, où chaque facette d'une information ou d'une opinion est présentée. Cette idée est soutenue par l'idée qu'un consommateur bien informé est moins susceptible d'être la proie de sources d'information douteuses. La diversité des points de vue et des opinions est doublement importante pour les produits culturels (films, livres, journaux, musique) ou pour les fournisseurs de contenu. En un sens, nous pouvons dire que notre objectif est d'augmenter la quantité de sérendipité à laquelle les utilisateurs sont exposés dans leurs interactions quotidiennes sur Internet.

Dans cette thèse, nous discutons des algorithmes pour les recommandations diverses dans les réseaux sociaux en ligne. Nous nous concentrerons sur la recommandation de fournisseurs de contenu aux utilisateurs d'un réseau social. Les réseaux sociaux sont un environnement où chaque utilisateur peut être considéré comme un créateur de contenu ou un émetteur de contenu, car ils sont composés de nœuds (les utilisateurs, qui sont à la fois consommateurs ou créateurs de contenu et réunis par des liens (les relations suivants/suivis).

Le problème devient alors assez intuitif. On nous donne un graphe social, et l'objectif est d'y ajouter des liens, de sorte que la diversité des informations propagées soit maximisée. Notre tâche consiste à recommander des liens entre les utilisateurs et les sources de contenu de manière à ce que le graphe soit divisé en un nombre de communautés aussi réduit que possible. Ceci, pour un utilisateur donné du réseau social, signifie qu'il sera exposé à autant d'autres communautés que possible.

Title : Enabling Diversity of Content Sources in Online Social Networks

Keywords : Web data, social networks, graph optimisation, data mining

Abstract : Recent events have shown how important the social Web has become to our daily lives. Indeed, most information no longer propagates through single, centralized, points of information, such as TV news or newspapers. Instead, it propagates through de-centralized means, through the large – sometimes global – social networks enabled by the Web. The de-centralized nature of the Web means that information of all sorts can spread instantly and without filtering, leading to both positive effects, such as when crowds spontaneously organize for common good goals, or negative effects, such as the spread of fake information which might be passed on as valid by otherwise well-meaning consumers.

How can fake information pass on as valid? This can occur when the consumers of information think they have all the knowledge due to their social neighborhood and that the information that is disseminated is representative of the world at large. The concept is well-known in the literature : we talk of filter bubbles or echo chambers, where consumers of information are stuck in an environment that reinforces their own point-of-view without challenging it. Compounding this problem are commercial recommendation algorithms that essentially focus on items one has already consumed or which are consumed by the immediate social neighborhood ; as such they do not focus on recommending truly diverse items from diverse sources. This is more evident when the social network is broken in several communities organized around similar interests, or even ideology-based communities that do not communicate with each other.

In this context, recommending new content to users should do more than reinforce past behavior. It is our thesis here that it should also be a tool for exposing users to new content, and hence to multiple viewpoints. In a way, the Internet should be seen as a global public square, in which content (including pro-

ducts, ideas, and pieces of information) are exchanged freely, and most importantly, where each facet of an information or an opinion is shown. This is supported by the idea that a well-informed consumer is less likely to fall prey to dubious sources of information. Diverse viewpoints and opinions are doubly important for cultural products (movies, books, newspapers, music) or for content providers. In a sense, we can say that we aim to increase the amount of serendipity that users are exposed to in their daily interactions on the Internet.

In this project, we will take the first steps in enabling diverse recommendations in online social networks. We will first focus on recommending content providers to users of a social network. To take an example of a well-known Web application where this is relevant, consider X (formerly Twitter). It can arguably be thought of as a sort of digital public square ; indeed, many events have started on Twitter or like applications (consider e.g., the Arab spring) ; in some cases, the news broke faster than on traditional media channels. Generally, the Twitter feed of a user of the application consists of the tweets of the users that they followed, or the retweets thereof. In a sense, each Twitter user can be considered either a content creator or a content transmitter (via re-tweets). The social network is hence composed of nodes (users, which are both consumers or creators of content), brought together by links (the following/follower relationships).

Then, the problem becomes rather intuitive. We are given a social graph, and the objective is to add links to it, such that the diversity of propagated information is maximized. Our task is to recommend links between users and content sources such that the graph is split in as few communities as possible. This, for a given social network user, means that they will be exposed to as many other communities as possible.

Résumé

Les réseaux sociaux occupent aujourd'hui une place centrale dans notre société et constituent une source d'information privilégiée pour une part croissante de la population. Leur rôle dans la diffusion et la circulation de l'information est ainsi devenu majeur. Contrairement aux médias traditionnels, fondés sur des points de diffusion centralisés et institutionnalisés, le Web social repose sur des structures décentralisées, au sein desquelles l'information se propage rapidement à travers de vastes réseaux d'utilisateurs. Cette transformation profonde des modes de circulation de l'information a des conséquences significatives sur la constitution et l'évolution des opinions individuelles et collectives.

L'exposition aux contenus diffusés sur les plateformes sociales influence les croyances et les jugements des individus, tant de manière active, lorsque ceux-ci consomment volontairement du contenu, que de manière passive, par la simple navigation lorsque le contenu suggéré est survolé. Ainsi, même les informations simplement vues, sans interactions explicites, participent à la construction des représentations et de la vision du monde des utilisateurs. Cette dynamique confère donc un pouvoir considérable aux algorithmes de recommandation, dont l'objectif principal est généralement d'optimiser l'engagement des utilisateurs. Pour ce faire, ces systèmes tendent à privilégier des contenus similaires aux intérêts déjà exprimés, tout en réduisant la visibilité de contenus jugés trop différents.

Dans un contexte où les utilisateurs s'appuient largement sur les recommandations algorithmiques pour découvrir de nouveaux contenus, par souci de simplicité ou par manque de temps, les systèmes de recommandations contribuent à limiter leur exposition à de l'information nouvelle et diverse. Cette dynamique favorise l'émergence de phénomènes bien documentés dans la littérature, tels que les bulles de filtres (filter bubbles) ou les chambres d'écho (echo chambers), au sein desquelles les individus évoluent dans un environnement d'information homogène, renforçant leurs points de vue préexistants sans les confronter à des perspectives alternatives. Cette situation est particulièrement problématique dans des réseaux sociaux fragmentés en communautés organisées autour d'intérêts ou d'idéologies similaires, qui communiquent peu entre elles.

La nature décentralisée du Web social, favorise une diffusion rapide et peu filtrée des informations, y compris de contenus erronés ou trompeurs. Dans des espaces d'informations homogènes, façonnés par les bulles de filtres et les chambres d'écho, ces informations peuvent être perçues comme crédibles, dès lors que les utilisateurs considèrent leur voisinage social et les contenus auxquels ils sont exposés comme représentatifs de la réalité globale. Dans

ce contexte, des contenus trompeurs peuvent être relayés et amplifiés par des individus pourtant bien intentionnés, qui ne disposent pas des éléments nécessaires pour en évaluer la véracité. L'absence de diversité des sources et des points de vue renforce ainsi cette dynamique, augmentant la vulnérabilité des individus face à des informations de faible qualité et mettant en évidence les limites des systèmes de recommandation centrés exclusivement sur l'engagement et la similarité.

Dans ce contexte, les systèmes de recommandation ne devraient pas se limiter à la reproduction ou au renforcement des préférences et des comportements passés des utilisateurs, une approche qui contribue directement à la formation de bulles de filtres et de chambres d'écho. À l'inverse, des recommandations intégrant explicitement des critères de diversité permettraient d'élargir l'espace informationnel des individus en les exposant à des contenus issus de sources variées. Une telle exposition contribue fondamentalement à limiter l'émergence et le renforcement des dynamiques de bulles de filtres et de chambres d'écho. Par ailleurs, l'accès à une information plus diversifiée offre aux utilisateurs une plus grande capacité de prise de recul, ce qui facilite l'évaluation critique des contenus et permet de mieux identifier les informations erronées ou trompeuses. Dans cette optique, les réseaux sociaux peuvent être envisagés comme de grandes places publiques mondiales, au sein desquelles la circulation équilibrée d'idées et d'informations contribue à renforcer l'esprit critique des utilisateurs et à limiter la propagation de fausses informations.

Notre étude se concentre sur la diversité de l'exposition à l'information et sur la manière dont celle-ci évolue dans un environnement de réseau social au fur et à mesure que les recommandations sont proposées aux utilisateurs. Plus précisément, lorsque les recommandations reposent sur une logique de similarité, la diversité des contenus auxquels les utilisateurs sont exposés tend à diminuer, ce qui peut renforcer les bulles de filtres et les chambres d'écho. En revanche, lorsque les recommandations suivent une logique de diversité, l'exposition à des points de vue et à des informations variés est favorisée, ce qui contribue à élargir le spectre informationnel des utilisateurs. Il s'agit donc de définir de manière rigoureuse la notion de diversité dans le contexte spécifique des réseaux sociaux, pour qu'elle soit mesurable, afin d'évaluer les performances des différents systèmes de recommandation en termes de diversité de l'exposition à l'information.

Pour ce faire, nous modélisons le réseau social sous la forme d'un graphe, où les nœuds représentent les utilisateurs et les arêtes les relations qui les relient. Les structures de communautés au sein de ce graphe sont utilisées pour représenter les différentes opinions présentes dans le réseau. En effet, deux individus appartenant à une même communauté tendent à partager des opinions plus proches que deux individus situés dans des communa-

tés distinctes. La propagation de l'information, et par extension des opinions, s'effectue au travers des arêtes reliant les utilisateurs. Par conséquent, l'exposition à l'information d'un utilisateur particulier dépend directement de la composition de son voisinage. Un utilisateur connecté à plusieurs communautés bénéficie ainsi d'une exposition à l'information plus diversifiée que celui qui reste confiné à sa propre communauté.

La topologie du graphe permet de quantifier cette exposition : nous pouvons calculer la distribution de l'exposition de chaque utilisateur à l'ensemble des communautés, puis en mesurer la diversité au moyen de l'entropie de cette distribution. L'entropie, concept issu de la théorie de l'information, fournit une mesure de la quantité d'information contenue dans un message ou, dans ce contexte, de la diversité des opinions auxquelles un utilisateur est exposé. Plus l'entropie est élevée, plus l'exposition est variée, reflétant ainsi une pluralité de perspectives au sein du réseau social.

Intuitivement, l'objectif consiste donc à maximiser l'entropie, afin de favoriser une exposition à l'information aussi diversifiée que possible. Toutefois, en s'appuyant sur des propriétés bien établies des fonctions d'entropie et sur le comportement mathématique de celle-ci, il est possible de reformuler ce problème de maximisation sous la forme équivalente d'un problème de minimisation. Cette transformation présente des avantages considérables, notamment en raison des propriétés de convexité qu'elle induit. Ces propriétés constituent le fondement de notre première approche, dans laquelle nous cherchons à optimiser l'exposition à l'information en utilisant une méthode de descente de gradient. Pour permettre l'application d'un tel algorithme, il est nécessaire de disposer d'une fonction objectif à la fois convexe et différentiable. En exploitant les formulations matricielles des graphes, il devient possible de représenter l'exposition à l'information sous la forme d'une fonction convexe différentiable, dont les dérivées sont non seulement bien définies, mais également calculables de manière explicite. La mise en œuvre de ce processus requiert toutefois des techniques d'optimisation avancées afin de réduire la complexité associée au calcul des dérivées des matrices.

Malgré ses avantages théoriques et computationnels, l'application d'un algorithme fondé sur la descente de gradient à un réseau social présente plusieurs limitations. Tout d'abord, ces méthodes reposent généralement sur des hypothèses de convexité et de différentiabilité qui peuvent simplifier excessivement la complexité des dynamiques sociales réelles, lesquelles sont souvent non linéaires, hétérogènes et surtout évolutives dans le temps. Par ailleurs, la structure des réseaux sociaux est intrinsèquement dynamique : les utilisateurs, leurs relations et les contenus évoluent continuellement, ce qui peut rendre une solution optimisée à un instant donné rapidement obsolète.

Par ailleurs, une optimisation exclusivement centrée sur la diversification de l'exposition à l'information peut entrer en conflit avec les exigences de per-

sonnalisation des recommandations. En particulier, il apparaît peu réaliste de proposer des contenus ou des liens sans tenir compte des préférences individuelles des utilisateurs, ni des objectifs de performance des plateformes, lesquels évoluent dans le temps et dépendent fortement du contexte d'usage. Ces tensions mettent en évidence les limites des approches statiques et unidimensionnelles et soulignent la nécessité de recourir à des méthodes plus flexibles, capables d'apprendre de manière continue les dynamiques propres au réseau social ainsi que les comportements et préférences de ses utilisateurs.

Dans ce contexte, l'apprentissage par renforcement constitue une approche particulièrement adaptée pour modéliser et optimiser les systèmes de recommandation dans les réseaux sociaux. Contrairement aux approches d'optimisation statiques, il repose sur l'interaction séquentielle entre un agent et un environnement dynamique, au sein duquel l'agent apprend progressivement une politique de décision à partir des retours qu'il reçoit sur les actions qu'il a pris. Cela permet d'intégrer simultanément des objectifs multiples et potentiellement contradictoires, tels que la personnalisation des recommandations et la diversité de l'exposition à l'information.

En apprenant de manière continue à travers l'interaction avec l'environnement, l'apprentissage par renforcement est en mesure de s'adapter de façon flexible à des environnements non seulement dynamiques, mais également partiellement observables, dont les caractéristiques internes n'ont pas nécessairement besoin d'être explicitement intégrées au processus d'optimisation du modèle. Cette propriété permet, d'une part, de répondre efficacement aux changements rapides et parfois drastiques qui caractérisent les réseaux sociaux. D'autre part, elle dispense le modèle de la nécessité de disposer d'une représentation exacte et explicite des opinions présentes au sein du réseau, contrairement aux méthodes d'optimisation statiques qui requièrent, en particulier dans l'étude des dynamiques d'opinion, une estimation précise des positions individuelles des utilisateurs. Or, une telle estimation est généralement difficile, coûteuse et sujette à de fortes incertitudes, ce qui renforce l'intérêt d'approches adaptatives fondées sur l'apprentissage par renforcement.

Par conséquent, notre seconde approche s'appuie sur une combinaison de l'apprentissage profond et de l'apprentissage par renforcement, à travers l'utilisation du modèle Structure2Vec Deep Q-Learning (S2V-DQN). Ce cadre permet d'exploiter des représentations latentes apprises automatiquement à partir de la structure du graphe, tout en optimisant une politique de décision par apprentissage par renforcement. En intégrant l'information topologique du réseau social dans un espace de représentation compact, S2V-DQN offre une approche adaptée à la complexité et à la dynamique des réseaux sociaux, sans nécessiter une modélisation explicite et exhaustive des opinions individuelles. Cette méthode constitue ainsi un compromis pertinent entre

expressivité du modèle, capacité d'adaptation et faisabilité computationnelle dans des environnements de grande dimension.

Avec cet outil, la modélisation de l'environnement joue un rôle central, et en particulier la définition de la fonction de récompense, qui conditionne directement le comportement appris par l'agent. Celle-ci doit refléter un compromis entre des objectifs antagonistes, en combinant à la fois la satisfaction des utilisateurs, estimé avec une sémantique de similitude, et la diversité de l'exposition à l'information. Une telle formulation permet d'encourager des politiques de recommandation qui restent pertinentes pour les utilisateurs tout en favorisant une ouverture à des contenus et des points de vue variés. En intégrant ces dimensions dans le signal de récompense, l'agent est ainsi incité à apprendre des stratégies équilibrées, capables de s'adapter aux dynamiques du réseau social tout en limitant les effets des bulles de filtres et des chambres d'écho.

Dans l'ensemble, cette thèse propose un cadre méthodologique visant à analyser et à améliorer la diversité de l'exposition à l'information dans les réseaux sociaux, en apportant des solutions innovantes ancrées dans des fondements mathématiques solides, tels que l'optimisation convexe et l'apprentissage par renforcement. Ces approches permettent de prendre en compte à la fois les dynamiques structurelles du réseau et les comportements des utilisateurs, offrant ainsi un cadre cohérent et adaptable pour l'étude et l'amélioration des systèmes de recommandation.

Table des matières

| | |
|---|-----------|
| Résumé | 5 |
| 1 General Introduction | 13 |
| 2 Related Work | 19 |
| 2.1 Recommender Systems | 19 |
| 2.1.1 Traditional Recommender Systems | 19 |
| 2.1.2 Deep learning Recommender Systems | 20 |
| 2.1.3 Reinforcement Learning Recommender Systems | 21 |
| 2.2 Graph Representation Learning | 23 |
| 2.2.1 Node embeddings | 23 |
| 2.2.2 Graph embeddings | 23 |
| 2.3 Influence Maximization | 24 |
| 2.3.1 Diffusion models | 24 |
| 2.3.2 SzV DQN | 25 |
| 2.4 Social Interactions on Social Media | 26 |
| 2.4.1 Online communities | 26 |
| 2.4.2 Graph diameter reduction | 27 |
| 2.4.3 Recommendation dynamics | 27 |
| 2.4.4 Fairness and diversity | 28 |
| 2.4.5 Opinion dynamics | 31 |
| 2.4.6 Polarization | 31 |
| 2.4.7 Misinformation Propagation | 32 |
| 2.4.8 Serendipity | 33 |
| 2.5 Entropy in Graphs | 33 |
| 2.6 The Industry's perspective | 34 |
| 3 Formal Setting | 37 |
| 3.1 Graph Framework for Social Networks | 37 |
| 3.2 Diverse Exposure Framework | 39 |
| 3.3 Notations | 43 |
| 4 Optimizing Diverse Information Exposure in Social Networks | 45 |
| 4.1 Introduction | 45 |
| 4.2 Problem definition and properties | 46 |
| 4.3 Convex optimization | 47 |
| 4.4 Greedy approaches | 50 |
| 4.5 Computational optimization | 52 |
| 4.6 The user-content graph extension | 54 |

| | | |
|----------|---|------------|
| 4.7 | Experiments | 58 |
| 4.8 | Conclusion | 66 |
| 5 | Reinforcement Learning for Optimizing Diverse Exposure | 67 |
| 5.1 | Introduction | 67 |
| 5.2 | Problem definition | 68 |
| 5.3 | Formulation | 69 |
| 5.3.1 | The general RL framework | 69 |
| 5.3.2 | The DQN framework | 70 |
| 5.3.3 | Using DQN with graphs | 71 |
| 5.4 | Implementation | 74 |
| 5.5 | Experiments | 76 |
| 5.5.1 | Base Settings | 78 |
| 5.5.2 | Transfer Learning | 79 |
| 5.5.3 | Satisfaction as condition for edge adoption | 82 |
| 5.5.4 | Recommendation with unfollowing | 83 |
| 5.5.5 | Conclusion | 86 |
| 6 | Conclusion and Perspectives | 89 |
| 6.1 | Summary of Findings | 89 |
| 6.2 | Perspectives and Future Work | 90 |
| 7 | Annexes | 109 |
| 7.1 | User-Content setting derivations | 109 |
| 7.2 | Supplementary Results for Chapter 4 | 112 |

1 - General Introduction

The rise of the social Web as a cornerstone of modern society has fundamentally altered the architecture of information dissemination. Indeed, traditional media structures such as newspapers, television broadcasts or radio programs are continuously losing ground to social media posts and podcasts. Once curated by official organisms and expert knowledge, news broadcast and documentaries are now in the hand of content creators and any user can share their personal point of view and opinions. This decentralized structure facilitates instantaneous and unfiltered propagation of information. On one hand, it yields significant benefits, exemplified by the rapid mobilization of crowds for a good cause or when time is of the essence in the context of an emergency. On the other hand, it introduces considerable risks, most notably the rampant spread of misinformation and disinformation, which can be unwittingly amplified by well-intentioned individuals, thereby eroding public trust. Indeed, fueled by anonymity, some users feel free to disseminate hate speech or fake news in spite of each social platform's terms of use and always finding ways to bypass their filters. This is further exacerbated by the recent popularization of generative AI. Fake accounts called "bots" appear more and more human and the practice of "botting" where people fake an audience or a movement to support their ideas, is becoming increasingly challenging to deal with. Public trust is being further damaged by generative AI's ability to artificially construct realistic pictures and videos as well as manipulating live-streams.

In parallel, social media platforms make use of recommender systems which function as sophisticated information filters, strategically curating the content presented to users to maximize platform engagement and avoid information overload [1, 2]. Indeed, the more a user spends time on a given platform the more revenue it generates through ads, subscriptions or promotional contents. These systems operate by constructing a dynamic personalized user profile derived from their explicit and implicit behaviors. Explicit criteria include clicks, likes, shares and followership while implicit signals revolve around dwell time or attention retention. This data is then used with various techniques presented in Section 2.1 to suggest contents with features similar to those a user has previously interacted with as these are the one they are the most likely to interact with in the future. At the same time, having no record of past interactions with content differing from what the user has already engaged with makes the system unable to predict user engagement with this new content, hence only capitalizing on semantics of similarity and optimizing accordingly to predict accuracy. Subsequently, recommender systems prioritize content that is similar to the users' current interests, while actively *hiding*

content that is too dissimilar. Consequently, users who depend on these systems for discovery, usually for reasons of convenience or efficiency, are not exposed to a genuinely diverse information landscape. Instead, their exposure is narrowed, reinforcing existing preferences and limiting serendipitous encounters with novel ideas.

This is the basis for a well-documented loop that leads to the formation of *echo chambers* and *filter bubbles*. An *echo chamber* is “a bounded, enclosed media space that has the potential to both magnify the messages delivered within it and insulate them from rebuttal” [3]. When recommender systems only provide like-minded content and fail to provide any counter-point, it gives a false confirmation bias. The trapped user believes they have a complete overview of the entire landscape of a given topic when in reality their point of view is actually limited. This applies to any idea : ambiguous, benign, malign, controversial, or widely accepted. The concept has been mostly studied in the context of political debates and discussions where it has clear immediate damaging effects [4, 5]. On the other hand, a *filter bubble* is a closed media space resulting from the personalized recommendations of the algorithmic ranking systems. While an echo chamber is largely a social construct fortified by user choice, a filter bubble is an algorithmic construct, actively created for the user by the platform’s underlying algorithm. Both phenomena, however, converge in their capacity to limit exposure to diverse viewpoints. As such, experts usually study echo chambers as they are linked to increasing polarization [6], a state where people are growing further apart until there is no more common ground. A fragmented social environment facilitates the propagation of fake news, disturbs public discourse and inflames public tensions.

Indeed, in such an environment, consumers of information wrongly believe to have full knowledge because they assume that the information that is disseminated is representative of the world at large. However, their immediate social neighborhood echos similar opinions and creates a confirmation bias giving a false sense of security in their opinions. In fact, they are stuck in an environment that reinforces their own point-of-view without challenging it. If fake news were to be propagated in such enclosed spaces, it would immediately be taken as true and remain so because no counter point would be provided.

Hence, recommender systems and their algorithms that maximize user engagement by providing content based on similarity have been blamed for these emerging issues. Even though some experts question this [7, 8], most do agree that there is an undeniable link between growing social fragmentation and recommender systems and that their semantics need to change.

Researchers have looked into algorithms that optimize networks for the minimization of polarization but these structural interventions are difficult to implement realistically. How would one force a user to befriend another or

to consume a specific content item? Similarly, attempts at content-level interventions have been studied but these approaches also suffer severely from their similarity to censorship [9, 10]. This exemplifies the difficulty of dealing with such problems. Indeed, moving the focus of recommender systems to a new semantic impedes their current goal, which can severely impact their performances leading to business costs for the social network companies. Even if it could also lead to totally new traffic by users interested in diverse and novel content.

In spite of these challenges, we believe that a new semantic for recommendation based on promoting diverse content would at least help alleviate these issues if not greatly decrease their effects. Indeed, promoting diverse content to users would continuously present novel and different ideas as well as similar ones. This would by definition reduce the *filter bubble* effect as well as limit the formation of echo chambers as multiple viewpoints would always be available. Making users exposed to more varied sources and opinions will eventually make them more informed and as such, less likely to fall prey to misinformation and disinformation. Diversity would also benefit content providers as even more niche interests would also be promoted. In other words, this would promote creativity and increase serendipity.

In this thesis, we will take algorithmic steps in enabling diverse recommendations in online social networks. We will first focus on recommending content providers to users of a social network. To take an example of a well-known Web application where this is relevant, consider X (formerly Twitter). It can arguably be thought of as a sort of a digital public square. Generally, the feed of an X user consists of the tweets written or shared by other accounts that they follow. In a sense, each X user can be considered either a content creator or a content transmitter (via re-tweets) and at the same time a content consumer. The social network is hence composed of nodes (users, which are both consumers and/or creators of content), brought together by links (the following/follower relationships). Thus, the problem becomes rather intuitive. We are given a social graph and the objective is to add links to it, such that the diversity of propagated information is maximized. Our task is to recommend links between users and content sources such that the graph is split in as few communities as possible. This, for a given social network user, means that they will be exposed to as many other communities as possible. An example is given in Fig. 1.1 : when the node identified by the red square gets linked to the red diamond nodes — part of other communities —, it is directly exposed to more, diverse, sources of information.

Our first contribution is a structural intervention algorithm that optimizes information exposure in two distinct social graphs settings. The first setting is the standard user-user graph where only users and their connections are represented. We extend our framework to also be applicable to the second

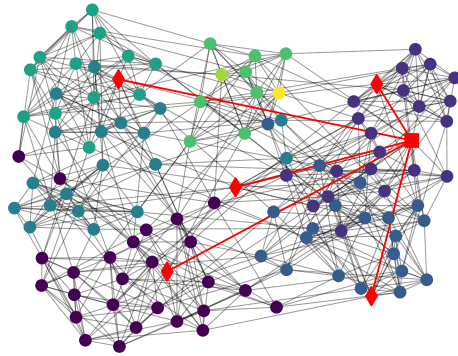


Figure 1.1 – Example of diverse recommendations (Football dataset). Node colors represent communities as computed by a greedy modularity algorithm. The red square is the target node and red diamonds are diverse recommendations, i.e., from other communities.

setting of user-item graphs. This setting is different in that there are now two distinct information propagation pathways. Using matrix representation of the graph we formulate the objective as a generally convex function and use an online gradient descent algorithm to efficiently compute the optimal solution, i.e. the edge that best improves information exposure for a given user. However, this approach shares a notable shortcoming with most polarization minimization algorithm in that it needs explicit opinion vectors for its optimization process which are private information in a real world setting. Similarly to other polarization minimization algorithm we question their applicability in an actual setting. This leads us to our second contribution : a framework that can optimize information exposure that stays agnostic of private opinion vectors. We formulate this problem a Markov Decision Process (MDP) and design a reinforcement learning framework that combines both a diversity objective and a satisfaction objective based on similarity. This combination encourages the agent to provide not only diverse recommendation, but also satisfying recommendation.

Thesis outline This thesis studies the broad subject of diversity in social networks. In particular, we propose a novel framework to measure diversity from the point of view of users by quantifying information exposure and introduce the problem of diverse information exposure maximization. Subsequently, we propose two new frameworks, inspired by state of the art techniques, to solve this problem. This manuscript is structured as follows :

- Chapter 2 provides an extensive presentation of the general landscape of social networks and the many fields of research relevant to the discussion of diversity in social networks. First, we present the central com-

ponent of recommender systems and how recommendations are generated. Starting with traditional techniques and how they evolved to the deep learning methods currently used. We also discuss the graph representation techniques that enable and improve them. Then we present influence maximization as a problem that deals with notions similar to those of diversity. Finally, we present the many social behaviors patterns and interactions that take place in social networks. Starting with how users behave and how it leads to communities. We also discuss how recommender systems are prone to various biases and how these biases are fought by enforced fairness. We introduce the opinion dynamics models that are used to study how opinions evolve in a social setting and how they are used to measure polarization in social networks. This will also explain how echo chambers and filter bubbles emerge in such environments. We will also briefly discuss the emerging topic of serendipity in recommender systems and its shortcomings.

- Chapter 3 is a formal introduction to the many mathematical tools and graph notions that are necessary for a proper understanding of the works presented in this thesis. We will also develop the framework we propose to measure information exposure which is the foundation of how we study diversity in social networks.
- Chapter 4 introduces the problem of diverse information exposure maximization and how it can be translated to a minimization problem using convex properties of the objective function. We also provide an extension to the setting of user-item graphs where multiple pathways of information propagation have to be considered. Subsequently, we propose a gradient descent based algorithm to solve these problems and compare it to various baselines from the related work, standard social networks approaches and other intuitive diversity methods.
- Chapter 5 continues the contribution of chapter 4 by addressing its limitations. To do so, we build upon the framework of structure2vec deep Q-learning network to formalize a multi-objective reinforcement learning setting and extensively test it.
- Chapter 6 summarizes this thesis's key discoveries and its contributions as well as suggesting promising avenues for extending this research and addressing its limitations.

Publications The research work during this thesis has led to three published papers : one international conference paper, one national paper and one international journal paper.

- International Papers
 1. **Jonathan Colin** and Silviu Maniu. "Optimizing Diverse Information Exposure in Social Graphs." 2024 IEEE International Conference on Big Data (BigData).
 2. **Jonathan Colin** and Silviu Maniu. "Maximizing Diverse Information Exposure in Content-Based Social Networks." 2025 Transactions on Large-Scale Data- and Knowledge-Centered Systems LIX Special Issue on Data Management - Principles, Technologies, and Applications (TLDKS)
- National Conferences
 1. **Jonathan Colin** "Enabling Diverse Recommendations in Social Graphs", 2023. PhD Poster at 39ème Conférence sur la Gestion de Données – Principes, Technologies et Applications (BDA)
 2. **Jonathan Colin** and Silviu Maniu. "Optimizing Diverse Information Exposure in Social Graphs.", 2024. 40ème Conférence sur la Gestion de Données – Principes, Technologies et Applications (BDA)

Contribution Our proposed methods are implemented in python and the code is publicly available at the following github page :

github.com/Jonathan-COLIN/

2 - Related Work

Social medias are platforms that encapsulate many complex interactions, processes and relationships, as such they have caught the attention of many researchers to deal with tasks across a large variety of domains. In this section we are going to present a list of problems and tasks most relevant to the issue of diversity in social media, beginning with recommender systems and other social tasks such as influence maximization, as well as current deep learning and reinforcement learning techniques. We will illustrate how these algorithms play a role in current concerns over social fragmentation in social networks and discuss various branches of research that aim at alleviating these issues.

2.1 . Recommender Systems

Recommender systems (RS) are at the core of many online platforms such as shops (Amazon), streaming services (Netflix) or social networks (Facebook). At a surface level, their task is to provide content but the main goal is to maximize platform engagement [11, 12]. To do so, they tailor content suggestion to each user depending on their interests as well as their history on the platform. Over the years many different techniques have been developed :

2.1.1 . Traditional Recommender Systems

Traditional recommendation methods are primarily divided into neighborhood models and collaborative filtering. Neighborhood models operate directly on a graph or interaction matrix by aggregating information from a user's or item's immediate local connections (e.g., the items a user has already interacted with) to generate suggestions [13, 14]. In contrast, collaborative filtering leverages the global pool of historical interactions across all users to infer preferences; it identifies latent patterns by comparing a user's behavior to similar users (user-based [15]), an item's audience to similar items (item-based [16, 17]) or a combination of both (hybrid methods [18, 19]) thereby uncovering affinities beyond direct local links. Collaborative filtering used to be one of the most successful techniques for recommender systems, we refer the readers to this survey [20] for a more in-depth overview of collaborative filtering's variants. However, collaborative filtering suffered from data sparsity as most users rate or interact with only a small fraction of available items and scalability issues as datasets grew.

Collaborative filtering was later improved with matrix factorization techniques [21, 22] that allowed the system to uncover hidden patterns and make

more accurate predictions by representing users and items as latent vectors in a shared low-dimensional space. Matrix factorization not only helped to generalize better but also enabled efficient large-scale optimization using algorithms like stochastic gradient descent [23, 24, 25]. Matrix factorization assumes linear relationships between users and items, limiting its expressiveness and struggling to incorporate contextual information. The latent representation it learns are hard to interpret, making the system less explainable. These techniques were particularly affected by the *cold-start* problem [26], in which recommendations for newly introduced users or items were often inaccurate. Additionally, these methods relied on simple similarity measures or linear latent models, which limited their ability to capture complex, nonlinear relationships in user behavior.

To overcome these constraints, researchers introduced model-based recommender systems that used machine learning techniques to learn predictive functions. Machine learning methods such as regression models [27] with multi-layered perceptrons [28] or auto-encoders [29, 30], clustering [31, 32] and support vector machines [33] could learn from diverse data sources including user demographics, item attributes and contextual signals. These models improved prediction accuracy, scalability and adaptability, allowing systems to capture nonlinear relationships and incorporate side information for more personalized recommendations.

2.1.2 . Deep learning Recommender Systems

We live in the information era [34], and the access to tremendous amount of data has led to Deep Learning techniques gaining in popularity for a vast array of modern tasks. These recent improvements in Deep learning techniques have also been very useful for recommender systems due to their powerful ability of learning complex and non-linear relationships. Initial deep learning approaches in recommender systems encompass two principal architectures : Convolutional Neural Networks (CNNs) [11, 35, 36], recognized for their efficiency in learning local patterns and feature extraction from unstructured data; and Recurrent Neural Networks (RNNs) [37, 38, 39], which are particularly effective at deriving predictions from sequential data.

RNN are particularly useful for session-based recommendations, where the system predicts a user's next action based only on the current session's sequence of interactions, without long-term user history. This can prove useful in contexts where users are anonymous or their past data are unavailable, such as e-commerce or media platforms. RNNs capture the order and context of interactions, learning dynamic patterns that traditional collaborative filtering cannot. As a result, RNN-based models significantly improve the accuracy and relevance of real-time, context-aware recommendations.

In combination with recent improvements in graph representation lear-

ning, CNNs have given rise to graph neural networks (GNN) which have become the current state-of-art for recommender systems [40]. Indeed, taking advantage of low-dimensional encoding of nodes, edges or entire graphs, GNNs are able to capture higher order relationships using information propagation. In other words they can capture even more complex user-item relationships. Several architectural variants exist, distinguished by their method of aggregating neighborhood information. Graph Attention Networks (GAT) employ an attention mechanism to weight the influence of neighbors based on their relevance [41, 42], whereas Graph Convolutional Networks (GCN) assign equal weight to all neighboring nodes [43]. To explore graph neural networks and their various extensions in greater depth, readers may consult the following surveys [44] for more recent and general discussion and [45] for specifically the application of GNNs in social networks.

2.1.3 . Reinforcement Learning Recommender Systems

While Deep learning techniques are powerful, especially when combined with graph encoding and embedding techniques, their training is targeted at immediate prediction accuracy, favoring immediate results instead of long term efficiency. It is well known that social platforms as well as the interactions held on them are fast evolving and this is why reinforcement learning (RL) is also quite popular for recommender systems. Indeed, RL optimizes long-term rewards and is able to identify complex unstructured relationships. RL agents learn by continuous exposure to and interaction with an environment, receiving a reward or a penalty depending on their actions. Subsequently, the agent develops a policy, i.e. a strategy to identify which action to take, through trial and error aiming to maximize the reward [46, 47].

RL is based on the Markov Decision Process (MDP) [48] framework which formalizes sequential decision making problems. It is based on 4 key components : state space, action space, transition function and a reward function. The state space contains all possible states the agent can be in, the action space consists of all the actions the agent can take, the transition function gives the probability of moving from one state to another and the reward function returns a scalar value for a given state and action which reinforces desired behaviors and discourages undesired behaviors. Eventually, the MDP finds a policy – a mapping that given a state returns an action – that maximizes the reward, in other words that maximizes desired behavior. Initial RL methods include multi-armed bandits, setting that views each item as an "arm" akin to a slot machine's arm that can be pulled and provide a reward [49, 50]. Contextual bandits builds upon the previous approach by providing item features to enabling a more personalized model [51, 52]. A defining characteristic of these algorithms is that they are essentially one-state algorithms.

Among the key algorithms in RL is Q-learning, which directly estimates the

optimal action–value function [53, 54]. The learned values define a greedy policy, in which the agent exploits its current knowledge by selecting, in each state, the action with the highest estimated value. Q-learning is an off-policy method, as it learns the value of the optimal policy independently of the agent’s behavior during learning, allowing the agent to follow exploratory actions without affecting the policy being learned. A prominent variant is the SARSA model (State–Action–Reward–State–Action) [55], an on-policy algorithm that instead learns the value of the same policy used to interact with the environment, thereby directly accounting for the effects of exploration when estimating action values. Other approaches include Monte Carlo methods, which estimate expected returns by averaging rewards obtained over complete episodes [56, 57, 58].

Traditional Reinforcement Learning techniques possess distinct advantages over non-RL approaches. They are inherently designed to optimize for long-term cumulative benefits and offer significant flexibility [59]; a well-crafted reward function can encapsulate multiple objectives, enabling their simultaneous optimization [60, 61, 62]. However, these techniques are also characterized by notable weaknesses, including the cold-start problem [63, 64](i.e. the system cannot provide useful recommendation when the user or item is new), scalability issues arising from the curse of dimensionality in large state-action spaces and a limited capacity to handle partially observable environments where the agent lacks full state information [59].

To address these shortcomings, Deep Learning tools have been implemented [65, 66], using a deep neural network to replace the state-action space and to estimate the policy and value function. This is particularly relevant in the context of RS as deep reinforcement learning models can handle large and high dimensional state spaces which is the case of social networks. The most popular approach is the deep Q-network (DQN) which is a combination of Q-learning and a deep neural network [67, 68]. It can take advantage of experience replay which is a memory buffer of past interactions (state-action reward) and is randomly sampled from during the training, meaning that consecutive interactions are decorrelated and that data can be used multiple times so that more useful instances of data can also be re-used. A second network is used as a target network, this decouples online training and prediction. Variants have been proposed to minimize the shortcomings of this framework such as double-DQN (DDQN) [54, 69] which separates action selection and evaluation which reduces the overestimation bias generally present in DQN. Dueling DQN methods [70] split the Q-function into two streams with one estimating the state value and the other the advantage of each action. Another popular framework is the actor-critic model [71] where two network are trained, one learning the policy and the other the value function. In other words, while the first network learns to take actions the second network learns to eva-

uate said actions. The most popular method from this framework are Deep Deterministic Policy Gradient (DDPG) models [72]. In addition to its actor and critic networks, it also has a target network for prediction as well as a replay memory.

2.2 . Graph Representation Learning

Graph representation learning is a branch of machine learning focused on learning low-dimensional representations of graph-structured data while preserving the inherent relationships between nodes and edges. Graphs are a natural way to model complex systems, capturing entities as nodes and their interactions as edges. Traditional machine learning struggles with graph data due to its irregular structure, but graph representation learning techniques enable the use of standard learning models by transforming nodes, edges, or entire graphs into dense vector embeddings that encode structural and semantic information. As such graph representation has become a cornerstone of Deep Learning for any graph-based tasks such as molecular property prediction [73], Traffic flow prediction [74] and in particular recommender systems. The compact and meaningful embeddings empower agents to make more informed decisions, ultimately improving generalization and scalability within complex, graph-structured environments.

2.2.1 . Node embeddings

Node embeddings aim to represent each individual node in a graph as a vector in a continuous space, capturing its local neighborhood structure and properties. Initial techniques are inspired from natural language processing and embeddings used for text. The foundational skipgram [75] and word2vec [76] inspired DeepWalk [77], comparing nodes in a graph as words in a sentence and sampling random walks to generate a node embedding. DeepWalk lead to node2vec [78] which extends DeepWalk with better control breadth-first or depth-first random walk in order to have embeddings relying more on either structural features or community features. More recent node embedding techniques use message passing with GNNs such as GraphSAGE [79] or PinSage [43] to aggregate information from the neighborhood in order to generate an encoding for all nodes.

2.2.2 . Graph embeddings

Unlike node embeddings, which focus on individual nodes and their neighborhood, graph embeddings aggregate information from all nodes and edges to form a holistic representation of the graph. Similarly to node embeddings, initial techniques were also inspired from natural language processing. Doc2vec [80] lead to graph2vec [81], comparing a document to a graph and words

to node, computing a distribution of words/nodes provides an embedding for the document/graph. Modern techniques use message-passing or aggregation mechanisms, as seen in models like Graph Convolutional Networks [82], Graph Attention Networks [42] and Graph Isomorphism Networks [83], which iteratively combine node features based on their connectivity patterns. The resulting embeddings preserve global characteristics such as community structure, connectivity and feature distribution.

2.3 . Influence Maximization

Acting as hubs for the dissemination of information and facilitating abundant interactions, social networks have been identified to be very useful for viral marketing, political campaigns and general health interventions. As such, there is extensive research on the subject of influence maximization (IM) [84, 85, 86], the NP-hard problem of identifying key individual in the network that maximizes the spread of influence of a campaign. IM includes many variations such as topic aware influence maximization (TIM) [87] which adds information about topics so that nodes are biased towards specific interests, competitive influence maximization [88] which simulates multiple campaigns against each other on the same network at the same time or influence blocking maximization [89] which aims at preventing or limiting the spread influence that is undesirable, like viruses, malwares, misinformation or hate speech.

2.3.1 . Diffusion models

The influence of an individual is given by a diffusion model which characterizes how influence spreads over a social network. The two most popular models in the academic literature are linear threshold (LT) [90] and independent cascade (IC) [84]. IC assumes that each activated node has a single chance to activate each neighboring node with a given probability and LT assumes that a node only activates if the combined influence from its neighbors exceeds a threshold. Both these propagation models include many variations to better capture real-world diffusion dynamics. Extensions include the Weighted Cascade [84], Generalized Threshold [90] and Continuous-Time IC models [91], which incorporate heterogeneous influence strengths, temporal factors and adaptive activation probabilities. In particular, topic-aware models such as the Topic-Aware Independent Cascade and Topic-Aware Linear Threshold models [92] which integrate semantic or contextual information by associating each edge or user with topic-specific influence probabilities. This allows the diffusion process to reflect content relevance and user interests, providing a more realistic representation of how information spreads in social and recommendation networks.

Subsequently, the spread of a given individual can be estimated through

Monte Carlo simulation [93]. Then a greedy strategy is used to select the best candidate with the highest spread at each step, such methods are guaranteed to be within $(1 - \frac{1}{e})$ of the optimal solution as the influence spread function is monotonous and submodular [94]. Another standard approach is to use heuristics such as the PageRank [95, 96], betweenness or degree centrality to provide a score and approximate a solution. However, these methods lack severely in scalability as well as suffering from unknown diffusion probabilities.

Learning-based approaches for influence maximization aim to overcome the limitations of traditional diffusion models (like Independent Cascade or Linear Threshold) by learning influence patterns directly from data rather than relying on predefined propagation probabilities. Previously mentioned frameworks such as graph neural networks, graph attention networks or graph convolutional networks are utilized to capture higher-order relational dependencies leading to more accurate and adaptive influence prediction in real-world social networks. We refer the reader to the following survey [97] for an in-depth study of such methods for influence maximization.

2.3.2 . S2V DQN

We will discuss the Structure2Vec Deep Q-Network (S2V DQN) reinforcement learning framework [98] for influence maximization. It combines two key components : Structure2Vec (S2V) for graph representation learning and a Deep Q-Network (DQN) for decision optimization. The influence maximization problem is then modeled as a sequential decision process, where the agent selects one seed node at a time to maximize the cumulative influence spread. Using message passing techniques, the S2V module generates node embeddings which encode both the graph topology and the current diffusion state. The resulting embedding of each node is then input into a Q-network, which estimates the expected marginal gain of selecting that node as the next seed. The DQN learns to approximate the Q-function using feedback from simulated diffusion outcomes, updating its parameters via experience replay and target networks, stabilizing learning in the same way as in standard DQN. Once trained, S2V-DQN can generalize to unseen graphs and efficiently infer near-optimal seed sets without explicit diffusion simulation. This framework's key innovation lies in its end-to-end learning of both graph embeddings and seeding policies, enabling it to outperform traditional greedy and heuristic algorithms, particularly on large or complex network structures. The first implementation of this framework for influence maximization is PIANO [99] and more relevant to our research is its application to topic aware influence maximization [100, 101]. Indeed, in this setting, S2V DQN integrates topic information into its graph representation and decision-making process. Each node's embedding thus captures both its structural role in the network and its semantic relevance to specific topics. The Deep Q-Network then receives

these enriched embeddings and learns a policy that dynamically selects seed nodes to maximize influence spread under topic constraints. In other words it chooses users who are both well-connected and topically influential. This is relevant to the study of diversity of opinions when you draw a parallel between topics and opinions as S2V DQN becomes a framework to infer both user's opinion and their influence spread.

2.4 . Social Interactions on Social Media

The combination of social mechanism and Recommender system over-tuned towards maximizing user retention creates a growingly hostile environment. Ironically, in the most connected era of our history we are the most divided. There are growing concerns about social fragmentation in social media. Even if not unanimously agreed upon [7, 8] it is considered that recommender systems play a major role in polarization [102, 103, 104, 105, 7].

Polarization in social network and opinion dynamics refers to the divergence of opinions towards increasingly extreme and opposite positions and become sharply divided, this translates to very dense and heavily segregated communities [8]. This in turns facilitates the propagation of fake news and rumors as no counterpoint can be seen, as well as exacerbating echo chamber mechanisms [106, 7]. Indeed, a regular recommender system will promote misinformation if it drives high user engagement and maintains user attention. This is particularly exacerbated by generative IA and how powerful and popularized it has become [107, 108]. To mitigate these issues, many lines of research have emerged. The most popular methods revolve around algorithmic approaches and structural interventions and aim at balancing out personalization with another socially responsible objective, generally a metric of fairness or diversity.

In the following, we will first present the dynamics models that are used to study how opinion evolve in a social network setting, then we will present algorithmic approaches based on fairness and diversity as well as structural intervention for polarization and disagreement minimization. We will also explain how polarization contributes to the propagation of misinformation and fakes news. Lastly, we will discuss the topic of serendipity with its many definitions, its issues and how it is linked to previously mentioned themes.

2.4.1 . Online communities

A foundational research area and one of the most studied is the task of community detection [109, 110] and remains a core aspect in most problems around social networks. A formal definition of a community is a densely connected set of nodes inside a network but it can also be defined as a subgroup of a network where information is trapped, some definitions consi-

der functional or feature similarities instead of structural similarity. Independently of which definition is used, an important feature of the resulting communities is how they interact with information propagation. Standard methods include the Girvan-Newman algorithm [111] which uses edge betweenness to determine communities, spectral clustering [112] which makes use of the spectrum (eigenvalues) of the Laplacian of the adjacency matrix to cluster it in a lower dimension yielding communities and modularity based approaches [113].

While communities play a role at a global level in term of information propagation, many social processes play a role at a local level. These include triadic closure [114], preferential attachment [115] and homophily [116]. Triadic closure is the social mechanism where there is a higher likelihood of forming a friendship between two people if they already share a common friend, preferential attachment is the increased likelihood of connecting individual that are mor popular or in more mathematical terms nodes with high degree. Lastly homophily is the tendency to connect to individuals that are similar to ourselves.

2.4.2 . Graph diameter reduction

An extensive body of research focuses on improving information propagation in graph networks through structural optimization called graph diameter reduction [117], also known as shortcut selection [118]. The foundation of these approaches is the small world property [119] of networks which refers to a characteristic of many real-world networks where most nodes can be reached from any other through a relatively small number of steps, despite the network being large and sparsely connected. Formally, such networks exhibit a high level of clustering and a short average path lengths between nodes. As such graph diameter reduction adds shortcuts, long-range edges connecting distant parts of the graph, which enhance the small-world property by significantly reducing the average path length while largely preserving local clustering. This makes information, influence, or signals propagate more efficiently across the network.

2.4.3 . Recommendation dynamics

Recommender systems provide recommendation to users based on predictive accuracy, in other words they suggest content that is the most likely to be consumed. Ultimately, it is up to the users to decide whether to actually engage with the given recommendation. There are various models of how to simulate the users' behavior which revolve around measuring satisfaction. These measures are necessary to curate insightful information for a reinforcement learning agent to learn properly and efficiently, as knowing about a user's satisfaction helps differentiate "good" from "bad" actions. Usually, en-

agement metrics are used to infer satisfaction values such as clicks or watch time [120]. But, these metrics are not available in the context of offline training, when there is a fixed dataset that is not update with new data in real time, which means that satisfaction values are inferred.

The models used to interpret these values are built on the foundation of the theory of reasoned action and fishbein models [121]. You can consider a binary threshold interpretation where all values under a threshold are negative and all values above are positive [122]. There is affective intensity where the positive value encodes how strong the reaction is and all negative reactions are grouped on the zero value [123]. Another semantic is qualitative categories using the likert scale (commonly used questionnaires) and associating numerical values to each category [124].

While recommendations try to add links (edges) to the network (graph), there are also some processes that leads to links being removed, in particular, the dynamic of unfollowing. Even if unfollowing and structural interventions that incorporate edge deletion are virtually the same as in they both end up with removal of an edge. The latter is frowned upon due to its similarity to censorship but the former is not as it captures a regular behavior in social network [125]. However, the study of unfollowing dynamics are important as they are linked to the formation of echo chambers [6, 114].

2.4.4 . Fairness and diversity

A fair RS does not systematically disadvantage or promote a specific group, user or item, an absolutely fair system is devoid of any biases. As such, the study of fairness, particularly in RS, revolves around minimizing biases and the bias amplification loop that may occur [126]. Although there are some approaches that aim to create models with a fair process [127, 128], most fairness-driven research focuses on the fairness of outcomes on the user-side, all users should have the same quality of recommendations, or item-side, all items should have equal chances of being recommended. There are many biases impacting each step of the recommendation pipeline as well as recommendation model training. We present here a concise list of the common biases that are studied; for a more in depth discussion on these biases, their effects and potential solutions we refer the readers to the following surveys [129, 130] :

selection bias also called “observation bias” refers to the bias of observed rating values. A real world example would be on Netflix, where ratings are only given for movies that users chose to watch, so the system never sees ratings for movies skipped, skewing the data that is eventually also used for the training of their model.

conformity bias happens as users tend to behave similarly to the others in a group. This impacts how users rate items either upwards if it already

received positive ratings or downwards if negative at the cost of the real opinion.

exposure bias refers to how users are only presented a part of specific items meaning that there are many unobserved interactions which are not equivalent to a negative interaction. A concrete illustration is the case of Spotify where songs that are not recommended to a user, will also not be played by that user but it does not inherently mean that this user also dislikes these songs.

position bias is the propensity of users interacting with items that are higher on a recommendation list regardless of relevance. A simple example happens when people search something on any internet browser and only consider the first few results on the first page and completely disregard all other results on pages two and onward even if the first result is an advertisement that does not answer the original search.

popularity bias, comparable to the Matthew effect [131], refers to popular items being recommended more often. As can be seen on platforms such as Youtube or Twitch where videos/streams with more views are promoted heavily.

inductive bias denotes all the assumptions that must be made for the recommendation model to better learn the objective function and to generalize beyond training data. For instance, collaborative filtering assumes that “users with similar past ratings will have similar future preferences”; this not an absolute truth and can fail in some instances.

Even though selection, conformity and exposure are user-level biases, they produce data imbalances and heavily impact the recommendation model via the training process, which also carries its own inductive biases. In turn these lead to item-level biases in the results, with position and popularity biases. This exemplifies the bias amplification loop that is prone to happen in most recommendation systems.

Although it is easy to visualize, diversity is actually quite challenging to conceptualize [132, 127]. This has led to many different definitions with different measures and objectives. There is individual diversity that caters to specific users and their personal recommendation list and aggregate diversity that aims to capture diversity across all recommendations in the system. The latter is relevant to the overall fairness of the system while the former is central in the experience of users. Individual diversity can be differentiated into two sub-groups : distance-based or category-based. The most widely used distance based definition is that of intra-list diversity which measures how dissimilar items in a single recommendation list are. This includes many variations depending on item embeddings or the distance function used. Category-based diversity considers coverage, redundancy and size-awareness of item

categories/genres [128]. While aggregate diversity considers global diversity and looks at how many different and unique items are being recommended, as such it is closely related to exposure fairness. The most intuitive approach is to simply count the number of completely different and diverse items being recommended. For a more expressive formulation that takes item distribution into account, it can be formulated using Shannon's entropy [133] or the Gini index.

Fairness and diversity methods can be injected into any step of the recommendation pipeline. First, there are many pre-processing techniques that mitigate biases through data refinement so that the models train on fair and unbiased data. To achieve this, many techniques can be implemented, such as feature modification [134] which identifies sensitive features and either removes them or decorrelates other features, distribution adjustments [135] which rebalance the data training set for underrepresented groups to appear more frequently, data reweighting [136] which assigns different weights to interactions depending on group membership.

Many methods have been proposed that incorporate fairness and diversity in their models. These in-process approaches can be divided into four frameworks. The first framework is constraint based optimization [128] where the model's scoring function is subject to constraints that must be respected. Secondly, regularization based approaches [137] integrate fairness constraints in the form of regularization terms in the loss function during the model's learning. This also includes learning to rank algorithms [138] where diversity regularization terms are used during the user/item representation learning step. Thirdly, multi-objective models [139] consider fairness and accuracy as separate objectives for which the models have to learn the trade-off between personalization quality and fairness. This also includes fusion based approaches [140] which combine multiple recommender systems to generate aggregated scores. The last framework is adversarial learning [141], consisting of three different models : the encoder, the predictor and the adversary. The encoder learns how to produce representation from the input and the predictor learns how to recommend. At the same time, both learning processes also try to prevent the third agent, the adversary, of predicting the sensitive features in the data.

Post-processing methods are popular as they defer the task of generating accurate and personalized recommendation to the recommender systems and they focus only on fairness/diversity. This also comes with the benefit of using a resulting recommendation list which is very small in comparison to the original data the RS has to parse in order to do its prediction. This means that methods with higher computational costs can be used enabling more expressive and complex algorithms. The most widely used approach is re-ranking which given a recommendation list re-orders the item according

[142]. Concerning fairness, there are three types of re-ranking algorithms, (i) slot-wise re-ranking which constructs the list sequentially for a specific user [143]. Then there is (ii) user-wise re-ranking [144] and (iii) group-wise re-ranking [145, 146], the former generates a single user's list at once while the latter generates the list for a group of users. A widely used approach for both is integer programming with fairness constraints [147, 144]. Concerning diversity, re-ranking methods usually focus on a greedy approach using Maximum Marginal Relevance (MMR) which considers the relevance of each item only in terms of how much new information it brings given the previous results [148]. Similarly, this type of content moderation has also been used to minimize polarization [149, 150, 151, 152].

2.4.5 . Opinion dynamics

The study of opinion dynamics within social settings is a well-established field, with foundational models such as the DeGroot model [153] dating back to 1974. This line of research is critical to social network science, providing a framework for understanding how individual opinions evolve, the underlying mechanisms and patterns that lead to polarization and the interplay between these dynamics and the hyperparameters of recommender systems [154].

The most popular model in the academic literature is the Friedkin-Johnsen opinion dynamic model [155] where the opinion of a node is a function of its neighbors' opinions, it also differentiates between personal innate opinion and expressed opinion of a given node. The innate opinion provides a certain bias, or stubbornness, that may hinder consensus. FJ is an extension of the DeGroot model [153] which only aggregate neighbors opinions without consideration for any innate personal opinion.

Other methods include Bounded Confidence Models [156] such as Hegselmann-Krause [157] and Deffuant-Weisbuch [158] where users only interact with others whose opinions are within a certain confidence threshold of their own and voter models [159] where users randomly select a neighbor to copy their opinion.

2.4.6 . Polarization

Computational models of opinion dynamics [155, 153, 157, 158], as well as empirical studies of social media platforms [160, 154, 5, 4, 6], suggest that mechanisms such as selective exposure and algorithmic personalization can amplify global polarization. As a result, polarization has become a central concern which required specific algorithms to reduce social fragmentation. Using an opinion dynamics model (usually FJ) opinions are represented by a value in the range of 0 to 1 (equivalent -1 to 1), polarization is conceptualized as the deviation of opinions from the average similarly to the mean squared error loss in machine learning. It has been studied that exposing users to opinion

that are too different from their own may actually have negative effects with increased rejection leading to more polarization [161]. As such, polarization is usually studied in combination with disagreement which is conceptualized as the average distance in neighboring opinions. Polarization and disagreement minimization is an optimization problem traditionally solved through structural interventions which include edge addition [10, 162, 163, 164], edge re-weighting [165], node re-weighting [165] and rewiring [166, 165]. Edge deletion is rarely used due to its controversial nature as it strongly resembles censorship and at the same time you cannot force a behavior on someone. Similarly, node re-weighting is also a debatable method as it assumes the system can change the innate opinion of users freely. Modern approaches are looking towards network-aware models that can balance maximizing engagement and minimizing polarization [167].

2.4.7 . Misinformation Propagation

The propagation of misinformation and disinformation in online and offline platforms has become a central concern in both social science and computational research [168]. Misinformation refers to false or misleading information shared without intent to deceive while disinformation is deliberate dissemination of crafted information to manipulate beliefs or behaviors [168, 169]. A prominent manifestation of these phenomena in online environments are fake news, broadly defined as false or misleading information designed to appear credible and often amplified through social media platforms. Understanding how fake news spread is critical, as it can shape public opinion, influence elections and exacerbate social fragmentation.

Classical modeling approaches often draw inspiration from epidemiology, using frameworks such as the Susceptible–Infected–Recovered (SIR) or Susceptible–Exposed–Infected–Recovered (SEIR) models [170], which conceptualize information as “contagious” and allow researchers to analyze the dynamics of adoption, exposure and decay over time. Network-based approaches, including threshold models and complex contagion frameworks [171, 172], further account for social reinforcement effects, demonstrating that repeated exposure from multiple neighbors can accelerate the spread of false information. More recent research employs probabilistic graphical models and machine learning techniques to identify influential nodes, predict viral content and simulate the impact of structural network changes [173, 174].

Despite these advances, significant challenges remain : accurately capturing the temporal evolution of misinformation, modeling dynamic network structures, accounting for user heterogeneity and quantifying the interaction between competing narratives are all areas that continue to complicate predictive modeling [175].

Addressing these challenges is critical for developing robust strategies to

mitigate the societal impact of both unintentional misinformation and deliberate disinformation campaigns. At the same time, the emergence of generative AI has further amplified the challenges associated with misinformation [176, ?] by enabling the rapid and scalable production of highly realistic texts, images and videos. This increased generative capacity complicates existing propagation models, as it blurs traditional credibility cues and accelerates diffusion dynamics, thereby requiring new modeling approaches that jointly account for content generation, network dissemination and human perception [177].

2.4.8 . Serendipity

The concept of serendipity fuses personalized recommendation and diversity as well as fairness. The Oxford dictionary defines serendipity as “the occurrence and development of events by chance in a happy or beneficial way” which, in the context of content recommendation, translates to recommendations that correspond to a user’s interests disregarding their knowledge of that content [178]. However this is particularly challenging to formalize mathematically as there are many different definitions. Most include the ideas of unexpectedness and usefulness but it can also include ideas of novelty, diversity or relevance [178]. Indeed, serendipity can sometime refer to an unknown item that is useful but it can also refer to a known item that was actually useful unbeknownst to the user [179]. This exemplifies the core challenge of qualifying, defining and measuring serendipity [180]. Furthermore, there is a lack of public datasets for serendipity as they are time-consuming and laborious to make because of their reliance on users volunteering to provide their input on serendipity which comes with its own issues. In spite of these challenges, interests in serendipitous recommendations are growing.

2.5 . Entropy in Graphs

Entropy is a fundamental concept that quantifies the amount of uncertainty, disorder, or information contained within a system. Originally used in the context of information theory, it serves as a measure of unpredictability: a high-entropy system is one where outcomes are highly uncertain, while a low-entropy system is more predictable. Shannon’s entropy [133], provides the mathematical foundation for this idea. It calculates the average information content in the possible outcomes of a random variable. Essentially, it measures the minimum number of bits required to encode a message, where a more probable outcome conveys less information and contributes less to the overall entropy.

The task of determining the entropy of relational structures, such as complex networks or graphs, is central to a wide array of fields, from social net-

work analysis and communications routing to the study of protein interactions, financial key players and critical infrastructure vulnerability [181]. Given the profound differences in the routes and flow types inherent to these diverse systems, a substantial body of research has produced a variety of specialized information entropy metrics. Each metric is designed to capture distinct characteristics of a network or its components and no single metric is universally applicable across all network types. Unfortunately, this expanding body of literature remains scattered across numerous disciplines, making it challenging to identify the available metrics and, crucially, to understand the specific contexts for which they are suited [181].

In the context of social network analysis, entropy provides a powerful lens to analyze a network's complexity. For instance, it can measure the diversity of a node's connections : a user with links to many distinct, non-overlapping social circles would have a high local entropy, indicating a rich and varied social environment. Usually based on node degrees [182, 183], it is used as a ranking criteria to identify influential nodes but it can also be based on neighbor degrees [184], betweenness [185] or closeness centrality [183]. Conversely, it can also gauge the centralization of influence across the entire network; used as a graph metric it can be useful for identifying key nodes as a low-entropy network might be dominated by a few highly connected influencers. While a high-entropy network would have influence more evenly distributed among many participants [186]. In short, entropy helps identify key nodes that bridge diverse groups and how information might spread.

2.6 . The Industry's perspective

Many leaders in the tech industry such as Google have recognized that, while accuracy is the standard for evaluating recommender systems, diversity and novelty are critical to users' perceived utility. As such, modern evaluation metrics need to make a trade-off between accuracy and diversity as recommender systems should account for both [187, 188].

These developments indicate a shift in the industry from seeing diversity as a benefit towards considering it a core component of recommender system design. Amazon recognizes that diversity is a crucial aspect of effective recommender systems, ensuring users are exposed not only to familiar or obvious choices but also to novel and unexpected items [189]. This balance between relevance and discovery is especially important given the company's vast catalog spanning countless product categories. However, systems need to adapt to users' intents by narrowing recommendations when they have a specific wish or by broadening suggestions when the idea is more unclear [189]. Such an approach reflects an understanding that long-term customer satisfaction depends not just on accuracy, but on helping users discover new

interests and product areas they might never have searched for on their own [190].

Netflix is also looking at this problem [191, 192] to improve user engagement in the long term [193, 194]. In particular, Netflix is actively exploring algorithmic solutions that predict user intentions to either provide accuracy driven recommendations or diversity and novelty driven suggestions to bypass the need for a trade-off [191]. Spotify explores new semantics for recommender system focused on social motivations for music listening deviating from traditional predictive accuracy [195]. At the same time, they also recognize the challenges that two-sided marketplaces such as their platform present in terms of fairness and actively explore how to balance out fairness, relevance and satisfaction [196, 197].

3 - Formal Setting

In this section we will present the foundation of network analysis as well as the necessary notions and tools to understand the basis on which the work of this thesis is built. The formal representation of social networks are graphs, where nodes correspond to individuals or entities, and edges represent relationships or interactions between them. This allows the application of graph theory principles to study structural properties such as connectivity, clustering and community structures. By abstracting complex social interactions into a mathematical object, graphs enable both qualitative and quantitative analysis of phenomena like influence propagation, information diffusion and group dynamics which, in turn, lets us study polarization, echo chamber formation and social fragmentation.

3.1 . Graph Framework for Social Networks

We represent the social network as a graph $G = (V, E)$, where V is the set of nodes and $E \subseteq V \times V$ the set of edges, $n = |V|$, $m = |E|$. An edge is a link between two users, it can be **undirected** or **directed** depending on whether the relation is symmetric. When two users are connected their are called **neighbors**. For each user, the number of links they have is called the **degree** $deg(u)$ and a sequence of consecutive links from user to user is called a **path**. In the case of a directed graph the **degree** depends on which direction of edges you consider for given user their *in-degree* $deg_{in}(u)$ is the number of edges that point towards them and their *out-degree* is the number of edges that start from them. An important feature of a path is its length, measured by the number of edges in its sequence, because it can quantify the **distance** $d(u, v)$ between two users which is given by the length of the shortest path between them. Subsequently, a graph is considered **connected** if for every pair of nodes there exists a path connecting them. If you have additional information on the edges, you can include a weight Matrix to the graph which becomes a **weighted** graph. These weights quantify the strength of a relationship between two users. This can be interpreted as the strength of their influence or the probability of information being shared. In this case, it can also be used to measure the distance between users which then becomes the sum of the weights along the edges on the shortest path between them.

For instance, Facebook can be represented by an undirected graph as friendship links are symmetric. This is opposed to X (formerly Twitter) which can be represented as a directed graph as followership is a directed relationship between users. Both social networks can be considered to be connected

if one disregards test accounts or users that do not follow anyone. By default, none of the two graphs are weighted but weights can be added through various methods. For instance, for these networks specifically, the number and frequency of interactions can provide a good basis for weights. A generic approach is to use the degree and use it to give an equal weight among all neighbors.

Similarly to the offline case, users of online social networks connect and interact with friends, colleagues; generally, persons that are similar to them. This leads to friend groups being formed and thus to the formation of **communities**. In a topological sense, in the context of social networks, a community is a group of nodes that are strongly connected between them and sparsely connected to others. Traditionally, communities are considered to form a disjoint partition of the graph as it simplifies models. However overlapping communities would arguably be a more realistic setting. The **clustering degree** of the graph which measures how densely interconnected its immediate neighbors are, indicating the presence of tight-knit local groups.

It is not uncommon to consider that people that are part of the same community to have similar opinions. Hence, in this thesis, we consider that opinions are clustered into communities, and that each community in the graph represents a specific opinion.

Graph properties Social networks exhibit several structural properties that are fundamental to understanding their dynamics. One key feature is the **small-world property**, characterized by high clustering coefficients combined with short average path lengths, which allows individuals to be connected through only a limited number of intermediaries. This is commonly illustrated by the fact that it takes less than five links to connect any two people on earth [198, 119]. In addition, many social networks conform to **scale-free models**, [115] wherein degree distributions follow a power-law and a small subset of nodes function as highly connected **hubs** that significantly influence network connectivity and information flow. At the same time, the vast majority of users have very few connections. Within this topology, relational strength is also critical : **strong ties** typically denote frequent, trust-based interactions within close social circles, while **weak ties** serve as **bridges** across communities, enabling the diffusion of novel information and fostering structural cohesion.

Node properties Beyond global properties, the structural role of individual actors within social networks can be assessed using centrality and clustering measures. **Betweenness centrality** captures the extent to which a node lies on the shortest paths between others, highlighting actors who serve as brokers or bridges across communities. **Closeness centrality** reflects how

close a node is to all others in terms of graph distance, emphasizing individuals with efficient access to information across the network. Finally, the **clustering coefficient** quantifies the tendency of an actor's neighbors to also be connected with one another, thereby measuring the degree of local cohesion within the network.

Graphs as matrices Graphs can also be represented as a matrix $A_G \in \mathcal{M}_{n,n}$ called adjacency matrix. The i -th line (or column) characterizes the node i and the value at the i -th line and j -th column represent the edge between nodes i and j . A strictly positive value indicates the presence of a link while a zero indicates its absence. If all values in A are either 1 or 0 then the graph is unweighted, otherwise positive values a_{ij} in the matrix represented edges that exist and have weight a_{ij} . In other words the adjacency matrix is a record of all neighbors (nodes of distance 1). However, if raised to the power A^n it also records all pairs of nodes that can be reached within distance n along with the number of paths that can reach them. We also have the degree matrix D_G (with $D_{ii} = \sum_j A_{ij}$) and we denote by \mathcal{N}_u the set of neighbor indices of u in G . When unambiguous the subscript G is omitted.

3.2 . Diverse Exposure Framework

Graph Partition To represent the fact that the social network is organized in communities (possibly having different semantics), we define the notion of a *graph partition* \mathcal{P} :

Definition 1 (Graph Partition). *A graph partition of a graph G , \mathcal{P}_G , is a set of subsets of nodes $\mathcal{X}_i \subset V$:*

$$\mathcal{P}_G = \left\{ \mathcal{X}_i \mid \mathcal{X}_i \subset V, \mathcal{X}_i \neq \emptyset, \bigcup^i \mathcal{X}_i = V \right\}. \quad (3.1)$$

The set of partitions to which a node u belongs is denoted as :

$$\mathcal{P}_G(u) = \{ \mathcal{X}_i \mid u \in \mathcal{X}_i \}. \quad (3.2)$$

Note that this definition allows for overlapping partitions. So in this sense, the use of the term “partitions” is incorrect; however, we keep it as such as most communities in graphs, either real or detected using algorithms, are disjoint.

In matrix form, we represent the partitions as a column-stochastic matrix P of dimension $p \times n$, with the property that $\sum_i P_{ij} = 1, \forall j$. The column encode the distribution of communities of opinions an individual node has.

Node exposure We define next the concept of *exposure to information* and give some examples of how it can be measured in a graph.

Definition 2 (Node Exposure). *The node exposure function is a function $\mathcal{E} : V \rightarrow \mathbb{R}^n$, representing the distribution of exposure of information from all nodes $v \in V$; moreover, we require that this is a distribution, i.e., $\mathbf{1}^\top \cdot \mathcal{E}(u) = 1, \forall u \in V$.*

From node exposure, we can naturally extend it to a *partition exposure* function, $\mathcal{E}_P : V \rightarrow \mathbb{R}^p$, such that

$$\mathcal{E}(u | \mathcal{P}_G) = \mathcal{E}(u) \cdot P.$$

It is easy to check by the properties of \mathcal{E} and \mathcal{P}_G that $\mathcal{E}(u | \mathcal{P}_G)$ is also a distribution.

Exposure Functions There exist various ways of defining the exposure function \mathcal{E} , and we present here some options.

Personalized Random Walk (PPR) This is the classic PageRank random walk process [96], rooted at a node u using transition weight matrix $W_u = \alpha D^{-1}A + (1 - \alpha)(\mathbf{1}_n \otimes \mathbf{u})$, where \mathbf{u} is the one-hot vector of u , and \otimes is the Kronecker product. The exposure distribution is then the stationary distribution of this process :

$$\mathcal{E}(u) = W_u^\infty = (1 - \alpha) \left(I - \alpha D^{-1}A \right)^{-1} \mathbf{u},$$

where \mathbf{u} is the one-hot encoding of u .

Friedkin-Johnsen Model (F-J) This is a model of opinion dynamics in social networks [155], where the opinion of a node u is an aggregation of opinions of neighbors. It is equivalent to a PPR on the transpose graph, if we consider that the initial vector of opinions is \mathbf{u} :

$$\mathcal{E}(u) = W_u^\infty = (1 - \alpha) \left(I - \alpha D^{\top-1} A^\top \right)^{-1} \mathbf{u}.$$

Note that when $\alpha = 0.5$ and $D = I$, this is equivalent to the stationary distribution found in [165, 199].

Breadth-First (BFS) This is a simple exposure model, where the exposure of a node u is inversely proportional to the depth at which each node v is found in the process of a BFS search.

$$\mathcal{E}(u) = \sum_i A^i$$

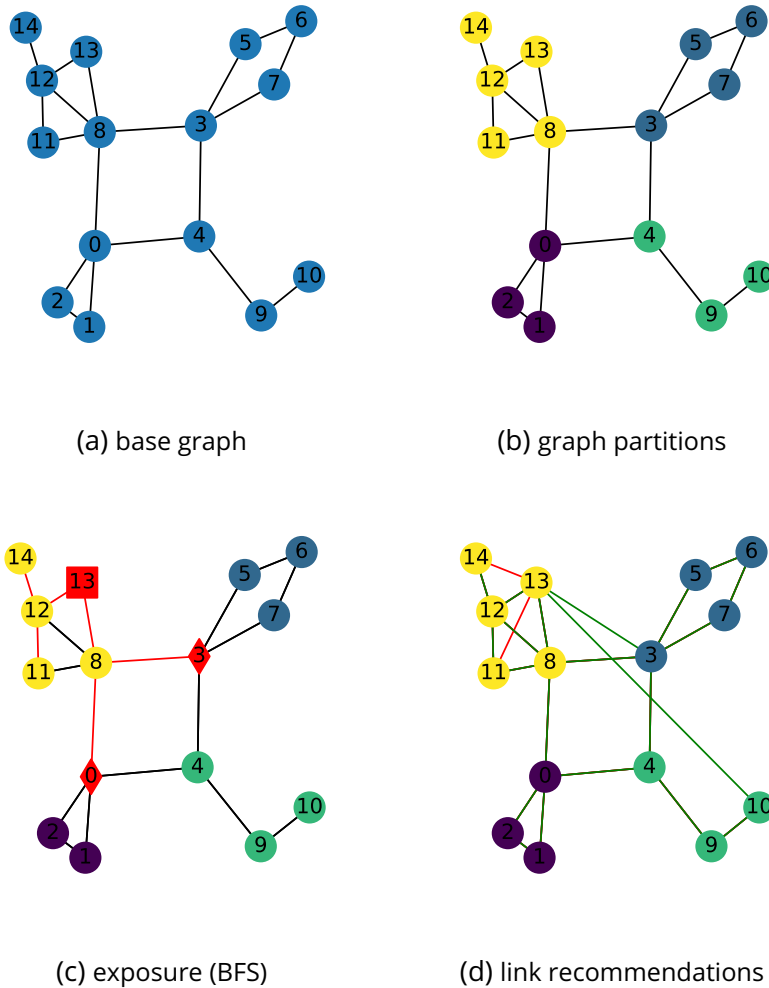


Figure 3.1 – Example social network : partitions, vision, and recommendations.

Example 1. Fig. 3.1a exemplifies a small social network of 15 nodes. In this example we consider the point of view of node $u = 13$. The network is divided into 4 partitions as shown in Fig. 3.1b – in this case, generated using the Louvain modularity algorithm. We exemplify node 13 and the BFS exposure (cut-off at depth 2 for legibility reasons), shown in Fig. 3.1c. The resulting distribution over the 4 partitions is : $\mathcal{E}(13 | \mathcal{P}_G) = \{0.125, 0.125, 0, 0.750\}$. This is because in the search, the blue partition is reached once (node 3 at depth 2), the purple partition once (node 0 at depth 2), the green partition never, and the “home” yellow partition 4 times (nodes 8, and 12 at depth 1, and 12 and 14 at depth 2).

Diversity of Exposure Given the probability distribution of the exposure function \mathcal{E} we can now define a diversity measure function $\sigma(\mathcal{E}(u | \mathcal{P}_G)) \in \mathbb{R}$

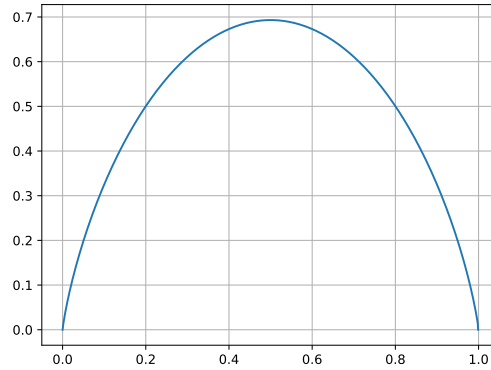


Figure 3.2 – Non-monotonicity of binary entropy.

which quantifies — given a node u in a graph G partitioned according to \mathcal{P}_G — how diverse u 's exposure $\mathcal{E}(u | \mathcal{P}_G)$ is.

Intuitively, if we view the partitions in \mathcal{P}_G as representing communities of opinion, the aim of diversity of exposure measure σ is to be as close as possible to a distribution in which all opinions / partitions are equally represented.

Entropy A reasonable candidate measure is then **Shannon's entropy** [133]— which is maximized when the distribution is equi probable :

$$\sigma(\mathcal{E}(u | \mathcal{P}_G)) := \mathcal{H}(\mathcal{E}(u | \mathcal{P}_G)) = - \sum_{\mathcal{X}_i \in \mathcal{P}_G} p(\mathcal{X}_i) \log(p(\mathcal{X}_i)). \quad (3.3)$$

Entropy makes intuitive sense because of its aim to measure the amount of information in a message. Entropy-based measures have been used before to measure diversity [200] and fairness [130, 142]. Moreover, [6] showed experimentally that the entropy can also quantify polarization in a graph.

Fig. 3.2 illustrates Shannon's entropy for a Bernoulli probability distribution meaning that there exists a probability of one outcome (success) p_1 and where the probability of the other outcome (failure) $p_2 = 1 - p_1$. On the x axis we have the value of p_1 and on the y axis the corresponding value of Shannon's entropy which in this binary setting is $\mathcal{H}(p_1, p_2) = -(p_1 \log(p_1) + p_2 \log(p_2))$. The entropy is equal to 0 in the context of guaranteed events, i.e. either p_1 or p_2 is 1 or 0. Also, the entropy is maximized for an equi-probable distribution, in other words when all outcomes are equally possible. This figure also illustrates that the entropy function is not monotonic : as the probability of one outcome increases, the entropy first increases until it reaches its maximum and then decreases.

Cross-entropy Another way to look at the problem is that we are aiming to minimize the distribution distance to a reference desired distribution Q . This can occur when we would like to minimize the exposure to e.g., toxic partitions while keeping the other partitions diverse. This can be measured using the **KL divergence** between \mathcal{E} and Q :

$$\begin{aligned} \sigma_Q(\mathcal{E}(u | \mathcal{P}_G)) &:= \mathcal{D}_{\text{KL}}(Q || \mathcal{E}(u | \mathcal{P}_G)) = & (3.4) \\ &= - \sum_{\mathcal{X}_i \in \mathcal{P}_G} q(\mathcal{X}_i) \log \left(\frac{q(\mathcal{X}_i)}{p(\mathcal{X}_i)} \right). \end{aligned}$$

When Q is the equi-probable distribution, Eq. (3.4) is equivalent to Eq. (3.3).

Example 2. We return to the example in Fig. 3.1. The BFS vision distribution is associated to an entropy value of 0.736. When one recommends using classic, triadic closure based, recommendation algorithms, node 13's community becomes denser (edges 13 – 14 and 13 – 11, in red in Fig. 3.1d), resulting in a more "biased" exposure : $\mathcal{E}(13 | \mathcal{P}_G) = \{0.1, 0.1, 0, 0.8\}$ and a smaller entropy value of 0.639. On the other hand, when recommending by optimizing entropy-based measures (edges 13 – 10 and 13 – 3, in green in Fig. 3.1d) we get a more balanced vision : $\mathcal{E}(13 | \mathcal{P}_G) = \{0.145, 0.285, 0.285, 0.285\}$ and a higher entropy of 1.351. Moreover, we can see that this is achieved by linking 13's original community, yellow, with the green and blue communities.

3.3 . Notations

Table 3.1 summarizes the main notations used in this thesis.

Table 3.1 – Main notations.

| Symbol | Definition |
|--------------------|--|
| G | A graph |
| V | The set of nodes |
| E | The set of edges |
| u | A node |
| (u, v) | The edge between nodes u and v |
| A_G | The adjacency matrix of G |
| $\mathcal{N}_G(u)$ | The set of neighbors of u |
| C | The adjacency matrix of the item graph |
| T | The transition matrix |
| Q | A distribution |
| \mathcal{P}_G | A graph Partition of G |
| \mathcal{E} | Exposure function |
| σ | Diversity measure |
| W | Exposure weight matrix |
| PPR | Pagerank |
| FJ | Friedkin-Johnsen |
| BFS | Breadth-First |
| \otimes | Kronecker product |
| \mathcal{S} | The set of states |
| s_t | State at timestamp t |
| \mathcal{A} | The set of actions |
| a_t | The action at timestamp t |
| \mathcal{R} | The reward function |
| S_G | The satisfaction function |

4 - Optimizing Diverse Information Exposure in Social Networks

In this chapter we introduce the problem of information exposure maximization and discuss various approaches to solve it. The following work was published at the 2024 IEEE International Conference on Big Data (IEEE Big-Data 2024) and in the 2025 Transactions on Large-Scale Data- and Knowledge-Centered Systems LIX Special Issue on Data Management - Principles, Technologies, and Applications (TLDKS) journal.

4.1 . Introduction

Ongoing research seeks to assess the full extent of the potential harms associated with the growing polarization and reliance on social media for news consumption, as traditional outlets continue to decline in influence. From an algorithmic perspective, various strategies have been proposed to optimize diversity and mitigate polarization in social graphs [10, 162, 165]. These approaches often focus on global interventions, either by minimizing a measure of overall polarization [199], or reducing average path lengths under the assumption that shorter distances enhance diversity [117].

However, most existing methods operate at the *global* level, implicitly assuming that modifications can be applied uniformly across the entire network. We argue that a focus on *local* optimization, i.e. adapting recommendations from the perspective of individual users, may prove more effective. Such an approach could yield more personalized content, reduce unintended global perturbations (given that not all users value diversity equally), and thereby increase both effectiveness and adoption. This vision requires developing complementary semantics for content discovery in social networks. One promising avenue is to maximize a user-specific measure of diversity, defined in terms of exposure to communities beyond one's own, thus fostering encounters with novel perspectives without imposing system-wide interventions. Intuitively, it also is a way to increase the *entropy of the information exposure* from the node's point of view.

In the following, we present an approach to optimizing the *diversity of information exposure* in the context of social networks based on a general formulation of exposure (presented in Section 3.2 as the distribution of opinions (represented as non necessarily distinct communities) a given node u is exposed to. This allows us to : (i) be flexible in the definition of the exposure function, and we take inspiration from well-studied approaches such as the *polarization* of opinions in the graph [165], but also from classic random-walk

and distance-based metrics; (ii) going beyond polarization, to model diversity of exposure over more than 2 polarized opinions; and (iii) define diversity not only as targeting uniform distributions of opinions but *any* distribution of opinions – crucially allowing us to incorporate user interests in the optimization. This framework aligns with theories of information diffusion, which emphasize the role of inter-community ties in fostering innovation and reducing echo chambers. Moreover, it provides a pragmatic balance between individual user agency in having more diverse recommendation and systemic goals in maximizing user engagement, acknowledging that diversity in information exposure needs not be universally imposed to yield meaningful collective benefits.

4.2 . Problem definition and properties

The studied problem can then be stated as follows :

Problem 1 (Single-Node Exposure Diversity Maximization). *Given a graph $G = (V, E)$, a node $u \in V$, a partition \mathcal{P}_G , an exposure function $\mathcal{E}(u)$, maximize the diversity measure $\sigma(\mathcal{E}(u | \mathcal{P}_G))$ for a fixed number k of edge additions to the graph starting at u .*

Properties The objective is *non-monotone* and *non-submodular*. This means that the greedy algorithm is not guaranteed to find the optimal solution; indeed, as we will show in the experimental results, there are diminishing returns when adding edges to the graph up until a point where it actually *decreases* the diversity of exposure. This can intuitively be visualized with the following example. Indeed, if we keep adding edges with the same community then this community's influence is expected to surpass and overshadow that of the others.

Moreover, the objective is not always convex. On the one hand, for the PPR and F-J functions, the objective is *convex*, as it has the same form as the polarization minimization in [165]. On the other hand, for the BFS function – and when expressing the exposure function as a sum of powers of the adjacency matrix – one can see that, from depth $d = 3$ on, the objective is not convex due to at least the term A^3 .

Definition 3 (Convex function). *Let $f : C \rightarrow \mathbb{R}$ be a function defined on a convex set $C \subseteq \mathbb{R}^n$. The function f is said to be **convex** if for any two points $x, y \in C$ and any $\lambda \in [0, 1]$, the following inequality holds :*

$$f(\lambda x + (1 - \lambda)y) \leq \lambda f(x) + (1 - \lambda)f(y)$$

The function $f(A) = (I - A)^{-1}$ is matrix convex when A is in a convex set and hence positive semi definite.

Definition 4 (Convex Set). A set $C \subseteq \mathbb{R}^n$ is called a **convex set** if for any two points $x, y \in C$ and any $\theta \in [0, 1]$, the point $\theta x + (1 - \theta)y$ also belongs to C . That is,

$$\forall x, y \in C, \forall \theta \in [0, 1] : \theta x + (1 - \theta)y \in C.$$

Consider the following set of graphs :

$$\begin{aligned} \mathcal{G}(A) = \{G' \mid & A'_{ij} > 0, \forall (i, j) \in E_G, \\ & A'_{ij} = 0, \forall (i, j) \notin E_G, i \neq u, \\ & \sum_j A'_{ij} = 1, \forall i \in V_G\}, \end{aligned} \quad (4.1)$$

i.e., all graphs that have their edge weights normalized, and for which $D = I$. Moreover, we allow only graphs that change only in row u (adding new links from u to other nodes). It is straightforward to show that this set is convex : a linear combination of row-stochastic matrices is also row-stochastic; it is also the set defined in [199, 165], restricted to u .

The aforementioned set 4.1 is also the feasible set of our problem. In mathematical optimization and computer science a feasible set is the set of all possible points, in our case of possible graphs, of an optimization problem that satisfy the problem's constraints. Having a convex feasible set with a convex objective function ensures that any local minimum is also a global minimum, which eliminates the risk of getting trapped in suboptimal solutions. This structure also enables efficient and reliable optimization algorithms with strong convergence guarantees such as Gradient-based methods.

Proposition 1. *The set of feasible graphs $\mathcal{G}_u(A)$ is convex.*

4.3 . Convex optimization

We know that both entropy and KL divergence are maximized at the equiprobable or the Q distributions, respectively. Assuming that we have a set of *feasible graphs* \mathcal{G} (defined later), the optimization problem can be stated as the minimization of the distance between the exposure distribution and these distributions :

$$\arg \min_{G \in \mathcal{G}} \|\mathbf{d}\|^2 = \|\mathcal{E}_G(u \mid \mathcal{P}) - \mathbf{q}\|^2 = \|\mathcal{E}_G(u) \cdot P \cdot \mathbf{u} - \mathbf{q}\|^2, \quad (4.2)$$

where \mathcal{E}_G denotes the exposure function computed on the graph G , \mathbf{u} the one-hot encoding of u , and \mathbf{q} is the desired target distribution.

Note that the formulation in Eq. 4.2 is similar to the formulation for minimizing polarization in a Friedkin-Johnsen model as the distance between the vector of opinions and the average opinion in the graph, as in [201, 199, 163]. We claim that our formulation is more general, as minimizing the distance to

the average opinion risks decreasing polarization but increasing extreme opinions, e.g., when the average opinion is extreme (note that this can also occur when the average opinion is 0-centered *a posteriori*, as claimed in [163]).

By Eq. 4.2 and 4.1 we can adapt gradient descent algorithms for Problem 1. For this, we have to define the gradient of the objective function, and the projection step.

Gradient Step The derivative of the objective depends on $\frac{\partial \mathcal{E}}{\partial A}$ in the following way :

Proposition 2. *Assuming that \mathcal{E} is a derivable function of the adjacency matrix A , the gradient of the objective function is given by :*

$$\frac{\partial \mathcal{O}}{\partial A} = 2\mathbf{d}(\mathbf{u}^\top \otimes P) \frac{\partial \mathcal{E}}{\partial A}. \quad (4.3)$$

Proof Sketch. Follows from writing \mathcal{O} in algebraic form, $\mathcal{O} = \mathbf{d}^\top \mathbf{d}$ and then using the product rule of matrix calculus. \square

This general form allows us to “plug-in” any exposure function and its derivative (or simply provide a numerical approximation thereof). For PPR and F-J exposure :

Proposition 3. *The gradient of the objective function for the **PPR** and **F-J** exposure functions is given by :*

$$\frac{\partial \mathcal{E}}{\partial A} = -\alpha(1 - \alpha)(W_\infty^{-\top} \otimes W_\infty^{-1}),$$

where $W_\infty = I - \alpha A$ for PPR, and $W_\infty = I - \alpha A^\top$ for F-J.

Proof Sketch. We use the facts that :

$$\begin{aligned} dW_\infty &= -\alpha dA, \\ d\mathcal{E} &= -(1 - \alpha)(W_\infty^{-1} dW_\infty W_\infty^{-1}) \\ &= (1 - \alpha)(W_\infty^{-\top} \otimes W_\infty^{-1}) dW_\infty. \end{aligned}$$

The final formula follows. \square

We can express the BFS exposure function as a sum of powers of A , limited to a depth of 3 for practical reasons :

$$\mathcal{E}^{(\text{BFS})} = \sum_{i=1}^3 A^i = A(I - A^3)(I - A)^{-1}.$$

Proposition 4. *The gradient of the objective function for the **BFS** exposure function at depth 3 is given by :*

$$\begin{aligned} \frac{\partial \mathcal{E}}{\partial A} = & I + (I \otimes A + A^T \otimes I) + \\ & + (A^{T^2} \otimes I + A^T \otimes A + I \otimes A^2). \end{aligned}$$

Proof Sketch. This follows from developing the $\mathcal{E} = \sum_{i=1}^3 A^i$ as $d\mathcal{E} = dA + dA^2 + dA^3$ and applying the rules of derivatives of powers of matrices. \square

Note that we could straightforwardly expand this to higher powers, but it would require both more involved formulas and would cost more computation time.

Once we have the matrix-function gradients, we can directly apply projected gradient descent algorithms, alternating gradient descent and projection in the feasible set (i.e., a modified graph). Algorithm 1 details the steps.

Algorithm 1: DescentDiverse

Data: graph G , target node u , exposure function \mathcal{E} , graph partition \mathcal{P}_G , target distribution \mathbf{q} , number of edges k , rate η

```

1  $G^{(1)} \leftarrow G, A^{(1)} \leftarrow A;$ 
2 for  $i \in \{1, \dots, k\}$  do
3   compute  $\mathcal{O}_i = \|\mathcal{E}_{G^{(i)}}(u | \mathcal{P}) - \mathbf{q}\|^2;$ 
4   gradient :  $\Delta A^{(i)} = \frac{\partial \mathcal{O}_i}{\partial A}$  according to Eq. 4.3;
5   descent :  $A' \leftarrow A^{(i)} - \eta \Delta A^{(i)};$ 
6   projection :  $\{v_i, A'_u\} \leftarrow \text{Project}(u, A'_u);$ 
7   add to graph :  $G^{(i+1)} \leftarrow G^{(i)} + (u, v_i), A^{(i+1)} = A, A_u^{(i+1)} = A'_u;$ 
8 end
9 return  $\{v_1, \dots, v_k\}$ 

```

Algorithm 2: Project

Data: target node u , set of neighbor indices \mathcal{N}_u , u 's row in the modified adjacency matrix A'_u

```

1 get the candidate set of nodes which have positive values :
    $\mathcal{C}(u) = \{v \mid v \notin \mathcal{N}_u, A'_{uv} > 0\};$ 
2 get  $v = \arg \max_{v \in \mathcal{C}(u)} A'_{uv};$ 
3 set all other edges to 0 :  $A'_{uj} = \mathbf{0}, \forall j \notin \mathcal{N}_u \cup \{v\};$ 
4 re-normalize the edge weights :  $A'_{ui} = 1/|\mathcal{N}_u \cup \{v\}|;$ 
5 return  $\{v, A'_u\};$ 

```

Projected gradient descent algorithms are used in online convex optimization, and are known to have a regret, i.e. how much worse an algorithm performs compared to the best fixed solution, of $O(\sqrt{k})$ for $\eta = O(1/\sqrt{k})$ [202], if the function \mathcal{O} is convex. For us, this is not generally the case, but we show that DescentDiverse is effective in practice.

Projection Step The projection step ensures that we only add edges from the candidate node u , and that the edge weights are normalized, thus keeping the modified graph in the feasible set \mathcal{G}_u . The steps are detailed in Algorithm 2.

4.4 . Greedy approaches

Another option, but which has a high computational cost, is to evaluate the objective over a set of *candidate edges* \mathcal{C} , which is potentially as large as all nodes $v \notin \mathcal{N}_u$, where \mathcal{N}_u is the set of neighbors of u .

Under this formulation, Problem 1 can be stated as maximizing over all possible configurations of k edges in \mathcal{C} . Our objective function is not general monotone or sub-modular, so we rely on heuristics which add one recommendation at a time. We present here two alternatives to solve this problem : a “partition boosting” algorithm and a greedy algorithm.

Partition Boosting This algorithm uses the intuition that, at any step, the best edge to add is one to the partition that is “farthest” from the objective \mathbf{q} , but only if that partition is under-represented. An interesting option would be to also allow the removal of edges from over-represented partitions, but this has obvious drawbacks in practice. Stated otherwise, we choose some partition from the set $\{\mathcal{X} \mid d_i < 0, \mathcal{X} \in \mathcal{P}_G\}$. We also assume that the set of candidates is the union of candidates per partition, i.e., $\mathcal{C} = \cup_i \mathcal{C}(\mathcal{X}_i)$.

Once the partition is chosen, we add the edge in $\mathcal{C}(\mathcal{X}_i)$ that maximizes the objective, and repeat the process until k edges are added, as detailed in Algorithm 3.

There are several ways to choose the partition in ChoosePartition, and we present three alternatives that we have evaluated experimentally :

1. *the most under-represented partition* : $\mathcal{X}_i = \arg \min_i d_i$;
2. *draw randomly from the under-represented partitions* : $\mathcal{X}_i \sim \mathcal{U}(\{\mathcal{X} \mid d_i < 0\})$;
3. *draw from a soft-max distribution* : $\mathcal{X}_i \sim \text{SoftMax}(\mathbf{d})$.

We provide the mathematical intuition that supports PartitionBoosting for the Single-Node Exposure Diversity Maximization Problem.

Algorithm 3: PartitionBoostingDiverse

Data: graph G , target node u , exposure function \mathcal{E} , graph partition \mathcal{P}_G , target distribution \mathbf{q} , number of edges k , candidates $\mathcal{C} \subseteq V$

```
1 for  $i \in \{1, \dots, k\}$  do
2    $\mathcal{X}_i = \text{ChoosePartition}(\mathbf{d});$ 
3    $v_i = \arg \min_{v \in \mathcal{C}(\mathcal{X}_i)} \|\mathcal{E}_{G+(u,v)}(u \mid \mathcal{P}_G) - \mathbf{q}\|^2;$ 
4   add edge  $(u, v_i)$  to  $G : G \leftarrow G + (u, v);$ 
5 end
6 return  $\{v_1, \dots, v_k\}$ 
```

In a graph G with a Partition \mathcal{P}_G with n parts, let d_0 be a weight vector that infers a probability distribution

$$d_0 = (p_i) = \left(\frac{\alpha_i}{t}\right) \leftrightarrow (\alpha_i) \text{ with } t = \sum \alpha_i$$

The impact of the addition of edge e can be represented by

$$d_0 + d_e = (\alpha_i + \beta_i)$$

$$d_e = (\beta_i \mid \exists i \text{ s.t. } \beta_i > 0 \text{ and } \forall j \neq i \mid \beta_j < \epsilon)$$

This means that d_e is a vector where all values are very close to zero with one exception, that of the targeted partition. In the following index i means the targeted partition, j every other partition and \mathcal{H} the entropy function.

We have that any edge addition will monotonously increase our objective function as long as the addition respects these following conditions :

- β_i must be positive, as we add an edge
- β_j must be zeros or as close to it as possible
- $\alpha_i + \beta_i$ may never be greater than the $1/n$: meaning that we cannot add an edge that will lead to a partition becoming overrepresented

With these constraints we have a clear limit for d when adding all possible edges :

$$\lim_{e \rightarrow \text{inf}} d_0 + d_e = \left(\frac{1}{n}\right)$$

In other words β_i 's increase for partition i will incur a decrease for all partitions j since $\mathcal{H}(d)$ is a distribution. However the overall decrease is always less powerful as for some values it either will produce no penalty (for values already close to 0) or an increase (for values higher than their target counterpart) and the decrease is shared among all partitions which lowers the impact.

Greedy Algorithm The objective function that we have defined, including the entropy and the KL-divergence, is not monotone. There is hence no hope of having an approximation algorithm using the greedy algorithm *in general*. However, it remains an interesting heuristic : at each step, we add the edge that maximizes the marginal gain in the objective function, as detailed in Algorithm 4.

Algorithm 4: GreedyDiverse

Data: graph G , target node u , exposure function \mathcal{E} , graph partition \mathcal{P}_G , target distribution \mathbf{q} , number of edges k , candidates $\mathcal{C} \subseteq V$

```

1 for  $i \in \{1, \dots, k\}$  do
2    $v_i = \arg \min_{v \in \mathcal{C}} \|\mathcal{E}_{G+(u,v)}(u \mid \mathcal{P}_G) - \mathbf{q}\|^2$ ;
3   add edge  $(u, v_i)$  to  $G : G \leftarrow G + (u, v)$ ;
4 end
5 return  $\{v_1, \dots, v_k\}$ 

```

4.5 . Computational optimization

Optimizing the Gradient Computation The gradient computation is the main bottleneck of Algorithm 1. However, we can note that the derivative of the objective function depends on the whole adjacency matrix A but we will only focus on the final distribution for node u . The Kronecker product $(\mathbf{u}^\top \otimes P)$ is in fact a (very) sparse matrix, only having P for the coordinates corresponding to u . Moreover, we can simplify each Kronecker product by identifying all indices corresponding to the target node u :

- $(\mathbf{u}^\top \otimes P)$ simply becomes P at indices $[u, u]$
- for **PPR** and **F-J**, we only need the square sub-matrix of size n at $[un, un]$ in the Kronecker product $W^{-\top} \otimes W^{-1}$ which becomes $W_{u,u}^{-\top} W^{-1}$, hence the local gradient at u becomes (abusing notation) :

$$\frac{\partial \mathcal{E}}{\partial A}[u] = -\alpha(1 - \alpha)(I - \alpha A)_{u,u}^{-\top} (I - \alpha A)^{-1};$$

- for **BFS**, the Kronecker products simplify naturally if we only consider the local gradient at u : (i) $I \otimes A$ becomes A , (ii) $I \otimes A^2$ becomes A^2 , (iii) $A^\top \otimes I$ and $A^\top \otimes A$ disappear in the sum because A 's diagonal values are zeros, and (iv) $A^{\top 2} \otimes I$ becomes $A_{u,u}^{\top 2}$, leading to :

$$\frac{\partial \mathcal{E}}{\partial A}[u] = I + A + A^2 + A_{u,u}^{\top 2}.$$

In order to avoid computing the inverse, we use the Biconjugate gradient stabilized method (BiCGSTAB) to approximate it. This approximation is given by solving the following equation for \mathbf{x} :

$$W\mathbf{x} = P^+\mathbf{q} + \mathbf{u}, \quad (4.4)$$

where W is either $I - \alpha A$ or $I - \alpha A^T$, P^+ is the Moore-Penrose inverse of P , and \mathbf{q} the target distribution. The resulting solution corresponds directly to the exposure of node u . This change of shape, from matrix $n \times n$ to a vector of length n , entails minimal changes to Algorithm 1, where the projected gradient descent is performed directly on the node's exposure. No changes are needed in the projection step, Algorithm 2.

Computational complexity For the DescentDiverse algorithm, the first computation of the inverse matrix for the PPR and F-J exposure is unavoidable and has a complexity of $O(n^3)$, but the subsequent updates have a complexity of $O(n^2)$, as we only need to update the row and column of the target node u . For the BFS exposure function at depth 3, the initial complexity is also $O(n^3)$, but subsequent computations are $O(n^2)$, as we only need to update the row and column of the target node u and the candidate node c . The overall complexity is thus $O(n^3 + kn^2)$.

The overall complexity of solving the linear equation system in Eq. 4.4 using the BiCGSTAB iterative method depends on the size of the system, the density of the graph and the convergence rate which depends on the condition number of $I - \alpha A$. We used the default and standard tolerance of 10^{-6} meaning that the residual error is lower than 10^{-6} . Hence, when using the BiCGSTAB method, the complexity of the gradient computation depends on the sparsity of the networks is $O(kRm)$, where R is the number of rounds to convergence.

Optimizing the greedy step computations Instead of computing the full objective function at each step, we can incrementally compute the changes in the exposure function.

For **BFS** exposure, instead of computing powers of the matrix A , we can simply update the matrix A considering only the node u and the index of the candidate that has been added in the previous step, c . Once the initial \mathcal{E} has been computed, we can accurately estimate the changes in the exposure function by taking advantage of the following intuitions : (i) at hop 2 we have to add the neighbors of u to c and vice versa, by transitivity, i.e., add the vector A_u to the column A_c and the column A_c^\top to the row A_u ; and (ii) at hop 3 we have to add the neighbors of the neighbors of u to the neighbors of c and vice versa.

For **PPR** and **F-J** exposure, we can incrementally update the inverses using the Sherman-Morrison formula.

Computational complexity For the PartitionBoostingDiverse and GreedyDiverse algorithms, the complexity is $\mathcal{O}(kn^2)$, as we need to compute the objective function for each candidate node v , by using the optimizations that have been described above. Moreover, we have to do it for each of the candidates at each step of the algorithm. However, the PartitionBoostingDiverse algorithm only needs to compute for at most $\mathcal{C}_{\max} := \max_i |\mathcal{C}(\mathcal{X}_i)|$ candidates. Hence the complexity is $\mathcal{O}(n^3 + k\mathcal{C}_{\max}n^2)$ for PartitionBoostingDiverse and $\mathcal{O}(n^3 + k\mathcal{C}n^2)$ for GreedyDiverse.

4.6 . The user-content graph extension

The previous model captures the framework of a user-user (social) graph, where we only have access to user information.

We extend our social setting to user-item graphs to include items created by users, which can be consumed (liked or shared) by other users. In this framework, we define $G = (U \times C) = (E_G = \{E_U, E_C, E_T\}, V_G = \{V_U, V_C\})$, where G represents the entire graph, U the user graph, and C the content (or item) graph. Edges are directed and categorized as E_U (between users), E_C (between contents), or $E_T = E_{V_U \cap V_C}$ (connecting users and contents).

Propagation model We keep the same two base assumptions as in the previous model. First, we still consider that information travels along all edges in the above hybrid graph, even if the relationship semantics underlying the link between users, items, or both might differ. This means that there are more pathways for the propagation of information. Secondly, we assume that the partitions are separate : the user partitions have the same meaning as the one discussed before, while the items are clustered in partitions only on the item-item graph (via similarity, for instance).

Problem statements The previously defined problem now has two variants : we optimize either along user edges or item. In other words, we recommend user nodes or item nodes :

Problem 2 (Single-Node User Exposure Diversity Maximization). *Given a graph $G = (U \times C)$, a node $u \in V$, a partition \mathcal{P}_U , an exposure function $\mathcal{E}(u)$ maximize the diversity measure $\sigma(\mathcal{E}(u | \mathcal{P}_G))$ for a fixed number k of edge additions in U to the graph starting at u .*

Problem 3 (Single-Node Item Exposure Diversity Maximization). *Given a graph $G = (U \times c)$, a node $u \in V$, a partition \mathcal{P}_U , an exposure function $\mathcal{E}(u)$ maximize the diversity measure $\sigma(\mathcal{E}(u | \mathcal{P}_G))$ for a fixed number k of edge additions in I to the graph starting at u .*

For the algorithms in Section 4.2 to work we have to redefine how exposure in the graph is estimated, and then we have to adapt the derivation step. This is because the exposure is now twofold : a given user is influenced by their connection (user neighbors) or by the content they consumed (item neighbors).

We model the graph $G = (U \times C)$ as a block matrix and use the same objective function :

$$G = \begin{bmatrix} U & T \\ T^\top & C \end{bmatrix}; P = \begin{bmatrix} P_U & 0 \\ 0 & P_C \end{bmatrix}; \mathbf{u} = \begin{bmatrix} \mathbf{u} \\ T^\top \mathbf{u} \end{bmatrix}; Q = \begin{bmatrix} \mathbf{q}_U \\ \mathbf{q}_C \end{bmatrix}$$

$$\arg \min_{G \in \mathcal{G}} \|\mathbf{d}\|^2 = \|\mathcal{E}_G(u | \mathcal{P}) - Q\|^2 = \|P \cdot \mathcal{E}_G(u) \cdot \mathbf{u} - Q\|^2,$$

This model does not require complete knowledge of the social network. If links between contents (or users) are unavailable, meaning C (or U) is empty, they can be substituted with the identity matrix. While the general case assumes two distinct partitions based on different semantics, either partition can be unknown (represented as an empty matrix) without affecting the optimization of the other. In the most general scenario, both partitions can coexist simultaneously.

We now state the new exposure functions for the user-item graph :

Proposition 5 (PPR Exposure). *Given the block matrix formulation of a graph $G = (U \times C)$ such that C has an invertible Schur complement S in G , then the PPR exposure is given by :*

$$\mathcal{E}_G^{PPR} = (1 - \alpha) \begin{bmatrix} S^{-1} \\ -C^{-1}T^\top S^{-1} \end{bmatrix} \quad (4.5)$$

with $S = (I - \alpha U - \alpha^2 T (I - \alpha C)^{-1} T^\top)$

Proposition 6 (FJ Exposure). *Given the block matrix formulation of a graph $G = (U \times C)$ such that C has an invertible Schur complement S in G , then the FJ exposure is given by :*

$$\mathcal{E}_G^{FJ} = (1 - \alpha) \begin{bmatrix} S^{-1} \\ -C^{-\top}T^\top S^{-1} \end{bmatrix} \quad (4.6)$$

with $S = (I - \alpha U^\top - \alpha^2 T (I - \alpha C^\top)^{-1} T^\top)$

Proposition 7 (BFS Exposure). *For any block matrix formulation of a graph $G = (U \times C)$, the BFS (limited to a depth of 3) exposure is given by :*

$$\mathcal{E}_G^{BFS} = \begin{bmatrix} U^3 + U^2 + U + UTT^\top + TT^\top U + TCT^\top + TT^\top \\ T^\top U^2 + T^\top U + T^\top TT^\top + CT^\top U + CT^\top + C^2 T^\top + T^\top \end{bmatrix} \quad (4.7)$$

cesses :

$$\mathcal{E}_G(u) = \theta_1 \mathcal{E}_U(u) + \theta_2 \mathcal{E}_C, \quad (4.8)$$

where θ_1 and θ_2 are hyperparameters that control the influence of each exposure, and \mathcal{E}_U is the previously defined exposure in the user graph.

The exposure function \mathcal{E}_U is given by :

$$\mathcal{E}_C^{PPR}(u) = W_u^\infty = (1 - \alpha) \mathbf{u} T (I - \alpha C)^{-1} T^\top,$$

with the resulting overall exposure function :

$$\begin{aligned} \mathcal{E}_G(u) &= \theta_1 \mathcal{E}_U(u) + \theta_2 \mathcal{E}_C(u) \\ &= \theta_1 (1 - \alpha_1) (I - \alpha_1 U)^{-1} + \theta_2 (1 - \alpha_2) \mathbf{u} T (I - \alpha_2 C)^{-1} T^\top. \end{aligned} \quad (4.9)$$

We can notice that this exposure has a similar form to that of the Schur complement of C in G using the block matrix formalization $G = (U \times I)$:

$$S = \left(I - \alpha U - \alpha^2 T (I - \alpha C)^{-1} T^\top \right).$$

This shows that we can implement a dual propagation model using block matrix formalization and that it coincides with our initial approach.

Derivations for user-item graphs The derivation formulas for the GreedyDiverse algorithm remain consistent with Eq. 4.3, but $\partial \mathcal{E}$ varies depending on whether the optimization is over users or contents, i.e., whether U or T is modified. Furthermore, PPR and F-J exposures allow for distinct propagation processes by using separate α parameters for each. For simplicity, we assume a single value of α for U , C , and T in the following.

Proposition 8 (PPR Exposure Derivative). *For the PPR exposure function, the derivatives corresponding to the two problems are given by :*

$$\frac{\partial \mathcal{E}_G^{PPR}}{\partial U} = -\alpha(1 - \alpha) \begin{bmatrix} (S_{u,u}^{-\top} S^{-1}) \\ (C^{-1} T^\top) (S_{u,u}^{-\top} S^{-1}) \end{bmatrix} \quad (4.10)$$

$$\frac{\partial \mathcal{E}_G^{PPR}}{\partial T} = (1 - \alpha) \begin{bmatrix} -\alpha^2 (S_{[u,u]}^{-\top} S^{-1}) (T(I - \alpha C)^{-1} + (T(I - \alpha C)^{-\top})) \\ -\alpha^2 ((C^{-1} T^\top) (S_{[u,u]}^{-\top} S^{-1}) (T(I - \alpha C)^{-1} + (T(I - \alpha C)^{-\top}))) + S_{[u,u]}^{-\top} C^{-1} \end{bmatrix} \quad (4.11)$$

Proposition 9 (F-J Exposure Derivative). *For the Friedkin-Johnsen exposure function, the derivatives corresponding to the two problems are given by :*

$$\frac{\partial \mathcal{E}_G^{FJ}}{\partial U} [u] = -\alpha(1 - \alpha) \begin{bmatrix} S_{u,u}^{-1} S^{-\top} \\ (C^{-\top} T^\top) (S_{u,u}^{-1} S^{-\top}) \end{bmatrix} \quad (4.12)$$

$$\frac{\partial \mathcal{E}_G^{FJ}}{\partial T}[u] = (1-\alpha) \left[\begin{array}{c} -\alpha^2 \left(S_{[u,u]}^{-\top} S^{-1} \right) \left(T(I - \alpha C^\top)^{-1} + (T(I - \alpha C^\top)^{-\top}) \right) \\ -\alpha^2 \left((C^{-\top} T^\top) \left(S_{[u,u]}^{-\top} S^{-1} \right) \left(T(I - \alpha C^\top)^{-1} + (T(I - \alpha C^\top)^{-\top}) \right) \right) + S_{[u,u]}^{-\top} C^{-\top} \end{array} \right] \quad (4.13)$$

Proposition 10 (BFS Exposure Derivative). *For the BFS exposure function, the derivatives corresponding to the two problems are given by :*

$$\frac{\partial \mathcal{E}_G^{BFS}}{\partial U}[u] = \left[\begin{array}{c} U_{u,u}^{T^2} I + U_{u,u}^\top U + U^2 + U + U_{u,u}^\top I + T T_{u,u}^\top I + T T^\top \\ T^\top U + U_{i,i}^\top T^\top + T^\top + C T^\top \end{array} \right] \quad (4.14)$$

$$\frac{\partial \mathcal{E}_G^{BFS}}{\partial T}[u] = \left[\begin{array}{c} (U T_{u,u} I) + (T_{u,u} U) + (T_{u,u} U^\top) + (U^\top T_{u,u} I) + T C + T C^\top + T \\ U_{u,u}^{2\top} I + U_{u,u}^\top I + T T_{u,u}^\top I + T^\top T + U_{u,u}^\top C + C + C^2 \end{array} \right] \quad (4.15)$$

Algorithms In this extended setting, our previous algorithms can be adapted with minimal changes. The projected gradient descent algorithm remains the same, utilizing the updated formulas for the objective function and gradient. The projection step depends on the optimization target : if optimizing over users (U), the projection is applied to the user matrix (U); if optimizing over contents, the projection is applied to the transition matrix (T), as links are created between user nodes and content nodes.

Baseline algorithms also remain unchanged, requiring only a modification in the candidate selection process (users or items). The primary computational challenge lies in matrix inversion. Specifically, we use Eq. 4.4 with $W = S$, which is the only matrix inverse that needs to be recomputed at each step. Other inverses, such as C^{-1} and $(I - \alpha C)^{-1}$, can be precomputed. Importantly, we do not need to invert the entire adjacency matrix of the graph; only subsets of this larger matrix are required. Consequently, the overall complexity of the DescentDiverse algorithm in the user-item graph setting is $O(x^3 + kx^2)$, where $x = \max(|U|, |C|)$. Typically, $|C| > |U|$, as there are generally more content items than users (e.g., on platforms such as X, the number of posts exceeds the number of users).

4.7 . Experiments

Implementation We implemented our experiments in Python using the NetworkX package for graph manipulation and the computational packages numpy and scipy.sparse. The GreedyDiverse and PartitionBoostingDiverse algorithms support multiprocessing for faster running time.¹

1. github.com/Jonathan-COLIN/Optimizing-Diverse-Information-Exposure-in-Social-Graphs

Table 4.1 – Dataset details. For the dataset marked with †, the communities represent the number of node features, and for datasets marked ‡ the number of communities found by the Louvain algorithm.

| Dataset | Nodes | Edges | Existing | #Comm. |
|------------------------|---------|---------|----------|------------------|
| <i>Polbooks</i> | 105 | 441 | ✓ | 3 |
| <i>Football</i> [203] | 115 | 613 | ✓ | 12 |
| <i>MovieLens</i> [204] | 1,680 | 264,718 | ✗ | 19 [†] |
| <i>Reddit</i> [205] | 34,671 | 123,570 | ✗ | 54 [‡] |
| <i>Facebook</i> [206] | 63,392 | 816,886 | ✗ | 74 [‡] |
| <i>Amazon</i> [207] | 334,863 | 925,872 | ✗ | 230 [‡] |

Datasets We evaluate our approach on several real graph datasets, described in Table 4.1.

The *MovieLens* dataset is built from the *Movielens100k* dataset, in which we consider only movies. Each movie has a distribution over 19 movie genres. We assign an edge between two movies if they share more than two thirds of their movie genres. After this linking process, each node’s genre vector is updated as the average of all genres of its neighbors. Using these vectors we can then build the partition matrix P , where each row corresponds to a genre and each column to the genre vector of a node. This allows us to exemplify the functioning of our algorithms on *non-discrete partition*.

Polbooks is a collection of US political books and their references, the dataset also records each book’s political alignment : liberal, neutral or conservative. This dataset features a clear separation between liberal and conservative books and is interesting to study when you use these alignments as ground-truth for communities. *Football* is a dataset of US college football representing the Division I games for the 2000 season. It also provides information about conferences, groups of colleges that often play football against each other, which can be used for ground-truth communities. Furthermore, we use the greedy Louvain modularity community detection algorithm [113] to infer community structures in datasets without ground-truth.

Target node selection The optimization process depends greatly on the initial situation of the target node. Indeed, high degree nodes have very different exposures compared to very low degree nodes. As such, in the following, target nodes are selected according to their degree. We batch nodes into three groups of high, medium and low degree and select an equal amount of nodes among these three groups.

Generating Non-Uniform Target Distributions The target distribution q is the main factor in how the exposure is optimized. Indeed, a standard uniform distribution would lead to an equal representation of each partition and a unit vector used as a distribution would lead to only promoting a single partition. To show the power of *non-uniform target distributions*, we can adapt q to take user interests into account. We assume that for all users, their initial exposure before the optimization process represents their own interest. From the initial exposure distribution, we scale this distribution by adding the uniform distribution and a normal distribution centered at u 's partition. This allows us to promote the partitions that are closer to the user's initial interest and reduce the probability of partitions that are further away. We do this for all datasets.

Targeting non-uniform partitions rather than uniform partitions allows us to present recommendation that are less "extreme" in their diversity, and which would allow the user to be gently nudged towards a more diverse exposure. Furthermore, if a given partition is recognized as harmful, we can improve the target distribution by setting the corresponding value to a very small value.

Baselines We compare our approach to the following baseline algorithms : random assignment of edges (Random), triadic closure (recommending based on common neighbors, TriadicClosure), polarization and disagreement minimization algorithms in the state-of-the-art, and a global and local diameter minimization algorithm. We compare the baselines with the Greedy, Partition-Boosting and Descent algorithms.

We used the Polarization and Disagreement minimization algorithm presented in [163], in particular their simpleGreedy method. Given a candidate set of edges, it iteratively selects the most promising one with regards to a standard measure of Polarization and Disagreement. Following their implementation we initialize our opinions uniformly between -1 and 1 . Since this is a global algorithm, operating on the entire graph, we constrain the candidate set to only include edges starting from the parameter node u . Additionally, we also compare our approach to a graph diameter minimization algorithm as presented in [117]. Their global method identifies center nodes of the graph and connects them together, we adapted it to always include the target node in the set of center nodes. Their local method solves a decision problem using linear programming in order to identify nodes to add. We identify these algorithms by GlobalDiameterReduction and SingleSourceDiameterReduction respectively.

All experiments choose 30 target nodes, equally distributed among high-degree, medium-degree and low-degree nodes in their respective graph. We set $\alpha = 0.05$ and $\eta = 0.1$.

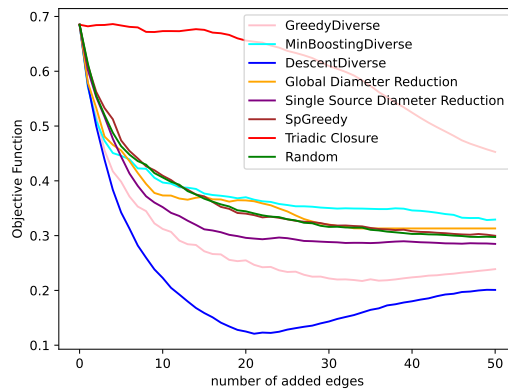


Figure 4.2 – Objective vs. number of edge additions, Polbooks dataset, ground truth communities, BFS exposure and uniform target distribution.

Ground-truth communities We start by presenting the results of the graphs for which we have ground-truth communities, *Polbooks* and *Football*, for both uniform and preferential target distributions, Fig. 4.2 shows the results on the *Polbooks* dataset as they are the most representative and the other plots are relegated to the appendix in Fig. 7.1.

We notice that *DescentDiverse* manages to achieve its objective faster than other approaches. Moreover, as expected, *TriadicClosure* is among the least effective approaches. Interestingly, *Random* is quite competitive (and always better than *TriadicClosure*), which can be attributed to the relatively small sizes of the graphs. Here, the *PartitionBoostingDiverse* manages to improve exposure but in a most cases, it does not manage to optimize past a certain point; indeed, once it can no longer find a partition to boost (it looks only for under-represented ones), it is liable to get “stuck” for the rest of the run. On the other hand, the diameter reduction approaches of [117], especially *Single-DiameterReduction*, are very competitive, even for preferential distributions.

One interesting, and potentially important, finding can be seen in the results on the BFS exposure function. One can notice that there exists a point in the process where adding new edges becomes counter-productive. This makes intuitive sense, seeing that the objectives are non-monotonous. Moreover, it means that one cannot indiscriminately add edges to maximize diversity, but that there is an optimal number of edges, which will depend on the graph and the node itself.

Inferred communities Where no ground truth is available, we generated partitions using a community detection algorithm. The results comparing *Random* and *TriadicClosure* to *DescentDiverse* on the *Amazon* datasets are shown in Fig. 4.3, results on the facebook and reddit datasets are presented

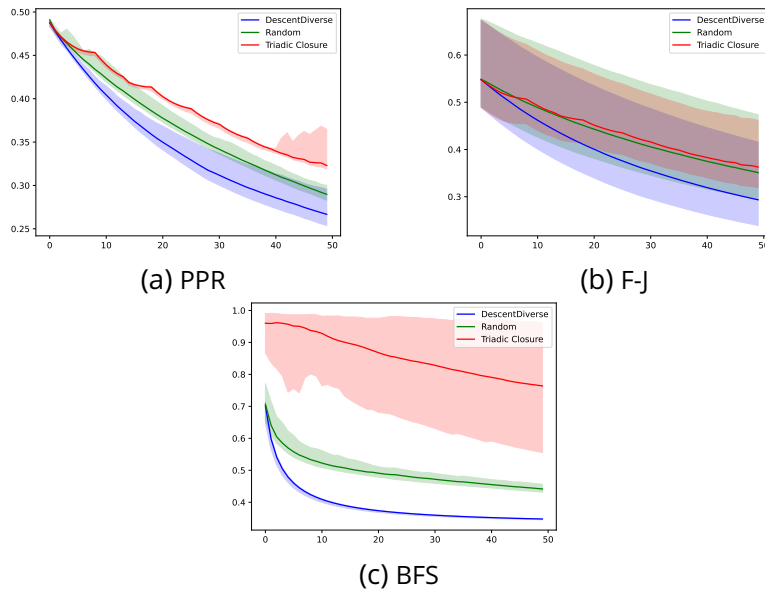


Figure 4.3 – Objective vs. number of edge additions, Amazon dataset, inferred communities, uniform target partition.

in the Appendix 7.2. The results are similar to those on graph having ground truth communities, with DescentDiverse being the most effective at the end of the recommendation process. Interestingly, recommending in the “classic” approach, TriadicClosure, is almost always worse than Random. This makes some intuitive sense as recommending based on common neighbors would get the user “stuck” in the same community, whereas recommending random links at least gives a chance of escaping their current community and thus increasing the diversity of exposure.

Effect of degree Fig. 4.4 shows the differences in the evolution of the optimization objective depending on the degree of the parameter node u . In *Polbooks*, higher degree nodes seem to be more diverse at the start; this may be because they tend to be connected to many nodes, and thus have a less focused exposure. Moreover, high degree nodes seem to not benefit as much from the optimization process as low degree nodes. Low degree nodes, conversely, start from a less diverse point (which may be an artifact of their low connectivity). All types of nodes seem to behave similarly through recommendations and achieve the same rate of improvement.

Conversely, in *MovieLens*, the high degree nodes seem to start less diverse, and low degree nodes more diverse. This can be attributed to how the dataset is constructed, by similarity to other movies and their genre, so high degree nodes are more likely to be connected to similar nodes. In general, the rate of improvement is similar, with the exception of the BFS exposure function,

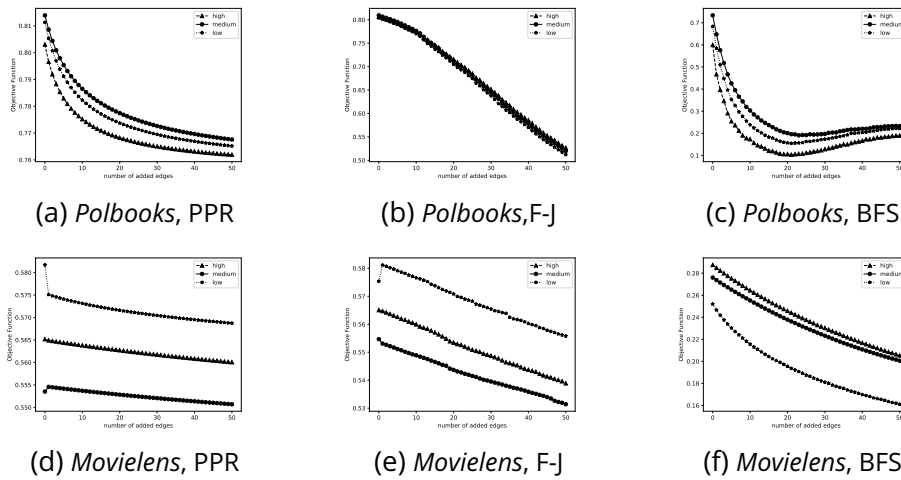


Figure 4.4 – Objective vs. number of edge additions, depending on node degree, DescentDiverse, uniform.

where low degree nodes improve faster and reach a higher diversity.

Non-discrete communities Our formulation also allows for optimizing in cases where partition are non-discrete or overlapping. As detailed above, we adapted the *MovieLens* dataset so that each movie is defined by a distribution over its movie genres, and optimize for uniform or preferential exposure. The results are shown in Fig. 4.5. Again, the DescentDiverse algorithm is the most effective, and TriadicClose the least effective, in line with the other results.

Recommendation on the user-item graph Figures 4.6 and 4.7 present the results for the user-user and user-item diverse recommendation objectives on the user-item graph. The results are similar to those on the original user-user network, with DescentDiverse achieving the best diversity measure. Notably, recommendations on the user-item graph converge faster than on the user-user graph. For PPR and F-J, the minimal objective is reached after approximately 30 recommendations, while for BFS, it is reached after only 10 recommendations. This speed-up in convergence is likely due to the additional paths in the graph available for optimizing exposure, such as user-item and item-item links. Note that this scenario cannot be directly compared to state-of-the-art baselines, as they are not designed for user-item graphs.

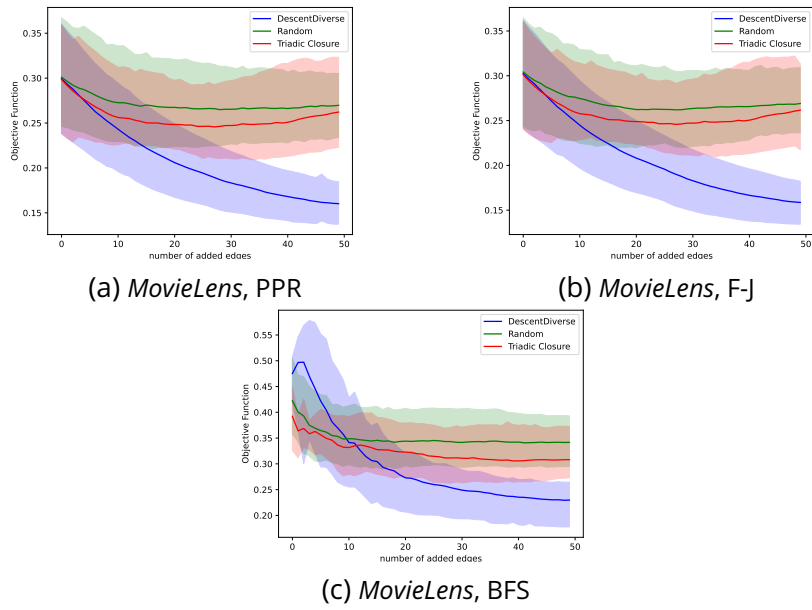


Figure 4.5 – Objective vs. number of edge additions, *MovieLens*, non-discrete communities.

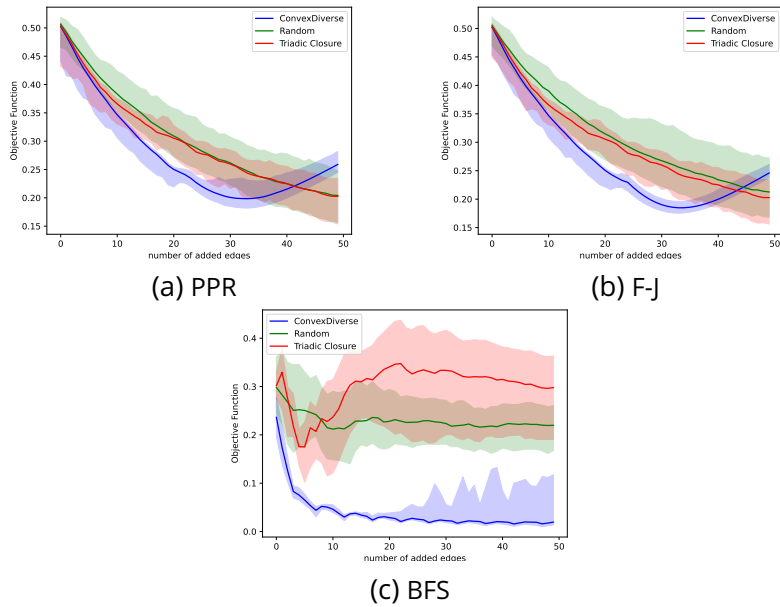


Figure 4.6 – Objective vs. number of edge additions, *MovieLens User-item*, user-user recommendation, discrete communities, uniform exposure objective.

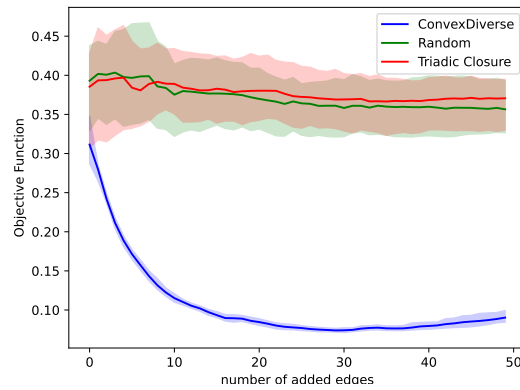


Figure 4.7 – Objective vs. number of edge additions, *MovieLens User-item*, BFS exposure, user-item recommendation, discrete communities, uniform exposure objective.

| Dataset | (density) | BFS | PPR | F-J | Greedy | Part. Boost | S.S. Diam. Min. | Global Diam. Min. | Pol. Min. |
|---------------------|--------------|--------------------------------|--------------------|--------------------|--------|-------------|-----------------|-------------------|-----------|
| Polbooks | 0.080 | 0.141 | 0.217 | 0.239 | 2.11 | 2.37 | 0.202 | 0.119 | 4.29 |
| Football | 0.092 | 0.155 | 0.247 | 0.266 | 2.23 | 2.03 | 0.176 | 0.033 | 4.99 |
| Movielens | 0.186 | 43.7 | 18.5 | 20.5 | 120 | 110 | 10.5 | 11.7 | 3064 |
| Reddit | $2.04e^{-4}$ | 182 | 136 | 157 | - | - | - | - | - |
| Facebook | $4.06e^{-4}$ | 656 | 513 | 599 | - | - | - | - | - |
| Amazon | $1.65e^{-5}$ | 830 | 2305 | 2441 | - | - | - | - | - |
| Movielens User-item | $6.09e^{-2}$ | (u-u) 3.62 (u-i) 4,17 | (u-u) 12.9 - | (u-u) 13.3 - | - - | - - | - - | - - | - - |

Table 4.2 – Running time per iteration in seconds. For *Movielens User-Item*, (u-u) represents user-to-user recommendation, while (u-i) represents user-to-item recommendation.

Running Time We measured the running time of all approaches and reported them in Table 4.2. Baselines could not be tested on larger networks as they scale poorly with graph size. In particular the polarization minimization algorithms. However, for our greedy and partition boosting algorithms this could be solved if provided a set of nodes to choose from, in other words if they are used as re-ranking at the end of a recommendation pipeline. We can see that our convex approach is somewhat scalable in spite of its matrix inversion which become extremely costly at greater dimensions.

4.8 . Conclusion

We presented in this chapter a formalization of diversity optimization when keeping into account a node's exposure to information, expressed as a distribution over partitions of opinions in the graph. Our formulation allows to express the problem of local edge additions as an optimization problem, over objectives derived from entropy-based measures. We discuss formalisms for social networks (user-user graphs) but also extend the methods and formalisms to hybrid graphs composed of a social network and a content network. We presented several algorithms to solve the problem, and showed that they outperform state-of-the-art algorithms in terms of diversity measures, and that they are scalable to large graphs.

The fact that our formulation of the objective function is non-monotone opens up the interesting problem of finding the optimal number of edges to add to maximize diversity. Moreover, one should account for the fact that some nodes might not belong to any opinion partition; it is not immediately apparent which the best semantics is to take into account such nodes. Another interesting avenue for research is when the partitions are not known in advance, and hence the diversity measure is a result of a user's feedback in an online algorithm.

5 - Reinforcement Learning for Optimizing Diverse Exposure

In this chapter we discuss a deep reinforcement learning framework that solves the information exposure maximization problem while addressing the shortcomings of our previous convex method.

5.1 . Introduction

A key component of most opinion dynamics models, in particular Friedkin-Johnsen is the knowledge of personal opinion vectors. As a consequence all structural intervention based on the FJ model that aim to minimizing polarization have the same need for private information to be accessible. Considering global concerns about privacy on the Internet, this core feature becomes a severe limitation and questions the applicability and relevance of such algorithms.

In our Convex optimization approach (Section 4), we assumed that opinions were clustered into communities according to a partition and assumed personal opinion vectors to be from ground truth community information or the results of community detection algorithm. However, the approach has the same flaw of needing the explicit input of personal opinion vectors in the optimization loop. Furthermore, this modification of the network's structure completely disregards whether new links are accepted by the users at each step. In other words, some recommendation may have been unsatisfying to the user and as such not adopted, meaning that the corresponding link should not have been added to the graph. In the following, we revisit the algorithm to include user adoption and satisfaction as a key component of the optimization process.

Hence, we require an algorithm that simultaneously maximizes exposure diversity and produces credible recommendations while not knowing potentially private user opinion vectors. In other words, the model must infer the user's hidden opinions from interactions and changes on the network and update those estimates online. Using these refined opinion estimates, the system can perform structural changes on the graph that both reduce polarization and improve the diversity of information exposure.

When confronted with such decision-making problems, i.e. which edge to add to the graph in order to optimize information exposure in a complex, uncertain and evolving environments such as social networks, one would naturally consider a reinforcement learning based approach. Indeed, reinforcement learning is designed for scenarios where the optimal strategy is not

known in advance but must be discovered as an agent interacts with an environment. This corresponds to our situation not because of the fact that personal opinion vectors are unknown but also that it can take into account user satisfaction and adoption. The agent can learn from the feedback it receives after each recommendation, allowing it to adapt its strategy over time based on what works best in reality.

5.2 . Problem definition

We need a framework able to deal with three key features. First, it should be able to infer opinion vectors. In other words, the framework should not incorporate the partition matrix \mathcal{P} into its optimization process. Secondly, it should optimize information exposure for diversity and lastly, any edge recommendations should comply with user satisfaction.

Using the exposure as defined in Section 3.2 which is a function that, for a given user, measures the amount of information they (the user) receive from each community present in the graph. We presented three ways to implement such function : based on pagerank, FJ opinion dynamics model and breadth-first search. Following the same development as in Problem 1, we aim to minimize the distance between the exposure and a target distribution, generally a uniform distribution. We define next the concept of *Satisfaction* and provide an implementation in Section 5.4 :

Definition 5 (Satisfaction). *The satisfaction function is a function $S_G : V \times V \rightarrow [0, 1]$ which for two users u and v returns the probability of the edge (u, v) being adopted by user u .*

The graph being directed means that the satisfaction function is not necessarily symmetric $S_G(u, v) \neq S_G(v, u)$. Furthermore, we consider not only edge additions but also edge reinforcements. Edge reinforcement means that we strengthen the relationship of an already present link in the graph. This essentially translates to increasing the corresponding edge weight. Small communities have fewer candidates for *unique* links which severely limits the maximum exposure. Subsequently, the balancing of exposure between small and large communities becomes impossible if one only considers unique edge additions. Thus, enabling the strengthening of links directly combats that unfairness towards disadvantaged groups.

We formally define the main problem as follows :

Problem 4 (Single-Node Dual Optimization). *Given a graph $G = (V, E)$ and a node $u \in V$, an exposure function $\mathcal{E}(u)$ and a satisfaction function S_G maximize the cumulative reward $R = f(\mathcal{E}, S_G)$ over a fixed number k of edge additions or reinforcement to the graph starting at u .*

The above problem contains two sub-problems :

Problem 5 (Single-Node Exposure Diversity Maximization). *Given a graph $G = (V, E)$ and a node $u \in V$, an exposure function $\mathcal{E}(u)$ maximize the diversity measure $\sigma(\mathcal{E}(u_G))$ for a fixed number k of edge additions or reinforcement to the graph starting at u .*

Problem 6 (Single-Node Satisfaction Maximization). *Given a graph $G = (V, E)$ and a node $u \in V$, a satisfaction function S_G maximize the satisfaction $S_G(u)$ for a fixed number k of edge additions or reinforcement to the graph starting at u .*

Our focus is on solving Problem 4 which is a combination of Problems 5 and 6 as multi-objective optimization. The two objectives are somewhat conflicting as improving satisfaction means augmenting similarity which in turns may reduce information exposure diversity. Hence, we need to find a balance between the two objectives. This is achieved by using a linear combination of both objectives as the reward function for our RL agent, as we will see next.

5.3 . Formulation

5.3.1 . The general RL framework

The standard RL system uses the agent-environment setting where the agent learns through interactions with an environment. Building such a system can be divided into three components : environment construction, state representation and policy learning. This is usually formalized as a Markov decision process (MDP) in the form of the tuple $(\mathcal{S}, \mathcal{A}, \mathcal{R}, P, \gamma)$. \mathcal{S} is the set of all possible states the environment can be in, \mathcal{A} is the set of all available actions the agent can make, \mathcal{R} is the reward function that provides a quantitative feedback to the agent on how good their selected actions were, P is the transition probability from one state and action to a new state and γ is the discount factor, i.e., how much the agent should prioritize short-term or long term rewards. With this formulation, the agent tries to learn an optimal policy π , a function that given a state returns the probability of taking a specific action, that maximizes discounted long-term reward :

$$\pi^* = \arg \max_{\pi} \mathbb{E}_{\pi} \left[\sum_{t=0}^{\infty} \gamma^t r_t \right]$$

It is important to carefully construct the environment and the state representation. In the setting of optimizing diverse information exposure locally for a specific user on a social platform, the environment is the social network itself. Subsequently, a state is the combination of the (possibly changed) graph and node and an action would be the addition of a new link or reinforcement

of a pre-existing link. The reward captures behaviors we want to incentivize and when complemented with a penalty it can also capture behaviors we want to discourage. In our case, the reward associated with this edge addition depends on two factors : satisfaction and diversity. Satisfaction is the probability that the user accepts and maintains the new edge while diversity measures the net gain in information exposure. This framework can be summarized in the following MDP $(\mathcal{S}, \mathcal{A}, \mathcal{R}, P, \gamma)$:

State s is given by the tuple $(G, u, \mathcal{H}_t(u))$, the graph is the environment and evolves along with the RL recommendation process, the node u is the target of the optimization and $\mathcal{H}_t(u)$ is the target's history of recommended nodes. Indeed, a recommendation becomes less attractive if it loses its novelty and is already known, and in our case we want to avoid recommending the same node multiple times.

Action a is given by the set of all possible nodes in the network.

Reward function $\mathcal{R}(s_t, a_t)$ is a combination of satisfaction and diversity gains, expressed as a linear combination of both components :

$$\mathcal{R}(s_t, a_t) = \alpha S_G(u, a_t) + \beta \Delta \sigma (\mathcal{E}(u | \mathcal{P}_G))$$

The satisfaction function given a node and an action returns a value between 0 and 1 with 0 meaning rejection and 1 meaning adoption.

Transition probabilities \mathcal{P} are unknown in practice but are a function of the adoption or not of a given recommendation. This is linked to the satisfaction with the recommended link.

Discount factor γ balances out immediate and future rewards, and is a parameter in our setting.

5.3.2 . The DQN framework

Traditional RL methods like tabular Q-learning store their knowledge in a massive Q-table. Each cell in this table contains the estimated value (Q-value) of taking a specific action in a specific state. For this, they need a discrete and finite state space and this explains why these methods struggle with complex high dimensional data. This is effectively the case of social network as graphs. Mathematically, a graph is represented by its adjacency matrix which is an exponentially large according to the number of nodes. However, this can be solved by taking advantage of neural network using deep Q-learning network (DQN). Indeed, neural networks are good at learning complex and non-linear relationships in the data, in our context the relationships between users.

Deep Q-Learning Network (DQN) is an approach that learns a Q-function which given a state and an action returns the expected reward. Q-learning is an off-policy value-based learning scheme for finding a greedy target policy. In

other words it means that it learns action values (i.e. the Q function) with one policy while still estimating the optimal values needed to act greedily under the best possible policy :

$$\pi(s) = \arg \max_a Q_\pi(s, a)$$

The DQN architectures consists of three components : an experience replay buffer, an online network and a target network. The online network is updated at every step of the training process to predict Q-values, while the target network, updated less frequently, provides stable targets for the Bellman updates. This separation reduces correlations and prevents divergence, ensuring more robust convergence of the Q-function. The DQN framework uses the online network to compute target values :

$$y = r + \gamma \max_{a'} Q(s', a'; \theta^-)$$

and it then updates its parameters according to the following loss :

$$L(\theta) = \mathbb{E}_{(s,a,r,s') \sim \pi} \left[(y - Q(s, a; \theta))^2 \right]$$

The loss is based on both the current reward and future reward as DQN is inherently designed to maximize cumulative reward. This allows it to learn strategies that may sacrifice a short-term reward for a longer-term benefit, an objective difficult to capture with one-step methods. It learns its policy directly from user feedback which allows the system to adapt to evolving user preferences and graph dynamics without relying on fixed, pre-defined rules, making it highly personalized and adaptable.

5.3.3 . Using DQN with graphs

In order to take advantage of deep learning models' ability to capture complex and non-linear relationships in the data, in our implementation we complement our DQN setting with graph representation techniques that provide embeddings for the whole graph structure of the network and also the target recommendation nodes. We use the generic Structure2Vec Deep Q-Learning Network (S2V-DQN) framework [98]. S2V-DQN builds upon the standard DQN framework by directly incorporating graph structure into the reinforcement learning process. It uses a graph neural network (S2V) to compute node embeddings that capture both local neighborhood information and global graph context. These structure-aware embeddings then serve as inputs to the Q-network, enabling the agent to make decisions based on rich topological relationships rather than just node features. This architecture allows S2V-DQN to naturally generalize across different graph regions and learn policies that understand complex structural patterns, making it particularly powerful for combinatorial optimization problems on graphs.

As described in Section 5.3.1, we represent our state as a triplet of graph, node and a history of past actions. This means that we need both graph embeddings and node embeddings to capture both local neighborhood information and global graph context for our Problem 4.

We use Graph2vec [81] for graph embeddings. Graph2vec is an unsupervised learning technique that generates embeddings for entire graphs, effectively creating a low-dimensional vector representation of a graph's global structure. Inspired by document embedding models like Doc2Vec, it treats a whole graph as a document and its rooted subgraph patterns as words. The model learns to embed graphs by predicting a graph's presence based on its constituent subgraph features, capturing structural similarities and differences. This allows dissimilar graphs to be mapped to distant vectors in the embedding space, while structurally similar graphs are positioned close together. This encoding strategy provides fixed-size embeddings for variable-size graphs and is often used as a useful feature extractor for downstream ML tasks, in our case deep reinforcement learning.

For node embeddings we used GraphSage (Graph Sample and Aggregate) [79] which is an inductive learning framework that generates embeddings for nodes by sampling and aggregating features from a node's local neighborhood. Unlike transductive methods that require the entire graph during training, GraphSAGE learns a function that generates embeddings by sampling a fixed number of neighbors and then iteratively combining their features with the target node's own features. This allows it to efficiently generalize to unseen nodes or entirely new graphs, making it highly scalable and practical for dynamic environments where the graph structure evolves over time.

The combination of these two specific algorithms for graph and node embedding provide our S2V-DQN framework with a powerful feature, that of transferability between the training graphs and inference graphs. This is due to the properties of our embedding techniques : Graph2vec can transform graphs of varying size into fixed size vectors and GraphSage can infer embeddings for never-seen-before nodes. This means that with minimal neural network modifications, as we shall discuss next, we can train our agent on one graph and then use it on another.

A summary of our entire S2V-DQN framework architecture is presented in Fig. 5.1.

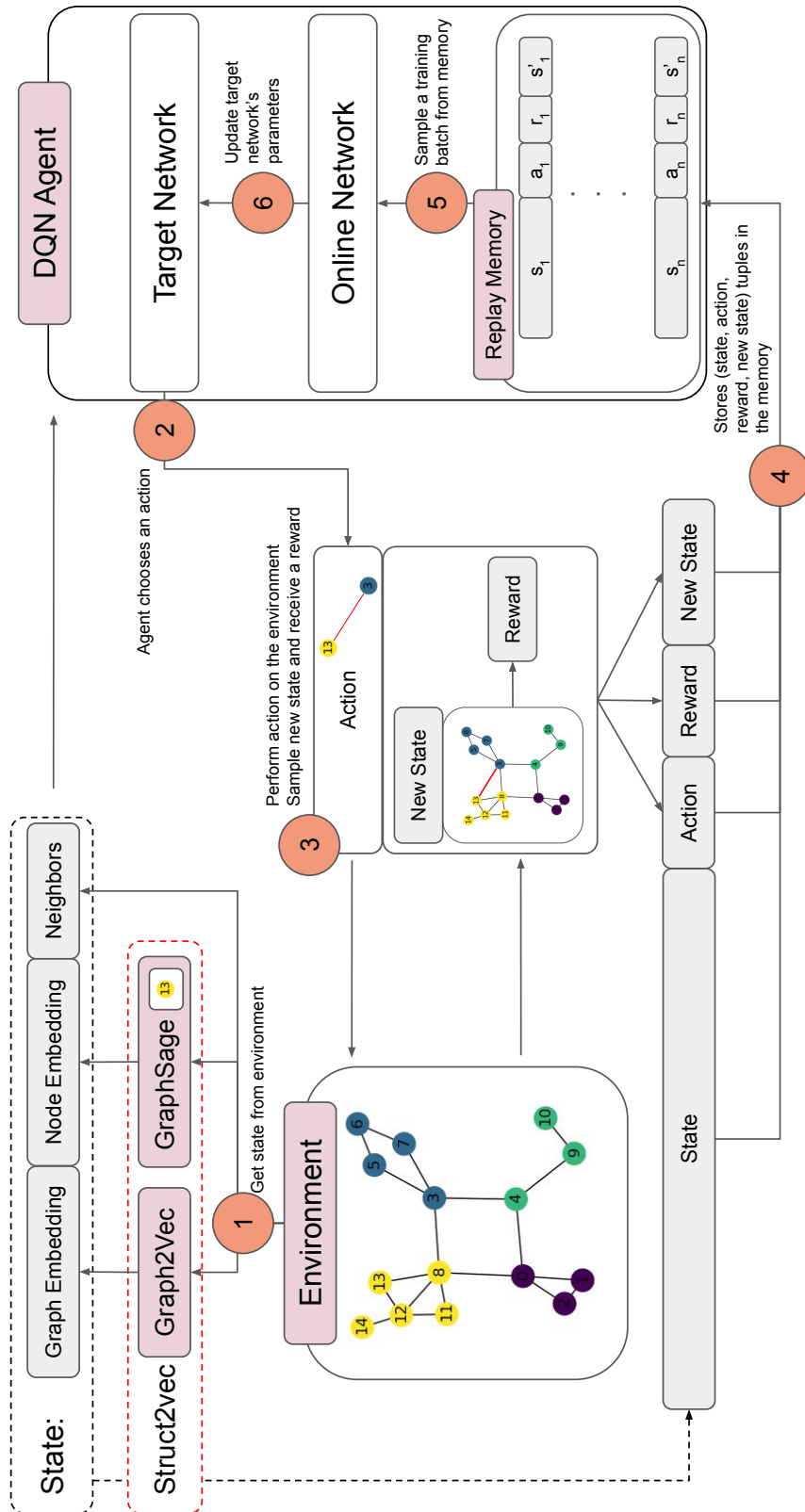


Figure 5.1 - S2V-DQN architecture adapted to our problem

5.4 . Implementation

Reward function details As explained previously, the reward function has to be precisely formulated to encourage wanted behaviors and deter negative actions. Our reward function is hence a linear combination of diversity and satisfaction :

$$\mathcal{R}(s_t, a_t) = \alpha S_G(u, a_t) + (1 - \alpha) \Delta \sigma (\mathcal{E} (u | \mathcal{P}_G))$$

We keep the same formulation of diversity as in Section 4, meaning that it is still the minimization of a diversity measure $\sigma (\mathcal{E} (u | \mathcal{P}_G))$. This is computed internally in the environment and the agent only receives the final value of the combination with the satisfaction value. This means that the agent does not have access to the graph partition \mathcal{P}_G . Satisfaction is based on a measure of similarity based on distance. We make the assumption that similarity is linked with closeness, e.g. content that is closer in the network is more likely to share more ideas than content that is further apart. Hence, content that is closer is more likely to be satisfying and as such adopted ; this is also in line with classic link recommendation algorithms.

We present our algorithm for the computation of the satisfaction score in Algorithm 5. Given a graph G a source node u , i.e. the target node of the MDP and the action node v the algorithm returns a distance score between 0 and 1. The algorithm is inspired by breadth-first search (BFS) and follows a similar structure. First we determine the target group to which the action nodes belong to. Then by iteratively multiplying the adjacency matrix with itself and multiplying the result with the target group vector we can know whether the source node can reach that group at a given depth. Once the depth has been identified at which the target group is reached we stop the process. The final return value is inspired by the sigmoid activation function, given by $f_\lambda(x) = \frac{1}{1+\exp^{-\lambda x}}$. This is because we want the return value to have a shape that follows a smooth logistic curve between 0 and 1. The final return value is given by $\frac{1}{1 + \exp \left((-1)^{(\text{value} > \sum T)} \cdot \frac{\text{value}}{\text{factor}} \right)}$.

The *value* variable is the count of each path that reaches the target group and the *factor* variable is the proportion between the adjacency to the power of *depth* and the base adjacency matrix. The factor variable represents how much connected the source node is at *depth*. As such the fraction $\frac{\text{value}}{\text{factor}}$ is a high value if the source node is well connected to the target group comparatively to the whole graph. Moreover, if the source node is connected enough then $\text{value} > 2 \sum T$ is true and $(-1)^{(\text{value} > \sum T)}$ is negative and $\frac{\text{value}}{\text{factor}}$ is a large value which makes the results closer to one. On the other hand if the source node is only sparsely connected to the target group then $\text{value} > 2 \sum T$ is false and $(-1)^{(\text{value} > \sum T)}$ is positive and $\frac{\text{value}}{\text{factor}}$ remains a positive value grea-

Algorithm 5: Satisfaction Score

Input: Graph G , source node vector u , action node vector v

Output: Distance score between target node and action node

```
1  $i \leftarrow \mathcal{P}_G(v)$ ;  
2  $T \leftarrow \mathcal{P}_{G_i}$ ;  
3  $A' \leftarrow A$ ;  
4  $reached \leftarrow \mathbf{false}$ ;  
5  $depth \leftarrow 0$ ;  
6 while  $\neg reached$  and  $depth < \max depth$  do  
7    $A' \leftarrow A' \cdot A$ ;  
8    $depth \leftarrow depth + 1$ ;  
9    $value \leftarrow \sum(T \cdot A' \cdot u)$ ;  
10  if  $value \geq \sum T$  then  
11  |  $reached \leftarrow \mathbf{true}$ ;  
12  end  
13 end  
14 if  $reached$  then  
15 |  $ratio \leftarrow \frac{\sum A'}{\sum A}$ ;  
16 | return  $\frac{1}{1 + \exp\left((-1)^{(value > 2 \sum T)} \cdot \frac{value}{ratio}\right)}$ ;  
17 end  
18 else  
19 | return 0;  
20 end
```

ter which makes the results closer to 0.

Transferability As mentioned in the previous section, specific graph representation techniques were chosen that enabled transferability between policies trained on one graphs to other graphs. In order to achieve that, we need to make node orders invariant. By default, in a given graph, a node id e.g., 1 is essentially a random node as the index 1 is just an identifier and does not say anything about the node. However, if we want our model to be able to transfer knowledge from one graph to another then there needs to be consistency in node labeling. Meaning that node 1 in one graphs is structurally the same as node 1 in another graph. This is achieved by re-labeling nodes according to a topological semantics. Intuitively, we want to use some centrality measures for the semantics; in our implementations, we use the descending sorting based on the total number of triangles each node is part

of. Using triangles is more expressive than simply using degree as it infers more information, especially local clustering information.

Lastly, to make our model transferable, we need it to have outputs that adapt to different graphs. Currently, the output of the model is a Q-value vector of expected reward values for each possible action; in other words, for each node in the graph. Meaning, that inherently, without any modifications, one could only transfer to graphs having the same number of nodes. To transfer the model from one graph to another we have to change the output layer. To do that, we freeze all the previous layers parameters and modify the output layer to have the correct output dimension then we execute a quick training step to learn the parameters of the output layer.

5.5 . Experiments

Training setup and implementation We use the standard approach for training learning-based models. First we split the data into training and testing sets using the standard ratio of 80/20 and we train the models over 100 episodes. Each episode iterates over all nodes in the training set and the models attempt to generate sequential edge recommendations up to a maximum number of nodes that is proportional to the size of the graph. The recommendations are then stored in an experience replay memory which records graph and node states (embeddings), the recommended edge and the reward returned by the environment. At each step, the replay memory is randomly sampled and the online network is used to predict Q-values which are then used to update the target network's parameters.

The experience replay memory is used to to break temporal correlations as well as improving data efficiency and better leverage good experiences. Furthermore we use the difference between predicted Q-values and actual values from the target network's parameters as an estimate for the usefulness of the stored transitions. If for a particular transition the predicted values are close to the actual ones then it shows that the model has properly learned how to act in this specific state. Consequently, such transitions provide little additional learning signal and should be sampled less often. Conversely, large differences between predicted and target Q-values suggest that the model has not yet fully learned from these experiences, and the associated transitions should therefore be sampled more frequently.

We implemented our environment, model and experiments in python. We used the `NetworkX` package for graph manipulation and `Gym` package for RL environment architecture to make our own custom RL environment. We took the `KarateClub` package's implementation of GraphSage and Graph2vec and pre-trained them on the same packages reddit dataset. Our RL agent's architecture is based on the `pytorch` package which enables CUDA optimization.

The code is publicly available.¹

Hyper parameters As is common with any deep learning framework there is a series of hyper parameters to fine-tune to your task. However, we did not go through a grid search to identify the best possible values and we chose commonly used values in the machine learning field. We use Adam optimizer with a learning rate of $1e - 4$ and the target network's update rate is 0.005 using a discount factor of 0.99. The replay memory stores 10000 transition tuples and is sampled by batches of 128 for training. We use an adaptive ϵ -greedy strategy for action selection which favors exploration at first and slowly converges toward exploitation depending on how well the agent performs. We use a graph embedding dimension of 128 and 16 for node embeddings.

Furthermore, there is a third component to the reward function which is the history penalty, i.e., the record of past actions and aiming for non-repetitiveness of link recommendations. Without this component, the agent would identify either a very satisfying action, an action that greatly improves information exposure or one that does both and keep repeating that specific action. This is undesirable behavior for the agent as the more of a specific content is recommended the less novel and attractive it becomes. To prevent such misbehavior, we add both a diminishing returns factor as well as a penalty depending on h , the count of a given action in the history $\mathcal{H}_t(u)$:

$$\mathcal{R}(s_t, a_t) = 0.7^{|1-h|} (\alpha S_G(u, a_t) + (1 - \alpha) \Delta \sigma (\mathcal{E}(u | \mathcal{P}_G))) - \left(\frac{|1-h|}{\text{len}(\mathcal{H}_t(u))} \right)$$

Using the absolute value $|1-h|$ makes it so that only repeated actions are discouraged and new actions are not punished. The exact value of 0.7 was arbitrarily chosen in testing. For example, if a content is recommended three times then $0.7^2 \approx 0.5$ meaning that the reward is cut in half or that the recommendation is about half as attractive as originally and there is also a penalty to take into account.

Datasets We evaluate our approach on several real graph datasets, described in Table 4.1.

Karate is a dataset representing a karate club, it is often used for baseline analysis. *Polbooks* is a collection of US political books and their references, the dataset also records each book's political alignment : liberal, neutral or conservative. This dataset features a clear separation between liberal and conservative books and is interesting to study when you use these alignments as

1. github.com/Jonathan-COLIN/

Table 5.1 – Dataset details

| Dataset | Nodes | Edges | Existing | #Comm. |
|-----------------------|-------|-------|----------|--------|
| <i>Karate</i> | 34 | 78 | ✗ | 4 |
| <i>Polbooks</i> | 105 | 441 | ✓ | 3 |
| <i>Football</i> [203] | 115 | 613 | ✓ | 12 |

ground-truth for communities. *Football* is a dataset of US college football representing the Division I games for the 2000 season [203]. It also provides information about conferences, groups of colleges that often play football against each other, which can be used for ground-truth communities. Furthermore, we use the greedy Louvain modularity community detection algorithm [113] for *Karate* as it does not provide ground-truth information for communities but we also use it for both *Polbooks* and *Football* to replicate evolving community structure dynamics.

5.5.1 . Base Settings

The main objective for our agent is to learn how to balance diversity and satisfaction in order to provide relevant recommendation that still improve information exposure. As such, it is important to identify the model’s ability to provide diverse and satisfactory recommendations independently. Hence, we train three agents (diversity, similarity and dual) on three environments that only differ in their reward but otherwise are the same, meaning that they have the same community structure and same exposure function.

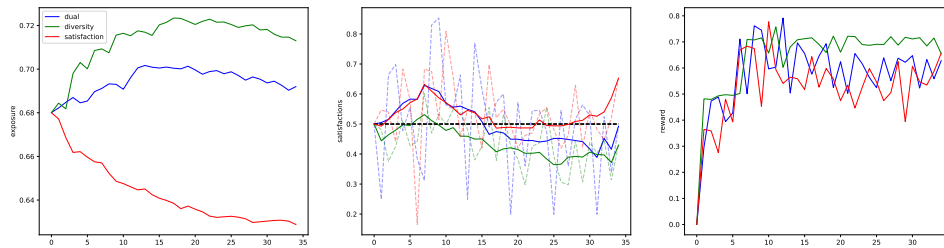


Figure 5.2 – Performance comparison between agents on the karate club dataset, Louvain modularity communities, FJ exposure

Results are presented in Fig. 5.2 and as expected models specialized in either diversity (drawn in green) or satisfaction (drawn in red) perform best in their respective metric. Interestingly, the diversity agent – despite only focusing on promoting diverse exposure – does so only at a minimal loss in satisfaction. Similarly, the dual agent (drawn in blue) manages to balance well diversity and satisfaction with only little performance loss in terms of diversity

and even staying comparable in satisfaction at first; however it starts losing some in the long term.

5.5.2 . Transfer Learning

Transfer learning is a machine learning approach where knowledge gained from training a model on one task or domain is reused to improve performance on a different but related task or domain. Instead of training a model from scratch, a pre-trained model is fine-tuned on a specific, task or domain, dataset. This approach reduces training time and lowers data requirements. In social networks, transfer learning can help adapt models trained on one platform or user population to another. In our setting, it helps transfer knowledge from one partition to another or one information propagation model (exposure function) to another. This captures shared behavioral and structural patterns.

In this section we will extensively study our model's ability to generalize its knowledge and how effective it is at transfer learning, both from one graph to another and from one partition/exposure function to another.

If the target graph has a different size than the one used for training then the model needs to be fine-tuned. This is done by freezing the model's parameters except the output layer as it needs to be adapted to the new graph. Indeed with the number of nodes changing, it also means that the number of possible actions also changes. On the other hand, the input layer does not need to be changed as embedding dimension do not change as explained in Section 5.4. Then once the new output layer is initialized, a quick training session is done with a smaller training set and number of episodes.

Synthetic training One possible application of such feature is to train the model on synthetic data. Indeed, training on fabricated data presents some advantages as the data itself can be generated in virtually unlimited quantities. At the same time, you can also generate training data that reduces training imbalances or biases which will help the models to have fairer recommendations. Furthermore, this process avoids privacy violations and reduces the need for personal data handling, which is crucial in sensitive domains such as healthcare or social networks. This is of interests as it is the main flaw of the FJ opinion dynamics model [155] as it assumes knowledge of private information to predict user opinions. Incidentally, it is also one of the shortcomings of our convex optimization framework presented in Section 4.

The synthetic graph generation is done using `NetworkX`'s configuration model [208] which allows us to replicate the degree distribution of the real-world dataset, which is known to be highly heterogeneous, with a small number of nodes exhibiting very high degree and the vast majority of nodes having low degree. The models are then trained as described previously and even-

tually tested on the real world graph the synthetic one is based on. Results are presented in Fig. 5.3 and we can see similar results to the base setting in Fig. 5.2 with the exception that the dual agent (drawn in blue) provides better information exposure improvement than the specialized agent (drawn in green). Again, only the similarity agent (drawn in red) manages to provide satisfactory recommendations in the long term where the other two only manage to do so at the beginning and then quickly start losing out on satisfaction.

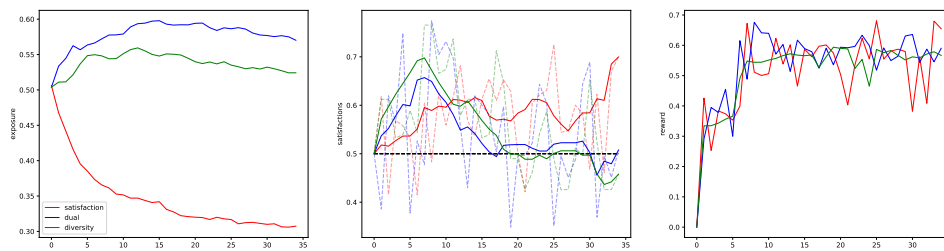


Figure 5.3 – Performance comparison between agents trained on synthetic version of the karate club dataset, louvain modularity communities, FJ exposure

Evolving partition and exposure function A key feature of social networks is how they evolve over time. As such it is important that the models are able to keep up with this evolution which in our setting is implemented by changing partitions or exposure function between training and testing.

Fig. 5.4 shows the results when changing partition while Fig. 5.5 shows the results after changing exposure function. Most notably, we can see that model trained for satisfaction (drawn in red) fails to generalize and performs poorly in both diversity and even satisfaction. Similarly, the dual agent also performs poorly in such settings, particularly in terms of satisfaction where it completely fails to provide anything relevant. In the context of changing exposure function, the dual agent also struggles to provide truly diverse recommendations which at first improve information exposure but end up reducing it. However, in spite of an evolving environment, the diversity agent still manages to provide diverse recommendations that improve information exposure. A main takeaway of this experiment is that diversity knowledge is easier to generalize than satisfaction in other words opinion clusters remain relatively similar through change. This means that if the focus is on diversity, inferred communities are a good substitute for actual communities.

Transfer learning over different graphs Real-world graphs, particularly in social networks, are too big to be processed in full memory. This also

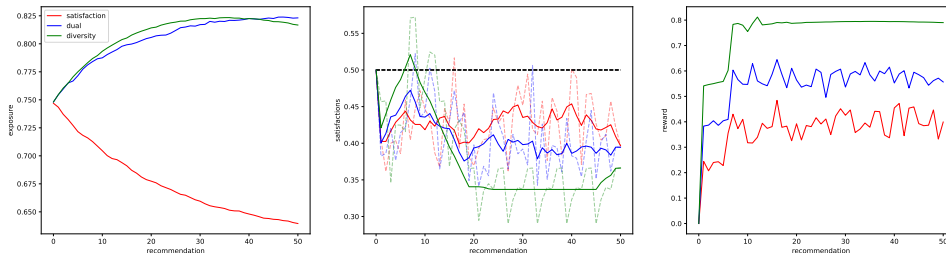


Figure 5.4 – Performance comparison between agents trained on the karate club dataset, louvain modularity communities, FJ exposure and evaluated on the football dataset, ground-truth communities, FJ exposure

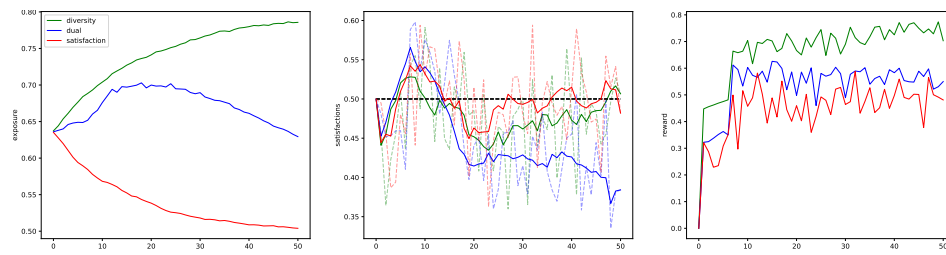


Figure 5.5 – Performance comparison between agents trained on the football dataset with louvain modularity communities and FJ exposure and evaluated on same dataset with same communities, BFS exposure

makes the computation of message passing, which is crucial to graph neural networks, expensive. This calls for a different strategy for training which in standard machine learning frameworks is done via graph sampling [209]. It is the process of selecting a subgraph from a larger graph while trying to preserve important structural or statistical properties of the original network. It mainly focuses on techniques based on graph topology such as neighborhood sampling because it helps message passing for Graph Neural Networks (GNN). However, this fails to capture diversity and the full span of information present in the network, meaning that standard graph sampling methods are not usable with our models. Another strategy, is transfer learning which focuses on the models' ability to generalize their knowledge. Training is done on a one dataset or environment and then used in another. This is useful as you can train on a small representative dataset and then transfer the learned knowledge to a large network or you can train on dataset with known information about your objective and then transfer it to another where it is unknown.

Fig. 5.7 presents the results of the three agents evaluated on the Football dataset when they were trained on the Karate dataset which is three times smaller. We can see the same patterns as in Fig. 5.6 which presents the baseline results of the three agents trained directly on the football dataset. All the

results are comparable to those of the previous experiments with the only difference being that the diversity and the dual agents perform the same in terms of exposure.

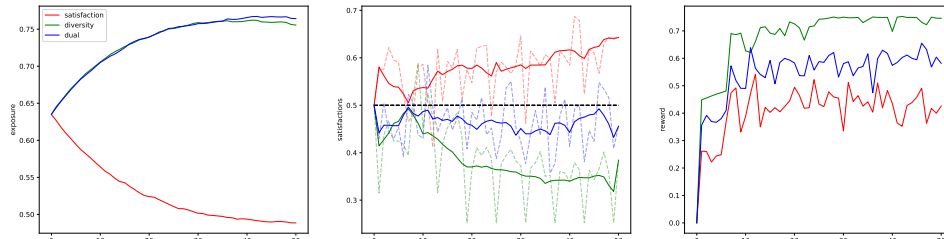


Figure 5.6 – Performance comparison between agents trained on the football dataset, ground-truth communities, FJ exposure

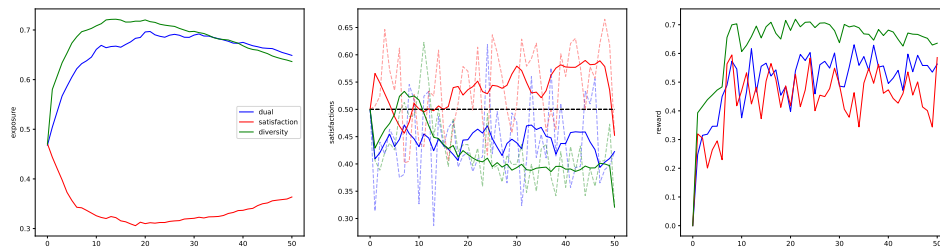


Figure 5.7 – Performance comparison between agents trained on the karate club dataset, Louvain modularity communities, FJ exposure and evaluated on the football dataset, ground-truth communities, FJ exposure

5.5.3 . Satisfaction as condition for edge adoption

Another element that needs to be accounted for in social networks is that users may or may not actually approve of a recommendation. This is crucial, as this means that some links may be refused and should not be added to the graph. Indeed, most polarization minimization and structural intervention algorithms completely disregard that fact. In these settings where opinions are one dimensional, disagreement (the distance between two opinions) may be interpreted as satisfaction but in our environment where opinions are multidimensional, this no longer is true. Indeed, if we consider two dimensional opinion vectors with a user with neutral opinions $(0.5, 0.5)$ and suggest two pieces of content with opinion vectors $(1, 0.5)$ and $(0.75, 0.25)$, in spite of both having an absolute distance of 0.5, the user might not agree with the first content's opinion value of 1 as it might be too extreme for them.

Hence, we need the models to account for the users' reaction to provided recommendations. To do so, we make edge addition depend on the satisfac-

tion of each recommendation. As explained in Section 5.4, our measure of satisfaction is on the $[0, 1]$ range with the middle value 0.5 being a neutral reaction, values over this are positive reactions and values under are negative reactions. Thus, we use a satisfaction value of 0.5 as a threshold for edge adoption. In case of refusal, there is a penalty for providing an unsatisfactory recommendation and there are two possible scenarios to consider, either the recommendation is completely rejected and there is no information crossover meaning that information exposure is not affected or the recommendation is partially rejected and there is some information crossover which leads to exposure still being affected but there is no edge adoption.

Initially, as discussed in Section 4.7 when optimizing for information exposure there is an optimal situation that may be missed or overshoot. As such, we provided our model with an exit action that would stop the recommendation process if the agent considered that he could no longer improve information exposure or no longer provide satisfactory recommendations. If the model chose to stop the recommendation process the corresponding reward would depend on the final exposure if diversity was accounted for or if only considering similarity, the final reward would depend on the average satisfaction. However, in our current context where edge adoption depends on satisfaction and there is no information crossover, all three agents would consistently converge on the same decision to stop the recommendation process quickly. For the two models that considered diversity, they did so because they could not provide satisfactory recommendations and stopping immediately prevents the accumulation of corresponding penalties. The satisfaction agent would also stop quickly because average satisfaction was high.

For the scenario with information crossover and with the option to stop recommendations disabled, the results shown in Fig. 5.8 show that even if information exposure is marginally improved by both, the dual model outperforms the specialized diversity model in spite of providing recommendations with similar satisfaction values. On the other hand, when no information crossover happens, we can observe a very unstable information exposure evolution as is shown in Fig. 5.9. In this setting, the dual agent seems to perform particularly well as not only it achieves the same results as the diversity agent but it also stays above the 0.5 satisfaction threshold throughout the entire recommendation process which it struggled to do in most of the previous experiments.

5.5.4 . Recommendation with unfollowing

An important aspect of social networks is the constant stream of new content being produced at all times. Combined with the limited time that people may spend on each given platform there are exist also the dynamics of unfollowing. This means that users' attention cannot be infinitely divided,

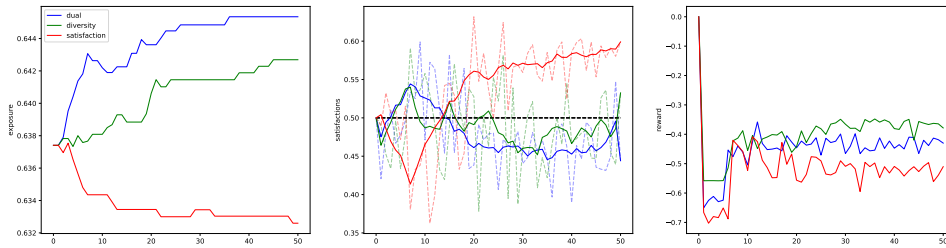


Figure 5.8 – Performance comparison between agents trained on the football dataset, Louvain modularity communities, FJ exposure with edge adoption depending on satisfaction and information crossover

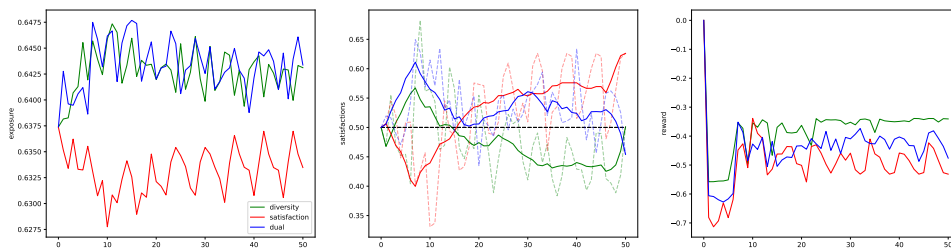


Figure 5.9 – Performance comparison between agents trained on the football dataset, Louvain modularity communities, FJ exposure with edge adoption depending on satisfaction and no information crossover

in other words, it is unrealistic to only add edges. As time progresses, trends evolve or opinions mature one's interests and links adapt accordingly meaning that some links are foregone in favor of new ones.

This can be implemented in various ways as there are two factors to consider : *when* does the edge get removed, and *which* edge is removed. There are several possible strategies to decide when and who to unfollow. There are many ways to decide when an edge gets removed and we chose three that we deemed representative of actual behaviors in social networks :

- **Static unfollowing** The node degree does not change which means that at each edge addition, an edge is also removed at the same time. If the edge adoption depends on satisfaction then the the removal is also dependent on it.
- **Ceiling unfollowing** There is a maximum degree that the node cannot exceed which means that at first there are no edge removals but once the maximum is reached then at each edge addition, an edge is also removed.
- **Satisfaction unfollowing** We capture attention span mechanism by also considering the satisfaction value as the probability of "forgetting" a past link (an edge).

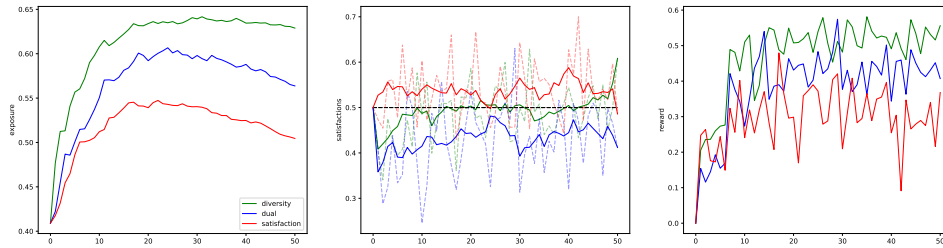


Figure 5.10 – Performance comparison between agents trained on the polbooks dataset, ground-truth communities, BFS exposure with static unfollowing of old links

Similarly, there are many ways to decide which edge needs to be removed and we also chose three relevant selection semantics :

- **Random selection** A neighbor is randomly chosen. This can be done in a truly random approach or with a biased random depending on satisfaction or neighbor degree.
- **Oldness selection** The oldest link with the target user is chosen.
- **Strength selection** The neighbor with the lowest satisfaction score is chosen.

In Figures 5.10, 5.11, 5.12 we present the results of the three agents on the polbooks dataset in the context of static unfollowing with oldness selection, ceiling unfollowing with random selection and satisfaction unfollowing with strength selection respectively. While the performances of all agents are similar to those of previous experiments, i.e. the specialized agents are the best with regards to their respective metric, there are two notable differences. First the dual agent (blue) appears to perform worse than the diversity agent when considering satisfaction despite performing better in the context of ceiling unfollowing with random selection. Secondly, the satisfaction agent appears to provide some exposure improvements before decreasing it in all three settings.

When considering ceiling unfollowing, the maximum is set as the minimum between a fixed value, in this case 10, and 1.5 times the user's degree at the start of the experiment. This means that the ceiling adaptively corresponds to a node's initial situation. For instance a user with an initial low degree of 5 will not suddenly grow up to a degree of 50, a ceiling of 10 means that he already doubles his initial degree and has to divide their attention correspondingly. This setting thus becomes unrealistic if the ceiling is set too high. On the other hand, if the ceiling is too low then it becomes unfair to nodes with a base degree higher than that ceiling, hence us using a second value that is proportional to the initial degree. Such adaptive conditions are important

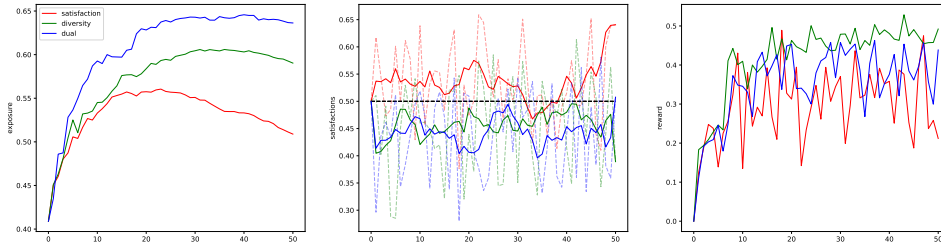


Figure 5.11 – Performance comparison between agents trained on the polbooks dataset, ground-truth communities, BFS exposure with ceiling unfollowing of random links

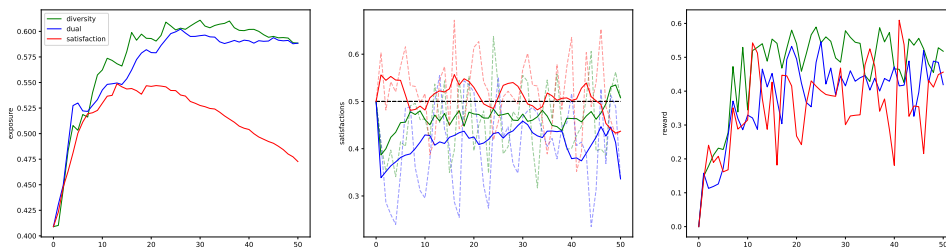


Figure 5.12 – Performance comparison between agents trained on the polbooks dataset, ground-truth communities, BFS exposure with satisfaction unfollowing of weak links

in the context of social networks with preferential attachment dynamics, resulting in few nodes with very high degree and a vast majority with very low degrees [115].

5.5.5 . Conclusion

In this chapter, we extended the information exposure maximization problem to take user satisfaction into account and presented a deep reinforcement learning (DRL) approach to solve it using the S2V DQN architecture. This approach has been built on the foundation of our convex optimization from Section 4 and addressed its shortcomings. In particular, the DRL framework can incorporate user feedback in its training loop in the form of satisfaction which the convex model could not. Furthermore, the DRL framework does not explicitly need to know the partition of the graph in its learning process. As a reminder, the graph partition is a representation of user opinions which, in a realistic setting, is not public information and cannot be used as is. We extensively tested our agent’s ability to provide diverse recommendations while respecting user satisfaction constraints. Most notably, we studied our model under the lens of transfer learning and showed that it can provide relevant recommendations despite changing partitions or exposure semantics. These experiments also showed that the model can be trained on a synthetic or a

small graph and then be used on a real or larger graph.

In spite of this extensive experimental study, a more thorough investigation of the model is still necessary. Similarly to the convex optimization algorithm, our DRL framework should be extended to be able to utilize other information diffusion pathways such as a content graph. Current experiments only study small graphs but large networks with more varied social dynamics should also be considered. Furthermore, a case study on temporal graphs is also needed in order to examine how the model deals with the cold-start problem as well as a network that evolves more dynamically.

6 - Conclusion and Perspectives

6.1 . Summary of Findings

The study of diversity in social media recommendations constitutes a multidisciplinary research domain that brings together a wide range of fields. It is the intersection of various areas such as recommender systems, fairness, polarization and serendipity, while also drawing from social science perspectives on group behaviors and opinion dynamics as well as network science with information diffusion and influence maximization. Moreover, it extends into theoretical domains like information theory through the lens of entropy.

In this thesis, we extended prior work on polarization and disagreement minimization and opinion dynamics to construct a flexible framework. This setting is capable of modeling a broad spectrum of opinions across communities, whether discrete or overlapping, and uses a generic definition that accommodates for different exposure functions and diversity measures. Building on this foundation, we formalized the problem of diverse information exposure which provides a rigorous framework for studying exposure diversity. Within this context, we defined the task of optimizing diverse information exposure in social networks and introduced a gradient descent algorithm that takes advantage of the convex properties of our problem as well as providing greedy heuristics as baselines.

We extend this framework to take user-item graphs into account, allowing for more realistic and adaptable representations of social media dynamics. Most notably, this provides a structure to study the separate information propagation channels. The experimental results confirmed the relevance of these approaches, highlighting their capacity to improve exposure diversity across different network structures and conditions.

We then study reinforcement learning techniques to answer the few shortcomings of our previous approach, notably the weak user-interests input and the necessity of using private information which is an inherited trait of prior work on polarization and disagreement minimization. We employed a popular deep reinforcement learning method boosted by graph representation techniques in the form of Structure to Vec Deep Q-Networks (S2v DQN). We conducted an extensive experimental evaluation of our model and showed that it is capable of effectively transferring knowledge from one network to another, even when the source network is synthetically generated. In addition, the model maintains strong performance in more dynamic environments that incorporate social group behaviors, such as unfollowing, as well as dynamic structural patterns resulting from evolving network partitions and exposures functions. Our Reinforcement learning architecture not only offered a flexible

framework for optimization but also enabled transfer learning scenarios, highlighting its promise for long-term applications in real-world systems.

In conclusion, the contributions of this work lie in their integration of theoretical rigor and practical applicability. By combining algorithmic optimization with adaptive, learning-based strategies, it offers new perspectives on how social networks can be designed to foster diversity of exposures as well as mitigating polarization and promoting fairness. While these advances in computational methods represent a significant step forward, they must be understood alongside the recognition that technical solutions alone are insufficient. Education probably remains one of the most effective means to cultivate awareness of echo chambers, confirmation biases, and the manipulation of information. Encouraging individuals to actively seek out new and different perspectives is as crucial as developing algorithms that promote them.

6.2 . Perspectives and Future Work

Future research can build on this work by exploring several promising directions. First, the development of new opinion dynamics models is necessary to better capture the complexity of current social interactions, which are shaped not only by direct exchanges but also by subtle mechanisms such as algorithmic curation, influencers, and ephemeral content. Indeed, traditional opinion dynamics models only consider a single channel of information propagation whereas social networks now combine many different pathways such as video content (short and long form), text posts, reacts or group pages. Each of these pathways have different mechanisms and lasting impact on users that use them.

Secondly, extending algorithms to account for these multiple information propagation pathways, spanning across different social media platforms, would offer a more realistic and comprehensive understanding of how ideas spread and interact in today's highly connected digital ecosystem. This would enable a better understanding on the latent mechanisms that lead to polarization and social fragmentation in turn providing better tools to combat these issues. At the same time, this would also help prevent the formation of echo chamber and filter bubbles.

Furthermore, future work could explore strategies based on targeted edge removal by identifying noisy or parasitic links that convey little novel information while disproportionately reinforcing existing exposure patterns. Although such links often attract minimal direct attention, they can cumulatively contribute to information redundancy by subtly influencing information flow and reshaping the global exposure landscape. Carefully pruning these connections may therefore help mitigate reinforcement effects such as echo chambers and promote more diverse information diffusion both at the local

and global level.

Finally, further investigation into dual-optimization frameworks that balance diversity with user satisfaction could help balance out the opposing objectives of exposing individuals to novel heterogeneous opinions and maintaining user engagement with personalized experiences. Together, these directions would lead to attractive serendipitous recommendations. Combining current recommender systems powered by predictive accuracy-driven models with socially responsible semantics would make the recommendations of tomorrow relevant, satisfying and diverse while ensuring fairness towards all groups and users as well as content providers.

Bibliographie

- [1] D. Bawden, C. Holtham, and N. Courtney, "Perspectives on information overload," in *Aslib proceedings*, 1999.
- [2] D. Bawden and L. Robinson, *Information overload : An overview*. Oxford University Press, 2020.
- [3] K. H. Jamieson and J. N. Cappella, *Echo chamber : Rush Limbaugh and the conservative media establishment*. Oxford University Press, 2008.
- [4] F. P. Santos, Y. Lelkes, and S. A. Levin, "Link recommendation algorithms and dynamics of polarization in online social networks," *Proceedings of the National Academy of Sciences*, 2021.
- [5] M. Hohmann, K. Devriendt, and M. Coscia, "Quantifying ideological polarization on a network using generalized euclidean distance," *Science Advances*, 2023.
- [6] K. Sasahara, W. Chen, H. Peng, G. L. Ciampaglia, A. Flammini, and F. Menczer, "Social influence and unfollowing accelerate the emergence of echo chambers," *Journal of Computational Social Science*, 2021.
- [7] A. Ross Arguedas, C. Robertson, R. Fletcher, and R. Nielsen, *Echo chambers, filter bubbles, and polarisation : A literature review*. Reuters Institute for the Study of Journalism, 2022.
- [8] Y. Lelkes, "Mass polarization : Manifestations and measurements," *Public Opinion Quarterly*, 2016.
- [9] B. Fedoruk, H. Nelson, R. Frost, and K. Fucile Ladouceur, "The plebeian algorithm : a democratic approach to censorship and moderation," *JMIR Formative Research*, 2021.
- [10] A. Matakos, E. Terzi, and P. Tsaparas, "Measuring and moderating opinion polarization in social networks," *Data Mining and Knowledge Discovery*, 2017.
- [11] P. Covington, J. Adams, and E. Sargin, "Deep neural networks for youtube recommendations," in *Proceedings of the 10th ACM conference on recommender systems*, 2016.
- [12] C. A. Gomez-Uribe and N. Hunt, "The netflix recommender system : Algorithms, business value, and innovation," *ACM Transactions on Management Information Systems (TMIS)*, 2015.
- [13] A. Töscher, M. Jahrer, and R. Legenstein, "Improved neighborhood-based algorithms for large-scale recommender systems," in *Proceedings of the 2nd KDD Workshop on Large-Scale Recommender Systems and the Netflix Prize Competition*, 2008.

- [14] A. N. Nikolakopoulos, X. Ning, C. Desrosiers, and G. Karypis, "Trust your neighbors : A comprehensive survey of neighborhood-based methods for recommender systems," *Recommender systems handbook*, 2021.
- [15] J. L. Herlocker, J. A. Konstan, A. Borchers, and J. Riedl, "An algorithmic framework for performing collaborative filtering," in *Proceedings of the 22nd annual international ACM SIGIR conference on Research and development in information retrieval*, 1999.
- [16] M. J. Pazzani and D. Billsus, "Content-based recommendation systems," in *The adaptive web : methods and strategies of web personalization*. Springer, 2007.
- [17] P. Lops, M. De Gemmis, and G. Semeraro, "Content-based recommender systems : State of the art and trends," *Recommender systems handbook*, 2010.
- [18] R. Burke, "Hybrid recommender systems : Survey and experiments," *User modeling and user-adapted interaction*, 2002.
- [19] P. Melville, R. J. Mooney, R. Nagarajan *et al.*, "Content-boosted collaborative filtering for improved recommendations," *Aaai/iaai*, 2002.
- [20] X. Su and T. M. Khoshgoftaar, "A survey of collaborative filtering techniques," *Advances in artificial intelligence*, 2009.
- [21] Y. Koren, R. Bell, and C. Volinsky, "Matrix factorization techniques for recommender systems," *Computer*, 2009.
- [22] D. Liang, J. Altosaar, L. Charlin, and D. M. Blei, "Factorization meets the item embedding : Regularizing matrix factorization with item co-occurrence," in *Proceedings of the 10th ACM conference on recommender systems*, 2016.
- [23] H.-F. Yu, C.-J. Hsieh, S. Si, and I. Dhillon, "Scalable coordinate descent approaches to parallel matrix factorization for recommender systems," in *2012 IEEE 12th international conference on data mining*, 2012.
- [24] H.-F. Yu, C.-J. Hsieh, S. Si, and I. S. Dhillon, "Parallel matrix factorization for recommender systems," *Knowledge and Information Systems*, 2014.
- [25] R. Mehta and K. Rana, "A review on matrix factorization techniques in recommender systems," in *2017 2nd International Conference on Communication Systems, Computing and IT Applications (CSCITA)*, 2017.
- [26] J. Gope and S. K. Jain, "A survey on solving cold start problem in recommender systems," in *2017 International Conference on Computing, Communication and Automation (ICCCA)*, 2017.
- [27] C. Basu, H. Hirsh, W. Cohen *et al.*, "Recommendation as classification : Using social and content-based information in recommendation," in *Aaai/iaai*, 1998.

- [28] X. He, L. Liao, H. Zhang, L. Nie, X. Hu, and T.-S. Chua, "Neural collaborative filtering," in *Proceedings of the 26th international conference on world wide web*, 2017.
- [29] S. Sedhain, A. K. Menon, S. Sanner, and L. Xie, "Autorec : Autoencoders meet collaborative filtering," in *Proceedings of the 24th international conference on World Wide Web*, 2015.
- [30] D. Liang, R. G. Krishnan, M. D. Hoffman, and T. Jebara, "Variational autoencoders for collaborative filtering," in *Proceedings of the 2018 world wide web conference*, 2018.
- [31] L. H. Ungar and D. P. Foster, "Clustering methods for collaborative filtering," in *AAAI workshop on recommendation systems*, 1998.
- [32] G.-R. Xue, C. Lin, Q. Yang, W. Xi, H.-J. Zeng, Y. Yu, and Z. Chen, "Scalable collaborative filtering using cluster-based smoothing," in *Proceedings of the 28th annual international ACM SIGIR conference on Research and development in information retrieval*, 2005.
- [33] D. H. Park, H. K. Kim, I. Y. Choi, and J. K. Kim, "A literature review and classification of recommender systems research," *Expert systems with applications*, 2012.
- [34] I. Cisco Systems, "The zettabyte era-trends and analysis." Cisco Systems, Tech. Rep., 2013.
- [35] R. Patel, P. Thakkar, and V. Ukani, "Cnnrec : Convolutional neural network based recommender systems-a survey," *Engineering Applications of Artificial Intelligence*, 2024.
- [36] Y. Deldjoo, T. Di Noia, D. Malitesta, and F. A. Merra, "A study on the relative importance of convolutional neural networks in visually-aware recommender systems," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2021.
- [37] Y.-J. Ko, L. Maystre, and M. Grossglauser, "Collaborative recurrent neural networks for dynamic recommender systems," in *Asian conference on machine learning*, 2016.
- [38] D. Z. Liu and G. Singh, "A recurrent neural network based recommendation system," in *International conference on recent trends in engineering, science & technology*, 2016.
- [39] A. A. Gunawan, D. Suhartono *et al.*, "Music recommender system based on genre using convolutional recurrent neural networks," *Procedia Computer Science*, 2019.
- [40] S. Wu, F. Sun, W. Zhang, X. Xie, and B. Cui, "Graph neural networks in recommender systems : a survey," *ACM Computing Surveys*, 2022.

- [41] C. Sun, C. Li, X. Lin, T. Zheng, F. Meng, X. Rui, and Z. Wang, "Attention-based graph neural networks : a survey," *Artificial intelligence review*, 2023.
- [42] P. Veličković, G. Cucurull, A. Casanova, A. Romero, P. Lio, and Y. Bengio, "Graph attention networks," *arXiv preprint arXiv :1710.10903*, 2017.
- [43] R. Ying, R. He, K. Chen, P. Eksombatchai, W. L. Hamilton, and J. Leskovec, "Graph convolutional neural networks for web-scale recommender systems," in *Proceedings of the 24th ACM SIGKDD international conference on knowledge discovery & data mining*, 2018.
- [44] V. Anand and A. K. Maurya, "A survey on recommender systems using graph neural network," *ACM Transactions on Information Systems*, 2025.
- [45] K. Sharma, Y.-C. Lee, S. Nambi, A. Salian, S. Shah, S.-W. Kim, and S. Kumar, "A survey of graph neural networks for social recommender systems," *ACM Computing Surveys*, 2024.
- [46] L. P. Kaelbling, M. L. Littman, and A. W. Moore, "Reinforcement learning : A survey," *Journal of artificial intelligence research*, 1996.
- [47] M. M. Afsar, T. Crump, and B. Far, "Reinforcement learning based recommender systems : A survey," *ACM Computing Surveys*, 2022.
- [48] M. L. Puterman, "Markov decision processes," *Handbooks in operations research and management science*, 1990.
- [49] A. Mahajan and D. Teneketzis, "Multi-armed bandit problems," in *Foundations and applications of sensor management*. Springer, 2008.
- [50] N. Silva, H. Werneck, T. Silva, A. C. Pereira, and L. Rocha, "Multi-armed bandits in recommendation systems : A survey of the state-of-the-art and future directions," *Expert Systems with Applications*, 2022.
- [51] T. Lu, D. Pál, and M. Pál, "Contextual multi-armed bandits," in *Proceedings of the Thirteenth international conference on Artificial Intelligence and Statistics*, 2010.
- [52] L. Zhou, "A survey on contextual multi-armed bandits," *arXiv preprint arXiv :1508.03326*, 2015.
- [53] C. J. Watkins and P. Dayan, "Q-learning," *Machine learning*, 1992.
- [54] H. Hasselt, "Double q-learning," *Advances in neural information processing systems*, 2010.
- [55] G. A. Rummery and M. Niranjan, *On-line Q-learning using connectionist systems*. University of Cambridge, Department of Engineering Cambridge, UK, 1994.
- [56] A. Barto and M. Duff, "Monte carlo matrix inversion and reinforcement learning," *Advances in neural information processing systems*, 1993.

- [57] Y. Wang, K. S. Won, D. Hsu, and W. S. Lee, "Monte carlo bayesian reinforcement learning," *arXiv preprint arXiv :1206.6449*, 2012.
- [58] T. Vodopivec, S. Samothrakis, and B. Ster, "On monte carlo tree search and reinforcement learning," *Journal of Artificial Intelligence Research*, 2017.
- [59] A. G. Barto, "Reinforcement learning : An introduction. by richard's Sutton," *SIAM Rev*, 2021.
- [60] M. Agarwal and V. Aggarwal, "Reinforcement learning for joint optimization of multiple rewards," *Journal of Machine Learning Research*, 2023.
- [61] C. F. Hayes, R. Rădulescu, E. Bargiacchi, J. Källström, M. Macfarlane, M. Reymond, T. Verstraeten, L. M. Zintgraf, R. Dazeley, F. Heintz *et al.*, "A practical guide to multi-objective reinforcement learning and planning," *arXiv preprint arXiv :2103.09568*, 2021.
- [62] R. Yang, X. Sun, and K. Narasimhan, "A generalized algorithm for multi-objective reinforcement learning and policy adaptation," *Advances in neural information processing systems*, 2019.
- [63] X. N. Lam, T. Vu, T. D. Le, and A. D. Duong, "Addressing cold-start problem in recommendation systems," in *Proceedings of the 2nd international conference on Ubiquitous information management and communication*, 2008.
- [64] B. Lika, K. Kolomvatsos, and S. Hadjiefthymiades, "Facing the cold start problem in recommender systems," *Expert systems with applications*, 2014.
- [65] J. X. Wang, Z. Kurth-Nelson, D. Tirumala, H. Soyer, J. Z. Leibo, R. Munos, C. Blundell, D. Kumaran, and M. Botvinick, "Learning to reinforcement learn," *arXiv preprint arXiv :1611.05763*, 2016.
- [66] X. Chen, L. Yao, J. McAuley, G. Zhou, and X. Wang, "Deep reinforcement learning in recommender systems : A survey and new perspectives," *Knowledge-Based Systems*, 2023.
- [67] V. Mnih, K. Kavukcuoglu, D. Silver, A. A. Rusu, J. Veness, M. G. Bellemare, A. Graves, M. Riedmiller, A. K. Fidjeland, G. Ostrovski *et al.*, "Human-level control through deep reinforcement learning," *nature*, 2015.
- [68] V. Mnih, K. Kavukcuoglu, D. Silver, A. Graves, I. Antonoglou, D. Wierstra, and M. Riedmiller, "Playing atari with deep reinforcement learning," *arXiv preprint arXiv :1312.5602*, 2013.
- [69] H. Van Hasselt, A. Guez, and D. Silver, "Deep reinforcement learning with double q-learning," in *Proceedings of the AAAI conference on artificial intelligence*, 2016.

- [70] Z. Wang, T. Schaul, M. Hessel, H. Hasselt, M. Lanctot, and N. Freitas, "Dueling network architectures for deep reinforcement learning," in *International conference on machine learning*, 2016.
- [71] V. Konda and J. Tsitsiklis, "Actor-critic algorithms," *Advances in neural information processing systems*, 1999.
- [72] T. P. Lillicrap, J. J. Hunt, A. Pritzel, N. Heess, T. Erez, Y. Tassa, D. Silver, and D. Wierstra, "Continuous control with deep reinforcement learning," *arXiv preprint arXiv :1509.02971*, 2015.
- [73] J. Gilmer, S. S. Schoenholz, P. F. Riley, O. Vinyals, and G. E. Dahl, "Neural message passing for quantum chemistry," in *International conference on machine learning*, 2017.
- [74] B. Yu, H. Yin, and Z. Zhu, "Spatio-temporal graph convolutional networks : A deep learning framework for traffic forecasting," *arXiv preprint arXiv :1709.04875*, 2017.
- [75] T. Mikolov, I. Sutskever, K. Chen, G. S. Corrado, and J. Dean, "Distributed representations of words and phrases and their compositionality," *Advances in neural information processing systems*, 2013.
- [76] T. Mikolov, K. Chen, G. Corrado, and J. Dean, "Efficient estimation of word representations in vector space," *arXiv preprint arXiv :1301.3781*, 2013.
- [77] B. Perozzi, R. Al-Rfou, and S. Skiena, "Deepwalk : Online learning of social representations," in *Proceedings of the 20th ACM SIGKDD international conference on Knowledge discovery and data mining*, 2014.
- [78] A. Grover and J. Leskovec, "node2vec : Scalable feature learning for networks," in *Proceedings of the 22nd ACM SIGKDD international conference on Knowledge discovery and data mining*, 2016.
- [79] W. Hamilton, Z. Ying, and J. Leskovec, "Inductive representation learning on large graphs," *Advances in neural information processing systems*, 2017.
- [80] Q. Le and T. Mikolov, "Distributed representations of sentences and documents," in *International conference on machine learning*, 2014.
- [81] A. Narayanan, M. Chandramohan, R. Venkatesan, L. Chen, Y. Liu, and S. Jaiswal, "graph2vec : Learning distributed representations of graphs. arxiv 2017," *arXiv preprint arXiv :1707.05005*, 2017.
- [82] T. Kipf, "Semi-supervised classification with graph convolutional networks," *arXiv preprint arXiv :1609.02907*, 2016.
- [83] K. Xu, W. Hu, J. Leskovec, and S. Jegelka, "How powerful are graph neural networks?" *arXiv preprint arXiv :1810.00826*, 2018.
- [84] D. Kempe, J. Kleinberg, and É. Tardos, "Maximizing the spread of influence through a social network," in *Proceedings of the ninth ACM SIGKDD international conference on Knowledge discovery and data mining*, 2003.

- [85] M. Jaouadi and L. B. Romdhane, "A survey on influence maximization models," *Expert Systems with Applications*, 2024.
- [86] A. Zareie and R. Sakellariou, "Influence maximization in social networks : a survey of behaviour-aware methods," *Social Network Analysis and Mining*, 2023.
- [87] S. Chen, J. Fan, G. Li, J. Feng, K.-l. Tan, and J. Tang, "Online topic-aware influence maximization," *Proceedings of the VLDB Endowment*, 2015.
- [88] S. Bharathi, D. Kempe, and M. Salek, "Competitive influence maximization in social networks," in *International workshop on web and internet economics*, 2007.
- [89] B.-L. Chen, W.-X. Jiang, Y.-X. Chen, L. Chen, R.-J. Wang, S. Han, J.-H. Lin, and Y.-C. Zhang, "Influence blocking maximization on networks : Models, methods and applications," *Physics Reports*, 2022.
- [90] M. Granovetter, "Threshold models of collective behavior," *American journal of sociology*, 1978.
- [91] M. Gomez-Rodriguez, J. Leskovec, and A. Krause, "Inferring networks of diffusion and influence," *ACM Transactions on Knowledge Discovery from Data (TKDD)*, 2012.
- [92] N. Barbieri, F. Bonchi, and G. Manco, "Topic-aware social influence propagation models," *Knowledge and information systems*, 2013.
- [93] C. Z. Mooney, *Monte carlo simulation*. Sage, 1997.
- [94] G. L. Nemhauser, L. A. Wolsey, and M. L. Fisher, "An analysis of approximations for maximizing submodular set functions—i," *Mathematical programming*, 1978.
- [95] S. Brin and L. Page, "The anatomy of a large-scale hypertextual web search engine," *Computer networks and ISDN systems*, 1998.
- [96] L. Page, S. Brin, R. Motwani, and T. Winograd, "The pagerank citation ranking : Bringing order to the web." Stanford infolab, Tech. Rep., 1999.
- [97] Y. Li, H. Gao, Y. Gao, J. Guo, and W. Wu, "A survey on influence maximization : From an ml-based combinatorial optimization," *ACM Transactions on Knowledge Discovery from Data*, 2023.
- [98] E. Khalil, H. Dai, Y. Zhang, B. Dilkina, and L. Song, "Learning combinatorial optimization algorithms over graphs," *Advances in neural information processing systems*, 2017.
- [99] H. Li, M. Xu, S. S. Bhowmick, J. S. Rayhan, C. Sun, and J. Cui, "Piano : Influence maximization meets deep reinforcement learning," *IEEE Transactions on Computational Social Systems*, 2022.

- [100] T. Halal, B. Cautis, B. Groz, and R. Gao, "Topic-aware influence maximization with deep reinforcement learning and graph attention networks," *Data Mining and Knowledge Discovery*, 2025.
- [101] S. Tian, S. Mo, L. Wang, and Z. Peng, "Deep reinforcement learning-based approach to tackle topic-aware influence maximization," *Data Science and Engineering*, 2020.
- [102] T. T. Hills, "The dark side of information proliferation," *Perspectives on Psychological Science*, 2019.
- [103] A. K. Ghoshal, N. Das, and S. Das, "Influence of community structure on misinformation containment in online social networks," *Knowledge-Based Systems*, 2021.
- [104] M. Cinelli, W. Quattrociocchi, A. Galeazzi, C. M. Valensise, E. Brugnoli, A. L. Schmidt, P. Zola, F. Zollo, and A. Scala, "The covid-19 social media infodemic," *Scientific reports*, 2020.
- [105] A. Efstratiou, J. Blackburn, T. Caulfield, G. Stringhini, S. Zannettou, and E. De Cristofaro, "Non-polar opposites : analyzing the relationship between echo chambers and hostile intergroup interactions on reddit," in *Proceedings of the International AAAI Conference on Web and Social Media*, 2023.
- [106] J. Möller, "Filter bubbles and digital echo chambers 1," in *The Routledge companion to media disinformation and populism*. Routledge, 2021.
- [107] E. Rodilosso, "Filter bubbles and the unfeeling : How ai for social media can foster extremism and polarization," *Philosophy & Technology*, 2024.
- [108] M. Coeckelbergh, "Democracy, epistemic agency, and ai : political epistemology in times of artificial intelligence," *AI and Ethics*, 2023.
- [109] X. Su, S. Xue, F. Liu, J. Wu, J. Yang, C. Zhou, W. Hu, C. Paris, S. Nepal, D. Jin *et al.*, "A comprehensive survey on community detection with deep learning," *IEEE transactions on neural networks and learning systems*, 2022.
- [110] P. Bedi and C. Sharma, "Community detection in social networks," *Wiley interdisciplinary reviews : Data mining and knowledge discovery*, 2016.
- [111] M. E. Newman and M. Girvan, "Finding and evaluating community structure in networks," *Physical review E*, 2004.
- [112] M. E. Newman, "Finding community structure in networks using the eigenvectors of matrices," *Physical Review E—Statistical, Nonlinear, and Soft Matter Physics*, 2006.
- [113] V. D. Blondel, J.-L. Guillaume, R. Lambiotte, and E. Lefebvre, "Fast unfolding of communities in large networks," *Journal of statistical mechanics : theory and experiment*, 2008.

- [114] M. S. Granovetter, "The strength of weak ties," *American journal of sociology*, 1973.
- [115] A.-L. Barabási and R. Albert, "Emergence of scaling in random networks," *science*, 1999.
- [116] N. Ferguson, "The false prophecy of hyperconnection : How to survive the networked age," *Foreign Affairs*, 2017.
- [117] E. D. Demaine and M. Zadimoghaddam, "Minimizing the diameter of a network using shortcut edges," in *Scandinavian Workshop on Algorithm Theory*, 2010.
- [118] N. Parotsidis, E. Pitoura, and P. Tsaparas, "Selecting shortcuts for a smaller world," in *Proceedings of the 2015 SIAM International Conference on Data Mining*, 2015.
- [119] D. J. Watts and S. H. Strogatz, "Collective dynamics of 'small-world' networks," *nature*, 1998.
- [120] T. Silveira, M. Zhang, X. Lin, Y. Liu, and S. Ma, "How good your recommender system is? a survey on evaluations in recommendation," *International Journal of Machine Learning and Cybernetics*, 2019.
- [121] M. Fishbein, "The search for attitudinal-behavioral consistency," *Behavioral science foundations of consumer behavior*, 1972.
- [122] J. R. Bettman, "A threshold model of attribute satisfaction decisions," *Journal of Consumer Research*, 1974.
- [123] M. Fishbein and I. Ajzen, "Predicting and understanding consumer behavior : Attitude-behavior correspondence," *Understanding attitudes and predicting social behavior*, 1980.
- [124] P. R. Grim, "A technique for the measurement of attitudes in the social studies," *Educational Research Bulletin*, pp. 95–104, 1936.
- [125] J. W. Thibaut, *The social psychology of groups*. Routledge, 2017.
- [126] B. Friedman and H. Nissenbaum, "Bias in computer systems," *ACM Transactions on information systems (TOIS)*, 1996.
- [127] Y. Ge, S. Liu, R. Gao, Y. Xian, Y. Li, X. Zhao, C. Pei, F. Sun, J. Ge, W. Ou *et al.*, "Towards long-term fairness in recommendation," in *Proceedings of the 14th ACM international conference on web search and data mining*, 2021.
- [128] G. K. Patro, A. Biswas, N. Ganguly, K. P. Gummadi, and A. Chakraborty, "Fairrec : Two-sided fairness for personalized recommendations in two-sided platforms," in *Proceedings of the web conference 2020*, 2020.
- [129] J. Chen, H. Dong, X. Wang, F. Feng, M. Wang, and X. He, "Bias and debias in recommender system : A survey and future directions," *ACM Transactions on Information Systems*, 2023.

- [130] Y. Wang, W. Ma, M. Zhang, Y. Liu, and S. Ma, "A survey on the fairness of recommender systems," *ACM Transactions on Information Systems*, 2023.
- [131] R. K. Merton, "The matthew effect in science, ii : Cumulative advantage and the symbolism of intellectual property," *isis*, 1988.
- [132] Y. Zhao, Y. Wang, Y. Liu, X. Cheng, C. C. Aggarwal, and T. Derr, "Fairness and diversity in recommender systems : a survey," *ACM Transactions on Intelligent Systems and Technology*, 2025.
- [133] C. E. Shannon, "A mathematical theory of communication," *The Bell system technical journal*, 1948.
- [134] H. Chen, W. Lu, R. Song, and P. Ghosh, "Counterfactual fairness through data preprocessing," 2021, this is an unpublished article.
- [135] M. D. Ekstrand, M. Tian, I. M. Azpiazu, J. D. Ekstrand, O. Anuyah, D. McNeill, and M. S. Pera, "All the cool kids, how do they fit in? : Popularity and demographic biases in recommender evaluation and effectiveness," in *Conference on fairness, accountability and transparency*, 2018.
- [136] H. Yang, Z. Liu, Z. Zhang, C. Zhuang, and X. Chen, "Towards robust fairness-aware recommendation," in *Proceedings of the 17th ACM Conference on Recommender Systems*, 2023.
- [137] J. Tang, S. Shen, Z. Wang, Z. Gong, J. Zhang, and X. Chen, "When fairness meets bias : a debiased framework for fairness aware top-n recommendation," in *Proceedings of the 17th ACM Conference on Recommender Systems*, 2023.
- [138] S. Rendle, C. Freudenthaler, Z. Gantner, and L. Schmidt-Thieme, "Bpr : Bayesian personalized ranking from implicit feedback," *arXiv preprint arXiv :1205.2618*, 2012.
- [139] Y. Zheng and D. Wang, "Multi-objective recommendations," in *Proceedings of the 27th ACM SIGKDD Conference on Knowledge Discovery & Data Mining*, 2021.
- [140] M. T. Ribeiro, A. Lacerda, A. Veloso, and N. Ziviani, "Pareto-efficient hybridization for multi-objective recommender systems," in *Proceedings of the sixth ACM conference on Recommender systems*, 2012.
- [141] L. Wu, L. Chen, P. Shao, R. Hong, X. Wang, and M. Wang, "Learning fair representations for recommendation : A graph-based perspective," in *Proceedings of the Web Conference 2021*, 2021.
- [142] E. Pitoura, K. Stefanidis, and G. Koutrika, "Fairness in rankings and recommendations : an overview," *The VLDB Journal*, 2022.
- [143] M. Mansoury, H. Abdollahpouri, M. Pechenizkiy, B. Mobasher, and R. Burke, "A graph-based approach for mitigating multi-sided exposure bias in recommender systems," *ACM Transactions on Information Systems (TOIS)*, 2021.

- [144] A. J. Biega, K. P. Gummadi, and G. Weikum, "Equity of attention : Amortizing individual fairness in rankings," in *The 41st international acm sigir conference on research & development in information retrieval*, 2018.
- [145] M. Mansoury, H. Abdollahpouri, M. Pechenizkiy, B. Mobasher, and R. Burke, "Fairmatch : A graph-based approach for improving aggregate diversity in recommender systems," in *Proceedings of the 28th ACM conference on user modeling, adaptation and personalization*, 2020.
- [146] Y. Li, H. Chen, Z. Fu, Y. Ge, and Y. Zhang, "User-oriented fairness in recommendation," in *Proceedings of the web conference 2021*, 2021.
- [147] H. A. Taha, "Integer programming," in *Mathematical Programming for Operations Researchers and Computer Scientists*. CRC Press, 2020.
- [148] J. Carbonell and J. Goldstein, "The use of mmr, diversity-based reranking for reordering documents and producing summaries," in *Proceedings of the 21st annual international ACM SIGIR conference on Research and development in information retrieval*, 1998.
- [149] S. Seymen, H. Abdollahpouri, and E. C. Malthouse, "A unified optimization toolbox for solving popularity bias, fairness, and diversity in recommender systems." in *MORS@ RecSys*, 2021.
- [150] T. Jambor and J. Wang, "Optimizing multiple objectives in collaborative filtering," in *Proceedings of the fourth ACM conference on Recommender systems*, 2010.
- [151] N. Li, B. Kang, and T. De Bie, "Content-agnostic moderation for stance-neutral recommendation," *arXiv preprint arXiv :2405.18941*, 2024.
- [152] P. Babvey, D. Borrelli, C. Lipizzi, and J. E. Ramirez-Marquez, "Content-aware galaxies : Digital fingerprints of discussions on social media," *IEEE Transactions on Computational Social Systems*, 2020.
- [153] M. H. DeGroot, "Reaching a consensus," *Journal of the American Statistical association*, 1974.
- [154] T. Evans and F. Fu, "Opinion formation on dynamic networks : identifying conditions for the emergence of partisan echo chambers," *Royal Society open science*, 2018.
- [155] N. E. Friedkin and E. C. Johnsen, "Social influence and opinions," *Journal of mathematical sociology*, 1990.
- [156] J. Lorenz, "Continuous opinion dynamics under bounded confidence : A survey," *International Journal of Modern Physics C*, 2007.
- [157] R. Hegselmann, "Opinion dynamics and bounded confidence : models, analysis and simulation," *The Journal of Artificial Societies and Social Simulation*, 2015.

- [158] G. Deffuant, D. Neau, F. Amblard, and G. Weisbuch, "Mixing beliefs among interacting agents," *Advances in Complex Systems*, 2000.
- [159] P. Clifford and A. Sudbury, "A model for spatial conflict," *Biometrika*, 1973.
- [160] E. Bakshy, S. Messing, and L. A. Adamic, "Exposure to ideologically diverse news and opinion on facebook," *Science*, 2015.
- [161] J. E. Lane, K. McCaffree, and F. L. Shults, "Is radicalization reinforced by social media censorship?" *arXiv preprint arXiv :2103.12842*, 2021.
- [162] H. A. Prasetya and T. Murata, "A model of opinion and propagation structure polarization in social media," *Computational Social Networks*, 2020.
- [163] L. Zhu, Q. Bao, and Z. Zhang, "Minimizing polarization and disagreement in social networks via link recommendation," *Advances in Neural Information Processing Systems*, 2021.
- [164] K. Garimella, G. De Francisci Morales, A. Gionis, and M. Mathioudakis, "Reducing controversy by connecting opposing views," in *Proceedings of the tenth ACM international conference on web search and data mining*, 2017.
- [165] C. Musco, C. Musco, and C. E. Tsourakakis, "Minimizing polarization and disagreement in social networks," in *Proceedings of the 2018 world wide web conference*, 2018.
- [166] F. Fabbri, Y. Wang, F. Bonchi, C. Castillo, and M. Mathioudakis, "Rewiring what-to-watch-next recommendations to reduce radicalization pathways," in *Proceedings of the ACM Web Conference 2022*, 2022.
- [167] S. Chandrasekaran, G. De Pasquale, G. Belgioioso, and F. Dörfler, "Mitigating polarization in recommender systems via network-aware feedback optimization," *arXiv e-prints*, 2024.
- [168] M. Pérez-Escolar, D. Lilleker, and A. Tapia-Frade, "A systematic literature review of the phenomenon of disinformation and misinformation," *Media and communication*, vol. 11, no. 2, pp. 76–87, 2023.
- [169] A. M. Guess and B. A. Lyons, "Misinformation, disinformation, and online propaganda," *Social media and democracy: The state of the field, prospects for reform*, vol. 10, pp. 10–33, 2020.
- [170] S. Govindankutty and S. P. Gopalan, "Epidemic modeling for misinformation spread in digital networks through a social intelligence approach," *Scientific Reports*, vol. 14, no. 1, p. 19100, 2024.
- [171] A. K. Ghoshal, N. Das, S. Das, and S. Dhar, "Minimizing spread of misinformation in social networks : a network topology based approach," *Social Network Analysis and Mining*, vol. 15, no. 1, p. 15, 2025.

- [172] J. Stein, M. Keuschnigg, and A. van de Rijt, "Network segregation and the propagation of misinformation," *Scientific Reports*, vol. 13, no. 1, p. 917, 2023.
- [173] R. Rodriguez-Garcia, R. Centeno, and A. Rodrigo, "Simulating misinformation diffusion on social media through convai : A textual-and agent-based diffusion model," 2025.
- [174] F. Ghisi, M. Gotelli, V. Solina, and F. Tonelli, "From misinformation to resilient communication : Strategic simulation of social network dynamics in the pharmaceutical industry," *Applied Sciences*, vol. 15, no. 21, p. 11734, 2025.
- [175] E. Aïmeur, S. Amri, and G. Brassard, "Fake news, disinformation and misinformation in social media : a review," *Social Network Analysis and Mining*, vol. 13, no. 1, p. 30, 2023.
- [176] A. Loth, M. Kappes, and M.-O. Pahl, "Blessing or curse? a survey on the impact of generative ai on fake news," *arXiv preprint arXiv :2404.03021*, 2024.
- [177] K. Rudra, N. Ganguly, J. Mifsud Bonnici, E. Müller-Budack, and R. Manu-
vie, "Disinformation and misinformation in the age of generative ai," in *Proceedings of the Eighteenth ACM International Conference on Web Search and Data Mining*, 2025, pp. 1122–1123.
- [178] R. J. Ziarani and R. Ravanmehr, "Serendipity in recommender systems : a systematic literature review," *Journal of Computer Science and Technology*, 2021.
- [179] X. Niu, F. Abbas, M. L. Maher, and K. Grace, "Surprise me if you can : Serendipity in health information," in *Proceedings of the 2018 CHI Conference on Human Factors in Computing Systems*, 2018.
- [180] D. Kotkov, A. Medlar, and D. Glowacka, "Rethinking serendipity in recommender systems," in *Proceedings of the 2023 Conference on Human Information Interaction and Retrieval*, 2023.
- [181] Y. M. Omar and P. Plapper, "A survey of information entropy metrics for complex networks," *Entropy*, 2020.
- [182] D. Ortiz-Arroyo and D. A. Hussain, "An information theory approach to identify sets of key players," in *European conference on intelligence and security informatics*, 2008.
- [183] E. Serin and S. Balcişoy, "Entropy based sensitivity analysis and visualization of social networks," in *2012 IEEE/ACM International Conference on Advances in Social Networks Analysis and Mining*, 2012.
- [184] T. Nie, Z. Guo, K. Zhao, and Z.-M. Lu, "Using mapping entropy to identify node centrality in complex networks," *Physica A : Statistical Mechanics and its Applications*, 2016.

- [185] V. Chellappan and K. M. Sivalingam, "Application of entropy of centrality measures to routing in tactical wireless networks," in *2013 19th IEEE Workshop on Local & Metropolitan Area Networks (LANMAN)*, 2013.
- [186] D. A. Hussain and D. Ortiz-Arroyo, "Locating key actors in social networks using bayes' posterior probability framework," in *European Conference on Intelligence and Security Informatics*, 2008.
- [187] J. Parapar and F. Radlinski, "Towards unified metrics for accuracy and diversity for recommender systems," in *Proceedings of the 15th ACM Conference on Recommender Systems*, 2021.
- [188] D. Stamenkovic, A. Karatzoglou, I. Arapakis, X. Xin, and K. Katevas, "Choosing the best of both worlds : Diverse and novel recommendations through multi-objective reinforcement learning," in *Proceedings of the fifteenth acm international conference on web search and data mining*, 2022.
- [189] B. Smith and G. Linden, "Two decades of recommender systems at amazon. com," *IEEE internet computing*, 2017.
- [190] K. Acharya, D. Wardrope, T. Korres, A. V. Petrov, and A. Uhrenholt, "Improving minimax group fairness in sequential recommendation," in *European Conference on Information Retrieval*, 2025.
- [191] S. D. Lamkhede and C. Kofler, "Recommendations and results-organization in netflix search," in *Proceedings of the 15th ACM Conference on Recommender Systems*, 2021.
- [192] S. Lamkhede and S. Das, "Challenges in search on streaming services : Netflix case study," in *Proceedings of the 42nd international ACM SIGIR conference on research and development in information retrieval*, 2019.
- [193] G. Tang, J. Pan, H. Wang, and J. Basilico, "Reward innovation for long-term member satisfaction," in *Proceedings of the 17th ACM Conference on Recommender Systems*, 2023.
- [194] H. Steck, "Calibrated recommendations," in *Proceedings of the 12th ACM conference on recommender systems*, 2018.
- [195] B. Lacker and S. F. Way, "Socially-motivated music recommendation," in *Proceedings of the international AAAI conference on web and social media*, 2024.
- [196] R. Mehrotra, J. McInerney, H. Bouchard, M. Lalmas, and F. Diaz, "Towards a fair marketplace : Counterfactual evaluation of the trade-off between relevance, fairness & satisfaction in recommendation systems," in *Proceedings of the 27th acm international conference on information and knowledge management*, 2018.
- [197] L. Beattie, D. Taber, and H. Cramer, "Challenges in translating research to practice for evaluating fairness and bias in recommendation sys-

- tems," in *Proceedings of the 16th ACM Conference on Recommender Systems*, 2022.
- [198] S. Milgram *et al.*, "The small world problem," *Psychology today*, 1967.
- [199] F. Cinus, A. Gionis, and F. Bonchi, "Rebalancing social feed to minimize polarization and disagreement," in *Proceedings of the 32nd ACM International Conference on Information and Knowledge Management*, 2023.
- [200] M. Kunaver and T. Požrl, "Diversity in recommender systems—a survey," *Knowledge-based systems*, 2017.
- [201] N. Bhalla, A. Lechowicz, and C. Musco, "Local edge dynamics and opinion polarization," *ACM Transactions on Intelligent Systems and Technology*, 2025.
- [202] M. Zinkevich, "Online convex programming and generalized infinitesimal gradient ascent," in *Proceedings of the 20th international conference on machine learning (icml-03)*, 2003.
- [203] M. Girvan and M. E. Newman, "Community structure in social and biological networks," *Proceedings of the national academy of sciences*, 2002.
- [204] F. M. Harper and J. A. Konstan, "The movielens datasets : History and context," *Acm transactions on interactive intelligent systems (tiis)*, 2015.
- [205] S. Kumar, W. L. Hamilton, J. Leskovec, and D. Jurafsky, "Community interaction and conflict on the web," in *Proceedings of the 2018 world wide web conference*, 2018.
- [206] B. Viswanath, A. Mislove, M. Cha, and K. P. Gummadi, "On the evolution of user interaction in facebook," in *Proceedings of the 2nd ACM workshop on Online social networks*, 2009.
- [207] J. Yang and J. Leskovec, "Defining and evaluating network communities based on ground-truth," in *Proceedings of the ACM SIGKDD workshop on mining data semantics*, 2012.
- [208] M. E. Newman, "The structure and function of complex networks," *SIAM review*, 2003.
- [209] L.-C. Zhang, *Graph sampling*. Chapman and Hall/CRC, 2021.
- [210] M. Brookes, *The matrix reference manual*. Imperial College London, 2005.

7 - Annexes

7.1 . User-Content setting derivations

Vectorized transpose matrix In the following we will use the Vectorized Transpose Matrix TVEC as defined in [210]. For any vectorized matrix $X \in \mathcal{M}_{m,n}$, the TVEC is the $nm \times nm$ permutation matrix :

$$X^\top = \text{TVEC } X. \quad (7.1)$$

General note Using the product rule and the following formulas for matrix derivatives [210], for $X \in \mathcal{M}_{m,n}$:

$$\begin{aligned} \frac{\partial(X^\top C X)}{\partial X} &= (I_m \otimes X^\top C) \partial X + (X^\top C^\top \otimes I_m) \partial X^\top \\ &= (I_m \otimes X^\top C + \text{TVEC}[m, m](I_m \otimes X^\top C^\top)) \partial X \end{aligned} \quad (7.2)$$

$$\frac{\partial(fg)}{\partial X} = (I \otimes f) \frac{\partial g}{\partial X} + \left((g^\top \otimes I) \frac{\partial f}{\partial X} \right) \partial X \quad (7.3)$$

Using $X^\top = T$ and adapting M for each exposure function, we then have :

$$\begin{aligned} \frac{\partial \mathbf{u} T M T^\top}{\partial T} &= (Id \otimes \mathbf{u}) \frac{\partial T M T^\top}{\partial T} + (T M T^\top)^\top \otimes Id \frac{\partial \mathbf{u}}{\partial T} \\ &= (Id \otimes \mathbf{u}) \left(Id \otimes T M + \text{TVEC}[n, n] (Id \otimes T^\top M^\top) \right) \end{aligned}$$

PPR Exposure.

$$\begin{aligned} \mathcal{E}_G^{PPR} &= W_G^{PPR} \\ &= (1 - \alpha) (I - \alpha G)^{-1} \\ (I - \alpha G) &= \begin{bmatrix} I - \alpha U & \alpha T \\ \alpha T^\top & I - \alpha C \end{bmatrix}, \\ (I - \alpha G)^{-1} &= \begin{bmatrix} S^{-1} & -\alpha S^{-1} T C^{-1} \\ -\alpha C^{-1} T^\top S^{-1} & C^{-1} + \alpha^2 C^{-1} T^\top S^{-1} T C^{-1} \end{bmatrix}, \\ \mathcal{E}_G^{PPR} &\approx (1 - \alpha) \begin{bmatrix} S^{-1} \\ -C^{-1} T^\top S^{-1} \end{bmatrix} \\ \text{with } S &= (I - \alpha U - \alpha^2 T (I - \alpha C)^{-1} T^\top) \end{aligned} \quad (7.4)$$

□

FJ Exposure.

$$\begin{aligned}
\mathcal{E}_G^{FJ} &= W_G^{FJ} & (7.5) \\
&= (1 - \alpha) (I - \alpha G^\top)^{-1} \\
(I - \alpha G^\top) &= \begin{bmatrix} I - \alpha U^\top & \alpha T \\ \alpha T^\top & I - \alpha C^\top \end{bmatrix}, \\
(I - \alpha G^\top)^{-1} &= \begin{bmatrix} S^{-1} & -\alpha S^{-1} T C^{-\top} \\ -\alpha C^{-\top} T^\top S^{-1} & C^{-\top} + \alpha^2 C^{-\top} T^\top S^{-1} T C^{-\top} \end{bmatrix}, \\
\mathcal{E}_G^{PPR} &\approx (1 - \alpha) \begin{bmatrix} S^{-1} \\ -C^{-\top} T^\top S^{-1} \end{bmatrix} \\
\text{with } S &= \left(I - \alpha U^\top - \alpha^2 T (I - \alpha C^\top)^{-1} T^\top \right)
\end{aligned}$$

□

BFS Exposure.

$$\begin{aligned}
\mathcal{E}_G^{BFS} &= W_G^{BFS} & (7.6) \\
&= G^3 + G^2 + G
\end{aligned}$$

$$\begin{aligned}
&= \begin{bmatrix} U^3 + U^2 + U + UTT^\top + TT^\top U + TCT^\top + TT^\top \\ T^\top U^2 + T^\top U + T^\top TT^\top + CT^\top U + CT^\top + C^2 T^\top + T^T \\ U^2 T + UT + UTC + TC + TT^\top T + TC^2 + T \\ T^\top UT + T^\top T + T^\top TC + CT^\top T + C^3 + C^2 + C \end{bmatrix} & (7.7)
\end{aligned}$$

$$\mathcal{E}_G^{BFS} \approx \begin{bmatrix} U^3 + U^2 + U + UTT^\top + TT^\top U + TCT^\top + TT^\top \\ T^\top U^2 + T^\top U + T^\top TT^\top + CT^\top U + CT^\top + C^2 T^\top + T^T \end{bmatrix}$$

□

Here are the details of the derivation process :

PPR Exposure Derivation Proof.

$$\mathcal{E}_G^{PPR} = (1 - \alpha) \begin{bmatrix} S^{-1} \\ -C^{-1} T^\top S^{-1} \end{bmatrix} \quad (7.8)$$

with $S = (I - \alpha U - \alpha^2 T (I - \alpha C)^{-1} T^\top)$

$$d\mathcal{E}_G^{PPR} = (1 - \alpha) \left[\begin{array}{c} (S^{-\top} \otimes S^{-1}) dS \\ (I \otimes C^{-1} T^\top) (S^{-\top} \otimes S^{-1}) dS + (S^{-\top} \otimes I) d(C^{-1} T^\top) \end{array} \right] \quad (7.9)$$

$$\begin{aligned}\frac{\partial \mathcal{E}_G^{PPR}}{\partial U} &= -\alpha(1-\alpha) \left[\begin{array}{c} (S^{-\top} \otimes S^{-1}) \\ (I \otimes C^{-1}T^\top) (S^{-\top} \otimes S^{-1}) \end{array} \right] \\ \frac{\partial \mathcal{E}_G^{PPR}}{\partial U}[u] &= -\alpha(1-\alpha) \left[\begin{array}{c} (S_{u,u}^{-\top} S^{-1}) \\ (C^{-1}T^\top) (S_{u,u}^{-\top} S^{-1}) \end{array} \right]\end{aligned}\quad (7.10)$$

$$\frac{\partial \mathcal{E}_G^{PPR}}{\partial T} = (1-\alpha) \left[\begin{array}{c} -\alpha^2 (S^{-\top} \otimes S^{-1}) X \\ -\alpha^2 \left((I \otimes C^{-1}T^\top) (S^{-\top} \otimes S^{-1}) X \right) + (S^{-\top} \otimes I) \text{TVEC}(I \otimes C^{-1}) \end{array} \right] \quad (7.11)$$

where $X = I \otimes T(I - \alpha C)^{-1} + \text{TVEC}(I \otimes T(I - \alpha C)^{-\top})$.

$$\frac{\partial \mathcal{E}_G^{PPR}}{\partial T}[u] = (1-\alpha) \left[\begin{array}{c} -\alpha^2 (S_{u,u}^{-\top} S^{-1}) (T(I - \alpha C)^{-1} + (T(I - \alpha C)^{-\top})) \\ -\alpha^2 \left((C^{-1}T^\top) (S_{u,u}^{-\top} S^{-1}) (T(I - \alpha C)^{-1} + (T(I - \alpha C)^{-\top})) \right) + S_{u,u}^{-\top} C^{-1} \end{array} \right] \quad (7.12)$$

□

FJ Exposure Derivation Proof.

$$d\mathcal{E}_G^{FJ} = (1-\alpha) \left[\begin{array}{c} (S^{-\top} \otimes S^{-1}) dS \\ (I \otimes C^{-\top}T^\top) (S^{-\top} \otimes S^{-1}) dS + (S^{-\top} \otimes I) d(C^{-\top}T^\top) \end{array} \right] \quad (7.13)$$

$$\begin{aligned}\frac{\partial \mathcal{E}_G^{FJ}}{\partial U} &= -\alpha(1-\alpha) \left[\begin{array}{c} \text{TVEC}(S^{-1} \otimes S^{-\top}) \\ (I \otimes C^{-\top}T^\top) \text{TVEC}(S^{-1} \otimes S^{-\top}) \end{array} \right] \\ \frac{\partial \mathcal{E}_G^{FJ}}{\partial U}[u] &= -\alpha(1-\alpha) \left[\begin{array}{c} S_{u,u}^{-1} S^{-\top} \\ (C^{-\top}T^\top) (S_{u,u}^{-1} S^{-\top}) \end{array} \right]\end{aligned}\quad (7.14)$$

$$\frac{\partial \mathcal{E}_G^{FJ}}{\partial T} = (1-\alpha) \left[\begin{array}{c} -\alpha^2 (S^{-\top} \otimes S^{-1}) X \\ -\alpha^2 \left((I \otimes C^{-\top}T^\top) (S^{-\top} \otimes S^{-1}) X \right) + (S^{-\top} \otimes I) (I \otimes C^{-\top}) \end{array} \right] \quad (7.15)$$

where $X = I \otimes T(I - \alpha C^\top)^{-1} + \text{TVEC}(I \otimes T(I - \alpha C^\top)^{-\top})$.

$$\frac{\partial \mathcal{E}_G^{FJ}}{\partial T}[u] = (1 - \alpha) \left[\begin{array}{c} -\alpha^2 \left(S_{[u,u]}^{-\top} S^{-1} \right) \left(T(I - \alpha C^\top)^{-1} + (T(I - \alpha C^\top)^{-\top}) \right) \\ -\alpha^2 \left((C^{-\top} T^\top) \left(S_{[u,u]}^{-\top} S^{-1} \right) \left(T(I - \alpha C^\top)^{-1} + (T(I - \alpha C^\top)^{-\top}) \right) \right) + S_{[u,u]}^{-\top} C^{-\top} \end{array} \right] \quad (7.16)$$

□

BFS Exposure Derivation Proof.

$$\frac{\partial \mathcal{E}_G^{BFS}}{\partial U} = \left[\begin{array}{c} U^{T^2} \otimes I + U^\top \otimes U + I \otimes U^2 + I \otimes U + U^\top \otimes I + TT^\top \otimes I + I \otimes TT^\top \\ I \otimes T^\top U + U^\top \otimes T^\top + I \otimes T^\top + I \otimes CT^\top \end{array} \right] \quad (7.17)$$

$$\frac{\partial \mathcal{E}_G^{BFS}}{\partial U}[u] = \left[\begin{array}{c} U_{u,u}^{T^2} I + U_{u,u}^\top U + U^2 + U + U_{u,u}^\top I + TT_{u,u}^\top I + TT^\top \\ T^\top U + U_{i,i}^\top T^\top + T^\top + CT^\top \end{array} \right]$$

$$\frac{\partial \mathcal{E}_G^{BFS}}{\partial T} = \left[\begin{array}{c} \text{TVEC}(UT \otimes I) + (T \otimes U) + \text{TVEC}(T \otimes U^\top + (U^\top T \otimes I) + \\ (U^{2^\top} \otimes I) + (U^\top \otimes I) + (TT^\top \otimes I) + \text{TVEC}(T^\top \otimes T) + \end{array} \right] \quad (7.18)$$

$$\begin{array}{c} +I \otimes TC + \text{TVEC}(I \otimes TC^\top + I + \text{TVEC}(I \otimes T)) \\ + (I \otimes TT^\top) + (U^\top \otimes C) + (I \otimes C) + (I \otimes C^2) \end{array}$$

$$\frac{\partial \mathcal{E}_G^{BFS}}{\partial T}[u] = \left[\begin{array}{c} (UT_{u,u} I) + (T_{u,u} U) + (T_{u,u} U^\top) + (U^\top T_{u,u} I) + TC + TC^\top + T \\ U_{u,u}^{2^\top} I + U_{u,u}^\top I + TT_{u,u}^\top I + T^\top T + U_{u,u}^\top C + C + C^2 \end{array} \right]$$

□

7.2 . Supplementary Results for Chapter 4

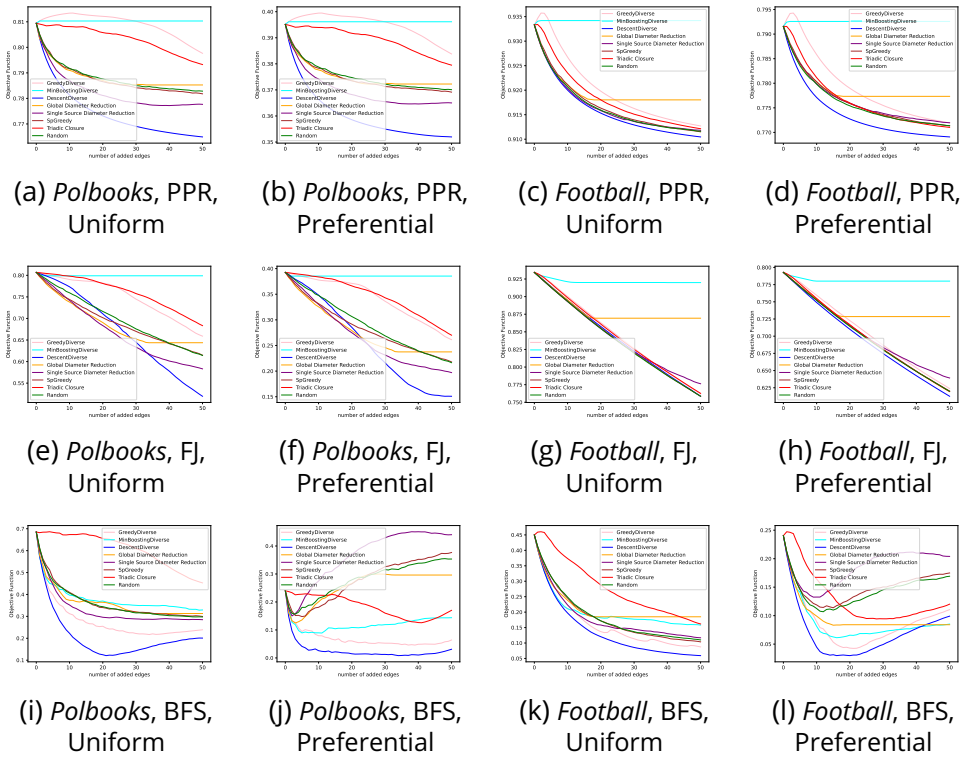


Figure 7.1 – Objective vs. number of edge additions, ground truth communities.

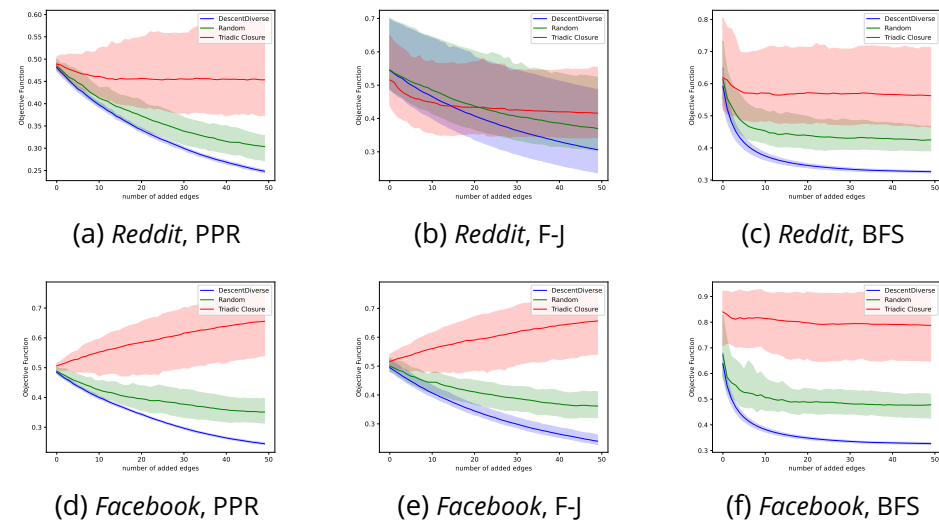


Figure 7.2 – Objective vs. number of edge additions, inferred communities, uniform target partition.