



République Algérienne Démocratique et Populaire
Ministère de l'Enseignement Supérieur et de la Recherche Scientifique
Université A.MIRA-BEJAIA
Faculté des Sciences Exactes
Département d'Informatique
Unité de Recherche LaMOS

THÈSE

Présentée par

Lyes BADIS

Pour l'obtention du grade de

DOCTEUR EN SCIENCES

Filière : Informatique

Option : Réseaux et Systèmes Distribués

Thème

**Contribution aux Performances des Réseaux Sociaux à
base d'Architectures P2P**

Soutenue le : 13 Novembre 2021.

Devant le Jury composé de :

Nom et Prénom

Grade

Mme Louiza BOUALLOUCHE	Professeur	Univ. de Béjaïa	Président
Mr Djamil AÏSSANI	Professeur	Univ. de Béjaïa	Rapporteur
Mr Mourad AMAD	Professeur	Univ. de Bouira	Rapporteur
Mr Fouzi SEMCHEDINE	Professeur	Univ. de Setif 1	Examineur
Mr Bilal SAOUD	M.C.A	Univ. de Bouira	Examineur
Mr Abdelmalek BOUDRIES	M.C.A	Univ. de Béjaïa	Examineur

Année Universitaire : 2021-2022

*Je dédie ce travail
A ma mère,
A mon père,
A ma femme et mes enfants,
A toute ma famille,
A tous mes amis.*

Table des Matières

Table des matières	iv
Liste des Tableaux	viii
Liste des Algorithmes	viii
Table des Figures	x
Remerciements	xi
Résumé	xii
Introduction Générale	1
1 Généralités sur les réseaux sociaux	3
1.1 Introduction	3
1.2 Historiques	3
1.3 Définitions	4
1.4 Fonctionnalités	6
1.4.1 Création de profil	6
1.4.2 Publication	6
1.4.3 Navigation	6
1.4.4 Interactions avec les autres utilisateurs	7
1.4.5 Connexion aux autres utilisateurs	7
1.4.6 Administration	8
1.4.7 Discussion	8
1.4.8 Jeux	8
1.4.9 Diffusion en direct du contenu multimédia	8
1.5 Applications des réseaux sociaux	9
1.5.1 Communication	9
1.5.2 Diffusion d'information	9
1.5.3 Réseautage d'affaires	9
1.5.4 Recherche d'information	9
1.5.5 Marketing	10
1.5.6 Gestion des catastrophes naturelles	10
1.5.7 Analyse des réseaux sociaux	10

1.6	Types de réseaux sociaux	11
1.6.1	Réseaux généralistes/ Réseaux spécialisés	11
1.6.1.1	Les réseaux généralistes	11
1.6.1.2	Les réseaux spécialisés	11
1.6.2	Réseaux de relations /réseaux de contenu	12
1.6.3	Classification à base des objectifs	12
1.6.4	Classification basée sur l'usage	13
1.6.5	Classification de Delcroix	15
1.7	Avantages et Inconvénients	15
1.8	Exemples de réseaux sociaux	17
1.8.1	Facebook	18
1.8.2	Linkedin	18
1.8.3	Twitter	18
1.8.4	YouTube	18
1.9	Conclusion	19
2	État de l'art sur les réseaux sociaux décentralisés	21
2.1	Introduction	21
2.2	Motivation	21
2.3	Architecture décentralisée pour les réseaux sociaux	22
2.3.1	Gestion du stockage	23
2.3.1.1	Stockage sur les machines des utilisateurs (<i>pair à pair</i>)	23
2.3.1.2	Stockage sur plusieurs serveurs	25
2.3.1.3	Stockage hybride (<i>serveur/pair à pair</i>)	26
2.3.2	Protection	26
2.3.2.1	Chiffrement symétrique/asymétrique	27
2.3.2.2	Chiffrement basé sur les attributs(<i>ABE</i>)	28
2.3.2.3	Listes d'accès sans chiffrement	28
2.3.2.4	Listes d'accès avec chiffrement	28
2.3.3	Communication	29
2.3.3.1	Communication gérée par les serveurs	29
2.3.3.2	Communication pair à pair	29
2.4	Classification des réseaux sociaux décentralisés	30
2.4.1	Réseaux sociaux Pair à Pair	30
2.4.2	Réseaux sociaux fédérés (<i>basés sur les serveurs</i>)	30
2.4.3	Réseaux sociaux hybrides	31
2.5	Fonctionnalités des réseaux sociaux décentralisés	31
2.6	Avantages et limites	31
2.6.1	Avantages	32
2.6.2	Limites	32
2.7	Pistes de recherche dans les réseaux sociaux décentralisés	34
2.8	Conclusion	34
3	Mise à jour des profils répliqués dans les réseaux sociaux décentralisés-un modèle basé sur le journal des événements	36
3.1	Introduction	36
3.2	Problématique	36

3.3	Travaux antérieurs	37
3.4	Solution Proposée	39
3.4.1	Réplication de profils	39
3.4.2	Recherche des Profils	40
3.4.3	Le journal des événements	42
3.4.4	Stockage des journaux des événements	47
3.4.5	Les algorithmes de mise à jour des profils	47
3.4.5.1	Mise à jour directe	47
3.4.5.2	Mise à jour synchrone	48
3.4.5.3	Mise à jour asynchrone	49
3.4.6	Illustration du modèle de mise à jour	49
3.4.7	Maintenance des journaux	50
3.5	Evaluation des performances	51
3.5.1	La disponibilité des profils	51
3.5.2	Équilibrage de charge dans les groupes de réplication	53
3.5.3	Discussion	55
3.6	Conclusion	55
4	P2PCF : un système de recommandation basé sur le filtrage collaboratif pour les réseaux sociaux pair à pair	57
4.1	Introduction	57
4.2	Problématique	57
4.3	Background et travaux antérieurs	59
4.3.1	Systèmes de recommandation	59
4.3.2	Propagation de contenu dans les réseaux sociaux P2P	60
4.3.3	Recommandation dans les réseaux P2P	61
4.4	Solution Proposée	63
4.4.1	Aperçu général	63
4.4.2	Scénario d'illustration	63
4.4.3	Stockage décentralisé des interactions	64
4.4.4	Les étapes de P2PCF	66
4.4.4.1	Sélection des amis	66
4.4.4.2	Sélection des publications	67
4.4.4.3	Agrégation des recommandations	69
4.4.5	Problème de démarrage à froid	70
4.4.5.1	Le cas des nouveaux utilisateurs	70
4.4.5.2	Le cas des nouvelles publications	71
4.5	Evaluation des performances	72
4.5.1	Le jeu de données	72
4.5.2	Détails d'implémentation	72
4.5.3	Rappel	73
4.5.4	Couverture	73
4.5.5	Sélection des amis	74
4.5.6	Sélection des publications	74
4.5.7	Discussion	75
4.6	Conclusion	76

Conclusion générale et perspectives	80
Bibliographie	83

Liste des tableaux

2.1	Top10 des réseaux sociaux fédérés	31
2.2	Les fonctionnalités fournies par les réseaux sociaux décentralisés	32
3.1	Exemple de distribution de priorité dans le groupe de réplication	41
4.1	Exemple de matrice de notation locale	66
4.2	Résumé des statistiques des données de test	72

Liste des algorithmes

1	Génération de la table des priorités	42
2	Recherche des profils	45
3	Mise à jour directe	49
4	Mise à jour synchrone	49
5	Mise à jour asynchrone	50

Table des figures

1.1	Panorama des médias sociaux	13
1.2	Les réseaux sociaux les plus populaires	19
2.1	Architecture type d'un réseau social décentralisé	23
2.2	Stockage sur les machines des utilisateurs	24
2.3	Stockage sur des serveurs	26
2.4	Architecture de PeerSon	30
3.1	Réplication des profils	40
3.2	Les listes ordonnées d'appareils	43
3.3	Le processus de recherche d'un profil	44
3.4	Exemple d'un journal des événements	46
3.5	Stockage des journaux des événements	48
3.6	Nouvel événement écrit dans le journal	51
3.7	Disponibilité des profils	52
3.8	Les machines utilisées pour accéder aux profils	53
3.9	Nombre moyen de profils servis simultanément par machine	54
4.1	Filtrage collaboratif basé sur la mémoire	60
4.2	Le flux de travail du processus de recommandation	65
4.3	Stratégie de sélection des amis	68
4.4	Sélection des publications basée sur les notes des amis similaires	69
4.5	Agrégation des publications et alimentation du fil d'actualité	71
4.6	Le Rappel moyen	74
4.7	Niveau de couverture	75
4.8	Rappel moyen en fonction du pourcentage d'amis sélectionnés	76
4.9	Niveau de couverture en fonction du pourcentage d'amis sélectionnés	77
4.10	Le rappel moyen en fonction du pourcentage de publications sélectionnées par chaque ami	78
4.11	Niveau de couverture en fonction du pourcentage de publications sélectionnées par chaque ami	79

Remerciements

D'abord je tiens à exprimer mes plus vifs remerciements à mon directeur de thèse, le professeur Djamil AÏSSANI (*Université de Bejaia*) et à mon co-directeur de thèse, le professeur Mourad AMAD (*Université de Bouira*), pour les nombreux conseils, orientations et encouragements qu'ils ont su me prodiguer durant les années de la réalisation de cette thèse.

Je remercie la présidente et les membres du jury d'avoir accepté de juger et d'évaluer ce travail de thèse.

Je remercie le docteur Sofiane ABBAR et tous les chercheurs de *l'institut de recherche en informatique de Doha (Qatar)* pour leur accueil chaleureux et leurs commentaires constructifs lors de mes stages dans leur institut.

Mes remerciements s'adressent aussi à tous mes collègues enseignants (*département d'informatique, Université de Bouira*) qui m'ont encouragé durant toutes les années de la thèse.

Merci à tous ceux qui ont contribué à la réalisation de ce travail de près ou de loin.

Béjaia, Juin 2021

Lyes BADIS

Résumé

Résumé : Les sites de réseaux sociaux sont très performants et très répandus. Cependant, la possibilité de détourner les données des utilisateurs vers des utilisations en dehors des accords déclarés est de plus en plus préoccupante. La recherche est orientée vers les architectures peer-to-peer afin de fournir les mêmes services sans avoir besoin d'une autorité centrale de stockage et de contrôle. Plusieurs propositions adoptent ce type d'architecture et y installent les fonctions de base des réseaux sociaux, mais ces actions sont loin de devenir des concurrents des sites centralisés actuels. Cela est dû au manque de fonctionnalités, car la plupart d'entre elles ne fournissent que des opérations de base telles que la publication et les commentaires sans fonctions avancées telles que la recherche et les systèmes de recommandation. D'autre part, les architectures proposées souffrent de lacunes en matière de performances (*disponibilité et temps de réponse*), en raison de leur dépendance aux machines des utilisateurs comme alternative aux serveurs centraux. Cette thèse vise à améliorer ces architectures décentralisées où nous apportons deux contributions. Dans la première contribution, nous proposons un nouveau modèle de mise à jour des profils des utilisateurs stockés sur les machines des utilisateurs au moyen du journal des événements normalement utilisé dans les bases de données. Cette contribution vise à réduire le coût des opérations de mise à jour, car on n'enregistre à chaque fois que les derniers changements intervenus sur les profils des utilisateurs. Dans la deuxième contribution, nous introduisons P2PCF : un nouveau système de recommandation basé sur le filtrage collaboratif, une technique qui repose sur le stockage centralisé des interactions des utilisateurs, mais nous l'avons modifié pour correspondre à la nature décentralisée des réseaux sociaux pair à pair.

Mots clés : *réseaux sociaux, pair à pair, système de recommandation, P2PCF, journal des événements, mise à jour, cohérence, réplication.*

Abstract : Social networking sites are among the most successful and widespread sites, whether considering the number of users or the areas in which they are present. However, concerns are growing about the fate of user data and the possibility of diverting it to uses outside of the announced agreements. This is why research is geared towards peer-to-peer architectures in order to provide the same services without the need for a central authority to store data and control the entire system. Several proposals have been published that adopt this type of architecture and install basic functions of social networks, but these actions are still far from becoming competitors of current sites based on a central model. This is mainly due to the lack of available features, as most of them only provide basic operations like posting and commenting without advanced features like search tools and recommender systems. On the other hand, the proposed architectures suffer from performance shortcomings, such as availability and response speed, due to their reliance on user machines as an alternative to large central servers. This thesis aims to improve these decentralized architectures where we make two contributions. In the first contribution, we propose a new model for updating user profiles stored on user machines by means of the event log normally used in databases. This contribution aims to reduce the cost of updating operations, because only the latest changes to user profiles are recorded. In the second contribution, we introduce P2PCF : a new recommender system based on collaborative filtering, a technique that relies on centralized storage of user interactions, but we modified it to match the decentralized nature of peer-to-peer social networks.

Key Words : *social networks, peer to peer, recommendation system, P2PCF, event log, update, consistency, replication.*

Introduction Générale

LES réseaux sociaux sont des applications web populaires et largement utilisées. Cela est dû à leur facilité d'utilisation et aux fonctionnalités attrayantes qu'ils offrent aux utilisateurs. Bien qu'ils aient été lancés dans un cadre de divertissement, leur succès leur a permis d'occuper une place importante dans des domaines plus importants tels que la politique, l'économie, et les affaires. Ces sites sont construits sur un modèle économique offrant un service gratuit en échange d'une exploitation commerciale des données des utilisateurs, notamment dans les domaines du marketing, politique et sécurité. Ce modèle a soulevé de nombreuses interrogations sur les menaces de la vie privée des utilisateurs à travers toute utilisation abusive du fournisseur de service ou encore sa capacité à protéger ces données contre les fuites et le piratage. Pour surmonter ces inquiétudes, la politique de confidentialité est fréquemment ajustée afin de rassurer les utilisateurs. D'un autre côté, les organes législatifs de plusieurs pays se sont efforcés de modifier les lois afin de protéger les droits des utilisateurs de ces sites.

Dans le milieu académique, les chercheurs ont proposé des architectures innovantes qui offrent les fonctionnalités des réseaux sociaux, mais sans avoir besoin d'une entité centrale pour stocker et faire fonctionner le système. Ces efforts ont conduit à la publication de plusieurs travaux qui démontrent le potentiel de s'appuyer sur l'ingénierie peer-to-peer pour implémenter des réseaux sociaux.

En étudiant les différentes propositions, nous remarquons qu'elles représentent une preuve de concept, mais elles comportent plusieurs lacunes liées aux performances. Ceci est dû à l'utilisation des machines des utilisateurs pour remplacer les serveurs. Ces machines ont des caractéristiques limitées (*mémoire, puissance de calcul, bande passante*). En plus, il n'est pas facile de maintenir ces machines connectées d'une manière permanente. Les recherches continuent pour concevoir des schémas de stockage et de réplication plus performants.

En plus, les réseaux sociaux P2P évitent tout stockage central des informations sur les interactions des utilisateurs ou sur la structure du graphe social. C'est la raison principale pour laquelle les réseaux sociaux P2P manquent de plusieurs fonctionnalités intéressantes telles que des outils de recherche et de recommandation de contenu qui sont responsables de l'acquisition et de la croissance des utilisateurs. En effet, ces systèmes vont rarement au-delà des simples fonctionnalités de publication, de commentaire et de partage. L'enrichissement des architectures P2P par ce genre de fonctionnalités sans violer les contraintes de confidentialité est un défi.

Notre thèse entre dans le cadre des réseaux sociaux basés sur les architectures pair

à pair. Notre objectif est de contribuer à l'amélioration des performances de ces architectures. Nous avons proposé deux contributions : (i) un schéma de réplication des profils avec optimisation des opérations de mise à jour et d'équilibrage de charge, (ii) un système de recommandation décentralisé pour les réseaux sociaux pair à pair basé sur le filtrage collaboratif. La thèse est organisée comme suit :

Dans le premier chapitre, nous avons présenté le domaine des réseaux sociaux. On a rappelé l'évolution historique du concept ainsi que les fonctionnalités offertes par les réseaux sociaux actuels. On a cité aussi les différents domaines où les réseaux sociaux sont utilisés. Les classifications des réseaux sociaux sont aussi illustrées avant de discuter les avantages et les inconvénients.

Le deuxième chapitre est un état de l'art sur les réseaux sociaux décentralisés, essentiellement ceux basés sur les architectures "pair à pair". On a expliqué les origines de cette piste de recherche. Ensuite, nous avons décrit comment la décentralisation est utilisée pour implémenter un réseau social. Il s'agit de trois aspects principaux : le stockage, la protection et la communication. Nous avons critiqué ces architectures en précisant leurs avantages et leurs limites. On a terminé ce chapitre par les pistes de recherche possibles dans ce domaine.

Dans le troisième chapitre, nous avons présenté notre première contribution (*Mise à jour des profils répliqués dans les réseaux sociaux décentralisés-un modèle basé sur le journal des événements*). Nous avons concentré sur le schéma de stockage sur les machines des utilisateurs. Dans cette situation, la disponibilité des profils est liée à la situation de la machine qui le stocke. La réplication sur plusieurs machines est conseillée pour améliorer la disponibilité. Cette solution engendre la question de cohérence des différentes répliques qui doivent être mises à jour correctement. Nous avons proposé un modèle de mise à jour basé sur le journal des événements pour optimiser la charge réseau générée par le processus de synchronisation des répliques. En plus, nous avons présenté un mécanisme de recherche des profils et de routage vers les répliques qui tente d'équilibrer la charge entre les machines qui stockent une copie du même profil.

Dans le quatrième chapitre, nous visons à enrichir les fonctionnalités des réseaux sociaux pair à pair en proposant ainsi un nouveau système de recommandation décentralisé baptisé (*P2PCF*). Notre objectif est de générer des recommandations raisonnables pour les utilisateurs sans compromettre leurs confidentialités, c'est-à-dire sans avoir accès à l'intégralité des journaux d'interactions générées dans le système par tous les utilisateurs. P2PCF adopte la technique de filtrage collaboratif et génère des recommandations en trois étapes : Premièrement, un utilisateur sélectionne un sous-ensemble d'amis pertinents à partir desquels demander des recommandations. Deuxièmement, chaque ami exploiterait ses connaissances locales sur les préférences du demandeur pour produire une liste de recommandations qui sont renvoyées au demandeur. Enfin, l'utilisateur qui a initialisé la demande, agrège toutes les recommandations en fonction de différents paramètres liés au contenu reçu et aux utilisateurs qui l'ont partagé.

Enfin, notre thèse se termine par une conclusion générale qui résume les résultats atteints et les perspectives envisagées à long terme et à court terme.

GÉNÉRALITÉS SUR LES RÉSEAUX SOCIAUX

1.1 Introduction

Les réseaux sociaux sont de plus en plus présents dans notre quotidien. Grâce à leur performance et leur facilité d'utilisation, ils ont dépassé les médias et les outils de communication classiques. Ils se basent sur une théorie sociologique ancienne en exploitant l'avancement des technologies web. L'utilisation des réseaux sociaux a commencé par des usages récréatifs. Ensuite, elle s'est étendue à d'autres aspects de notre vie, tels que l'éducation, le marketing et la santé. Ces applications sont plutôt présentes dans tous les aspects de la vie. On va présenter dans ce chapitre une vue générale sur ces systèmes.

1.2 Historiques

Le phénomène des sites web de réseautage social est très récent et il date du début du troisième millénaire mais la notion du réseau social est plus ancienne. Il faut donc, faire la différence entre "réseau social" dans le domaine de la sociologie et les sites web qui offrent le service de réseautage social qui sont connus sous le nom de "réseaux sociaux numériques" ou "médias sociaux".

La notion du "réseau social" (*social network*) a été inventée par John A. Barnes pour décrire des ensembles d'individus et les relations qu'ils entretiennent les uns avec les autres, ce qui était jusque-là connu sous divers noms : structures, systèmes, cercles, groupes [27]. Le travail de Barnes a étudié la relation entre chaque individu et la société dans une petite île Norvégienne et il a montré que tous les individus étaient indirectement liés entre eux par une chaîne de relations de quatre maillons au maximum. Il a proposé que ce principe relie chaque individu avec le reste du pays et du monde entier grâce aux liens de parenté ou d'amitié.

Milgram, en 1967 qui a testé cette théorie dans le cadre de l'expérience dite «le phénomène du petit monde» [119], a cherché à évaluer le nombre moyen d'intermédiaire entre un individu et un autre au sein de la société américaine; aboutissant au chiffre de 5,2 voire 6 degrés de séparation.

Cette théorie a inspiré l'acteur américain *Kevin Bacon* pour concevoir un jeu qui se base sur la relation entre lui et les différents acteurs de *Hollywood*. Le principe de ce jeu est que *Bacon* est relié avec un degré de 1 avec tous les acteurs qui ont joué avec

lui ; les acteurs avec un degré 2 n'ont jamais travaillé avec lui mais ils ont joué avec un acteur d'ordre 1. Le défi du jeu est de trouver un chemin inférieur à 6 entre *Kevin Bacon* et n'importe quel autre acteur de cinéma. Le succès de ce jeu a conduit à la création d'une organisation de charité nommée «SixDegrees.org».

Cette théorie a ouvert la voie d'une longue série de travaux empiriques cherchant à valider ou invalider les hypothèses de *Barnes et Milgram*, travaux qui, dans les années suivantes, ont fait d'Internet leurs terrains d'expériences [62].

L'implémentation de ce concept était difficile avec les anciennes technologies web qui comportent des pages statiques rarement modifiées. Dans ce contexte (*web 0.1*), l'utilisateur est plus consommateur que producteur d'information avec la difficulté de gestion qui demandait l'intervention des experts (*programmeurs ou administrateurs*). Le web a ensuite évolué en introduisant la notion des sites dynamiques dont les pages sont chargées à la demande à partir des bases de données.

Arrivant au *web 2.0* qui a permis de transformer les utilisateurs en acteurs actifs. *Philippe Torloting* résume ainsi : «dans le *web 1.0* l'internaute était seulement passif ; à partir du *web 2.0* l'internaute devient actif : il crée son propre contenu, partage des informations, se crée un réseau, etc. ; en particulier via les médias sociaux dont les réseaux sociaux» [156].

Dans le *web 2.0*, les applications sont plus accessibles et disposent d'interfaces interactives permettant aux utilisateurs de produire, modifier et partager des informations qui sont par la suite enrichies par d'autres utilisateurs [128]. Dans ce contexte d'interaction et de collaboration, les réseaux sociaux numériques sont nés. On peut dire que les forums de discussion, les plateformes de blogs et les sites collaboratifs tel que *Wikipédia* constituent un premier pas dans cette tendance.

On peut considérer «Friendster» lancé en 2002 comme le premier réseau social au sens actuel du mot. C'était un site destiné aux jeux qui utilisait pour la première fois la notion de cercles et réseaux d'amis. Ensuite, d'autres plateformes ont été mises en service. On peut citer «MySpace» lancé en 2003 qui dépassa «Friendster» en avril 2004 en termes de nombre de pages affichées. Les réseaux sociaux professionnels tels que «LinkedIn» et «XING» ont vu le jour en 2003. En 2004, «Facebook» a été lancé pour servir à la communication entre les étudiants de l'université de Harvard mais son succès fait de lui le réseau social le plus populaire jusqu'à nos jours. Plusieurs réseaux de partage de contenu ont ensuite apparus : «Flickr» en 2003, «Youtube» en 2005 et «Slide Share» en 2006. *Google* a essayé de joindre ce monde en offrant 30 millions dollars pour acheter «Friendster » et en achetant «Youtube» en 2006 et en lançant son propre réseaux «Google+» en juin 2011.

L'invention des smartphones et le développement des applications mobiles a permis aussi de créer des réseaux sociaux pour l'environnement mobile. On peut citer : «WhatsApp» en 2009, «Instagram» en 2010 et «TikTok» en 2017.

1.3 Définitions

Un réseau social est un ensemble d'entités (*individus ou organisations*) connectées via plusieurs types de relations (*familiale, amitié, ou professionnelle*). Les définitions proposées aux réseaux sociaux d'une manière générale sont concentrées sur ces deux

caractéristiques : les entités et les liens qui les connectent.

Selon *Michel Forsé*, «un réseau social est un ensemble de relations entre un ensemble d'acteurs» [57].

Les relations sont de plusieurs types (*collaboration, soutien, amitié, contrôle, conseil, échange d'informations*) et les acteurs peuvent être des individus, des groupes ou des organisations [75, 105].

Lemieux Vincent précise que le réseau social est un ensemble qui peut être organisé (*une entreprise par exemple*) ou non (*comme un réseau d'amis*) et ses relations peuvent être de nature fort diverse (*pouvoir, échanges de cadeaux, etc.*), spécialisées ou non, symétriques ou non [108].

Une autre définition s'intéresse à l'utilité de ces structures : «C'est un moyen efficace pour les individus de bénéficier d'avantages et d'opportunités au travers des liens établis au sein du réseau social» [34].

Même si les définitions citées ci-dessus parlent du concept sociologique, elles sont cohérentes dans le contexte des réseaux sociaux numériques qui peuvent être définies comme une implémentation de la théorie des réseaux sociaux.

Boyd et Ellison décrit les sites des réseaux sociaux : «On définit les sites des réseaux sociaux comme des services web qui permettent aux individus de : (1) construire un profil public ou semi-public au sein d'un système délimité (2) articuler ce profil avec une liste des autres utilisateurs avec qui ils partagent une connexion (3) afficher et naviguer à travers leurs listes de connexions et celles des autres au sein du système» [32].

La définition suivante éclaire aussi bien le fonctionnement des relations dans les médias sociaux : «Un réseau social représente des entités et des connexions entre eux. Les entités sont généralement des individus connectés par des relations personnelles, des interactions et le suivi des activités de leurs contacts sur le réseau» [94].

D'autres essayent d'aller plus loin pour mettre en évidence la différence entre les réseaux sociaux avant et après internet. Ils essayent de définir les médias sociaux non seulement comme un ensemble d'entités reliées par des relations sociales mais en mettant la lumière sur les services échangés et les centres d'intérêts qui regroupent les utilisateurs.

On peut donner par exemple la définition de *Yahoo* qui appartient aux propriétaires des réseaux sociaux grâce à son service «Yahoo!360». Selon *Yahoo*, un réseau social est «un terme assez large qui désigne des sites Internet qui aident leurs utilisateurs à créer leur propre profil Internet et à partager une partie de leurs contenus préférés, y compris des photos et de la musique» [59].

Dupin [66] différencie les réseaux et médias sociaux de la façon suivante :

- Les réseaux sociaux reposent sur un lien social,
- Les médias sociaux reposent sur l'ensemble des sites proposant une interaction sociale.

«Dans le premier cas, c'est donc l'individu qui est au centre des échanges alors que pour le second, c'est l'ensemble des objets présents qui favorise l'interaction. Dans cette logique, les réseaux sociaux sont une partie des médias sociaux. Ils sont la plus pure représentation du terme "*social*", qui connote la relation entre différents individus et dont l'expression se centralise sur un profil utilisateur».

On peut dire que les médias sociaux ont permis de construire des réseaux sociaux plus larges, plus diversifiés et souvent plus concentrés autour des thèmes ou des centres d'intérêts bien définis. En plus, ces technologies ont permis aux utilisateurs de mieux bénéficier de la puissance des relations.

1.4 Fonctionnalités

Les réseaux sociaux se partagent les fonctionnalités de base mais chaque réseau développe un ou plusieurs services comme il peut bloquer définitivement une fonctionnalité donnée. On résume dans ce paragraphe les différentes fonctionnalités des réseaux sociaux.

1.4.1 Création de profil

C'est une étape indispensable que chaque utilisateur doit faire pour pouvoir participer dans les réseaux. Il s'agit de remplir une fiche d'identité contenant essentiellement :

- Un identifiant qui est en général une adresse mail ou un numéro téléphone.
- Le mot de passe qui sécurise l'accès au compte.
- Un pseudo nom qui représente l'utilisateur dans le réseau et qui peut être changé par la suite.
- Les informations personnelles tel que le nom, le prénom, et la date de naissance. La vérification de la fiabilité n'est pas exigée dans la plupart des médias sociaux.

On peut ajouter d'autres informations telles que la photo d'identité du profil, une image de couverture ou une citation introductive.

1.4.2 Publication

Une fois le profil créé, l'utilisateur peut contribuer dans les réseaux en publiant différents types d'information. Certains réseaux limitent la taille de la publication comme Twitter qui limite la publication à 140 caractères (*étendu en 2017 à 280 caractères*). D'autres n'acceptent qu'un seul type de données, qui est le cas des plateformes de partage : YouTube pour les vidéos, SlideShare pour les présentations, SoundCloud pour l'audio. Les réseaux généralistes tels que Facebook et Google+ ne mettent aucune restriction sur le contenu publié. Les publications peuvent être organisées sous différentes formes (*album de photo, playlist de vidéos, collection, ..*).

1.4.3 Navigation

L'utilisateur peut consulter le contenu présent dans le réseau selon différentes manières :

- Visualiser sa page personnelle qui ne contient que l'information qu'il a publié auparavant.
- Visualiser les pages des autres utilisateurs.
- Souvent les réseaux proposent ce qu'on appelle un fil d'actualité (*timeline*) qui contient un contenu varié extrait des pages des différents utilisateurs. Ce contenu est généré par un système de recommandation qui repose sur les intérêts et les interactions de l'utilisateur pour lui proposer un contenu approprié. Les systèmes de recommandation sont basés sur des algorithmes intelligents dont la notion d'apprentissage est nettement présente [167].

1.4.4 Interactions avec les autres utilisateurs

L'interactivité est une propriété de base dans les médias sociaux et dans les applications du web 2.0. L'utilisateur peut réagir aux publications des autres utilisateurs par :

- **l'appréciation** : il s'agit de donner un avis positif ou négatif sur un contenu. Ceci est réalisé par les boutons de type «like/dislike» (*le cas de YouTube ou Yahoo question/réponse*) ou par un vote (+1) comme le cas de Google+. Ces mentions d'appréciation ont un effet important sur la propagation du contenu dans le réseau.
- **Commentaire** : L'avis sur un contenu peut être exprimé par un commentaire textuel ou visuel (*fonctionnalité ajoutée par Facebook en juin 2013*).
- **Partage** : Le partage signifie que l'utilisateur souhaite de sa part publier le contenu provenant d'un autre membre. La publication partagée aura la même propriété d'une nouvelle publication : elle apparaît dans la page personnelle de l'utilisateur et elle peut être appréciée, commentée ou même partagée une nouvelle fois.

1.4.5 Connexion aux autres utilisateurs

Les liaisons entre les membres d'un réseau social peuvent prendre plusieurs formes :

- **Liens symétrique (*amitié*)** [31] : Dans ce type de liens, les deux parts ont le même rôle. Si le membre *Ali* envoie une demande au membre *Mohamed* et que *Mohamed* accepte cette invitation, *Ali* est l'ami de *Mohamed* et de même *Mohamed* est l'ami d'*Ali*.
- **Lien asymétrique (*abonnement*)** [31] : Cette liaison a un seul sens. Il s'agit de la fonctionnalité d'«abonner» ou «suivre» qui représente la relation éditeur/lecteur. Si par exemple *Ali* s'abonne pour suivre les publications de *Mohamed*, *Mohamed* n'est pas automatiquement abonné au compte d'*Ali*.

- **Groupes d'utilisateurs** : Plusieurs utilisateurs regroupés autour d'une thématique ou d'un centre d'intérêt bien défini. Les membres ne sont pas forcément des amis mais ils peuvent publier librement dans l'espace réservé au groupe. On peut différencier une sous-catégorie de membres dans le groupe : les administrateurs ou les modérateurs, ils disposent de quelques privilèges par rapport aux autres membres. Ils interviennent pour modifier le contenu et les paramètres du groupe comme ils peuvent décider de l'ajout ou de la suppression des membres.

1.4.6 Administration

L'utilisateur est le maître de son espace personnel. Il dispose de quelques autorisations pour la gestion de son contenu. En plus de la publication, l'utilisateur peut :

- Supprimer une publication ou un commentaire.
- Définir le public qui peut visualiser une publication donnée.
- Autoriser ou interdire un autre membre à interagir avec lui.
- Autoriser ou interdire les invitations d'amitié.

1.4.7 Discussion

La position de cette fonctionnalité diffère d'un réseau social à un autre. Il existe des réseaux qui ignorent ce service. D'autres mettent en place un service de messagerie entre les membres. La messagerie peut être statique ou instantanée, elle supporte ou non la communication audiovisuelle. Il est à noter aussi qu'il existe des médias sociaux dont la messagerie constitue la fonctionnalité de base.

1.4.8 Jeux

Les jeux en ligne est l'une des fonctionnalités les plus anciennes dans les réseaux sociaux avec Friendster. Aujourd'hui, il existe des réseaux dédiés uniquement aux jeux. Certains réseaux généralistes -tel que Facebook- intègrent des applications de jeux en ligne.

1.4.9 Diffusion en direct du contenu multimédia

En plus du partage des vidéos, il existe des réseaux sociaux qui offrent une fonctionnalité plus avancée. Il s'agit de la diffusion des vidéos en direct. On cite par exemple : *Meerkat*, *Periscope*, *Blab*. *Facebook* a intégré ce service en mars 2016.

1.5 Applications des réseaux sociaux

L'utilisation des réseaux sociaux ne s'est pas arrêtée à la communication et au divertissement, mais s'est étendue à de nombreux aspects. Les plus importantes de ces utilisations sont énumérées ci-dessous.

1.5.1 Communication

Les réseaux sociaux ont permis aux utilisateurs de rester en contact avec leurs proches ou amis éloignés. En plus, ils ont permis de communiquer avec des personnes célèbres qui sont difficiles à contacter dans la vie réelle. Cette application a explosé avec l'émergence des Smartphones et les applications mobiles. D'après [109, 153], les applications mobiles de messagerie instantané ont surpassé les appels vocaux, les e-mails et même la communication en face-à-face, pour devenir le moyen de communication le plus populaire chez les jeunes. Les statistiques montrent que les adultes américains passent deux fois plus de temps par jour sur les applications de messagerie mobile en 2019, par rapport à 2015 [69, 153].

1.5.2 Diffusion d'information

Les réseaux sociaux offrent au grand public un outil facile et puissant de diffusion d'information. Sans avoir besoin d'un grand budget ou un contact avec les médias classiques (*journaux, chaînes télévisés, sites web informatif*), les internautes peuvent publier leur avis ou leur talent en toute liberté et leurs publications se propagent ensuite dans les pages des amis et celles des amis des amis. Cette efficacité à créer un nouveau type de presse électronique. Ce sont les pages et les comptes des réseaux sociaux qui sont devenus un concurrent important aux anciens médias. Ces derniers n'ont pas tardé de constater la puissance des médias sociaux comme source d'information [170]. En plus, les institutions des médias utilisent les réseaux sociaux comme moyen de diffusion d'information à côté des canaux classiques (*papier, radio, Tv ou sites web*).

1.5.3 Réseautage d'affaires

Grâce aux liaisons qui existent entre les utilisateurs et la propagation rapide des informations, les réseaux sociaux permettent de conduire des affaires avec succès, que ça soit le Marketing [19], l'analyse des marchés [38, 98], ou même pour la gestion efficace des ressources humaines [29, 88, 106]. Ceci est réalisé par une présence indispensable des entreprises dans les réseaux sociaux populaires notamment les réseaux professionnels.

1.5.4 Recherche d'information

En plus des moteurs de recherche, l'utilisateur repose sur les médias sociaux pour rechercher un contenu ou une personne. Cette tendance a conduit vers l'apparition d'une nouvelle classe de systèmes de recherche d'information : Les systèmes de recherche d'information sociaux (*SIR social information retrieval*) [31]. Ces systèmes se basent sur la fonctionnalité d'indexation personnelle (*dit folksonomie*). Les utilisateurs joignent à leurs publications des marques «*Tag*». Ces marques proposées par les utilisateurs

constituent un index collaboratif qui peut servir aux outils de recherche de relier une requête avec un contenu adéquat. Cette technique a rendu la recherche plus efficace par rapport aux différents algorithmes d'indexation utilisés dans les systèmes de recherche d'information classiques. Les résultats de la recherche d'information sociale sont basés sur des propositions des utilisateurs. Cette collaboration entre les utilisateurs pour répondre à une requête d'un autre utilisateur peut être observée dans les réseaux sociaux basés sur les questions et les réponses : les utilisateurs répondent et notent les autres réponses pour indiquer au chercheur de l'information les réponses les plus pertinentes.

1.5.5 Marketing

Pour promouvoir leurs produits et leurs services, les entreprises donnent plus d'importance à la publicité dans les réseaux sociaux. Selon une étude emarketer, marketingland, et recode, «70% des annonceurs ont augmenté leur budget publicitaire sur les médias sociaux en 2015» [48]. Grâce à l'analyse des données utilisateurs, les réseaux sociaux offrent aux entreprises un moyen plus efficace pour lancer des campagnes ciblées. Les entreprises proposent leur produit ou leur service à une population bien définie (*distribution géographique, age, niveau d'études, genre, moyens financière, ...*). En plus, l'évaluation des actions publicitaires (*compagnes, promotions, ...*) est plus facile en considérant les réactions des utilisateurs [110]. Ceci peut être réalisé par l'analyse des conversations des utilisateurs pour comprendre comment les consommateurs voient une entreprise ou ses actions [141].

1.5.6 Gestion des catastrophes naturelles

L'attention a été attirée sur l'importance de publier en temps réel des informations et des images sur les accidents et les catastrophes naturelles par les utilisateurs des réseaux sociaux. Ces informations peuvent être très utiles aux services de secours qui interviennent pour aider les gens et réduire les effets de ces accidents. L'avantage de ces informations est qu'elles sont généralement fournies par des personnes qui se trouvent à proximité des lieux de l'événement, et elles sont rapidement publiées avant même que l'aide n'arrive ou que les nouvelles n'atteignent les médias traditionnels. On peut citer, par exemple, *AIDR (Artificial Intelligence for Digital Response)* [91] qui est une plateforme pour filtrer et classer les messages des réseaux sociaux liés aux urgences, aux catastrophes et aux crises humanitaires. *AIDR* utilise l'intelligence humaine et machine pour baliser automatiquement jusqu'à des milliers de messages par minute. Ce système, supporté par l'*ONU*, a été utilisé durant plusieurs catastrophes naturelles dans plusieurs pays.

1.5.7 Analyse des réseaux sociaux

L'analyse des réseaux sociaux est un nouveau domaine de recherches qui visent à analyser des tendances sociologiques, économiques, ou politiques à partir des données publiées par les utilisateurs. Les opinions des citoyens sur les événements d'actualité, les sentiments de colère ou de bonheur peuvent être déduits à partir des publications et des commentaires [40]. Des techniques d'analyse du discours sont aussi développées pour détecter la croissance du discours de haine et de discrimination raciale [72, 116]. En plus

de l'analyse et de la détection de ces phénomènes, les études prédictives peuvent prévoir des phénomènes avant leur arrivée. On peut par exemple, détecter des cas susceptibles de devenir déprimés ou de tenter de se suicider [137]. On peut même détecter la possibilité d'infection par certaines maladies organiques [11]. L'analyse des réseaux sociaux se base sur plusieurs techniques [35, 117] : statistique, linguistique, des techniques de fouille de données, des algorithmes intelligents basés sur l'apprentissage automatique.

1.6 Types de réseaux sociaux

Même si chaque réseau social possède ces propres caractéristiques, il existe une intersection entre plusieurs réseaux ce qui a poussé les chercheurs à définir des classes des réseaux sociaux. Ces travaux ont résulté plusieurs approches basées sur différents critères (*usage, objectifs,...*). On résume ci- dessous les classifications les plus rencontrés dans la littérature.

1.6.1 Réseaux généralistes/ Réseaux spécialisés

La classification la plus présente dans la littérature considère deux grandes catégories des réseaux sociaux. Les réseaux généralistes et les réseaux spécialisés

1.6.1.1 Les réseaux généralistes

Ils sont les réseaux populaires dont on peut trouver la plupart des fonctionnalités des réseaux sociaux. On peut citer par exemple : Facebook et MySpace qui sont basés sur plusieurs utilisateurs connectés par des liaisons d'amitié et échangeant plusieurs types de données.

1.6.1.2 Les réseaux spécialisés

Ils sont des réseaux concentrés sur une thématique bien définie. D'où on trouve plusieurs sous-catégories :

- **Les réseaux communautaires** : Ils regroupent des utilisateurs partageant un même centre d'intérêt autour d'une thématique bien définie comme c'est le cas des ventes aux enchères sur *eBay* ou encore la mise à disposition des internautes de différents et divers registres musicaux sur *Boompa*.
- **Les réseaux professionnels** : Ce sont des réseaux permettant à leurs membres de se valoriser sur le marché de l'emploi en créant leurs curriculum vitae en ligne, de chercher des opportunités d'emploi ou des opportunités de partenariat comme c'est le cas par exemple de Viadeo, LinkedIn, ou 6nergies.
- **Les réseaux académiques** : Ce sont des plateformes destinées aux chercheurs (*ResearchGate, Academia.edu, Mendelej,...*). Le contenu partagé est limité aux publications scientifiques et aux projets de recherche. Ils constituent aussi un bon forum de questions/réponse entre spécialistes. Les membres sont reliés par des relations classiques (*suivi, abonnement*) ou par des liens spécifiques au contexte

académique (*citations, co-crédation*).

- **Les réseaux de partage de contenus** : Des réseaux sociaux orientés vers le partage d'un certain type de contenu multimédia (vidéos, son) tels que YouTube, Dailymotion ou Sound Cloud. Ces réseaux permettent la publication et le partage des contenus, l'échange autour de ces contenus sous forme de commentaires mais ne proposent pas de fonctionnalité de mise en relation.

1.6.2 Réseaux de relations /réseaux de contenu

Dans cette classification [36, 107], les réseaux sociaux sont observés selon la fonctionnalité la plus importante : les relations entre individus ou le partage de contenu.

Les réseaux structurés autour des individus et les relations qui les connectent. Ces relations peuvent être de plusieurs types mais l'essentiel est que le plus important dans ces réseaux ce sont les liens d'amitié ou d'abonnement. On peut citer par exemple : Facebook ou LinkedIn.

L'autre classe contient des réseaux centrés autour des contenus partagés. La navigation des profils est réalisée beaucoup plus suivant les données qu'il partage même s'ils permettent de visualiser une page ou une chaîne d'un utilisateur bien déterminé.

Ziryeb Marouf [114] opte pour cette classification (*ils nomment les deux classes : «profil centric» et «content centric»*) en ajoutant une troisième catégorie : «*Collaborative Centric*» qui regroupe les plateformes collaboratives qui proposent les mêmes fonctionnalités qu'un groupware d'entreprise tel que le partage des documents, les agendas partagés, les forums de discussions. Il s'agit ici d'un groupware grand public accessible par un grand nombre d'internautes tel que Google Agenda.

1.6.3 Classification à base des objectifs

Thelwall [155] catégorise les réseaux sociaux selon leurs trois objectifs : socialisation, réseautage et navigation :

- **Les réseaux sociaux de socialisation** : L'objectif de ces réseaux est la communication sociale entre les utilisateurs. Ils permettent de rechercher et d'afficher les amis et de se connecter à de nouveaux amis. MySpace et Facebook sont à la tête de cette catégorie.
- **Les réseaux sociaux de réseautage** : Ils permettent de naviguer pour trouver de nouveaux contacts et entrer en connexion avec des personnes inconnues auparavant comme c'est le cas de LinkedIn ou Viadeo.
- **Les réseaux sociaux de navigation** : Ce sont des sites de partage de liens Internet (connus sous le nom de social bookmarking). Leur objectif est de faciliter la recherche et l'accès à l'information. Les membres partagent des liens ou des informations. Chaque utilisateur reçoit une page principale contenant un extrait des partages des membres avec les mentions d'appréciations des utilisateurs qui

peuvent servir comme indice d'importance. L'utilisateur peut également consulter les contributions d'un membre donné. C'est le cas de Digg ou Delicio.us.

TheWall différencie également les sites pour lesquels les fonctionnalités de réseaux sociaux sont principales (*de type Facebook et LinkedIn*) ou secondaires (*Youtube, Flickr, Deezer*).

1.6.4 Classification basée sur l'usage

En février 2012, *Frédéric Cavazza* propose le graphique suivant (*Figure 1.1 [39]*) qui résume sa vision à la typologie des médias sociaux basée sur l'usage :



FIGURE 1.1 – Panorama des médias sociaux

- D'après Cavazza [39], les médias sociaux peuvent être répartis en dix catégories :
- **Forum** : Un espace de discussion public où les messages sont affichés par ordre chronologique. La consultation est libre, mais l'inscription est obligatoire pour pouvoir répondre. La modération des discussions se fait à priori ou a posteriori. Exemples de gros forums français : Doctissimo, Forum-auto, Cyberbricoleur, MagicMaman, Comment ça marche, etc. Exemples de plateformes de forum : PHPbb, Phorum, bbPress, etc.

- **Blogue** : Un outil de publication simplifié où les articles sont affichés par ordre chronologique et triés dans des catégories. Les lecteurs peuvent déposer des commentaires qui sont modérés à postériori. Le flux RSS permet de facilement exporter le contenu vers des agrégateurs et des lecteurs. Exemples de plateformes de blogues : Blogger, WordPress, Typepad, etc.
- **BlogueWiki** : Une base de connaissance en ligne où les internautes rédigent et corrigent eux-mêmes le contenu. Les wikis sont constitués d'un ensemble de pages sans système de navigation cohérent. Chaque page dispose d'un historique des modifications et peut être commentée. La modération est assurée par des équipes organisées de façon pyramidale. Exemple de wikis célèbres : Wikipedia, Wookipedia, Brickipedia, etc. Exemples de plateformes de wiki : MediaWiki, Wikia, Wetpaint, etc.
- **Service de partage** : Service en ligne où les internautes peuvent publier des photos, vidéos, liens. Chaque élément publié est rattaché à un membre et peut être commenté et noté. La communauté ou les annonceurs peuvent créer des chaînes et des groupes pour fédérer des micro-communautés. Exemples : YouTube, Flickr, Delicious, Deezer, Slideshare, etc.
- **Réseau social** : Site à l'accès restreint où chaque utilisateur possède un profil. Les membres sont liés de façon bilatérale ou au travers de groupes. Certains réseaux proposent également des fonctionnalités plus sophistiquées (messagerie, publication, partage de contenus, etc.) ainsi que la possibilité d'héberger des applications tierces (*plateforme*). Exemples : Facebook, Orkut, Friendster, Tagged, etc.
- **Microblogue** : Service de publication, de partage et de discussion reposant sur des messages très courts. La consultation des messages et profils ne requiert pas d'inscription et peut se faire sur le web, les terminaux mobiles ou au travers d'applications. Chaque membre possède un profil public où sont listés les derniers messages. Les membres peuvent s'abonner aux profils des autres pour recevoir leurs messages dans un flux unique. Exemples : Twitter, Google Buzz, etc.
- **Agrégateur** : Service en ligne permettant de regrouper l'ensemble des publications d'un utilisateur des médias sociaux (*social stream*). De très nombreuses formes de contributions sont acceptées (*RSS, photos, vidéos, liens, email, etc.*). Les utilisateurs peuvent s'abonner aux flux des autres membres. Exemples : Posterous, FriendFeed, etc.
- **FAQ collaborative** : Service en ligne d'entraide où les questions et les réponses sont publiées par les utilisateurs. Les réponses sont commentées et notées, le membre qui a publié la question sélectionne la réponse la plus satisfaisante afin de clôturer les échanges et récompenser l'auteur avec un système de points. Exemples : Quora, StackOverflow, etc.
- **Jeux sociaux** : Jeux en ligne reposant sur une plateforme sociale exploitant

les profils des membres pour proposer différentes interactions sociales entre les joueurs (*tableau publics des meilleurs scores, système d'invitation et de défis, objectifs ne pouvant être réalisés en solo, etc.*). Exemples : Farmville, Mafia Wars, Texas HoldEm Poker ...

- **Service de géolocalisation** : Applications permettant de publier, partager et discuter sur des terminaux mobiles. Les articles ou photos publiés sont rattachés à un lieu afin de leur donner un contexte géographique. Chaque membre dispose d'un profil où sont listées ses dernières publications ainsi que les lieux qu'il a visités. Chaque lieu dispose également d'une page où sont listés les membres qui s'y sont signalés (*check-in*). Exemples : Foursquare, Facebook Places, Gowalla, etc.

1.6.5 Classification de Delcroix

Une autre classification proposée par *Eric Delcroix* [59], consultant, spécialiste et expert en communication web, web2.0, réseaux sociaux. Il détermine 07 catégories.

- **Réseau d'Affaires et d'Emplois** : Viadeo, LinkedIn.
- **Réseau Communautaire et Thématique** : Wikkio, Scoopeo.
- **Réseau de Jeunes, «bloglike»** : MySpace, Skyrock.
- **Réseau «Identité Numérique»** : Ziki, MyBlogLog.
- **Réseau «Micro»** (*micro-blogging, micro-vidéo, etc.*) : Twitter, Tumblr.
- **Réseau «Privé»** : sur invitation.
- **Réseau Spécialisé** (*vidéo, images...*) : Dailymotion, YouTube, Flickr.

1.7 Avantages et Inconvénients

On peut résumer les aspects qui font que les réseaux sociaux numériques sont plus performants que les sites web classiques dans les points suivants :

- **Omniprésence** : Grâce à la notion des notifications l'utilisateur est informé en temps réel de ce qui se passe dans son réseau. Cette caractéristique est plus claire lorsque l'utilisateur accède au service via un appareil mobile.
- **Accès l'information** : Grâce aux réseaux sociaux, l'information circule d'une manière très rapide et très souple. Il suffit d'être abonné à une page ou dans un groupe de discussion ou tout simplement suivre un amis pour recevoir les informations qui vous intéressent dans un temps très réduit.

- **Facilité d'utilisation et d'administration** : Les réseaux sociaux ont permis aux grands publics de publier différents formats d'information sans avoir besoin de payer un hébergeur ou faire appel à un spécialiste en informatique.
- **Liberté d'expression** : Les réseaux sociaux présentent un milieu idéal des échanges de pensées loin de tout type de contrôle ou de censure. Ils ont permis aux internautes de s'exprimer et de discuter leurs problèmes et de proposer des solutions qui ont été souvent mises en action. A titre d'illustration les révolutions arabes connues sous le nom de «printemps arabe» est un résultat de plusieurs débats et campagnes dans les réseaux sociaux notamment sur Facebook et Twitter [73].
Par contre, l'expansion des médias sociaux a déclenché plusieurs études socio-économiques ou même psychologiques pour mettre le point sur les effets de l'utilisation des médias sociaux. On cite quelques inconvénients dus à la mauvaise utilisation des médias sociaux :
- **Une vie virtuelle** : Les soupers entre amis ou les rencontres familiales sont souvent altérés par les gens qui sont connectés à leurs comptes des médias sociaux. Il est donc très facile de vivre dans le virtuel et de ne pas vivre tout à fait le moment présent.
- **Crédibilité des informations** : Il est facile de publier et l'information se propage rapidement mais rien n'empêche de publier des informations erronées ou des avis dans des domaines dont le rédacteur n'est pas expert. Il se peut alors que l'information que nous trouvons sur les réseaux sociaux ne soit pas exacte (*fake news*). Les fournisseurs des médias sociaux ont créé des équipes pour veiller et superviser le contenu informatif en associant des étiquettes aux publications contenant des informations fausses ou non confirmées [73]. En plus, Plusieurs initiatives sont lancées pour détecter et limiter la propagation des rumeurs [1–3].
- **Diminution de productivité** : Les médias sociaux nous incitent à être de plus en plus multitâches. Nous regardons souvent notre compte Facebook alors que nous sommes en train de travailler sur autre chose. On pourrait croire que cela nous permettrait d'accomplir deux choses en même temps, mais au contraire, notre concentration diminue et notre rythme de productivité baisse. Des études récentes ont analysé ce problème [133, 166].
- **Réputation professionnelle** : Employeurs et chasseurs de têtes utilisent de plus en plus les réseaux sociaux pour dénicher des candidats. Mais aussi pour vérifier leur profil avant un entretien. Un sondage publié sur le site de recrutement en ligne CareerBuilder révélait que des employeurs qui consultent Facebook et compagnie ont déjà écarté des candidats à cause du contenu de leurs pages personnelles [37].
- **Collecte et exploitation des données** : Les fournisseurs des réseaux sociaux collectent et stockent les informations des utilisateurs dans de grands centres

de données. Ces données ne se limitent pas aux informations personnelles que l'utilisateur a saisi lors de l'ouverture de son compte et ses publications et interactions dans le réseau, mais elles englobent toutes les localisations géographiques à travers lesquelles il a utilisé le réseau, ainsi que toutes les dates d'entrée et de sortie, en plus des noms de contact sur l'appareil qu'il utilise, comme le téléphone, par exemple. Ces données sont par la suite analysées pour construire un profil utilisateur. Ce profil est utilisé pour l'amélioration de l'expérience des utilisateurs en proposant des contenus et des relations adéquates à leur intérêt. Il est aussi exploité par les entreprises pour une conduite efficace du marketing ciblé et l'analyse des marchés. Ceci constitue la source principale des revenus des réseaux sociaux. D'une autre part, rien n'assure que ces données sont protégées contre un piratage ou une utilisation illégitime. Un exemple d'une telle utilisation qui peut être qualifié comme malicieuse est l'affaire Cambridge Analytica [139] : En 2016, la campagne électorale de l'ancien président américain Trump a exploité des données de Facebook pour influencer la décision des électeurs. Plusieurs affaires ont été traduites en justice pour empêcher les entreprises des médias sociaux d'utiliser abusivement les données des utilisateurs [45, 124]. En plus, Les gouvernements ont commencé à promulguer des lois réglementant la collecte et l'exploitation des données des utilisateurs. Par exemple, les développeurs d'applications sont obligés de demander l'autorisation pour collecter les données des utilisateurs de l'Union Européen selon la loi GDPR (General Data Protection Regulation) [127]. Techniquement, les architectures décentralisées sont proposées comme alternative pour éviter un stockage centralisé des données.

- **Intimidation et harcèlement en ligne** : Le cyber harcèlement sur les réseaux sociaux est un problème qu'il ne faut pas prendre à la légère, il peut avoir de graves conséquences. En effet, le cyber harcèlement peut revêtir plusieurs formes comme : la création de faux profils, l'usurpation d'identité, la diffusion de rumeurs infondées ou encore l'envoi de messages d'insultes. Ces agressions répétées sur le long terme peuvent prendre des proportions importantes et impacter directement la vie des victimes. De plus, ces messages, photos et vidéos publiées et échangées via les canaux numériques à grande échelle, laissent des traces même après que le harcèlement cesse. Les adolescents sont tout particulièrement touchés par ce phénomène. La détection ou la prédiction de ce phénomène est un problème de recherche récent dans les réseaux sociaux.

1.8 Exemples de réseaux sociaux

On expose dans ces paragraphes des réseaux sociaux populaires de différentes catégories :

1.8.1 Facebook

Facebook est un réseau social généraliste. Ils supportent toutes les fonctionnalités citées ci-dessus. Facebook est né en 2004 à l'université Harvard ; d'abord réservé aux étudiants de cette université. Il s'est ensuite ouvert à d'autres universités américaines avant de devenir accessible à tous en septembre 2006. En décembre 2015, il compte 1,04 milliard d'utilisateurs actifs quotidiens sur un total de 1,59 milliard d'utilisateurs actifs mensuels [4].

1.8.2 LinkedIn

Un réseau social professionnel en ligne créé en 2003 à Mountain View (*Californie*). LinkedIn est principalement utilisé par ses membres pour trouver du travail, des clients, des fournisseurs et développer ses affaires. Il fonctionne sur le principe de la connexion, ce qui signifie que pour entrer en contact avec un professionnel, vous ne pouvez le faire que par l'entremise de votre réseau. Si ce professionnel n'est connu par aucun membre de votre réseau, il vous sera impossible de le rejoindre, ou de lui envoyer un message. LinkedIn divise les connexions en trois niveaux distincts :

- Les contacts directs (*le premier degré*)
- Les contacts de notre réseau (*le deuxième degré*)
- Les contacts de nos contacts du deuxième degré (*le troisième degré*)

1.8.3 Twitter

Twitter est un outil de micro-blogging géré par l'entreprise Twitter Inc. Il permet à un utilisateur d'envoyer gratuitement de brefs messages, appelés tweets, sur internet, par messagerie instantanée ou par SMS. Ces messages sont limités à 140 caractères. Les utilisateurs peuvent apprécier, commenter ou partager (*retweet*) les publications. Les liaisons entre les utilisateurs sont basés sur le principe d'abonnement. Twitter a été créé le 21 mars 2006 par *Jack Dorsey, Evan Williams, Biz Stone et Noah Glass*, et lancé en juillet de la même année. Le service est rapidement devenu populaire, jusqu'à réunir plus de 500 millions d'utilisateurs dans le monde fin février 2012 [95]. Twitter compte 320 millions d'utilisateurs actifs par mois en avril 2016 [5].

1.8.4 YouTube

YouTube est un site web d'hébergement de vidéos sur lequel les utilisateurs peuvent envoyer, évaluer, regarder, commenter et partager des vidéos. Il a été créé en février 2005 par *Steve Chen, Chad Hurley et Jawed Karim*, trois anciens employés de *PayPal* et racheté par Google en octobre 2006 pour la somme de 1,54 milliard de dollars. En Avril 2016, YouTube était classé en deuxième position dans la liste des sites les plus visités [10].

La figure 1.2 [148] illustre les réseaux sociaux les plus populaires dans le monde, classés par nombre d'utilisateurs actifs (*janvier 2021*).

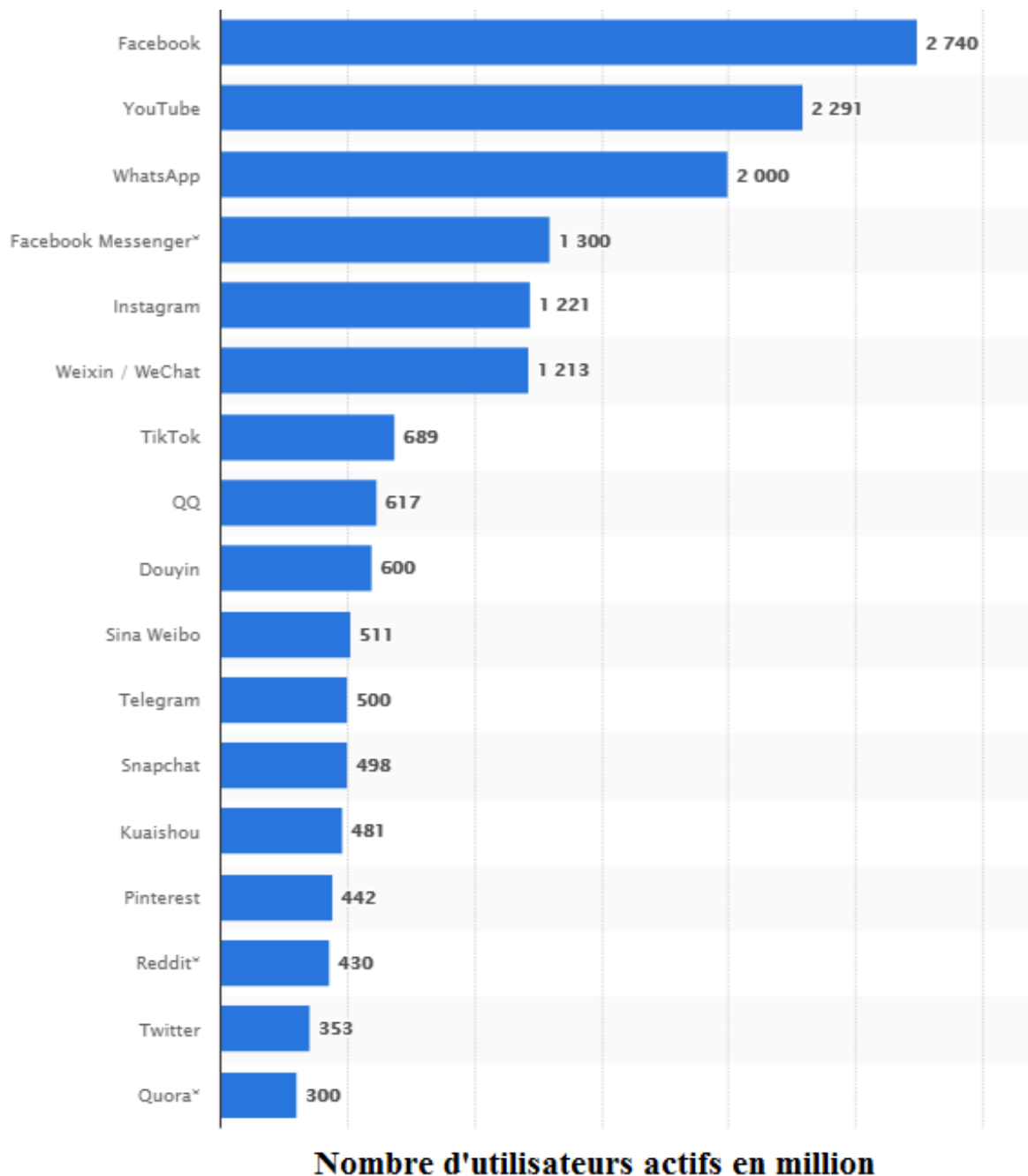


FIGURE 1.2 – Les réseaux sociaux les plus populaires

1.9 Conclusion

LES réseaux sociaux ont changé notre vie et ont facilité l'accès à l'information pour tous. Ils sont actuellement au centre des différentes activités humaines : éducation, santé, commerce, etc. Malgré leur force, leur attractivité et leur efficacité dans la gestion de ces activités, les préoccupations grandissent quant à la mauvaise utilisation et à la sur utilisation. D'autre part, la collecte et l'analyse des données ainsi leur partage avec

des tiers a provoqué l'émergence d'un grand débat sur la protection de la vie privée des utilisateurs. Toutes ces préoccupations nécessitent une large sensibilisation pour rationaliser l'utilisation, d'une part, et le développement d'un système juridique qui protège les utilisateurs, d'une autre part. En plus, la recherche continue pour développer de nouvelles architectures qui vont au-delà des inconvénients des architectures actuelles qui dépendent d'un point de contrôle central. Le chapitre 2 présente un état de l'art sur les réseaux sociaux décentralisés.

ÉTAT DE L'ART SUR LES RÉSEAUX SOCIAUX DÉCENTRALISÉS

2.1 Introduction

L'ÉMERGENCE de la quantité de données publiées sur les réseaux sociaux offre une grande source de richesse pour les propriétaires mais constitue une source de menace de la vie privée des utilisateurs. Dans les architectures actuelles des réseaux sociaux, les fournisseurs ont accès à toutes les données des utilisateurs et ils contrôlent et observent leurs relations et leurs comportements.

Les réseaux sociaux décentralisés sont des réseaux sociaux fondés sur une architecture qui essaye de libérer les données des utilisateurs du contrôle des propriétaires des réseaux sociaux. Cette tendance a comme objectif principal de protéger la vie privée des utilisateurs contre les différentes menaces y compris la mauvaise utilisation des fournisseurs des réseaux sociaux SNP (*Social network provider*). Dans ce chapitre, on présente les fondements de cette approche.

2.2 Motivation

Les réseaux sociaux sont des outils très puissants de communication et de partage d'informations. Les données des utilisateurs sont stockées sur des serveurs centraux. Cela a permis aux fournisseurs de ces services d'assurer des performances élevées (*disponibilité, temps de réponse, et espace de stockage*) et de développer des fonctionnalités très avancées (*recherche d'informations sociales [31, 93], systèmes de recommandation [167]*).

Le premier problème avec ce modèle de stockage centralisé est l'intérêt des fournisseurs pour la monétisation des données des utilisateurs (*données de profil, traces de communication, tout le contenu téléchargé et toutes les traces d'interaction [115]*). Ces données sont mises à la disposition des tiers pour développer et connecter leurs applications sur le site [53] ou pour les analyser et les utiliser à des fins diverses : économique, sociologique, politique, etc.

En fait, les utilisateurs perdent le contrôle de leurs propres informations une fois qu'ils les publient sur un réseau social. En outre, les publications ou amis suggérés par les systèmes de recommandation montrent que les fournisseurs peuvent en savoir sur les utilisateurs plus qu'ils ne l'envisagent. Toutes les données et relations sont gérées

par un fournisseur de services, ce qui menace la confidentialité de l'utilisateur [159].

Ces préoccupations ont conduit à une polémique qui a incité les fournisseurs à modifier les politiques de confidentialité. Cependant, il est peu probable que les fournisseurs agissent en faveur des utilisateurs [173]. Cela les obligerait à renoncer à l'accès aux données des utilisateurs, et une telle concession reviendrait à renoncer à un certain nombre d'avantages économiques [101].

Un deuxième problème des architectures centralisées est la capacité des systèmes de supporter la montée en charge. Cinq sources de ce problème ont été identifiées [113,115] : (i) Grand nombre d'utilisateurs hautement connectés, (ii) Problèmes d'infrastructure, (iii) Trafic réseau interne, (iv) Gestion du contenu généré par les utilisateurs, (V) Évolutivité de la base de données.

Les grands fournisseurs qui disposent des moyens financiers nécessaires ont affronté ces problèmes par le renforcement de leur infrastructure et le développement de nouvelles technologies de stockage (*Cassandra* [104] et *Haystack* [28] de *Facebook*, *Bigtable* [41] et *Megastore* [26] de *Google*).

Le dernier problème qu'on peut observer est la dépendance à une entité centrale qui impose ses règles de censure. Elle peut à tout moment changer ses règles de monétisation [120] ou demander des frais d'utilisation d'un service supposé gratuit [101]. Le fournisseur peut même décider d'arrêter le service définitivement [9, 140].

Les réseaux sociaux décentralisés (*DOSN*) constituent une alternative pour remédier à ces problèmes. Le concept principal est d'éviter le stockage des données des utilisateurs sur des serveurs non fiables. Le choix évident est l'utilisation des architectures «pair à pair» à cause de la nature collaborative des réseaux sociaux, car les utilisateurs d'un réseau social génèrent du contenu pour d'autres utilisateurs (*par exemple, leurs amis ou abonnés*) et non pas pour tout tiers comme un fournisseur [101]. En plus, les problèmes de confidentialité et de montée en charge peuvent être réglés par l'exploitation des ressources de chaque utilisateur (*espace de stockage, bande passante forfaitaire et inutilisée, cycles de calcul*) [115] et par des politiques de sécurité basées sur le chiffrement et le contrôle d'accès [130].

2.3 Architecture décentralisée pour les réseaux sociaux

Les réseaux sociaux décentralisés sont des systèmes proposés pour fournir les mêmes fonctionnalités du réseau social centralisé en éliminant ou minimisant le rôle d'une entité centrale. Ils sont proposés pour garantir aux utilisateurs plus de contrôle sur leurs données. L'architecture comporte plusieurs composants dont la décentralisation peuvent être introduite : (i) le stockage des données, (ii) la protection des données, (iii) la communication et (iv) les fonctionnalités. La figure 2.1 montre une architecture type d'un réseau social décentralisé.

L'architecture doit définir les schémas de stockage, de protection et de communication. Ensuite, il faut implémenter les différentes fonctionnalités d'un réseau social. Ces fonctionnalités peuvent être basiques comme la publication et la messagerie ou avancées telles que la recommandation et la recherche. Une interface utilisateur doit être implémentée pour permettre à l'utilisateur d'utiliser les fonctionnalités, de manipuler ces informations sociales ou même d'administrer le niveau bas : modifier les paramètres

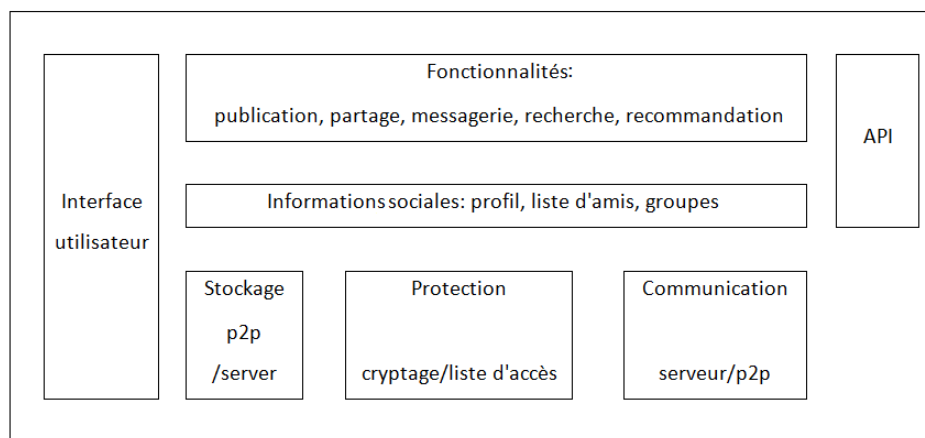


FIGURE 2.1 – Architecture type d'un réseau social décentralisé

de stockage ou de sécurité. Une API est une option qui permet de brancher d'autres systèmes (*exemple, d'autres réseaux sociaux*) ou de développer des applications au-dessus du système.

On va expliquer ci-dessous, comment les architectures proposées dans la littérature ont décentralisé les trois aspects : stockage, protection et communication.

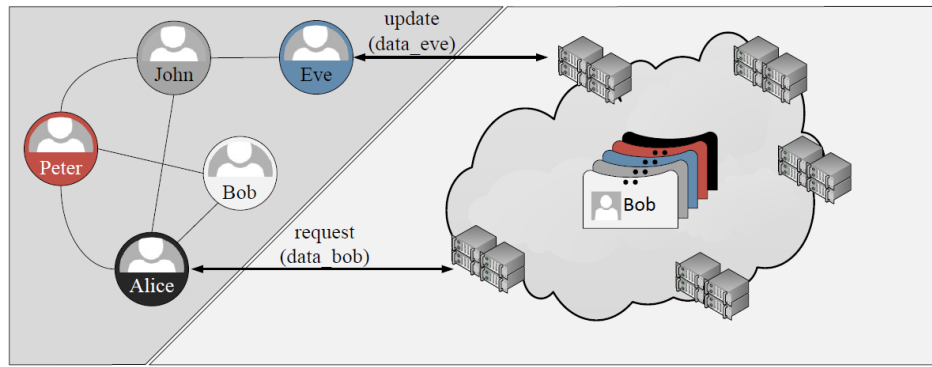
2.3.1 Gestion du stockage

Pour donner plus de contrôle sur les données, des schémas de stockage décentralisés sont proposés. Les utilisateurs peuvent stocker localement leur données ou sur des machines dont ils font confiance : c'est le principe du pair à pair ou chaque nœud dans le réseau est à la fois serveur et client. Le stockage sur des serveurs est souvent utilisé avec plus de précaution : crypter les données avant de les stocker ; donner à l'utilisateur le choix de son serveur ; stocker des données non sensibles seulement sur des serveurs. On peut donc différencier trois pistes pour le stockage décentralisé : pair à pair, serveur, hybride.

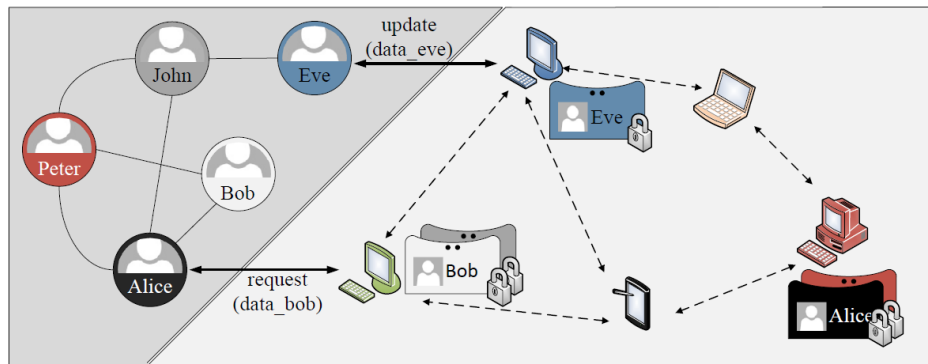
2.3.1.1 Stockage sur les machines des utilisateurs (*pair à pair*)

L'élimination d'une entité de stockage centrale nécessite de compenser ses ressources par des ressources des utilisateurs qui souhaitent utiliser le service. Chaque utilisateur doit contribuer dans le système par un espace de stockage pour héberger l'application et les données. L'idée triviale est que la machine de chaque utilisateur joue le rôle d'un serveur en stockant les données qu'il souhaite partager ainsi que son graphe social (*liste d'amis, abonnement, ..*). La figure 2.2 [101] décrit le stockage dans ce schéma, les données des utilisateurs sont stockées dans leurs propres machines.

Une machine peut stocker les données d'un ou de plusieurs utilisateurs (*les données de Bob et John sont stockées dans la même machine*). Ce schéma implique plusieurs puretés : les machines des utilisateurs doivent être puissantes pour servir une montée de requêtes et doivent être connectées en permanence pour que les données soient disponibles.



a- Réseau social centralisé



b- Réseau social décentralisé

FIGURE 2.2 – Stockage sur les machines des utilisateurs

Plusieurs schémas de stockage et de réplification issus de ce schéma initial sont proposés. Selon la stratégie de réplification des données, trois schémas de stockage pair à pair sont identifiés [101] :

- **a- Stockage sur le réseau overlay** : Dans cette approche, la couche de communication pair à pair (*en général basés sur les tables de hachage distribuées DHT*) est utilisée comme un support de stockage. Cette architecture auto organisée englobe un protocole déterministe pour le placement et la recherche des données [17]. En plus, certains DHT fournissent un mécanisme de réplification intégré ce qui améliore la disponibilité [101]. Parmi les architectures qui utilisent cette technique, on cite : Prometheus [102], LifeSocial.KOM [78], DECENT [92], Cachet [126], HorNet [90], Megaphone [132].
- **b- Stockage sur les machines des amis** : Dans cette approche, l'utilisateur choisit un sous ensemble de nœuds pour stocker ses données. Ce choix est basé essentiellement sur la notion de confiance. La technique la plus utilisée est le stockage sur les nœuds des amis (*ou une partie des amis*). Cette technique améliore la disponibilité et la confidentialité. En plus, elle a un impact positif sur les performances de fait que le contenu partagé par un utilisateur intéressent ces

amis plus que les autres utilisateurs ; et il est disponible sur leurs machines. Le stockage sur les machines des amis est adopté par : Friendstore [157], PeerSoN [33], Safebook [49], ProofBook [30], MyZone [112], DiDuSoNet [81], LotusNet [14].

- **c- Stockage selon le meilleur effort** : Dans cette stratégie, le choix des nœuds pour le placement des répliques n'est pas limité aux nœuds de confiance. L'objectif est d'exploiter les réseaux pour une meilleure disponibilité avec un nombre minimale de répliques. Autrement dit, au moins une des répliques devrait être disponible à tout moment pour les demandes d'autres utilisateurs, mais le nombre total de répliques dans le système doit être aussi petit que possible [101]. Le choix de ces répliques peut combiner plusieurs critères tels que : la puissance des machines, la stabilité ou les habitudes quotidiennes de connexion [143]. L'adoption de cette technique de réplication nécessite de renforcer la politique de protection du fait que les nœuds choisis ne sont pas forcément dignes de confiance. GEMSTONE [154] et SOUP [100] sont des exemples d'architectures qui utilisent cette approche.

Quelle que soit la technique de réplication utilisée, la haute disponibilité nécessite l'augmentation du nombre de répliques ainsi qu'un temps de connectivité suffisant. En plus, la nature dynamique des données des réseaux sociaux nécessitent un effort supplémentaire pour garantir la cohérence des répliques (*échange des messages de synchronisation*). Ceci peut surcharger le réseau par un trafic supplémentaire et dégrader les performances.

2.3.1.2 Stockage sur plusieurs serveurs

Dans ce deuxième schéma se trouve le modèle semi centralisé où certains nœuds (*super peers*) jouent un rôle plus important que les autres. Le principe consiste à donner à l'utilisateur la possibilité de monter son serveur d'hébergement ou de choisir un serveur de confiance. Cette stratégie est qualifiée comme décentralisée car les données des utilisateurs d'un même réseau social sont stockées sur plusieurs serveurs qui ne sont pas gérés par le même fournisseur. Dans ce cas, le problème de disponibilité ne se pose pas et la réplication n'est pas nécessaire [130]. Le principe de cette approche est reporté dans la figure 2.3 [101].

Deux approches peuvent être différenciées dans ce schéma :

- **a- Stockage sur des serveurs personnels** : Ces solutions supposent généralement que chaque utilisateur dispose d'un serveur web disponible pour stocker et récupérer ses données. Cette approche nécessite des compétences techniques souvent avancées pour pouvoir configurer un serveur distant ou monter un serveur local. Par exemple, Anderson et al. [18], Persona [20], Diaspora [7] proposent des architectures qui suivent ce schéma.
- **b- Stockage sur le cloud** : L'avancement du «cloud computing» a poussé vers la proposition des architectures où l'utilisateur stocke ses données dans un espace loué chez un fournisseur de service. Dans ce cas, il faut envisager des frais dépensés par l'utilisateur pour pouvoir participer au réseau social. Ce point peut être primordial dans le choix du fournisseur en plus des autres critères tels que :

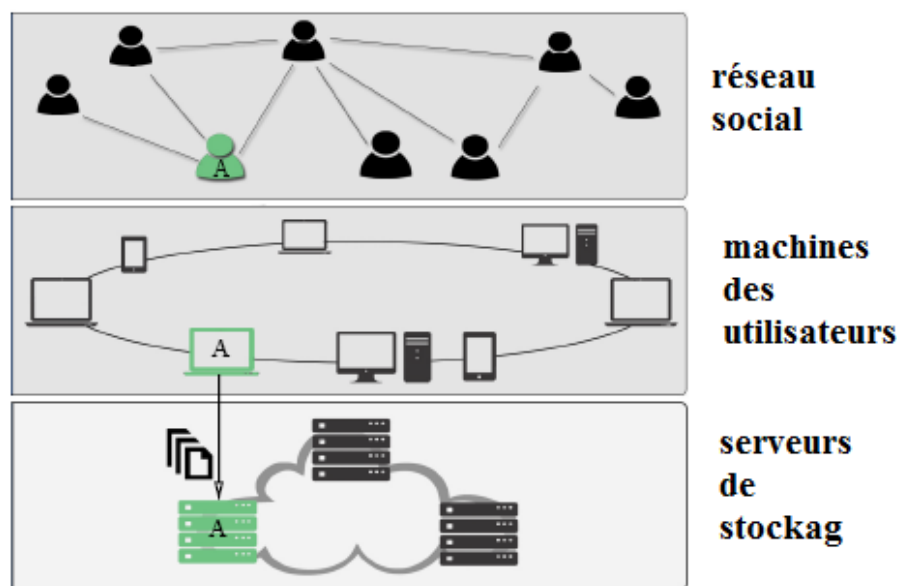


FIGURE 2.3 – Stockage sur des serveurs

taille de stockage, bande passante et fiabilité [130]. A titre d'exemple, on peut citer : PrPl [142], Vis-à-Vis [144], Contrail [149], POSN [70].

2.3.1.3 Stockage hybride (*serveur/pair à pair*)

Dans ce cas, les deux schémas précédents sont utilisés. L'objectif est de tirer profit des avantages des deux techniques. Un premier schéma est le stockage d'une partie des données dans des serveurs et d'autres données sur le réseau pair à pair. Une deuxième possibilité est de séparer le stockage du contrôle (*indexation et adressage*), un aspect peut être géré par des serveurs et l'autre dans le réseau pair à pair. Cette hybridation est observée dans : SuperNova [111], Confidant [144], Cuckoo [168], Polaris [165], Vegas [68], Lilliput [129].

2.3.2 Protection

Afin d'atteindre l'objectif principal (*confidentialité*), des politiques de contrôle d'accès sont introduites. Elles sont basées sur des listes d'accès, des schémas de chiffrement ou une combinaison des deux techniques [56, 130]. Les RSD assurent une confidentialité des données en donnant à l'utilisateur la possibilité de définir lui-même sa politique. Cette politique se résume à des déclarations simples sur les droits attribués aux autres utilisateurs pour accéder au contenu. Ensuite, différents mécanismes sont appliqués pour réaliser cette politique.

En observant les architectures proposées, on peut remarquer que l'utilisateur peut dans un premier lieu spécifier si le contenu est public ou protégé. Le contenu public

est accessible par tous les utilisateurs. Par contre, pour un contenu protégé, seuls les utilisateurs autorisés peuvent y accéder. Dans ce deuxième cas, plusieurs déclarations sont adoptées :

- Le contenu est accessible par les amis,
- Le contenu est accessible par un sous ensemble d'amis (*famille, collègue de travail, ..*),
- Les utilisateurs autorisés sont cités dans une liste d'accès (*liste blanche*),
- Les utilisateurs non autorisés sont cités dans une liste d'interdiction (*liste noire*),
- Le contenu est destiné à un groupe d'utilisateurs,
- Les utilisateurs autorisés sont caractérisé par une information présente dans leurs profils (*location, intérêt*),
- Les droits d'accès sont définis par type de contenu (*post, image, commentaire, ...*),
- Les droits d'accès sont définis pour chaque utilisateur sous forme d'un certificat de subvention (*Grant certificate*).

Il est à noter que plusieurs type d'accès sont autorisés ou refusés : lecture, écriture, téléchargement et ajout (*commenter par exemple*).

Pour mettre en œuvre ces hypothèses de confidentialité, des mécanismes de sécurité sont employés. Ils sont basés sur le chiffrement (*symétrique, asymétrique*) et sur les listes de contrôle d'accès. Dans plusieurs architectures, les deux techniques sont employées en même temps pour renforcer la protection des données. Le choix des mécanismes à utiliser est affecté par le schéma de stockage et de réplication. Les données sont souvent stockées en clair dans les nœuds de confiances. Le chiffrement devient indispensable lorsqu'on stocke les données sur des nœuds qui ne disposent pas l'autorisation pour y accéder.

2.3.2.1 Chiffrement symétrique/asymétrique

Le principe du chiffrement asymétrique est que chaque utilisateur dispose d'une paire de clé (*privé, public*). Ceci est idéal pour l'envoi des messages sécurisés entre deux utilisateurs. Pour le partage d'un contenu avec un groupe, une clé de groupe doit être utilisée. Le chiffrement symétrique est utilisé dans ce cas. Le principe est de générer une clé symétrique pour chaque groupe. Cette clé est utilisée pour chiffrer le contenu destiné aux membres du groupe. La clé du groupe est envoyée aux membres par le biais d'un chiffrement asymétrique. Cette combinaison des deux schémas de chiffrement est l'approche utilisée par plusieurs architectures : Safebook [49], PeerSoN [33], ProofBook [30], SuperNova [111], Contrail [149], Vegas [68].

2.3.2.2 Chiffrement basé sur les attributs (*ABE*)

Un autre schéma de chiffrement aussi employé est le chiffrement basé sur les attributs (*ABE*) [77]. Le principe de cette technique est d'associer des attributs à chaque utilisateur (*dans le cas des réseaux sociaux, c'est la nature de la relation : amis, famille, collègue*). En conséquence, chaque utilisateur dispose d'une clé secrète *ABE* (*équivalente à la clé privé*) qui englobe ses attributs. Ensuite, la clé utilisée pour chiffrer un contenu englobe aussi des attributs d'une manière qui permet aux utilisateurs avec les attributs demandés de déchiffrer le contenu.

A titre d'exemple, DECENT [92] combine cette technique avec le chiffrement symétrique. Le contenu créé par l'utilisateur pour ses amis est chiffré avec une clé symétrique aléatoire. Le propriétaire du contenu crée une référence au contenu dans son profil et il crypte la clé symétrique pour les attributs corrects (*selon la politique d'accès*). Lorsqu'un utilisateur souhaite lire le contenu, il trouve la référence au contenu et décrypte la clé de contenu symétrique avec sa clé secrète *ABE* obtenue à partir du propriétaire. Ensuite, l'utilisateur peut récupérer l'objet à partir du DHT et décrypte les champs chiffrés à l'aide de la méthode symétrique. Le chiffrement par attributs est aussi employé dans : Soup [100], Persona [20], SocialGate [99], GEMSTONE [154].

2.3.2.3 Listes d'accès sans chiffrement

Cette approche est adoptée dans le cas où les données sont stockées dans des nœuds de confiance. Le contenu est stocké sans chiffrement et l'utilisateur définit la politique de confidentialité en précisant pour chaque contenu la liste des utilisateurs autorisés. Par exemple dans Diaspora [7], les amis d'un utilisateur sont organisés en catégories dites (*aspects*). Lorsque l'utilisateur souhaite partager un contenu, il précise si le contenu est public ou bien destiné à un sous ensemble d'amis en précisant l'aspect concerné. Polaris [165], Vis-à-Vis [144] et Friendica [74] utilisent aussi les listes d'accès sans chiffrer le contenu.

2.3.2.4 Listes d'accès avec chiffrement

Une autre approche exploite les deux techniques (*liste de contrôle d'accès et le chiffrement*). On peut citer par exemple LifeSocial.KOM [78] qui crypte le contenu par une clé symétrique. Ensuite cette clé est cryptée en utilisant les clés publiques des amis autorisés. L'objet à publier est composé du contenu crypté suivi d'une liste des clés qui correspondent aux utilisateurs autorisés. Cet objet est stocké dans n'importe quel nœud dans le réseau. L'utilisateur autorisé doit déchiffrer la clé secrète en utilisant sa clé privé pour pouvoir déchiffrer le contenu. Ceci n'est possible que si une copie de la clé secrète a été cryptée par sa clé publique. En plus, le nœud qui stockent la donnée ne peut pas savoir quel sont les utilisateurs autorisés. On peut citer aussi Prometheus [102] comme architecture qui utilise les listes d'accès et le chiffrement asymétrique.

Ces techniques semblent protéger efficacement les données. Cependant, cela n'empêche pas l'existence de carences. L'idée de faire confiance au nœud de stockage (*pair ou serveur*) pourrait constituer des failles potentielles d'épuisement de la confidentialité. Même avec le chiffrement de contenu, les informations concernant les liens d'amitié et la constitution des groupes peuvent être considérés comme information sensible à protéger.

L'étude [79] soulève également la possibilité d'extraire des informations sensibles en analysant le contenu crypté ou les métadonnées divulguées en clair. Dans une autre étude [56], nous trouvons une estimation du coût du cryptage, que ce soit à travers des processus de génération de clés ou des processus de décryptage au niveau du récepteur. Ces mécanismes nécessitent des efforts de calcul supplémentaires et augmentent la taille des données à stocker et à échanger, cela affecte la charge du réseau et le temps de réponse déjà faible.

2.3.3 Communication

Un troisième aspect aussi important est la gestion de la communication entre les utilisateurs. Les fonctions de base sont les échanges des messages et la découverte du contenu partagé par les amis. Particulièrement pour un réseau social, un système de notification et la gestion des invitations sont souhaitables. La gestion de communication englobe l'indexation des identités, le routage des échanges et la recherche des données et des utilisateurs. Le choix du mécanisme de communication est très lié au schéma du stockage adopté. On peut facilement différencier deux techniques : communication basée sur les serveurs et communication pair à pair.

2.3.3.1 Communication gérée par les serveurs

Cette technique est utilisée lorsque le réseau est organisé autour des serveurs jouant le rôle du Super Peer. Chaque utilisateur est lié à un serveur et tous ses échanges sont gérés par ce serveur. En cas où les deux utilisateurs communicants sont liés à deux serveurs différents, les techniques de communication inter-serveurs sont utilisées (*exemple : XMPP*). Une autre possibilité est l'appel à d'autres tiers pour gérer la communication [111, 165]. Dans cette approche, la communication est gérée par des entités performantes. Par contre, une autorité centrale est toujours en mesure d'apprendre par exemple : les intérêts, les relations sociales et la popularité des utilisateurs et de leurs profils. Par conséquent, ces approches exigent que les utilisateurs aient confiance à l'autorité à un certain niveau [130].

2.3.3.2 Communication pair à pair

Dans cette deuxième technique, on souhaite éviter de passer par un serveur. Plusieurs architectures (*Safebook* [49], *DECENT* [92], *Cachet* [126], *LotusNet* [14], *LifeSocial.KOM* [78], *Vis-à-Vis* [144]) utilisent une couche DHT pour la découverte du contenu et pour l'acheminement des données du nœud de stockage au demandeur.

PeerSoN [33] offre la possibilité d'une communication directe entre les utilisateurs une fois le contenu trouvé par la DHT (*voir figure 2.4*). Cuckoo [168] utilise, en plus de la couche DHT, la technique d'inondation pour optimiser la recherche du contenu populaire.

Une autre technique basée sur un modèle d'abonnement (*publish/subscribe*) est employée dans FETHR [138] et Litter [96]. Dans Litter, le créateur d'un contenu communique directement avec les abonnés. Litter offre aussi la possibilité de propager le contenu par la technique d'inondation si l'utilisateur souhaite partager ses publications avec les autres utilisateurs. FETHR exploite un protocole simple basé sur des requêtes

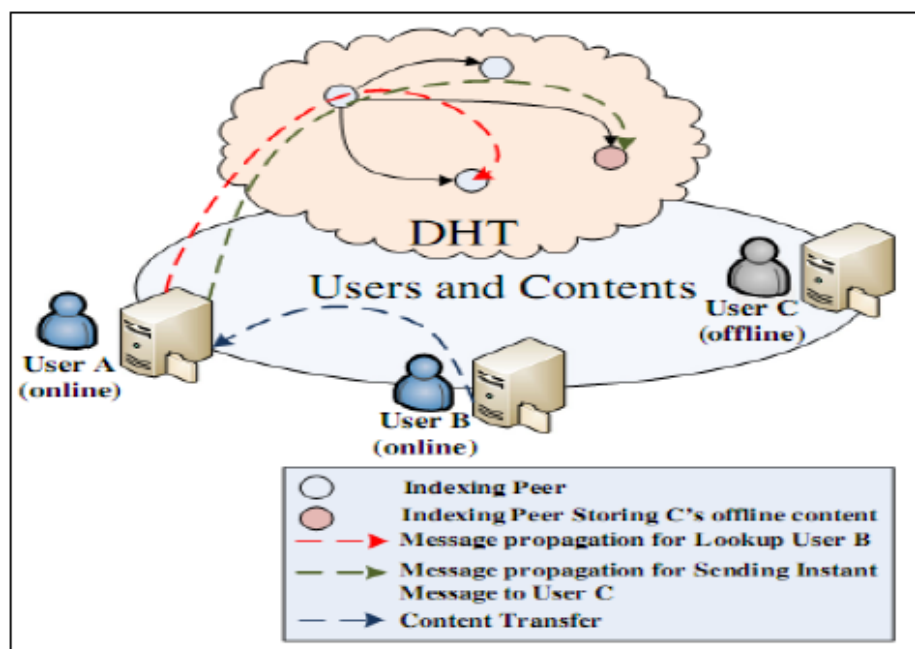


FIGURE 2.4 – Architecture de PeerSon

HTTP (*GET*, *POST*) pour enregistrer des abonnements et pour transmettre les publications directement aux abonnés.

Dans cette deuxième approche, il n'est pas nécessaire de faire confiance à une entité centrale mais la déconnexion de certains nœuds peut troubler la communication.

2.4 Classification des réseaux sociaux décentralisés

Selon la décentralisation dans les trois aspects de l'architecture (*essentiellement le stockage*), trois classes ont été identifiées dans [130] et [101].

2.4.1 Réseaux sociaux Pair à Pair

Le réseau est conçu sur une collaboration entre les utilisateurs sans aucune intervention d'une entité centrale.

2.4.2 Réseaux sociaux fédérés (*basés sur les serveurs*)

Des architectures semi centralisées ou plusieurs serveurs jouent le rôle des super peer. Une couche de fédération gouverne la communication entre les serveurs et l'utilisateur peut choisir et changer à tout moment son serveur.

2.4.3 Réseaux sociaux hybrides

Un mélange des deux technologies (*clients/serveur et pair à pair*).

2.5 Fonctionnalités des réseaux sociaux décentralisés

Les réseaux sociaux décentralisés sont proposés comme une alternative des réseaux sociaux centralisés. Donc, ils doivent assurer les mêmes fonctionnalités que ces derniers. Cependant, l'avancement actuel du domaine démontre que les architectures proposées sont toujours coincées dans le monde académiques avec peu de fonctionnalités supportées. En plus, la majorité des architectures ne présentent qu'une simple preuve de concept à l'exception de quelques architectures qui viennent avec un prototype [68, 78, 96, 168]. On doit exclure les réseaux sociaux fédérés de ce constat car ils existent plusieurs projets qui sont déployés et utilisés par un public important. Le tableau 2.1 [8] montre les statistiques des 10 meilleurs réseaux sociaux fédérés.

Projet	Nœuds	Utilisateurs	Site web
Mastodon	3,415	3,035,360	joinmastodon.org
Matrix (Synapse)	2,518		matrix.org
Pleroma	1,028	79,501	pleroma.social
PeerTube	816	121,839	joinpeertube.org
Write Freely	301	21,116	writefreely.org
Friendica	298	7,799	friendi.ca
diaspora*	184	753,433	diasporafoundation.org
Misskey	172		join.misskey.page/en
Pixelfed	164	33,402	pixelfed.org
Hubzilla	133	4,622	hubzilla.org

TABLE 2.1 – Top10 des réseaux sociaux fédérés

En ce qui concerne les réseaux purement pair à pair, la pénurie des fonctionnalités est clairement observée. Le tableau 2.2 [44] illustre les fonctionnalités présentes dans un ensemble de réseaux décentralisés.

2.6 Avantages et limites

Les architectures décentralisées sont proposées pour éviter les problèmes liés à la dépendance d'une autorité centrale. Notamment, le problème de la mauvaise utilisation des données. Malgré les efforts continus des communautés académiques et open-source, les réseaux sociaux décentralisés sont loin de concurrencer les réseaux sociaux centralisés. Les utilisateurs continuent toujours d'utiliser les réseaux sociaux centralisés à cause de leurs attractivités et leurs performances. Les architectures décentralisées sont innovantes et offrent plus de contrôle et de protection mais elles viennent avec plusieurs limites qui n'encouragent pas la migration des utilisateurs. On discute ci-dessous les avantages et les limites des architectures décentralisées.

Service	PeerSon	Safebook	PrP1	Vis-à-Vis	Super-Nova	Cachet	FETHR	Cuckoo
Micro-publication	Oui	Oui	Oui	Oui	Oui	Oui	Oui	Oui
Commentaires	Oui	Non	Non	Non	Oui	Non	Non	Non
Partage de contenu multi-média	Oui	Oui	Oui	Oui	Oui	Oui	Non	Non
Fil d'actualité	Non	Oui	Oui	Oui	Oui	Non	Oui	Oui
Messagerie instantanée	Oui	Non	Non	Non	Oui	Non	Non	Non

TABLE 2.2 – Les fonctionnalités fournies par les réseaux sociaux décentralisés

2.6.1 Avantages

- **Contrôle** : Les utilisateurs ont plus de contrôle sur leur données du fait qu'ils ont la possibilité de paramétrer le stockage et les droits d'accès.
- **Protection** : Les mécanismes de sécurité employés garantissent un niveau élevé de confidentialité. Seuls les utilisateurs autorisés peuvent consulter les données partagées. En plus, il n'existe pas un point unique avec une vision globale sur le graphe social (*amitiés, groupes, communautés,...*).
- **Liberté** : Aucune politique de censure ou de blocage centralisée. Les utilisateurs peuvent se regrouper dans des communautés et définissent eux-mêmes leurs politiques.

2.6.2 Limites

- **Infrastructure** : L'infrastructure centralisée est remplacée par les machines des utilisateurs pour distribuer la charge à travers un grand nombre de nœuds et éviter la dépendance à un seul point de défaillance. Cependant, les ressources de ces machines sont limitées (*on parle même des Smartphones*). Ces machines peuvent chuter brusquement sous un flux élevé de requêtes ce qui provoque l'indisponibilité des données et perturbe le protocole de communication pair à pair. En plus, l'exploitation des appareils d'amis comme moyen fiable de stockage peut devenir une cause de fuite de données en cas de vol ou de piratage.
- **Performances** : Les architectures décentralisées sont moins performantes à cause des limites des machines des utilisateurs, des protocoles de recherche et de communication pair à pair et des messages de synchronisation des données répliquées. En réalité, il y a un compromis entre la latence de recherche et de

restauration de données d'une part et le taux de disponibilité et de sécurisation de données. La disponibilité est améliorée par la réplication ce qui engendre une surcharge du réseau par les messages de synchronisation. Les opérations de cryptage/décryptage augmentent le temps nécessaire pour l'envoi et la réception des données. En plus, certaines architectures nécessitent des échanges de certaines clés ou listes d'accès ce qui affecte la surcharge des réseaux [56].

- **Attractivité** : L'un des points forts des réseaux sociaux centralisés est leur facilité d'utilisation. Un utilisateur avec peu de connaissance en informatique peut s'inscrire et utiliser le réseau. Par contre, plusieurs architectures décentralisées nécessitent un effort technique de la part des utilisateurs pour configurer un nœud ou pour paramétrer les préférences de stockage et de sécurité. Par exemple, dans Diaspora, pour gérer son propre serveur, un utilisateur devrait être en mesure de configurer le serveur physiquement et puis pour installer Ruby, SQLite3, OpenSSL et plusieurs autres bibliothèques requises pour monter une instance du réseau Diaspora [101].
- **Fonctionnalité** : Les architectures proposées ont prouvé la possibilité d'implémenter les fonctionnalités de base d'un réseau social tels que (*publier, commenter et communiquer*). Par contre, les fonctionnalités avancées tels que la recherche, la recommandation et les notifications sont plus difficiles à mettre en œuvre [130].
- **Modèle économique** : Les plates-formes centralisées des médias sociaux offrent leurs services gratuitement aux utilisateurs, en échange de l'utilisation de leurs données pour réaliser des profits significatifs. Ce modèle a attiré des millions d'utilisateurs et a également attiré des investisseurs. En revanche, les architectures décentralisées viennent pour éliminer la possibilité d'investir dans les données des utilisateurs, rendant difficile l'obtention du financement. D'autre part, certains systèmes obligent l'utilisateur à payer pour un service d'hébergement ou à dépenser de l'argent pour les infrastructures nécessaires à la participation dans le réseau (*stockage, puissance de calcul, bande passante*), ce qui est une raison supplémentaire pour le désintérêt des utilisateurs.

Il est à noter que la présence de ces avantages et limites diffère entre les classes des RSD (*p2p, fédérés, hybrides*) et même entre des architectures dans la même classe. Les architectures fédérées sont plus riches de fonctionnalités et assurent mieux la disponibilité des données mais engendrent les risques des architectures centralisées. Les architectures pair à pair offrent aux utilisateurs plus de contrôle mais sont moins performantes et moins attractives. Les architectures hybrides essaient de combiner les avantages des deux approches et évitent leurs limites. En particulier, ils peuvent offrir une haute disponibilité des données utilisateur grâce à l'utilisation de serveurs, tout en gardant la communication et les données privée entre les utilisateurs [101].

2.7 Pistes de recherche dans les réseaux sociaux décentralisés

Malgré les promesses des architectures décentralisées des réseaux sociaux sur la protection et la liberté, elles n'ont pas attiré un nombre important d'utilisateurs ou de fournisseurs. On cite ci-dessous, des pistes d'améliorations possibles pour concurrencer les systèmes centralisés.

- Continuer les efforts pour concevoir des schémas de stockage décentralisés qui assurent la disponibilité et des performances similaires à celles garantis par des serveurs. Il s'agit de proposer de nouvelles approches de stockage, de réplication et de mise à jour qui garantissent la disponibilité des données et la cohérence des répliques tout en optimisant les effets sur les performances.
- Développer des fonctionnalités avancées telles que les moteurs de recherche et les systèmes de recommandation sans violer la confidentialité. Ces systèmes reposent sur des index centraux. Le défi est d'offrir ces fonctionnalités avec un minimum de collecte centralisée des données.
- Mettre à niveau les protocoles pair à pair destinés au partage pour considérer les caractéristiques des réseaux sociaux (*liens sociaux, intérêts [25], confiance, communauté, proximité géographiques, etc.*).
- Développer des fonctionnalités en mode pair à pair dans des réseaux sociaux centralisés. L'idée est d'exploiter les capacités des systèmes centralisés et limiter la décentralisation à certaines fonctions telles que : la messagerie, la diffusion multimédia [15, 16], les jeux en ligne, etc.
- Proposer des réseaux sociaux décentralisés spécialisés. L'idée est que la décentralisation peut être plus efficace dans certaines applications : messagerie sécurisée, collaboration scientifique et gestion des catastrophes.
- Exploiter la réussite de la technologie BlockChain. Cette approche est actuellement utilisée pour récompenser les utilisateurs. Plusieurs systèmes offrent aux utilisateurs la possibilité de gagner de l'argent (*monnaie numérique*) en contribuant dans le réseau social [80]. Cette technologie est aussi utilisable pour gérer les identités et l'authentification. Le projet Stacks [6] est promoteur dans ce sens.

2.8 Conclusion

LES réseaux sociaux ont attiré l'attention des utilisateurs par les fonctionnalités qu'ils proposent et par la haute performance garantie. La contribution des utilisateurs dans ces systèmes par une variété de données (*souvent sensibles*) a posé beaucoup de questions sur l'exploitation malveillante de ces données par les fournisseurs et sur

leur capacité de les protéger contre le piratage ou la fuite. Les architectures décentralisées sont proposées comme alternative offrant aux utilisateurs plus de contrôle sur leurs données.

Les solutions proposées adoptent essentiellement les architectures pair à pair. Ces derniers ont connu une réussite dans d'autres domaines d'application tels que : le partage, la communication, la diffusion multimédia, etc. Pour concevoir un réseau social sur une architecture pair à pair, il faut définir : un schéma de stockage, une politique de protection, et un protocole de communication. Nous avons présenté dans ce chapitre, les approches proposées dans la littérature.

En analysant les avantages et les limites de la décentralisation des réseaux sociaux, nous avons constaté les problèmes de recherche qui doivent être étudiés. Dans le prochain chapitre, nous présentons notre première contribution : Mise à jour des profils répliqués dans les réseaux sociaux décentralisés.

MISE À JOUR DES PROFILS RÉPLIQUÉS DANS LES RÉSEAUX SOCIAUX DÉCENTRALISÉS-UN MODÈLE BASÉ SUR LE JOURNAL DES ÉVÉNEMENTS

3.1 Introduction

LES réseaux sociaux décentralisés sont proposés pour offrir plus de garantie de confidentialité aux utilisateurs. Ils évitent de stocker des données sur des serveurs centraux. Les profils sont des objets importants et critiques pour ce type de réseau. Ils sont stockés sur les machines des utilisateurs. Afin de garantir une plus grande disponibilité, chaque profil est répliqué sur plusieurs machines. Les différentes copies d'un profil donné, stockées dans différents appareils, sont correctement mises à jour avec toutes les écritures qui se sont produites dans une seule copie. Étant donné que les profils subissent des écritures fréquentes, une charge réseau importante est générée pendant le processus de mise à jour. Dans ce chapitre, nous proposons un modèle de mise à jour basé sur l'utilisation d'un fichier journal pour stocker les écritures. Cela réduit la taille des données échangées entre les machines stockant les mêmes profils.

3.2 Problématique

Le concept principal des réseaux sociaux décentralisés est d'éviter le stockage des données des utilisateurs sur des serveurs non approuvés. Le schéma selon lequel les profils sont stockés affecte les performances du système.

Le premier paramètre directement concerné par ce choix est la disponibilité des profils. Pour donner aux utilisateurs la possibilité de gérer leurs propres données, la première idée est de stocker les données sur la machine de l'utilisateur. Dans cette situation, la disponibilité des profils est conditionnée par la présence en ligne de la machine.

Pour améliorer la disponibilité des profils, le mécanisme de réplication est introduit. Ainsi, chaque profil est dupliqué dans de nombreuses machines. Différents schémas de réplication ont été proposés dans la littérature. Certains sont hérités des techniques de réplication des systèmes P2P traditionnels en les stockant dans des super pairs (*par exemple supernova [146]*) ou en utilisant DHT (*DECENT [92] et LifeSocial.Kom [78]*).

D'autres systèmes explorent les relations sociales (*amis* [49, 111], *communautés* [55, 83], ..) pour assurer une réplication sûre.

La réplication améliore clairement la disponibilité des profils, mais elle peut dégrader les performances du réseau. Les copies du même profil doivent être mises à jour afin de servir un contenu cohérent. Le mécanisme de mise à jour peut conduire à une situation de surcharge du réseau. Ceci est affecté par la nature des profils des réseaux sociaux par rapport aux données stockées dans les systèmes P2P classiques : des changements fréquents et une taille croissante.

Un autre aspect à observer est l'équilibrage de charge entre les machines. Les données sont stockées sur les machines des utilisateurs qui ont des performances réduites. Les machines qui stockent de nombreux profils populaires peuvent échouer à supporter une montée des requêtes des utilisateurs.

Dans ce chapitre, nous proposons un nouveau modèle pour gérer la réplication des profils dans un réseau social décentralisé. Nous proposons de stocker tous les profils dans de nombreuses machines des utilisateurs, y compris la machine du propriétaire du profil. Pour assurer la cohérence de toutes les copies d'un même profil, nous proposons d'appliquer deux mécanismes de mise à jour :

- Une mise à jour synchrone basée sur des messages de synchronisation.
- Une mise à jour asynchrone basée sur un journal qui stocke tous les événements survenus sur un profil donné.

Le service de recherche est basé sur une liste des priorités qui permet d'équilibrer la charge entre les machines stockant les mêmes profils. Les principaux objectifs de la solution proposée sont :

- Améliorer la disponibilité des profils,
- Assurer la cohérence des profils répliqués,
- Réduire la surcharge générée par le processus de mise à jour ; ceci est dû à l'utilisation du journal des événements,
- Équilibrer la charge entre les machines d'un même groupe de réplication.

3.3 Travaux antérieurs

Les auteurs de [143] présentent S-DATA (*Structured approach for Diurnal Availability by Temporal Assemblage*) ; un protocole de regroupement et de réplication de contenu basé sur le temps et qui exploite les habitudes quotidiennes de connexion des utilisateurs. Ils introduisent également le concept de Bêta-disponibilité (*au moins les Bêta membres d'un groupe de réplication seront en ligne à tout moment*). Les pairs sont regroupés en fonction de leur modèle de disponibilité quotidienne de manière à garantir la Bêta-disponibilité. L'évaluation montre qu'un niveau de 2-disponibilité assure la

disponibilité satisfaisante avec une surcharge optimale. Le modèle de cohérence n'est pas traité dans ce travail. De plus, le placement des répliques est choisi uniquement en fonction de la disponibilité, les auteurs ne prennent pas en compte d'autres critères tels que la confiance et la capacité.

Dans [46], un système de stockage dynamique est proposé. Les profils sont stockés dans un sous-ensemble d'amis uniquement. Les appareils des amis choisis pour stocker une copie du profil sont modifiés de manière dynamique. Le sous-ensemble d'amis qui participent à la réplication change également en fonction des interactions entre amis. La disponibilité assurée par ce système est intéressante. Cependant, les auteurs ne tiennent pas compte de la surcharge réseau causée par le changement du lieu des répliques. Ils supposent que les pages de profil sont généralement petites. Cette hypothèse ne peut pas rester vraie pour une longue période. Avec le temps, la taille du profil devient importante et il sera très coûteux de le copier à chaque mise à jour.

La réplication dirigée par l'utilisateur est proposée dans [123]. Les utilisateurs interviennent dans la sélection des amis de confiance qui stockent les répliques. Cet ensemble d'amis appelé TPS (*Trusted Proxy Set*) est ensuite optimisé selon deux paramètres : la localisation géographique et la période de présence en ligne. Pour le processus de mise à jour, nous ne devons copier que les nouveaux événements. Cela minimise la charge de travail des nœuds, mais l'algorithme suppose le chevauchement des périodes de présence en ligne des nœuds pour se synchroniser. Malheureusement, ce n'est pas toujours garanti.

Ivy [122] propose un système de fichiers peer-to-peer multi-utilisateurs. Il gère les écritures simultanées séparément. Chaque utilisateur enregistre ses modifications dans un journal stocké et répliqué sur la base d'un DHT. L'utilisateur, lors de la lecture, collecte tous les journaux et crée une version de cohérence. Cette approche évite l'utilisation de tout verrou pour protéger les données et ne nécessite aucune confiance des autres rédacteurs (*les utilisateurs peuvent choisir les journaux à utiliser ou à ignorer*). Nous ne pouvons pas appliquer ce système dans le contexte du travail en réseau social. Ceci est dû à la dépendance temporelle entre les écritures des utilisateurs : l'ordre d'écriture est crucial pour la cohérence de l'ensemble de la discussion. Aussi, dans un réseau social, l'utilisateur peut exprimer librement ses opinions et le lecteur ne doit pas exclure les contributions des autres.

Logoot est un autre système d'édition collaboratif présenté dans [162]. Il est conçu sur la base d'architecture P2P. Le système traite des fichiers texte organisés en couples (*ligne, texte*) et permet deux opérations (*insérer une ligne et supprimer la ligne*) avec spécification de position. Le principal problème traité est de savoir comment collecter des opérations de mise à jour parallèles à partir de différentes répliques afin de créer une version finale cohérente. Wikipedia est utilisé pour les tests. Logoot-undo [163] est une version extensible qui prend en charge les opérations d'annulation.

La même idée est traitée dans un autre système d'édition décentralisé [13] où la position d'insertion ou de suppression est représentée avec un nombre réel de manière à empêcher la génération de la même position deux fois. Ces systèmes d'édition collaboratifs peuvent être adaptés aux réseaux sociaux décentralisés. Au contraire, le modèle de cohérence éventuel n'est pas adapté à la nature omniprésente du contenu du réseau social (*publication et commentaires*).

Lilliput [129] est un schéma de réplication qui sépare le stockage de données statiques

en bloc (*vidéos et albums photo*) de la couche sociale essentielle (*par exemple les informations de profil de base, les mises à jour fréquentes, les notifications et les messages personnels*). Il vise à accroître la disponibilité et de réduire les coûts des opérations de mise à jour. Les données en masse sont censées être rarement utilisées. Elles sont cryptées et stockées sur un serveur central. Les données récentes et le graphe social sont répliqués dans la couche P2P. Cette solution augmente clairement la disponibilité, mais la protection de la vie privée des utilisateurs n'est pas complètement protégée puisque le choix des répliques ne considère pas l'amitié ou la confiance. De plus, le serveur central gère les données et permet de se faire une idée sur les utilisateurs et leurs échanges [79].

Dans ce travail, nous gérons la réplication des profils dans un réseau social décentralisé. Nous proposons d'organiser le réseau en groupes de réplication. Pour garantir la cohérence des profils répliqués, nous utilisons un journal inspiré des systèmes d'édition collaboratifs. Ce journal permet de réduire le coût de l'opération de mise à jour. De plus, l'utilisation du fichier journal garantit de transmettre les mises à jour sans qu'il soit nécessaire de chevaucher les périodes de connexion. La solution proposée est détaillée dans la section suivante.

3.4 Solution Proposée

Dans cette section, nous proposons un nouveau modèle pour gérer la réplication des profils dans les réseaux sociaux décentralisés (*voir [21, 23]*). Pour garantir une haute disponibilité, les appareils des utilisateurs sont regroupés dans des groupes de réplication. Les profils des utilisateurs sont ensuite dupliqués dans tous les appareils appartenant au groupe. Nous utilisons deux modèles pour mettre à jour les différentes copies du même profil :

- Un modèle synchrone basé sur des messages de mise à jour directe. Ce modèle est appliqué pour synchroniser les machines en ligne.
- Un modèle asynchrone pour mettre à jour les machines après une déconnexion. Il est basé sur un fichier appelé «journal des événements».

Nous proposons de stocker, toutes les écritures subites par un profil dans ce fichier. Ces modèles de mise à jour évitent de copier la totalité des profils à chaque opération de mise à jour. Seulement, les nouvelles écritures sont échangées. Le processus de recherche est basé sur un ordre de priorité au sein du groupe de réplication afin d'équilibrer la charge entre les machines.

3.4.1 Réplication de profils

Nous proposons de dupliquer les profils des utilisateurs pour augmenter leur disponibilité. Chaque profil est stocké dans la machine de l'utilisateur et d'autres copies sont répliquées dans les machines des autres utilisateurs.

Les machines des utilisateurs sont regroupées en groupes de réplication. Chaque machine ne peut appartenir qu'à un seul groupe. La réplication symétrique [103] est alors adoptée : si les utilisateurs (U_1, U_2, \dots, U_n) construisent un groupe de réplication ;

chaque machine de ce groupe stocke une copie de chaque profil associé (P_1, P_2, \dots, P_n).

Selon cette architecture, un profil est disponible, si au moins une machine du groupe de réplication est en ligne. De plus, si un appareil est en ligne, tous les profils affiliés au groupe sont disponibles. Sur la figure 3.1, un groupe de réplication est représenté.

A titre d'exemple, le niveau de réplication est limité à 5. Les utilisateurs *Ali*, *Sami*,

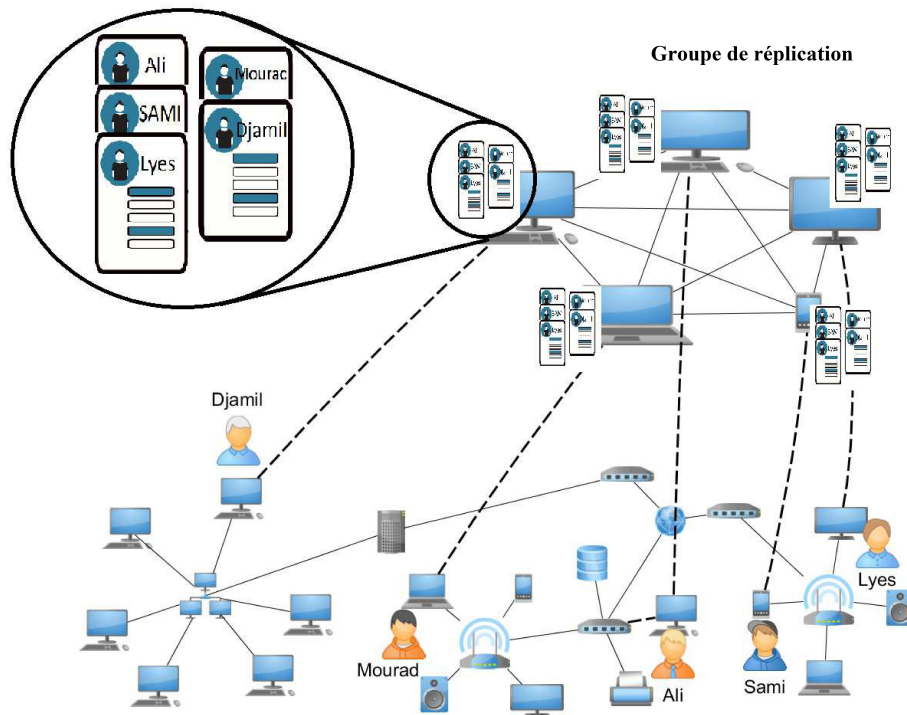


FIGURE 3.1 – Réplication des profils

Lyes, *Mourad* et *Djamil* collaborent pour stocker leurs profils.

Par exemple, le profil de *Ali* est stocké dans la machine de *Ali* et dupliqué en 4 copies dans les machines des autres membres. Le profil *Ali* peut être atteint si l'un des cinq nœuds est en ligne. De plus, *Ali* stocke dans son appareil une copie de chaque profil appartenant au groupe. La construction du groupe de réplication est critique. Elle peut prendre en compte différents critères : les performances, la confiance, et les habitudes de connexion au cours de la journée.

3.4.2 Recherche des Profils

Pour consulter ou interagir avec un profil donné, les utilisateurs doivent connaître les machines qui le stockent. Le service de recherche dirige les utilisateurs vers le groupe de réplication responsable pour le profil souhaité. Une copie du profil est stockée dans chaque machine du groupe. De nombreux nœuds peuvent être en ligne en même temps. Le choix de la machine pour consulter le profil a un effet important sur les performances

du système. Deux stratégies peuvent être envisagées :

- Avec un choix aléatoire, ce profil peut être consulté et modifié dans différents nœuds simultanément. Cela générera des écritures simultanées dans les journaux d'événements. La mise à jour asynchrone est plus compliquée.
- Une deuxième approche est de définir une priorité entre les nœuds. Si deux machines ou plus sont en ligne, la machine prioritaire est choisie. Avec cette stratégie, la contribution des différentes machines dans le groupe de réplication ne sera pas équilibrée : une machine à haute priorité et à longue connectivité est exploitée plus que les autres machines.

Compte tenu de cela, nous proposons de définir un ordre de priorité pour chaque profil. Une machine avec la priorité la plus élevée pour servir un profil donné doit avoir une priorité faible pour un autre profil. Une table de priorité est créée et remplie lors de la construction du groupe. Le **tableau 3.1** illustre cet ordre de priorité pour l'exemple présenté à la figure 3.1.

Machine de	profil de Ali	profil de Mourad	profil de Djamil	Profil de Lyes	Profil de Sami
Ali	1	5	4	3	2
Mourad	2	1	5	4	3
Djamil	3	2	1	5	4
Lyes	4	3	2	1	5
Sami	5	4	3	2	1

TABLE 3.1 – Exemple de distribution de priorité dans le groupe de réplication

Pour chaque profil, une colonne du tableau définit la priorité des machines à utiliser comme serveur de ce profil. Les machines sont contactées selon la priorité définie dans ce tableau. Une valeur faible désigne une priorité élevée.

Par exemple, pour consulter le profil *Lyes* nous contactons la machine de *Lyes* (*priorité 1*). Si cet appareil est hors ligne, nous contactons la machine de *Sami* (*priorité 2*). Dans ce cas, l'appareil *Djamil* servira le profil *Lyes* si toutes les autres machines sont hors ligne. La priorité de chaque machine est changée pour chaque profil pour qu'une machine avec une priorité importante pour servir un profil ait une priorité faible pour servir d'autres profils. La création et le remplissage du tableau peuvent être résumés dans l'algorithme 1.

Avec cette stratégie, les lectures et les écritures sur un profil donné à un moment donné ont lieu dans le même périphérique et la charge est équilibrée entre les différents périphériques du groupe de réplication.

La table de priorité est alors exploitée pour générer une liste ordonnée pour chaque profil. Les éléments de cette liste contiennent des liens vers les périphériques stockant le profil. La figure 3.2 illustre les listes ordonnées générées à partir du **tableau 3.1**.

Pour chaque profil, nous ne stockons que la liste ordonnée des machines (*il n'est pas nécessaire de stocker l'intégralité du tableau des priorités*). Puisqu'il n'y a pas de serveur central ou d'index, nous proposons de stocker ces listes dans une structure DHT. L'identificateur de profil est utilisé comme une clé pour la fonction de hachage et pour

Algorithm 1 Génération de la table des priorités

```

1 : Début
// P : Profil
// R : degré de répllication
// T : table de priorité
// Chaque ligne représente une machine et chaque colonne représente un profil
2 : Pour P = 1 à R faire
// Priorité élevée (1) à la machine propriétaire
3 :   T [P] [P] = 1;
// Augmenter la priorité jusqu'à la fin de la colonne
4 :   Priorité=2;
5 :   Pour i = P + 1 à R faire
6 :     T [i] [P] = Priorité;
7 :     Priorité = Priorité + 1;
8 :   Finpour
// Diminuer la priorité jusqu'au début de la colonne
9 :   Priorité = R;
10 :  Pour i = P-1 jusqu'à 1 faire
11 :    T [i] [P] = Priorité;
12 :    Priorité = Priorité - 1;
13 :  Fin pour
14 :  Fin pour
15 :Fin

```

chaque clé (*Profil*), la liste des appareils stockant le profil est associée.

Dans la phase de connexion, les utilisateurs communiquent l'identifiant du profil souhaité à la DHT. La DHT renvoie la liste des machines stockant chacune une copie du profil. Ensuite, les utilisateurs contacteront ces machines (*si elles sont en ligne*) pour consulter et interagir avec le profil. La figure 3.3 illustre un exemple du service de recherche.

Comme le montre la figure 3.3, l'utilisateur qui recherche un profil, contacte une structure DHT (*étape 1*). La fonction de hachage utilise l'identificateur de profil pour trouver son groupe de répllication. La réponse à cette demande est une liste ordonnée (*étape 2*). Chaque élément de la liste est un lien vers une machine qui stocke une copie du profil. L'utilisateur parcourt la liste et contacte les machine selon l'ordre fixé (*étape 3*). Une fois, une machine contactée est en ligne, l'utilisateur se connecte au profil (*étape 4*). La navigation dans la liste est alors arrêtée.

Dans l'exemple, l'utilisateur contacte 3 machines avant de trouver une copie en ligne. L'algorithme 2 suivant explique le processus de recherche.

3.4.3 Le journal des événements

Les différentes machines d'un groupe de répllication devraient communiquer pour garantir la cohérence des différentes copies d'un même profil. Ils échangent les différentes écritures qui se produisent dans les profils. Une communication directe (*en mode*

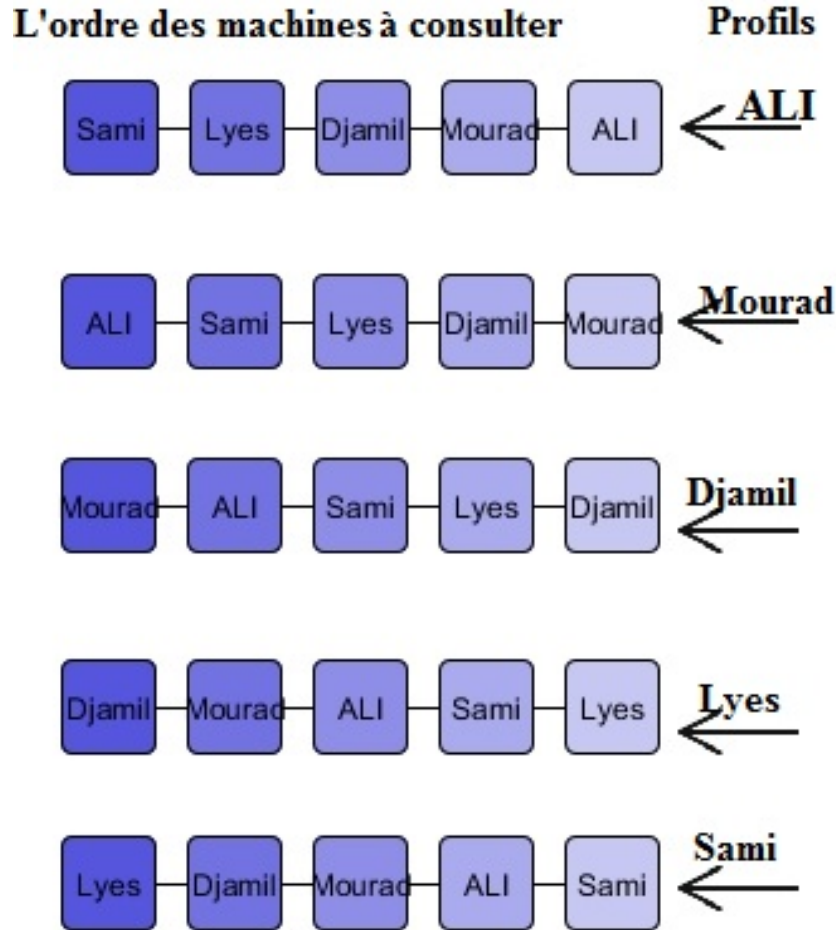


FIGURE 3.2 – Les listes ordonnées d'appareils

synchrone) est possible lorsque deux machines sont en ligne. Au contraire, si deux machines sont en ligne pendant des périodes séparées, ils ne peuvent pas être mis à jour correctement. Nous proposons donc d'utiliser un fichier journal pour stocker toutes les écritures effectuées sur chaque profil. Ce fichier (*nous l'appelons journal des événements*) permet une communication asynchrone au sein du groupe de réplication. Il est utilisé pour mettre à jour les copies des profils avec les écritures effectuées lorsque les machines sont hors ligne.

Le journal des événements est un fichier associé à chaque profil utilisateur. Il est utilisé pour stocker tous les événements se produisant dans les nœuds en ligne. Il sera utilisé ultérieurement pour mettre à jour les versions stockées sur les nœuds déconnectés. Ce fichier journal est conçu pour éviter la duplication de l'ensemble du profil à chaque opération de mise à jour. Ce mécanisme peut réduire la surcharge due à la réplication. La taille du journal est très petite par rapport à la taille du profil. Les événements résident pendant un temps suffisant pour être copiés dans toutes les versions.

Afin d'assurer la cohérence du contenu du profil, les événements sont estampillés. Le mécanisme de mise à jour peut insérer les différents événements en fonction du temps

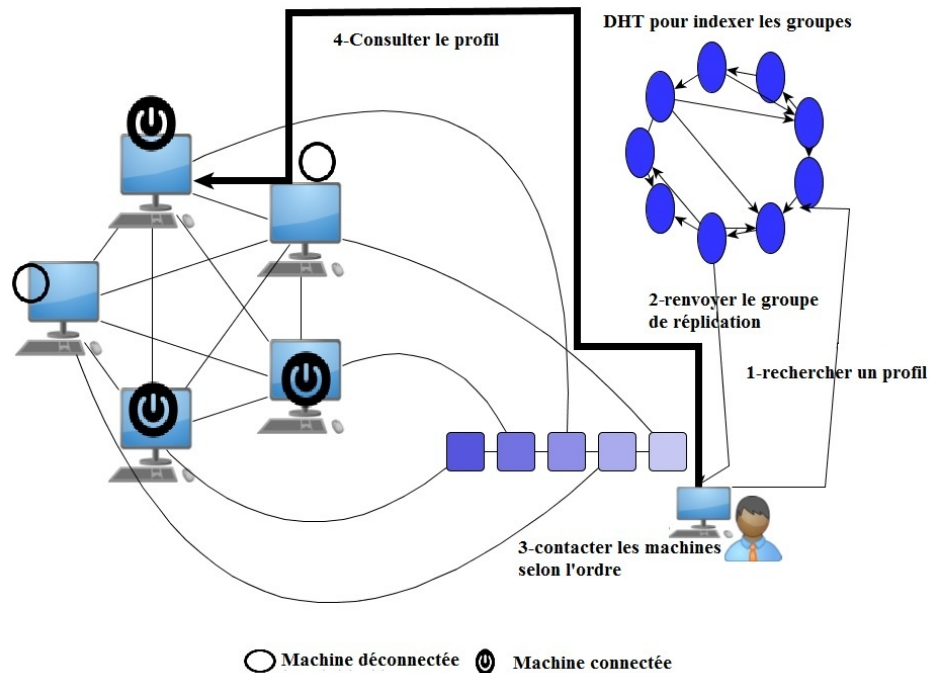


FIGURE 3.3 – Le processus de recherche d'un profil

noté dans le journal. Le journal des événements est une série d'événements structurés qui concernent un profil. Chaque événement contient les informations suivantes :

- **Type** : publier, commenter, aimer , ...
- **Auteur** : l'identifiant de l'utilisateur qui a généré l'événement.
- **Heure et date** : pour insérer l'événement dans un ordre correct dans le profil.
- **Le contenu** : en général un texte.
- Le commentaire ou la publication concernée par l'événement.

Nous présentons quelques exemples des événements stockés dans le fichier journal.

1. Évènement de publication

```
<event type="post ">
<date> 11/11/2016 </date>
<time> 14 :55 :25 </time>
<writer> IDuser1 </writer>
<idpost>idpost2 </idpost>
<post>"this is a new post "</post>
</event>
```

Algorithm 2 Recherche des profils

```

1 : Début
2 : Liste = DHT.Lookup ( ID_Profile )
3 : Trouv = FAUX ;
4 : Tant que ( Liste != NULL ) et ( Non Trouv ) faire
5 :   Contacter Liste
6 :   Attendre (timeout)
7 :   Si ACK alors
8 :     Connectez à Liste ;
9 :     Trouv = vrai ;
10 :  Sinon
11 :    Liste = Liste.suivant
12 :  Fin tant que
13 :  Si ( non trouv ) alors
14 :    Ecrire ("Le profil est actuellement indisponible");
15 : Fin

```

2. Événement de commentaire

```

<event type="comment ">
<date> 11/11/2016 </date>
<time>14 :30 :25 </time>
<writer> IDuser2 </writer>
<idpost>idpost1</idpost>
<idcomment> idcomment </idcomment>
<comment>" my comment " </comment>
</event>

```

3. Événement de réaction (Like)

```

<event type="Like">
<date> 11/11/2016 </date>
<time> 14 :35 :25 </time>
<writer> IDuser1 </writer>
<idpost>idpost1</idpost>
<idcomment> idcomment </idcomment>
</event>

```

Un exemple de journal complet est illustré sur la figure 3.4.

```
<log>
  <profile>ID_user1</profile>
  <replicas>
    <replica>
      <device>iddevice1</device>
      <Last_log_use_time>
        <date> 11/11/2016 </date>
        <time> 11:25:00 </time>
      </Last_log_use_time>
    </replica>
    <replica>
    <replica>
    <replica>
    <replica>
  </replicas>
  <events>
    <event type="comment ">
      <date> 11/11/2016 </date>
      <time>14:30:25 </time>
      <writer>IDuser2 </writer>
      <idpost>idpost1</idpost>
      <idcomment> idcomment </idcomment>
      <comment>" my comment " </comment>
    </event>
    <event type="Like">
      <date> 11/11/2016 </date>
      <time> 14:35:25 </time>
      <writer> IDuser1 </writer>
      <idpost>idpost1</idpost>
      <idcomment> idcomment </idcomment>
    </event>
    <event type="post ">
      <date> 11/11/2016 </date><time> 14:55:25 </time>
      <writer> IDuser1 </writer>
      <idpost>idpost2 </idpost>
      <post>"this is a new post " </post> </event>
  </events>
</log>
```

FIGURE 3.4 – Exemple d'un journal des événements

Le fichier journal contient les informations suivantes :

- Information sur le profil correspondant,
- Information sur les machines qui stockent les répliques : l'identifiant de la machine et le temps de la dernière utilisation du journal par cette machine,
- Un ensemble d'événements horodatés.

3.4.4 Stockage des journaux des événements

Sur la base de l'hypothèse que la taille du journal est inférieure à la taille du profil, le fichier journal est stocké dans un schéma de réplication plus grand. Notre objectif est qu'un nœud déconnecté puisse être informé de tous les événements survenus lors de sa déconnexion. Une très haute disponibilité doit être assurée pour le fichier journal. Ceci est nécessaire pour synchroniser deux nœuds qui ne se rencontrent jamais.

Dans notre cas, nous choisissons de construire une organisation basée sur un algorithme DHT (*voir Figure 3.5*). Nous utilisons une fonction de hachage qui calcule l'emplacement pour stocker le journal. Ensuite, les nœuds de réplication sont automatiquement déduits [136]. La même fonction de hachage est utilisée pour trouver le fichier journal afin de mettre à jour le profil. Dans notre cas, la fonction de hachage utilise l'identifiant de profil comme entrée. Tout algorithme DHT peut être utilisé sans effet significatif sur les performances du système. La solution proposée est générique en ce qui concerne l'architecture sous-jacente du P2P.

3.4.5 Les algorithmes de mise à jour des profils

Chaque version d'un profil utilisateur évolue selon trois algorithmes de mise à jour. La situation du nœud de stockage par rapport aux interactions des utilisateurs définit l'algorithme à appliquer.

- **Mise à jour directe** : sur le nœud recevant une modification directement d'un utilisateur.
- **Mise à jour synchrone** : appliquée aux nœuds en ligne du groupe de réplication.
- **Mise à jour asynchrone** : exécutée après le retour d'un nœud déconnecté au réseau.

3.4.5.1 Mise à jour directe

Cet algorithme est exécuté sur le nœud actif (*le nœud utilisé pour consulter le profil*). Il est lancé lorsqu'un utilisateur (*lecteur, ami, visiteur ou propriétaire du profil*) essaye

Une structure DHT pour stocker les journaux

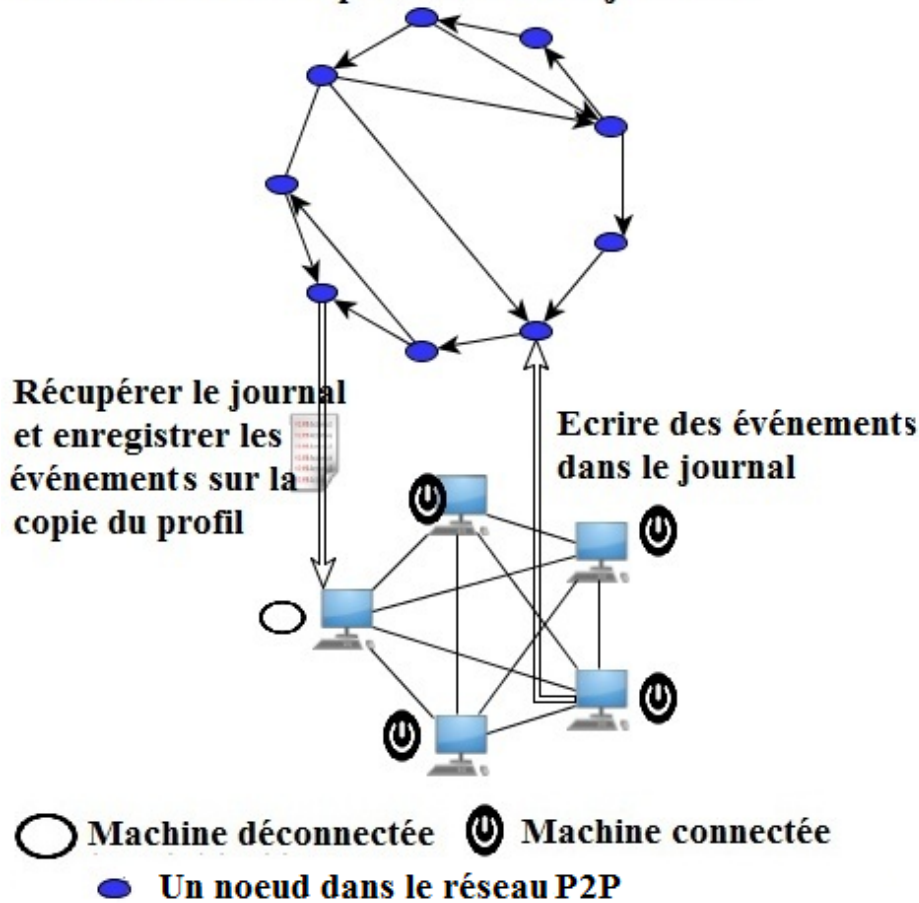


FIGURE 3.5 – Stockage des journaux des événements

de publier ou de réagir avec un ancien contenu. La modification est notée directement dans cette version du profil et les autres nœuds doivent en être informés. Lorsqu'un utilisateur effectue une écriture, elle est directement appliquée à la version en ligne du profil. Le nœud stockant cette version doit informer les autres membres du groupe de réplication. Les nœuds en ligne reçoivent un message contenant l'événement. L'événement est également stocké dans le journal des événements pour être pris en compte par les nœuds déconnectés. La mise à jour directe est illustrée dans l'algorithme 3.

3.4.5.2 Mise à jour synchrone

Pour consulter un profil, les utilisateurs sont dirigés vers un nœud en ligne du groupe de réplication. Ainsi, tous les nœuds en ligne doivent stocker la même version de chaque profil. Après avoir reçu une nouvelle écriture (*nouvelle publication, commentaire ...*), le nœud actif informe les autres nœuds en ligne sur cet événement (*voir Algorithme 3, Etape 2*). Cet événement est immédiatement noté en exécutant l'algorithme de mise à

Algorithm 3 Mise à jour directe

1 : Début
2 : Stockez l'événement sur le profil ;
3 : Envoyez l'événement à tous les nœuds (*en ligne*) du groupe de réplication ;
4 : Stockez l'événement dans le journal des événements ;
5 : Temps_Dernière_mise_à_jour = Temps_actuel ;
6 :Fin

jour synchrone. Les nœuds en ligne d'un groupe de réplication échangent des messages contenant les écritures sur les profils qu'ils servent. Lors de la réception d'un événement (*message de synchronisation*), le nœud l'applique sur sa version. Il n'est pas nécessaire de le stocker dans le fichier journal. Cette opération est effectuée par le premier nœud qui reçoit l'écriture (*Etape 4 de l'Algorithme 3*).

Algorithm 4 Mise à jour synchrone

1 : Début
2 : Stockez l'écriture sur le profil ;
3 : Temps_Dernière_mise_à_jour = Temps_actuel ;
4 :Fin

3.4.5.3 Mise à jour asynchrone

Cet algorithme doit faire partie de la procédure de reconnexion. Une fois reconnecté, un nœud devient en ligne ; mais il ne peut pas servir les utilisateurs tant que tous les profils stockés ne sont pas mis à jour. La mise à jour asynchrone n'est pas effectuée dans le groupe de réplication de profil (*par des messages de synchronisation*), mais elle utilise les journaux des événements stockés dans l'architecture P2P sous-jacente.

Après un certain temps de déconnexion, les fichiers stockés dans un nœud ont probablement d'autres versions modifiées. Le processus de mise à jour asynchrone informe le nœud qui se reconnecte de ces modifications. Cet algorithme est exécuté pour chaque profil stocké dans ce nœud. Seuls les événements survenus après l'instant de la dernière mise à jour sont pris en compte. Le temps de cette mise à jour asynchrone est aussi enregistré.

3.4.6 Illustration du modèle de mise à jour

Nous présentons ici un scénario d'exécution des différents algorithmes de mise à jour. On suppose que seuls deux utilisateurs : *Ali* et *Sami* sont en ligne. Les cinq profils sont consultés à partir des machines d'*Ali* et *Sami*.

- Si *Ali* affiche une nouvelle publication, cette publication est enregistrée à l'aide de l'algorithme de mise à jour directe dans la machine de *Ali*. Ainsi, *Ali* stocke

Algorithm 5 Mise à jour asynchrone

```

1 : Début
2 :   Pour chaque profil stocké dans le nœud faire
3 :     Ouvrir le fichier journal ;
3 :     Extraire tous les événements horodatés après Temps _Dernière _ mise _
à _ jour ;
3 :     Stockez les événements sur le profil ;
3 :     Temps _dernière _ mise _ à _ jour = Temps _actuel ;
3 :     Stocker dans le journal(Temps _derniere _utilisation _du _journal =
Temps _actuelle)
3 :   FinPour
46 : Fin

```

la publication dans sa version et envoie l'événement à *Sami*.

- *Sami* lance l'algorithme de mise à jour synchrone et enregistre la publication dans la version du profil «*Ali*» qu'il stocke.
- De plus, *Ali* enregistre cette publication comme nouvel événement dans le journal des événements correspondant à son profil (*voir figure 3.6*).
- Après un certain temps, *Sami* et *Ali* se déconnectent. Ensuite, *Mourad* rejoint le réseau.
- *Mourad* collecte les journaux des événements de tous les profils qu'il stocke et utilise l'algorithme de mise à jour asynchrone.

En conséquence, la version du profil «*Ali*» stocké dans la machine de *Mourad* contient la dernière publication de *Ali* même si cela se produisait lors de sa déconnexion sur la machine de *Ali* et *Ali* et *Mourad* n'ont jamais été en ligne simultanément.

3.4.7 Maintenance des journaux

Le journal des événements stocke toutes les écritures qui se sont produites sur les différentes versions du profil. Les événements les plus anciens doivent être éliminés pour réduire la taille du journal. Nous ne devons supprimer que les événements inutiles. Ce sont des événements déjà appliqués par les nœuds hors ligne.

Lors de la reconnexion, le nœud applique la mise à jour asynchrone et stocke le temps actuel dans le journal : Temps _derniere _utilisation _du _journal (*Algorithme 5, étape 5*). Ensuite, le journal contient une valeur Temps _derniere _utilisation _du _journal pour chaque version du profil.

La procédure de maintenance des journaux supprime tous les événements qui se sont produits avant l'instant le plus ancien du Temps _derniere _utilisation _du journal.

```

<log>
  <profile>Ali</profile>
  <replicas>
    <replica>
    <replica>
    <replica>
    <replica>
    <replica>
  </replicas>
  <events>
    <event type="post ">
      <date> 08/08/2018 </date><time> 14:00:25 </time>
      <writer> Ali </writer>
      <idpost>idpost12500 </idpost>
      <post>"What a beautiful summer day"</post>
    </event>
    <event type="comment ">
    <event type="comment ">
    <event type="Like">
    <event type="post ">
    <event type="comment ">
  </events>
</log>

```

L'événement ajouté par Ali

FIGURE 3.6 – Nouvel événement écrit dans le journal

3.5 Evaluation des performances

Dans cette section, nous discutons les performances de l'architecture proposée. Nous nous concentrons principalement sur deux paramètres majeurs :

- **La disponibilité du profil** : Chaque profil est stocké en plusieurs copies dans les machines de différents utilisateurs. Un profil est considéré comme disponible si au moins une machine de stockage est en ligne. Nous mesurons la disponibilité des profils sur une période de 24 heures en faisant varier deux facteurs : le degré de réplification (*nombre de machines stockant le profil*) et la période de connexion (*la durée de connectivité en 24 heures*).
- **L'équilibrage de charge entre les machines** : Nous testons la stratégie de recherche présentée dans (*Algorithme 2*). Nous souhaitons que toutes les machines collaboratrices contribuent par la même charge pour servir les utilisateurs.

3.5.1 La disponibilité des profils

Nous observons la disponibilité des profils selon deux facteurs : le degré de réplification et le temps moyen de connexion de différentes machines. Nous varions le degré

de réplication entre 4 et 8 répliques. Ensuite, le temps de disponibilité en 24 heures est mesuré avec l'hypothèse que toutes les machines sont en ligne pour la même durée. Cette durée de connexion varie aussi entre 4 à 12 heures par jour et elle est distribuée de manière aléatoire dans un intervalle de 24 heures. Selon notre architecture, un profil est disponible si au moins un nœud du groupe de réplication est en ligne. La figure 3.7 illustre le résultat de la simulation.

Il est clair que le taux de disponibilité augmente avec la taille du groupe de répli-

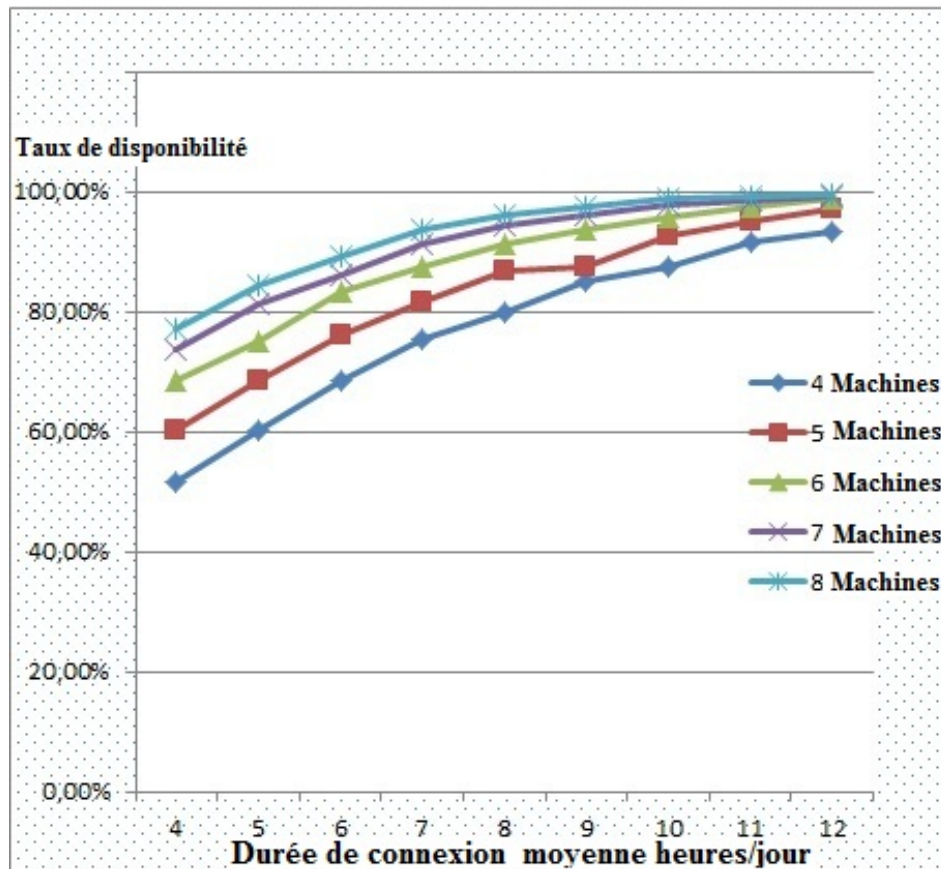


FIGURE 3.7 – Disponibilité des profils

cation. Un plus grand nombre de machines stockant le profil augmente les chances que le profil soit accessible. Cependant, un grand groupe de réplication risque de surcharger le réseau par les messages de mise à jour. Nous recherchons donc d'autres critères qui peuvent garantir une disponibilité suffisante même avec un nombre de répliques plus petit.

On observe que le temps de connexion moyen est plus efficace : avec une période de connexion courte, l'ajout de nouvelles répliques n'a pas d'effet important. Nous avons constaté que le choix des utilisateurs pour construire un groupe de réplication est plus important que la création d'un groupe avec un grand nombre de machines.

3.5.2 Équilibrage de charge dans les groupes de réplication

Nous observons ici l'effet de la stratégie de recherche au sein du groupe de réplication sur l'équilibrage de charge entre les nœuds collaborateurs. Sur la figure 3.8, nous déterminons le nœud responsable de chaque profil.

Avec un groupe de réplication de 5 appareils qui se connectent 8 heures par jour,

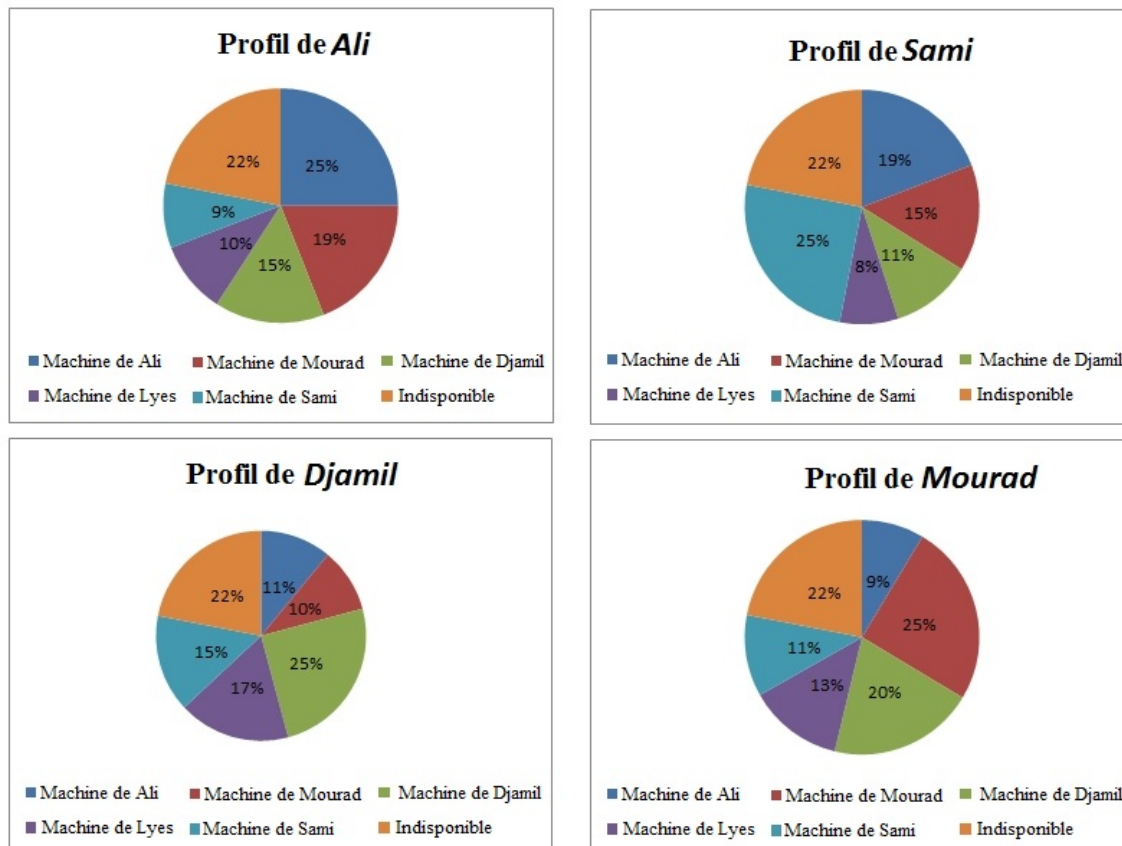


FIGURE 3.8 – Les machines utilisées pour accéder aux profils

les profils sont supposés être en ligne 86% du temps. On observe pour chaque profil la répartition de son temps de disponibilité sur les machines du groupe de réplication.

Par exemple, le profil de *Ali* est plus consulté à partir de la machine de *Ali* que le profil de *Djamil*. Pour illustrer l'équilibrage de charge entre les différentes machines, nous avons observé le nombre de profils servis simultanément par les différentes machines du groupe de réplication. Les résultats sont représentés à la figure 3.9.

Nous faisons varier la durée de connexion par jour et nous calculons pour chaque machine, le nombre moyen de profils servis simultanément. Nous observons que la stratégie de recherche équilibre la charge entre les machines. Pour une période de connexion donnée, les machines sont utilisées pour accéder aux même nombre de profils. Il est clair que la charge diminue lorsque la durée de connexion des machines augmente.

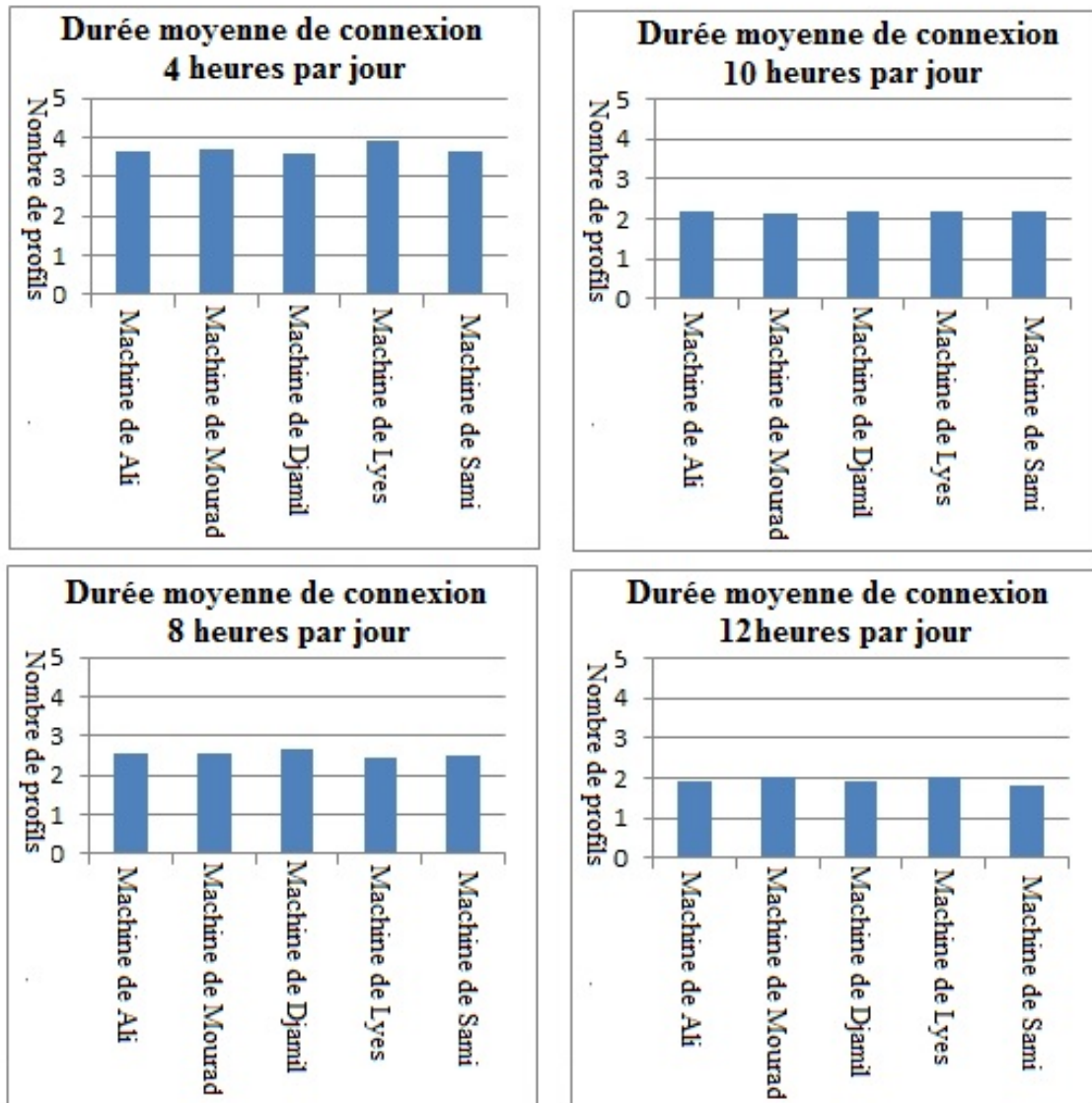


FIGURE 3.9 – Nombre moyen de profils servis simultanément par machine

3.5.3 Discussion

Nous évaluons la disponibilité des profils et l'équilibrage de charge en fonction de la taille du groupe de réplication et de la durée moyenne de connexion. Le taux de disponibilité augmente en fonction du nombre de répliques. L'effet est clair pour les petits groupes : l'ajout d'une nouvelle machine améliore le taux de disponibilité jusqu'à 10%. L'effet de l'ajout d'une nouvelle machine diminue si le groupe est grand. Nous pouvons en arriver au point que l'ajout d'une nouvelle machine n'améliore pas le taux de la disponibilité : la période de connexion de la machine ajoutée est déjà couverte par les autres machines.

Le deuxième facteur qui affecte la disponibilité est la durée de connexion (*le temps que les machines des utilisateurs soient en ligne*). On a remarqué que si les utilisateurs ont une durée de présence suffisante, le taux de disponibilité est acceptable même dans le cas d'un petit groupe de réplication. Nous concluons que la création du groupe de réplication doit être basée principalement sur les informations de connectivité.

Il est déconseillé d'essayer d'étendre le groupe de réplication. Cela a un petit effet sur la disponibilité. De plus, cela peut dégrader la charge du réseau. Chaque opération de mise à jour sur un profil utilisateur génère un message de synchronisation envoyé à toutes les machines en ligne. En outre, les machines doivent rechercher les journaux à chaque démarrage de session. Ils doivent restaurer un journal pour chaque profil stocké.

L'équilibrage de charge est observé pour évaluer la stratégie de recherche. La simulation montre que les machines d'un même groupe de réplication contribuent avec une charge équilibrée. La charge est quantifiée comme le nombre de profils servis simultanément. Bien que les machines soient chargées de stocker un certain nombre de profils, elles agissent rarement comme un serveur pour tous ces profils. Nous admettons que les machines servant des profils plus populaires peuvent supporter une charge supplémentaire. Un mécanisme d'équilibrage de charge qui tient compte de la popularité des profils ou des publications est une piste intéressante.

3.6 Conclusion

DANS ce chapitre, nous avons proposé un modèle pour mettre à jour les profils répliqués dans un réseau social décentralisé. Chaque membre du réseau stocke son profil sur sa propre machine. Ensuite, ce profil peut être répliqué sur d'autres appareils afin d'améliorer sa disponibilité.

De nombreux schémas de réplication sont proposés. Nous nous sommes concentrés sur la charge réseau générée lors de la mise à jour des différentes copies du même profil. Nous avons proposé d'utiliser un fichier journal pour stocker toutes les écritures effectuées sur un profil donné. Chaque machine qui stocke une copie de ce profil doit consulter ce journal après chaque période de déconnexion.

Grâce à ce journal, les nouvelles écritures sont copiées dans toutes les répliques sans copier l'ensemble du profil (*une taille importante au bout d'un moment*). De plus, le fichier journal permet la synchronisation avec des répliques stockées dans deux machines qui ne se rencontrent jamais (*jamais mis en ligne simultanément*).

Afin d'équilibrer la charge entre les machines stockant le même ensemble de profils, nous avons introduit un ordre de priorité. Cet ordre régule le nombre de profils servis simultanément par n'importe quelle machine.

Le prochain chapitre présente une deuxième contribution qui est un nouveau système de recommandation pour les réseaux sociaux P2P.

P2PCF : UN SYSTÈME DE RECOMMANDATION BASÉ SUR LE FILTRAGE COLLABORATIF POUR LES RÉSEAUX SOCIAUX PAIR À PAIR

4.1 Introduction

LES incidents récents de confidentialité rapportés dans les principaux médias à propos des réseaux sociaux mondiaux ont soulevé de réelles préoccupations du public concernant les architectures centralisées. Les réseaux sociaux P2P constituent un paradigme intéressant pour redonner aux utilisateurs le contrôle de leurs données et relations. Alors que les fonctionnalités de base des réseaux sociaux telles que les commentaires, l'abonnement, le partage et la publication de contenu sont largement disponibles, des fonctionnalités plus avancées liées à la recherche d'informations et à la recommandation sont toujours difficiles à implémenter. Cela est dû à l'absence d'un serveur central qui a une vue complète du réseau.

Dans ce chapitre, nous proposons un nouveau système de recommandation pour les réseaux sociaux pair à pair appelé P2PCF.

4.2 Problématique

Contrairement aux réseaux sociaux centralisés qui enregistrent toutes les interactions des utilisateurs dans une grande matrice de notation centralisée, dans laquelle les lignes sont des utilisateurs, les colonnes sont des éléments (*contenu publié dans notre cas*), et chaque cellule rapporte la force de notation d'un utilisateur particulier sur un élément particulier, les réseaux sociaux P2P évitent une telle centralisation et fait pression pour que la matrice soit distribuée entre les nœuds utilisateurs. C'est la raison principale pour laquelle ces systèmes manquent de plusieurs fonctionnalités intéressantes telles que des moteurs de recherche et des recommandations de contenu qui sont responsables de l'acquisition et de la croissance des utilisateurs [130]. En effet, ces systèmes vont rarement au-delà des simples fonctionnalités de publication, de commentaire et de partage.

Dans ce chapitre, nous visons à combler cette lacune et proposons un nouveau cadre pour les systèmes de recommandation décentralisés nommé P2PCF. Notre objectif est

de générer des recommandations pour les utilisateurs sans compromettre leur confidentialité, c'est-à-dire sans avoir accès à l'intégralité des journaux d'interaction générés dans le système par tous les utilisateurs.

Le principal défi rencontré dans les systèmes de recommandation décentralisés est de savoir comment calculer des recommandations significatives à partir de données partielles. Ce problème a limité la portée des techniques de recommandation possibles que l'on peut utiliser. En effet, dans la littérature, les systèmes de recommandation sont classés en trois classes principales [167] : recommandation basée sur le contenu, filtrage collaboratif et systèmes hybrides. Cependant, les systèmes de recommandation basés sur le filtrage collaboratif sont les plus largement utilisés car ils ont obtenu un plus grand succès. Nous distinguons deux types d'approches de filtrage collaboratif : (i) les techniques basées sur la mémoire qui conservent les interactions de l'utilisateur avec les éléments dans une matrice de notation, puis prédisent les évaluations manquantes en utilisant l'approche des k plus proches voisins (Knn). Ces techniques sont assez efficaces en termes de temps d'exécution. (ii) des techniques basées sur des modèles qui ont été introduites plus récemment et reposent principalement sur des techniques d'apprentissage automatique pour prédire les évaluations des utilisateurs. Des exemples de telles techniques comprennent la factorisation matricielle [161], l'analyse sémantique latente [89] et les réseaux de neurones profonds [71]. Cependant, malgré le récent succès obtenu en appliquant des techniques d'apprentissage automatique à différents domaines de recherche tels que le transport [12], la médecine [51,52] et la cyber-sécurité [147], il est largement admis que la majeure partie de ce succès provient de l'apprentissage supervisé qui a profité de la disponibilité récente de grandes quantités de données étiquetées. Bien que cela convienne parfaitement aux systèmes de recommandation centralisés, il est en effet très difficile d'envisager un niveau de réussite similaire dans le contexte de systèmes décentralisés, où seules de petites fractions de données sont disponibles localement pour l'apprentissage automatique. Ainsi, nous avons décidé d'adapter le filtrage collaboratif basé sur la mémoire dans notre système P2PCF, moins gourmand en données.

Les questions de recherche auxquelles nous voulons répondre tout au long de ce chapitre sont les suivantes :

- Est-il possible de générer des recommandations dans un système décentralisé similaires à celles générées dans un système centralisé ? Sinon, quel est l'impact de la décentralisation (*c.-à-d. La préservation de la vie privée*) sur la qualité des recommandations que voient les utilisateurs ?
- Est-il possible de sélectionner un nombre limité d'amis parmi lesquels demander des recommandations ? Si oui, combien et comment les sélectionner ? Ceci est particulièrement important pour réduire le trafic réseau dans les systèmes P2P.
- Comment pouvons-nous regrouper et trier des ensembles de recommandations générées par différents amis avant de les afficher sur le fil d'actualité de l'utilisateur ?

4.3 Background et travaux antérieurs

4.3.1 Systèmes de recommandation

Les systèmes de recommandation sont des techniques utilisées pour proposer et recommander un contenu pertinent aux utilisateurs. Le contenu recommandé peut être un film, une chanson, un livre, un produit sur une plateforme de commerce électronique ou une publication dans un réseau social. Le système de recommandation explore les opinions précédentes (*notes*) des utilisateurs sur un contenu donné pour mieux prédire leurs futures interactions (*notes*) avec le contenu et suggérer le plus approprié. Il existe trois types de systèmes de recommandation : (i) basé sur le contenu, (ii) filtrage collaboratif (*FC*) et (iii) approches hybrides [152].

Dans les systèmes de recommandation basés sur le contenu, le contenu des éléments notés est analysé pour créer des profils et des préférences utilisateur. Ensuite, de nouveaux éléments sont comparés à ces préférences pour recommander les plus intéressants. Les systèmes de filtrage basés sur le contenu génèrent généralement des recommandations basées sur les profils d'utilisateurs pré-construits en mesurant la similarité du contenu avec ces profils à l'aide de certaines métriques vectorielles, telles que la similarité cosinus [167]. Cette approche a été appliquée dans plusieurs ouvrages en particulier ceux qui visent à suggérer un contenu textuel [87, 160].

Dans le filtrage collaboratif, nous utilisons les anciennes évaluations pour prédire la note des utilisateurs sur un nouvel élément. L'idée principale des systèmes basés sur le filtrage collaboratif est que : si les utilisateurs se sont mis d'accord dans le passé (*mêmes éléments, mêmes notes*), ils sont plus susceptibles de se mettre d'accord sur des éléments invisibles à l'avenir [152]. Récemment, les approches basées sur le filtrage collaboratif sont devenues les méthodes les plus pratiques et les plus efficaces dans les systèmes de recommandation, comme en témoignent un certain nombre de produits ou systèmes commerciaux [167], notamment *Netflix*, *Spotify* et *Youtube*. Ce développement a conduit à distinguer deux types de systèmes de filtrage collaboratifs : (i) le filtrage collaboratif basé sur la mémoire et (ii) le filtrage collaboratif basé sur un modèle. Dans le filtrage collaboratif basé sur la mémoire, toutes les évaluations sont stockées dans une matrice (*Utilisateurs/Objets*). Chaque ligne correspond à un utilisateur et chaque colonne correspond à un objet. Les cellules sont des évaluations utilisateur/objet. Lorsque nous souhaitons recommander un nouvel objet à un utilisateur, nous prédisons sa note. Cette prédiction est basée sur les notes attribuées par cet utilisateur à des objets similaires (*FC basée sur les objets*) ou sur la note attribuée au nouvel objet par des utilisateurs similaires (*FC basée sur les utilisateurs*). La figure 4.1 illustre le FC basé sur la mémoire.

Dans un filtrage collaboratif basé sur un modèle, l'idée consiste à dériver un modèle à partir des données de notation historiques. Pour dériver le modèle caché, une variété d'algorithmes d'apprentissage automatique sont utilisés, tels que les réseaux bayésiens, les réseaux de neurones, le clustering et l'analyse sémantique latente (*LSA*) pour n'en citer que quelques-uns [167].

La troisième classe de systèmes de recommandation est hybride, dans laquelle plusieurs techniques sont combinées [121, 131, 145, 169].

Plus récemment, certains systèmes de recommandation ont commencé à utiliser les

	Objet ₁	Objet ₂	Objet _j	Objet _m
Utilisateur ₁	0.35	0	0.65	1
Utilisateur ₂	0.65	1	0	1
⋮	⋮	⋮	⋮	⋮	⋮	⋮
Utilisateur _i	0.35	0.4	???	0.25
⋮	⋮	⋮	⋮	⋮	⋮	⋮
Utilisateur _n	0.35	1	0.65	0.35

FC basé sur les utilisateurs considère les notes des autres utilisateurs sur le même objet

FC basé sur les objets considère les notes de l'utilisateur sur les autres objets

Une note inconnu

FIGURE 4.1 – Filtrage collaboratif basé sur la mémoire

relations sociales comme facteur supplémentaire dans le processus de recommandation. Celles-ci sont appelées systèmes de recommandation sociale. Les relations sociales peuvent être des relations de confiance, des amitiés, des appartenances ou des relations de suivi [152]. Cette définition a été étendue pour inclure «tout système de recommandation qui cible les domaines des médias sociaux» [84]. La nouvelle définition couvre les systèmes recommandant des objets dans les domaines des médias sociaux tels que les publications, les hashtags, les liens, les personnes et les communautés [61, 152]. Il a été démontré que les considérations sociales améliorent les systèmes de recommandation car il existe un chevauchement important entre les intérêts des utilisateurs et leurs relations [135, 151, 164].

4.3.2 Propagation de contenu dans les réseaux sociaux P2P

Dans les réseaux sociaux P2P, les données sont stockées sur les machines des utilisateurs [33, 99] ou répliquées dans des machines de confiance [49]. La découverte des données est l'un des problèmes des systèmes P2P, car il est très difficile d'obtenir une vue d'ensemble de toutes les ressources du réseau [85]. Les techniques de propagation des données dans les réseaux sociaux P2P sont classées en deux grandes classes : (i) diffusion active [118], et (ii) demande-réponse [82].

La diffusion active signifie qu'un utilisateur qui produit des mises à jour les pousse

vers le réseau. Dans [118], les auteurs proposent un protocole de potins (Gossip protocol) où les utilisateurs envoient périodiquement du contenu à leurs voisins. Ce protocole est amélioré par l'introduction de techniques heuristiques pour sélectionner les voisins cibles [60]. Un protocole communautaire est proposé dans [47]. Il s'appuie sur un index central pour détecter les communautés. Les mises à jour sont envoyées à un voisin de chaque communauté. Ensuite, les mises à jour sont diffusées au sein des communautés. GoDisco [54] exploite le contexte sémantique pour sélectionner les amis concernés par chaque message de mise à jour. Cette méthode nécessite que les utilisateurs informent également leurs voisins de leurs intérêts. Enfin, dans [150], les interactions des utilisateurs sont utilisées pour modéliser la force des relations dans un domaine d'intérêt donné.

Dans la diffusion Demande-Réponse, un utilisateur à la recherche de mises à jour doit envoyer des demandes pour trouver les profils de ses amis. La structure réseau (*essentiellement DHT*) est chargée de diriger cette demande vers le nœud stockant le profil (*appareil utilisateur ou répliques actives*). Plusieurs systèmes utilisent ce protocole, notamment : PeerSon [33], LifeSocial.KOM [78], DECENT [92] et SOUP [100]. Les auteurs de [82] considèrent Cachet [126] comme une technique hybride qui introduit la mise en cache sociale pour améliorer les performances de DECENT. Dans DECENT [92], un algorithme de recherche est lancé via DHT pour collecter toutes les mises à jour disponibles. Ensuite, un fil d'actualité est créé et présenté à l'utilisateur. Dans Cachet, un algorithme de mise en cache sociale basé sur les potins est également exploité : lorsqu'un utilisateur disponible est invité à effectuer des mises à jour, il renvoie ses mises à jour appropriées et d'autres mises à jour d'amis en cache.

Dans notre travail, nous adoptons le mécanisme de demande-réponse pour notre système de recommandation de filtrage collaboratif P2P. Les utilisateurs répondent avec un contenu spécifique à chaque demande qu'ils reçoivent. De plus, les interactions des utilisateurs sont stockées localement et utilisées pour recommander des mises à jour pertinentes.

4.3.3 Recommandation dans les réseaux P2P

Les auteurs de [65] proposent P2PRec, un algorithme de recommandation hybride pour améliorer la récupération d'informations dans le système de partage de fichiers P2P combinant le filtrage collaboratif et le filtrage basé sur le contenu. Le système proposé extrait les sujets des documents et les utilise pour modéliser l'expertise des utilisateurs. Lorsqu'une requête est faite, les sujets concernés sont d'abord déduits, puis la demande n'est acheminée qu'aux utilisateurs qualifiés comme experts dans ces sujets. Dans F2Frec [64], les relations sociales sont considérées pour améliorer les résultats. Les amis les plus pertinents sont sélectionnés pour recommander des documents. Dans ce système, la distance entre les utilisateurs est calculée en combinant les deux éléments : les réseaux d'amitié et les sujets d'intérêt. Les deux systèmes, F2Frec et P2PRec, sont une amélioration des algorithmes d'inondation classiques [64]. Cependant, ils ont tous deux besoin de calculs supplémentaires pour extraire les sujets des demandes et des documents. Ce processus est plus compliqué sur les publications sur les réseaux sociaux (*texte court et abréviations*).

Dans [58], les auteurs proposent un système de recommandation décentralisé pour les appareils mobiles. Ici, la note des utilisateurs est diffusée à leurs voisins (*point de*

vue distance). Chaque utilisateur stocke et gère l'évaluation des autres utilisateurs sur son appareil et explore sa base de données locale pour prédire les évaluations. Ainsi, aucun serveur central n'est nécessaire. Le système est moins coûteux car basé sur la communication directe, mais n'a pas de profondeur historique : seules les notes de la même période sont capturées. En outre, ce système est implémenté pour recommander des objets à voir dans un musée. Dans [97], l'échange de fichiers multimédias est amélioré par l'introduction du concept de communauté d'intérêts. À partir d'un routage d'inondation, les pairs sont classés en communautés, selon la sémantique de leurs fichiers partagés ; les demandes sont ensuite acheminées vers les communautés concernées uniquement. Le défi de ce système est la composition décentralisée des communautés lorsque le nombre d'utilisateurs est important. En outre, les systèmes doivent suivre l'évolution des intérêts des utilisateurs, ce qui nécessite la reconstruction des communautés. Les auteurs dans [86] proposent un système de recommandation décentralisé pour faire face aux problèmes de montée en charge. Leur solution est de partitionner la matrice de notation et de stocker ses différents fragments dans un réseau P2P basé sur DHT. Ce système évite de surcharger un serveur central mais la confidentialité des utilisateurs n'est pas garantie puisque leurs notes sont copiées en clair dans le réseau P2P.

Pour recommander des produits dans une plateforme de commerce mobile, un autre système de recommandation P2P est proposé dans [158]. L'auteur propose de mettre en œuvre un filtrage collaboratif à l'aide de Gnutella (*un algorithme basé sur l'inondation*). L'évaluation des utilisateurs/produits est stockée sous forme de vecteur dans chaque pair et l'algorithme d'inondation est utilisé pour collecter des vecteurs similaires. Chaque pair qui reçoit le vecteur de requête évalue la similarité entre la requête et les vecteurs mis en cache avant que des vecteurs similaires ne soient renvoyés. L'utilisateur à la recherche de recommandations recueille des vecteurs similaires et leur applique l'algorithme de filtrage collaboratif. Cette approche est une bonne adaptation du concept de filtrage collaboratif pour les systèmes P2P, cependant, elle hérite de nombreux inconvénients des algorithmes d'inondation tels que le fait d'être non déterministe et de consommer beaucoup de données sur le réseau. En outre, le message mis en cache peut contenir des données expirées.

L'inondation des vecteurs de notation est également utilisée dans [172]. Les utilisateurs stockent leurs vecteurs de notation et la requête est également un vecteur de notation. Une fois la requête diffusée sur le réseau, les éléments recommandés sont renvoyés dans le même chemin. Chaque pair doit collecter et agréger les évaluations des articles avant de les envoyer à son voisin. Cette approche peut être critiquée en raison du temps d'attente considérable dans chaque nœud avant de renvoyer l'élément recommandé à son voisin.

En tant qu'implémentations P2P du filtrage collaboratif basé sur un modèle, des approches de factorisation matricielle décentralisée sont proposées dans [67] et [42]. Cette méthode pousse le calcul du modèle de recommandation vers les appareils des utilisateurs. De plus, les pairs ne doivent pas partager toute la notation stockée dans leur matrice locale.

4.4 Solution Proposée

Nous proposons une adaptation de la technique de filtrage collaboratif au contexte des réseaux sociaux P2P (voir [22, 24]). Le nouveau système est décrit en détails dans les paragraphes suivants.

4.4.1 Aperçu général

Dans cette section, nous décrivons l'architecture générale de notre solution P2PCF : un système de recommandation basé sur le filtrage collaboratif pour les réseaux sociaux pair à pair. Lorsqu'un utilisateur rejoint le réseau, P2PCF se lance pour sélectionner les publications les plus intéressantes publiées par ses amis. Les publications collectées sont utilisées pour créer un fil d'actualité similaire à ce que nous voyons dans les réseaux sociaux traditionnels tels que *Facebook* et *Twitter*. Pour cela, P2PCF utilise une version adaptée du filtrage collaboratif, la technique la plus répandue dans le domaine des systèmes de recommandation sociale.

Le premier défi est l'absence d'une entité centrale où stocker les avis des utilisateurs sur les publications (*matrices de notation*). De plus, les réseaux sociaux P2P sont basés sur des contraintes de confidentialité strictes.

Pour résoudre ces défis, nous proposons tout d'abord de stocker les notes dans un schéma décentralisé dans lequel chaque utilisateur stocke une matrice locale de notes ou d'interactions sur son contenu. Ensuite, le processus de recommandation est défini en trois étapes :

- **Sélection des amis** : exécutée du côté du demandeur (*l'utilisateur pour qui nous générons des recommandations*) pour sélectionner un ensemble de ses amis qui contribueraient aux recommandations.
- **Sélection des publications** : à exécuter du côté des amis. Chaque ami sélectionné répond avec un ensemble de publications jugées pertinentes aux intérêts du demandeur. Cette sélection est basée sur la matrice locale de notation stockée au sein de chaque ami, c'est-à-dire que chaque ami évalue la pertinence des publications en fonction des notes précédentes données par le demandeur.
- **Agrégation des recommandations** : cette étape est responsable de la fusion de toutes les recommandations reçues de différents amis. Alors que chaque ami répond avec une liste de publications et leur score de pertinence, calculé à partir de leur matrice de notation locale, P2PCF doit homogénéiser ces scores en intégrant le concept de force d'amitié, afin de choisir les publications qui formeraient le fil d'actualité de l'utilisateur.

4.4.2 Scénario d'illustration

P2PCF est exécuté dans chaque pair sans aucune entité centrale. Le processus de recommandation est divisé en plusieurs étapes :

- L'utilisateur u sélectionne un sous-ensemble d'amis pour contribuer à la recommandation.
- L'utilisateur u envoie des demandes séparées aux amis sélectionnés.
- Chaque ami v qui reçoit la demande applique un filtrage collaboratif local pour sélectionner un ensemble de publications à recommander.
- Les publications sélectionnées sont ensuite renvoyées à l'utilisateur u .
- L'utilisateur u recueille les réponses des amis, les agrège et crée un fil d'actualité.

La figure 4.2 illustre le flux de travail de ce processus de recommandation.

Dans les sous-sections suivantes, nous détaillerons les différents composants de P2PCF. Nous commençons par expliquer comment les données sont stockées dans le réseau P2P. Ensuite, nous expliquons le processus de recommandation. Enfin, nous discutons la façon dont le problème de démarrage à froid est géré.

4.4.3 Stockage décentralisé des interactions

Le principal défi de notre travail est de produire des recommandations sur une matrice de notation distribuée entre pairs. Dans ce cas, chaque pair a accès à une matrice qui rapporte les notes faites par ses amis sur ses propres publications uniquement. Le Tableau 4.1 montre un exemple d'une telle matrice. En d'autres termes, les interactions que fait un utilisateur John sont stockées dans les matrices de ses amis d'une manière qu'aucun de ses amis n'a accès à toutes les interactions et préférences de John. Il est important de souligner le caractère décentralisé des matrices de notation locales car aucune entité centrale n'a accès à l'ensemble des données générées dans le réseau social, ce qui constitue une différence majeure avec les systèmes de recommandations standards mis en œuvre dans les réseaux sociaux largement utilisés. En fait, notre approche attend de chaque utilisateur qu'il stocke les réactions se déroulant sur son mur, ce qui augmente considérablement la confidentialité des utilisateurs.

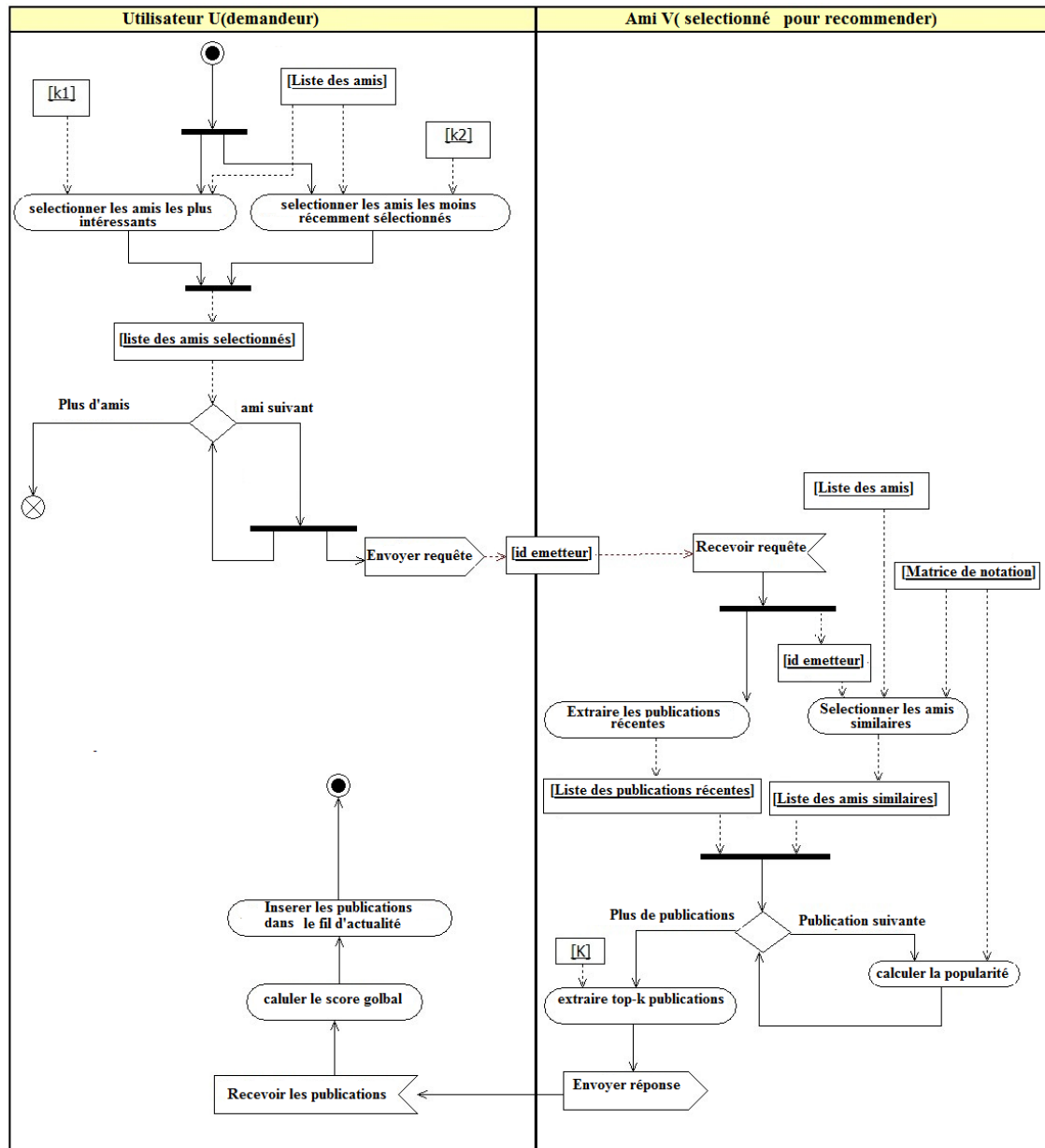


FIGURE 4.2 – Le flux de travail du processus de recommandation

TABLE 4.1 – Exemple de matrice de notation locale

Amis	Publications					
	P1	P2	P3	P4	P5	P6
John	0.65	0	0	0.30	0	1
Sarah	0	0	1	0.66	0.35	0
Brahim	0.75	0.4	0	1	0.65	0

Les notes de cette matrice représentent un score global des différentes actions que les utilisateurs effectuent sur chaque publication. Ces actions comprennent : aimer, partager, et commenter. Pour se conformer au schéma CF adopté dans cette recherche, nous proposons de traduire les réactions des utilisateurs (*partage*, *commentaire*, *like*) en notes numériques.

Une pratique courante dans les réseaux sociaux consiste à attribuer des poids différents à différentes actions en fonction du niveau d’engagement qu’elles induisent [134]. Par exemple, il est largement admis que partager une publication est plus engageant que publier un commentaire dessus, ce qui est encore plus engageant que de l’aimer [134].

Notre système de stockage distribué et préservant la confidentialité, associé à la méthodologie de représentation et de conversion des interactions des utilisateurs en valeurs numériques, nous permettra de générer plus facilement des recommandations significatives, comme nous le verrons dans la sous-section suivante.

4.4.4 Les étapes de P2PCF

Comme mentionné précédemment, une fois que les interactions sont capturées au sein des différents pairs sous forme de matrices de notation locales, on peut exploiter le réseau d’amis P2P pour générer des recommandations comme suit.

4.4.4.1 Sélection des amis

La première étape de notre processus de recommandation est la sélection des amis, qui a lieu du côté du demandeur. Les réseaux sociaux distribués existants tels que DECENT [92] et Cachet [126] chargent le fil d’actualité des utilisateurs avec un flux de publications générées par tous ses amis, ce qui conduit à la recommandation d’un contenu peu important. En effet, il est bien établi que les utilisateurs des réseaux sociaux n’accordent pas le même intérêt aux contenus publiés par tous leurs amis [46, 150, 171].

Ainsi, P2PCF s’appuie sur ces résultats pour ne pas envoyer la demande de recommandation à tous les amis. L’idée est de supposer que si un utilisateur u a exprimé son intérêt pour le contenu d’un utilisateur v dans le passé, il y a une forte attente que u sera intéressé par le contenu de v dans le futur. Les réactions des utilisateurs à un contenu donné peuvent être une bonne indication de leur niveau d’intérêt. Ces réactions doivent être enregistrées pour valoriser les amis qui partagent un contenu intéressant. Ainsi, nous proposons de créer un compteur pour chaque ami : *Reaction_count*. Ce compteur est incrémenté à chaque fois qu’un utilisateur réagit à une publication d’un ami donné. Ceci est particulièrement important, car les réactions réelles de l’utilisateur sur les publications de ses amis sont stockées du côté des amis, ce qui rend difficile pour le système de récupérer les décomptes si aucun compteur n’est utilisé.

Étant donné que les intérêts des utilisateurs peuvent changer au fil du temps, le système de recommandation doit prendre en compte les intérêts réels (*observés dans les réactions récentes*) plus que les réactions archivées. Notre solution à ce problème consiste à enregistrer le temps de la dernière réaction de l'utilisateur sur le mur de chacun de ses amis : *Last_reaction*. Cela permet au système de capturer des amis émergents qui ont un nombre d'interactions inférieur à celui des autres.

Pour chaque ami, nous calculons le niveau d'intérêt comme le *Reaction_count* décrétement en fonction de l'intervalle entre *Last_reaction* et l'heure actuelle. Nous proposons la formule (1) pour calculer ce niveau d'intérêt.

$$Interest_level = Reaction_count \times e^{-\alpha \times Delta(t)} \quad (4.4.1)$$

Où $Delta(t)$ est le temps en secondes depuis la dernière réaction, $Reaction_count(u,v)$ est l'accumulation d'interactions faites par l'utilisateur u sur les publications de v . α est un paramètre ajustable.

Les amis sont ensuite triés en fonction de leur niveau d'intérêts (*Interest_level*). Les amis ayant le niveau d'intérêt le plus élevé (*Top-k1*) sont sélectionnés pour être contactés pour des recommandations de contenu.

Cette stratégie alimente le fil d'actualité des utilisateurs avec des publications récentes publiées par leurs meilleurs amis. Cependant, cela peut conduire au problème de la «*bulle de filtres*» [125], dans laquelle l'utilisateur voit les publications du même sous-ensemble d'amis. Pour éviter cette situation, nous devons donner une chance à tous les amis afin de contribuer au processus de recommandation, ce qui augmentera à terme la diversité et la couverture des recommandations. Pour cela, nous proposons de consolider la liste des amis sélectionnés par d'autres candidats choisis en fonction de leur historique de sélection. C'est-à-dire que pour chaque ami, nous stockons la date de la dernière fois qu'il a été sélectionné/sollicité pour des recommandations. Les amis de cette liste sont triés du moins récemment sollicité au plus récemment sollicité, les *Top-k2* des utilisateurs de cette liste sont ajoutés à la liste initiale des utilisateurs.

Comme le montrent les figures 4.2 et 4.3, la liste des amis sélectionnés est la fusion de deux listes : *top-k1 des amis les plus intéressants et top-k2 des amis les moins récemment sélectionnés*. k_1 et k_2 sont des valeurs empiriques.

4.4.4.2 Sélection des publications

Dans cette section, nous expliquons comment sélectionner les publications les plus pertinentes de chaque ami. Nous supposons que l'utilisateur v a été sélectionné dans la phase précédente, et a reçu une demande de recommandation d'un utilisateur u . Dans ce cas, v devrait répondre avec ses publications qui ont été bien notées par des utilisateurs similaires à u . C'est le principe de base du filtrage collaboratif. La similarité entre u et les utilisateurs de la matrice de notation de v est calculée de manière locale. En effet, deux utilisateurs u_1 et u_2 peuvent être très similaires dans la matrice de notation locale de v_1 , alors qu'ils peuvent devenir très différents au sein de la matrice de notation locale de v_2 . C'est par exemple le cas lorsque deux utilisateurs partagent le même intérêt pour les genres musicaux ; par conséquent, ils sont similaires au sein de la matrice de leur ami commun expert en musique, alors qu'ils sont fans de différents clubs sportifs, ce qui se traduit par une faible similarité dans la matrice de notation de leur ami expert

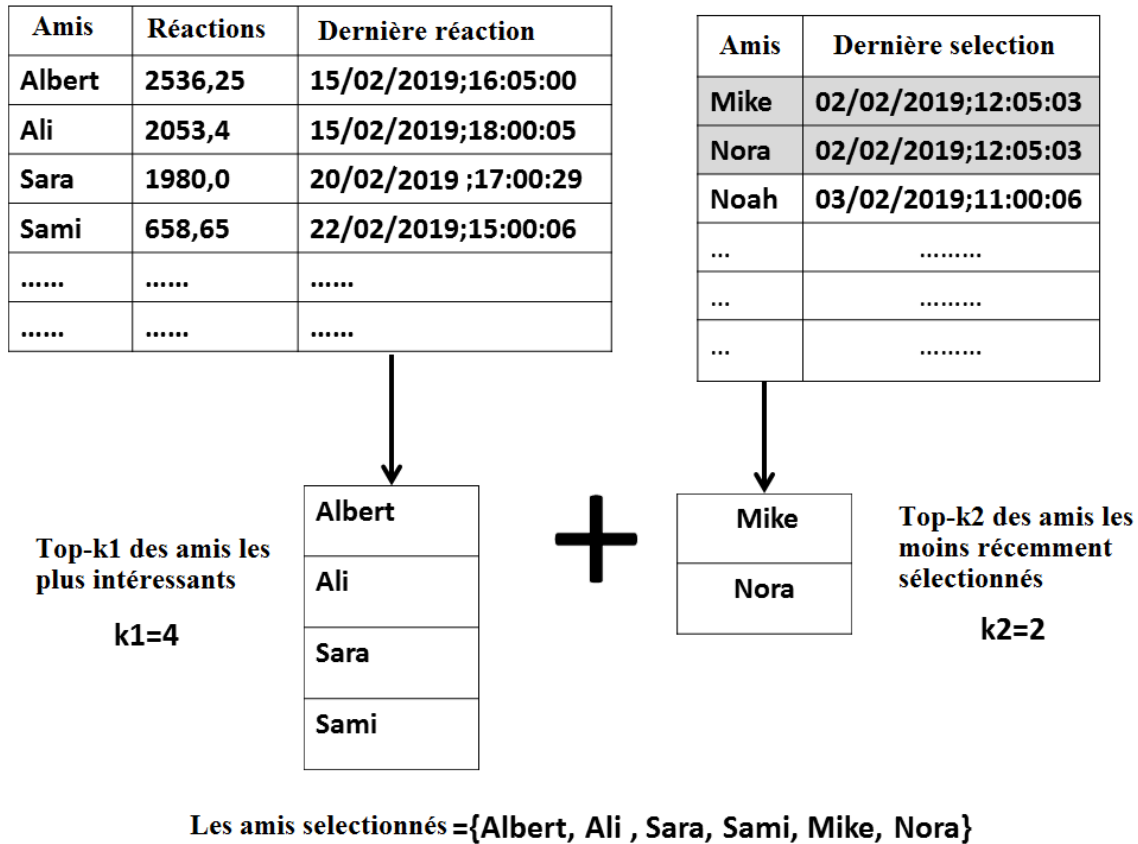


FIGURE 4.3 – Stratégie de sélection des amis

en sport. Il s'agit d'une fonctionnalité très intéressante qui n'est pas capturée dans les schémas de filtrage collaboratifs centralisés.

Pour calculer la similarité entre les utilisateurs, nous adoptons la formule de similarité cosinus [43](2) :

$$Sim(u, f) = \frac{\sum_{P \in RRp} Rating(u, p) \times Rating(f, p)}{\sqrt{\sum_{P \in RRp} Rating(u, p)^2} \times \sqrt{\sum_{P \in RRp} Rating(f, p)^2}} \quad (4.4.2)$$

Où u est le demandeur, f est un ami de l'utilisateur v recevant une demande, RRp est l'ensemble des publications récemment notées par le demandeur u sur le mur de l'utilisateur v . $Rating(u, p)$ est la note donnée par l'utilisateur u à la publication p .

Après réception d'une requête, la formule ci-dessus est utilisée pour calculer la similarité entre l'utilisateur demandeur u et tous les amis connectés au récepteur v . On considère les utilisateurs avec une similarité supérieure à un seuil prédéfini ($MinSim$); ainsi, nous créons SF : l'ensemble des amis les plus similaires.

$$SF(u) = \{ f \in friends(v) \text{ s.t. } sim(u, f) > MinSim \}$$

L'étape suivante consiste à sélectionner le sous-ensemble pertinent de publications

produites par v , en fonction des notes fournies par l'ensemble d'amis similaires SF . À cette fin, nous introduisons la popularité des publications que nous définissons comme une moyenne des notes fournies par chaque utilisateur de l'ensemble SF pondérée par la similarité de cet utilisateur avec le demandeur u . Le score de popularité est calculé selon l'équation (3) :

$$PopSim(u, p) = \frac{\sum_{f \in SF(u)} Sim(u, f) \times Rating(f, p)}{\sum_{f \in SF(u)} Sim(u, f)} \quad (4.4.3)$$

Où $PopSim(u, p)$ est la popularité de la publication p selon les amis similaires du demandeur u ; $Sim(u, f)$ est le niveau de similarité entre le demandeur u et un ami f (formule (2)) et $Rating(f, p)$ est la note attribuée par f à la publication p .

Comme le montre la figure 4.4, l'étape suivante consiste à trier les publications en fonction de leurs popularités. Les publications avec les scores de popularité les plus élevés sont renvoyées au demandeur.

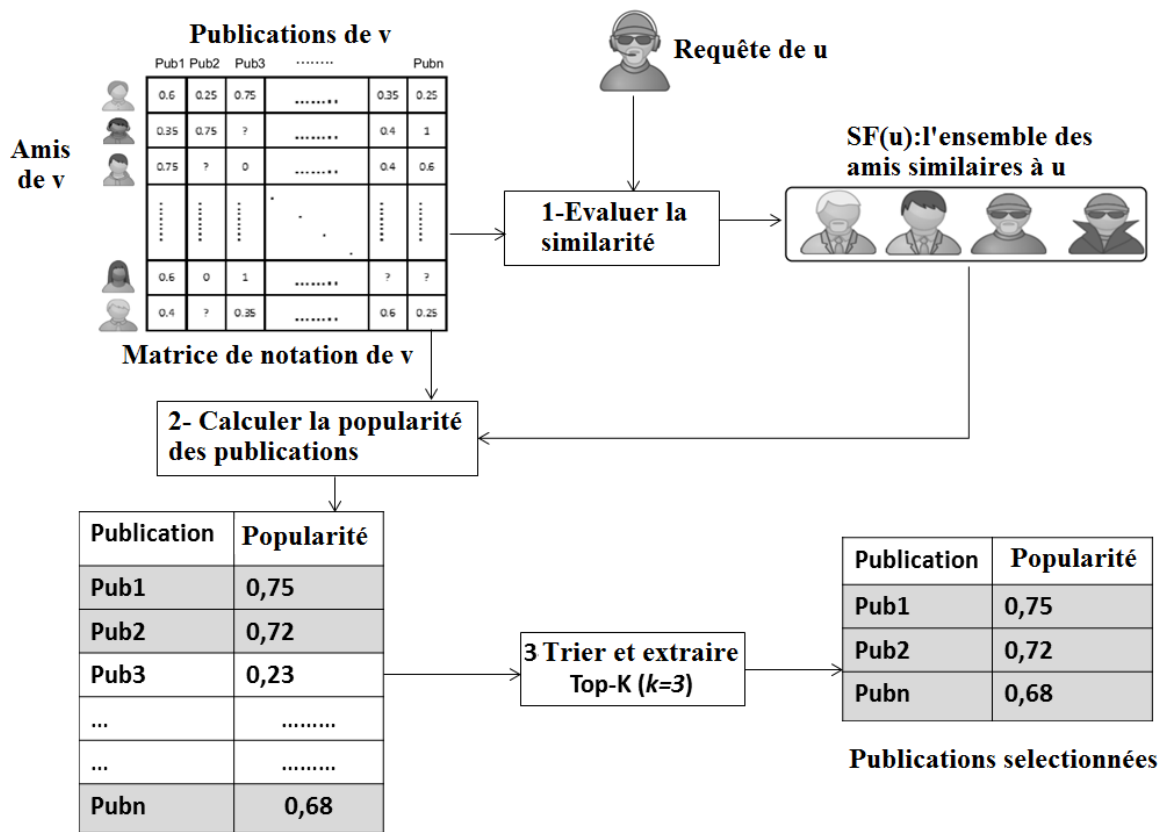


FIGURE 4.4 – Sélection des publications basée sur les notes des amis similaires

4.4.4.3 Agrégation des recommandations

Les publications recommandées à partir de différents (*amis*) doivent être évaluées et organisées avant d'être insérées dans le fil d'actualité des utilisateurs. Une évaluation

globale doit être appliquée pour exposer le contenu le plus pertinent à l'utilisateur qui a demandé des recommandations. Une autre préoccupation est de détecter les recommandations injustes. Par exemple, dans [50], une solution utilisant des règles d'association est proposée pour détecter les recommandations injustes basées sur le concept de confiance. Dans notre cas, nous nous basons sur le niveau d'intérêt calculé à l'aide de la *(formule(1))* qui intègre certains aspects de la confiance. En effet, il est facile de vérifier que les niveaux de confiance sont proportionnels à la fréquence des interactions. Nous proposons de calculer un score global de popularité pour mettre à profit les publications recommandées par les utilisateurs pour lesquels l'utilisateur demandeur a plus d'intérêt (*amis proches*). Nous proposons de pondérer la popularité calculée par chaque ami (*formule(3)*) par le niveau d'intérêt donné par le demandeur à l'ami selon la *formule(4)* :

$$Score(p) = PopSim(P) \times Interest_Level(u, v) \quad (4.4.4)$$

Compte tenu de cela, une publication avec un score de popularité élevé recommandée par un ami moins intéressant sera décalée au profit d'une publication recommandée par un ami plus intéressant.

Lorsque les utilisateurs communiquent via l'infrastructure P2P, les réponses des amis arrivent de manière asynchrone. Cela est dû à la structure du réseau et aux différentes capacités de calcul hétérogènes des appareils utilisateur. Pour rendre la latence transparente pour l'utilisateur qui reçoit des recommandations, les publications sont placées dans une file d'attente à priorités en fonction du score calculé. Ensuite, les publications sont périodiquement insérées dans le fil d'actualité de l'utilisateur (*voir Figure 4.5*).

4.4.5 Problème de démarrage à froid

L'un des principaux défis des systèmes de recommandation de filtrage collaboratif est le problème du démarrage à froid [76]. On est confronté à ce problème lorsque les informations nécessaires qui permettent la recommandation sont rares ou introuvables. Dans cette section, nous décrivons un nouveau mécanisme dans notre solution proposée pour surmonter le problème de démarrage à froid qui peut survenir à la fois pour les nouveaux utilisateurs et pour les publications nouvellement publiées.

4.4.5.1 Le cas des nouveaux utilisateurs

L'utilisateur en démarrage à froid dans les réseaux sociaux est illustré dans [63] comme un utilisateur qui n'a pas d'amis ou de publications notées. De nombreuses approches ont été proposées pour résoudre ce problème [76], où l'idée principale est d'exploiter des informations externes dans le processus de recommandation. Cependant, comme les utilisateurs des réseaux P2P ne partagent pas beaucoup d'informations sur leur identité ou leurs intérêts, les approches existantes pour résoudre le problème du démarrage à froid ne sont pas applicables.

Dans notre situation, tout nouvel utilisateur qui rejoint le système est considéré comme un utilisateur en démarrage à froid. Nous considérons également les utilisateurs qui ont moins de x amis avec une similarité supérieure à un seuil prédéfini en tant qu'utilisateurs en démarrage à froid. Dans ce cas, la *formule (3)* de la popularité des publications n'est pas pertinente. Le récepteur v n'a pas suffisamment d'informations

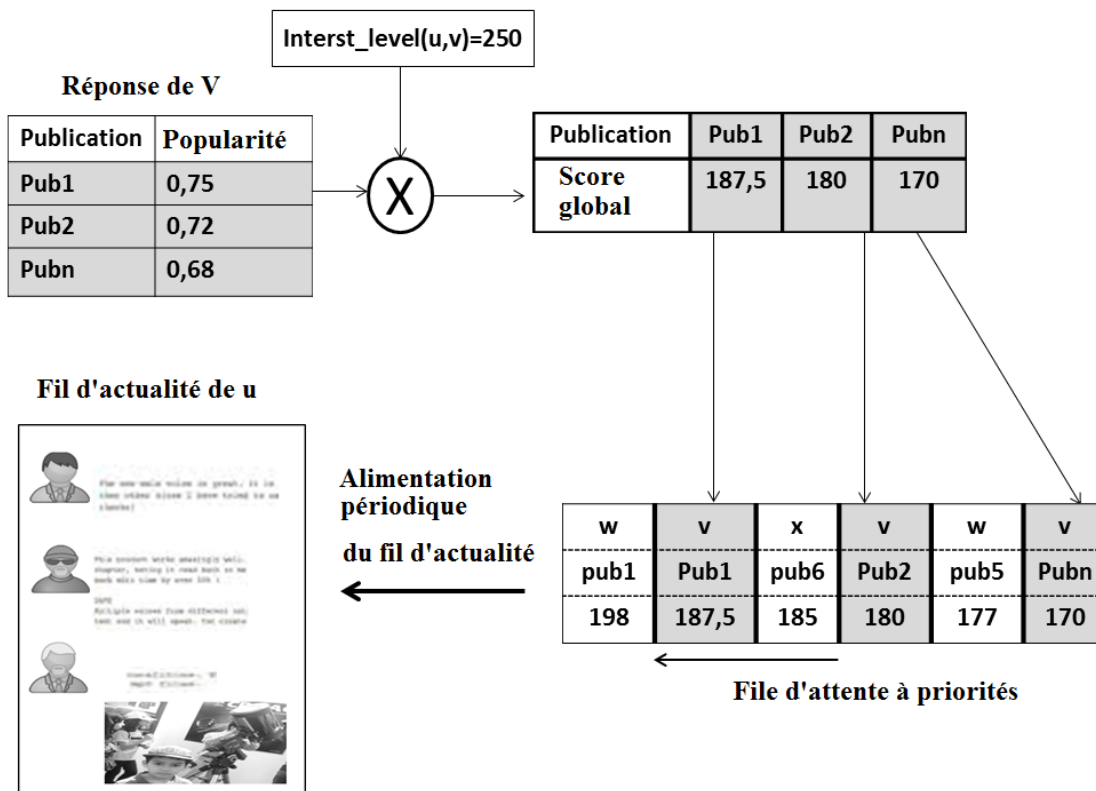


FIGURE 4.5 – Agrégation des publications et alimentation du fil d’actualité

sur les intérêts du demandeur u pour calculer les popularités et recommander des publications.

En guise de solution, nous proposons que le destinataire de la requête, v , renvoie les publications très appréciées par tous ses amis. Cela renverra le contenu le plus populaire au demandeur. Ainsi, nous proposons que l'utilisateur v calcule un score de popularité pour ses propres publications, en considérant les notes de tous ses amis. Ceci est illustré dans la formule (5) :

$$Pop(u, p) = \frac{\sum_f Rating(f, p)}{Nbf} \tag{4.4.5}$$

Où Nbf est le nombre d'amis connectés à l'utilisateur v . Notez que cette formule considère tous les amis (f) de l'utilisateur v .

4.4.5.2 Le cas des nouvelles publications

Afin de donner aux nouvelles publications une chance d'être recommandées, il est important de mettre en place un processus qui contourne les équations (3) et (5). En effet, lorsqu'une publication est publiée pour la première fois, elle n'a aucune note. Dans ce cas, chaque nouvelle publication obtient un score de popularité aléatoire compris entre 0 et 1. En même temps, nous définissons un compteur pour suivre le nombre de

fois où la publication a été recommandée. Une fois que le compteur atteint un nombre prédéfini de vues (*par exemple, 5*), nous remplaçons l'évaluation aléatoire initiale par les évaluations réelles que la publication a reçu. De cette manière, nous nous assurons que les nouvelles publications ne sont favorisées que dans les premiers stades sans pénaliser les anciennes publications.

4.5 Evaluation des performances

Dans cette section, nous évaluons la mise en œuvre décentralisée proposée du système de recommandation basé sur le filtrage collaboratif pour les réseaux sociaux P2P (*P2PCF*). Notre objectif principal ici est d'étudier le comportement de notre technique décentralisée par rapport à un système de filtrage collaboratif centralisé classique. Pour ce faire, nous nous intéressons à deux aspects : (i) le rappel qui mesure la capacité du système à restituer les publications attendues, (ii) la couverture, qui est le pourcentage de publications sélectionnées par le système de recommandation.

4.5.1 Le jeu de données

L'ensemble de données utilisé est extrait à partir de *Twitter*. Nous avons filtré un échantillon de tweets traitant les événements politiques en Algérie en 2019. Pour chaque tweet, les données collectées contiennent l'utilisateur qui a publié le tweet et les utilisateurs qui ont retweeté (*partagé*) ce tweet. Les tweets collectés sont ensuite utilisés pour construire un réseau social. Nous générons un lien d'amitié entre deux utilisateurs *A* et *B* si *A* a retweeté au moins un tweet publié par *B*. Le tableau 4.2 décrit certaines caractéristiques du réseau social.

TABLE 4.2 – Résumé des statistiques des données de test

Nœuds (<i>tous les utilisateurs</i>)	40,122
Auteurs de publications	5,714
Liens (<i>amitié</i>)	57,222
Publications(<i>tweets</i>)	13,158
Réactions(<i>retweets</i>)	73,002

Le réseau social est composé de 40 122 utilisateurs au total, dont 5 714 utilisateurs sont auteurs de publications.

4.5.2 Détails d'implémentation

Pour mettre en œuvre le système de filtrage collaboratif, nous devons créer la matrice de notation. Dans un système centralisé, une matrice globale est nécessaire. Chaque ligne de la matrice représente un utilisateur et chaque colonne représente une publication (*un identifiant de tweet dans notre cas*). La note est une valeur booléenne : 1 si l'utilisateur a retweeté la publication, 0 sinon. Dans le système de recommandation décentralisé, chaque utilisateur gère sa propre matrice locale, qui se limite aux notes de

ses amis sur ses publications (*tweets*).

Dans le cadre du RS-P2P, la recommandation se limite aux publications d'amis. Nous respectons cette restriction même dans la mise en œuvre centralisée. Autrement dit, pour chaque utilisateur, l'algorithme de recommandation explore toute la matrice de notation, mais ne recommande que les publications des amis.

Les différences fondamentales entre P2PCF et la référence centralisée sont mises en évidence ci-après :

- **Similarité entre le demandeur et les autres utilisateurs** : la similarité est calculée en fonction de la note précédente des utilisateurs sur l'ensemble du réseau. Dans P2PCF, les amis destinataires des demandes utilisent leurs matrices locales pour calculer la similarité locale entre le demandeur et d'autres utilisateurs similaires.

- **Décision de démarrage à froid** : certains utilisateurs en démarrage à froid (*en situation décentralisée*) ne sont pas qualifiés en démarrage à froid dans l'approche centralisée. Dans un schéma décentralisé, un utilisateur ayant moins d'interactions avec un ami donné est considéré en démarrage à froid par cet ami. Le serveur central gère toutes les interactions des utilisateurs dans le réseau, il a donc accès à plus de données sur l'utilisateur.

4.5.3 Rappel

Il est important d'observer le rappel de notre système par rapport à l'approche centralisée. Dans cette expérience, et afin d'avoir une comparaison équitable, tous les amis sont sélectionnés. Nous évaluerons plus tard l'effet de la sélection d'un sous-ensemble d'amis sur les performances de l'algorithme.

Nous imputons 20% des publications notées par chaque utilisateur, choisies au hasard. Nous exécutons ensuite le processus de recommandation. Le rappel est la capacité du système à recommander ces publications supprimées. Il est calculé comme le pourcentage des publications attendues (*les publications imputées*) qui sont correctement recommandées par le système. Les publications recommandées sont triées selon la formule (4) et le rappel est calculé pour différentes tailles d'ensembles de recommandations. Le rappel est calculé pour chaque utilisateur et la moyenne des rappels est indiquée à la figure 4.6.

4.5.4 Couverture

Nous calculons le niveau de couverture comme le pourcentage de publications qui ont été recommandées (*au moins une fois*). Cette métrique montre la capacité du système de recommandation à donner une chance à toutes les publications d'être vues par les utilisateurs. La figure 4.7 montre le niveau de couverture dans les systèmes centralisés et décentralisés.

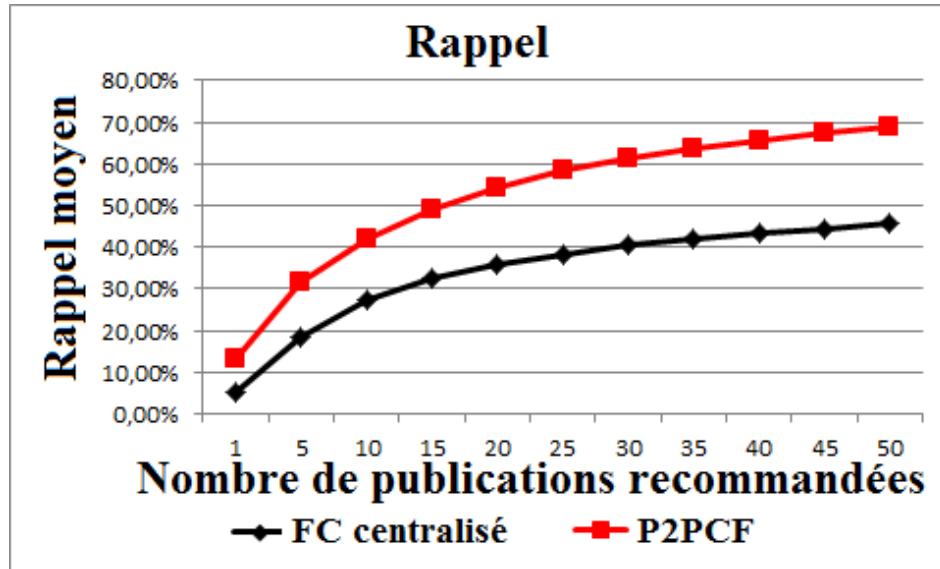


FIGURE 4.6 – Le Rappel moyen

4.5.5 Sélection des amis

Dans les deux expériences précédentes, nous avons comparé les approches centralisée et décentralisée dans des conditions similaires. Nous n'avons pas limité le nombre d'amis concernés par les recommandations. De plus, toutes les publications pertinentes à recommander sont renvoyées au demandeur. En pratique, envoyer une demande de recommandation à tous les amis coûte cher en termes de surcharge du réseau. Ainsi, nous évaluons dans cette section l'effet du nombre d'amis sélectionnés sur le système de recommandation. La figure 4.8 montre le rappel moyen en fonction du pourcentage d'amis sélectionnés.

De même, nous rapportons le niveau de couverture en fonction du pourcentage d'amis sélectionnés dans la figure 4.9.

4.5.6 Sélection des publications

Lorsqu'un ami est sollicité pour une recommandation, il doit répondre en envoyant au demandeur un ensemble de publications pertinentes. Le nombre de publications renvoyées affecte la surcharge du réseau et la qualité de la recommandation. Dans cette expérience, nous étudions l'effet de la taille des ensembles de publications recommandées sur les performances de notre système. La figure 4.10 montre les scores de rappel en fonction du pourcentage de publications sélectionnées par chaque ami sollicité.

Pour montrer l'effet de la limitation du nombre de publications à recommander sur la visibilité des publications, nous reportons sur la figure 4.11 la fonction de niveau de couverture du nombre de publications demandées.

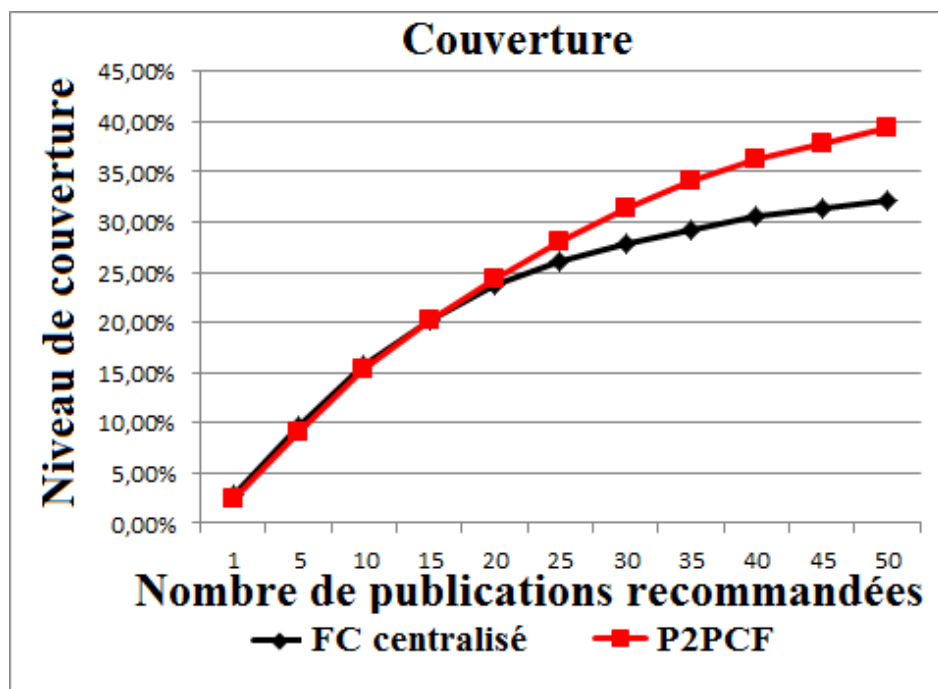


FIGURE 4.7 – Niveau de couverture

4.5.7 Discussion

Le premier constat que nous pouvons faire ici est que l’algorithme de filtrage collaboratif peut être efficacement implémenté de manière décentralisée pour enrichir les réseaux sociaux pair à pair.

Dans la première expérience, nous définissons l’algorithme avec les meilleurs paramètres : tous les amis sont sélectionnés et toutes les publications pertinentes sont renvoyées à l’utilisateur. La figure 4.6 montre que le rappel du P2PCF est meilleur que le CF centralisé. L’amélioration de la couverture est également illustrée à la figure 4.7. Cette expérience prouve l’utilité de répartir la tâche de filtrage entre les utilisateurs. Tous les amis contribuent en envoyant leurs publications au demandeur. Cela améliore la couverture en donnant la possibilité à des publications d’utilisateurs moins populaires. Par rapport à l’implémentation centralisée où le filtrage est effectué sur une matrice globale, les publications et amis peu populaires peuvent être marginalisés par rapport aux publications populaires. Certains utilisateurs peuvent avoir des intérêts spécifiques pour ces publications moins populaires (*publications de vrais amis ou cousins*). Dans ce cas, le FC décentralisé peut mieux accéder à ces publications en parcourant tous les amis.

Malheureusement, la configuration précédente est très coûteuse en termes de surcharge réseau. Notre algorithme propose que le demandeur sélectionne un sous-ensemble de ses amis pour leur demander des recommandations. Dans la deuxième expérience, nous avons joué sur le pourcentage d’amis à sélectionner en observant le rappel et la couverture. En analysant la figure 4.8 (*rappel*) et la figure 4.9 (*couverture*), nous notons qu’au moins 50% des amis devraient être sélectionnés pour obtenir des résultats

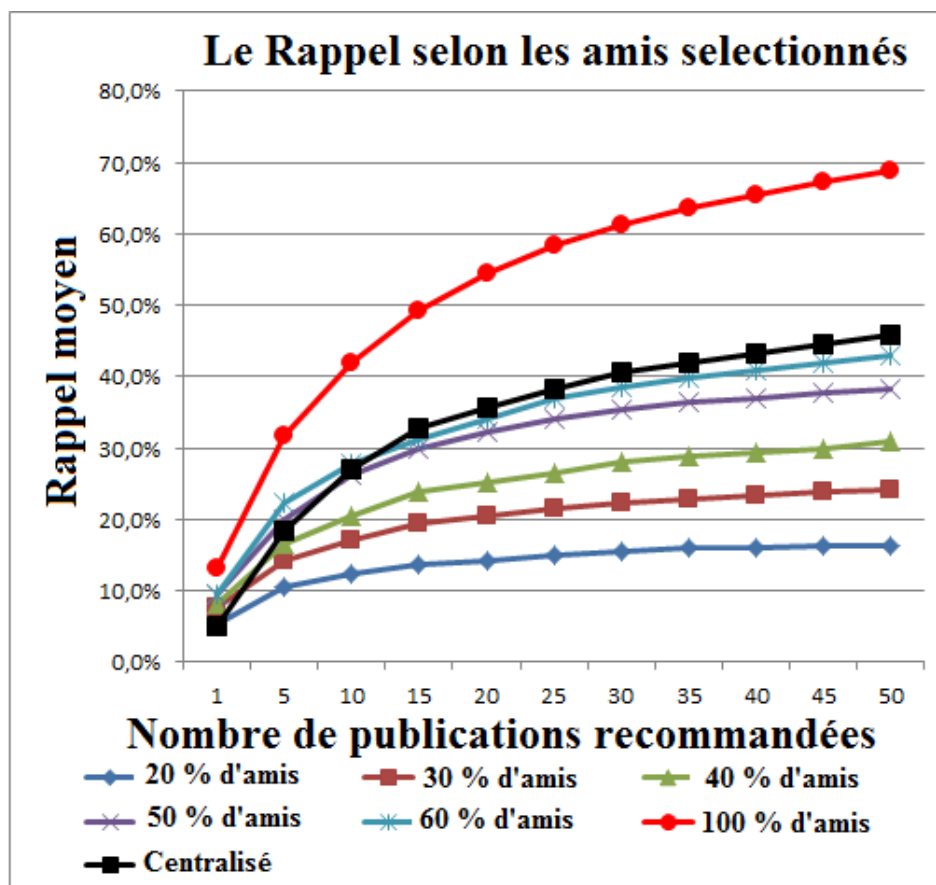


FIGURE 4.8 – Rappel moyen en fonction du pourcentage d'amis sélectionnés

similaires à ceux obtenus par le système centralisé de base.

La dernière expérience concerne le nombre de recommandations renvoyées par des amis. Nous avons constaté qu'il suffit de renvoyer 40% des recommandations de chaque ami pour obtenir un rappel comparable à celui du système centralisé (*Figure 4.10*). En termes de couverture (*Figure 4.11*), au moins 90% des publications devraient être recommandées pour couvrir le même pourcentage de publications que l'algorithme centralisé.

4.6 Conclusion

LES systèmes de recommandations représentent l'une des fonctionnalités les plus intéressantes des réseaux sociaux, car ils servent à attirer l'utilisateur et à le garder le plus longtemps possible dans le système. Ces systèmes fournissent un contenu personnalisé pour chaque utilisateur en fonction de son domaine d'intérêt. Pour y parvenir, une grande quantité d'informations doit être collectée sur l'utilisateur et ses relations et interactions antérieures avec le contenu. L'analyse de ces données améliore l'expérience de l'utilisateur, mais elle soulève des questions sur le degré d'observation auquel il est soumis lors de l'utilisation du système.

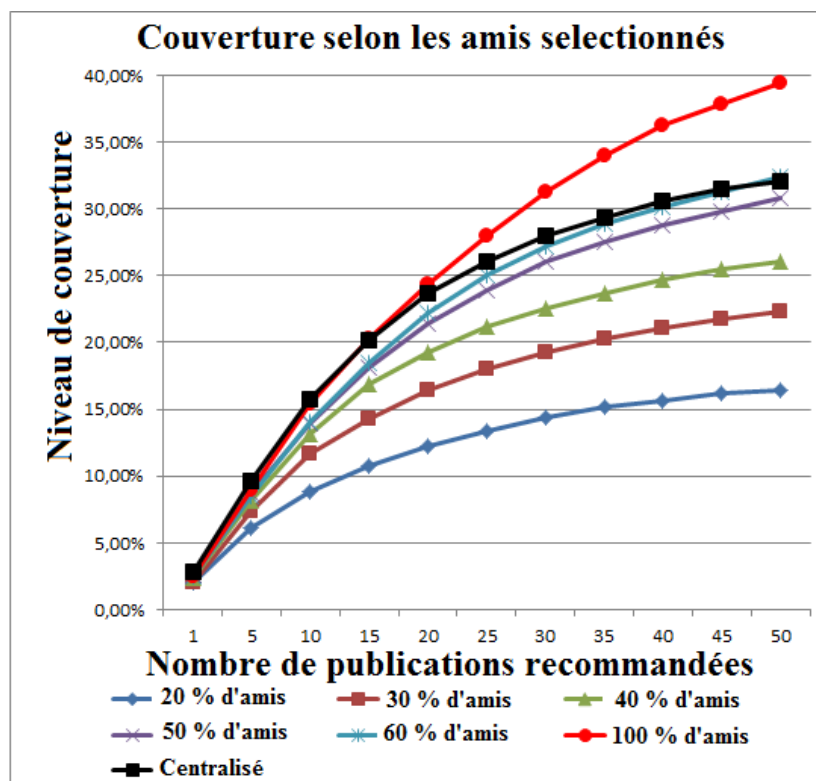


FIGURE 4.9 – Niveau de couverture en fonction du pourcentage d'amis sélectionnés

Les réseaux sociaux pair à pair en sont venus pour limiter la collecte centralisée des données des utilisateurs pour protéger leur vie privée. Cette approche a rendu difficile la construction d'un système de recommandation sans vision globale du réseau.

Dans ce chapitre, nous avons présenté P2PCF - un système de recommandation basé sur un filtrage collaboratif pour les réseaux sociaux P2P - qui se concentre principalement sur la protection de la confidentialité des données. Pour respecter les restrictions de confidentialité des réseaux sociaux P2P, nous supposons que la matrice de notation (*interactions*) est distribuée au sein des pairs, aucun pair n'ayant accès à la matrice globale.

Autrement dit, chaque utilisateur gère localement les réactions survenues sur son mur sans échanger ces informations avec les autres utilisateurs. Lorsqu'un utilisateur demande des recommandations, une demande est soumise à ses amis qui génèrent des listes d'éléments pertinents localement. Ces listes sont agrégées à la réception et hiérarchisées en fonction de la force des liens entre le demandeur et ses amis.

Nous avons évalué l'efficacité de notre système et l'avons comparé au système de recommandation centralisé, en adaptant les données réelles de Twitter. Notre évaluation a révélé une amélioration significative en termes de rappel et de couverture lorsque tous les amis sont impliqués dans le processus de recommandation. Nous avons également constaté que demander des recommandations à seulement 50% de ses amis suffit à obtenir des scores similaires à ceux rapportés par le système centralisé.

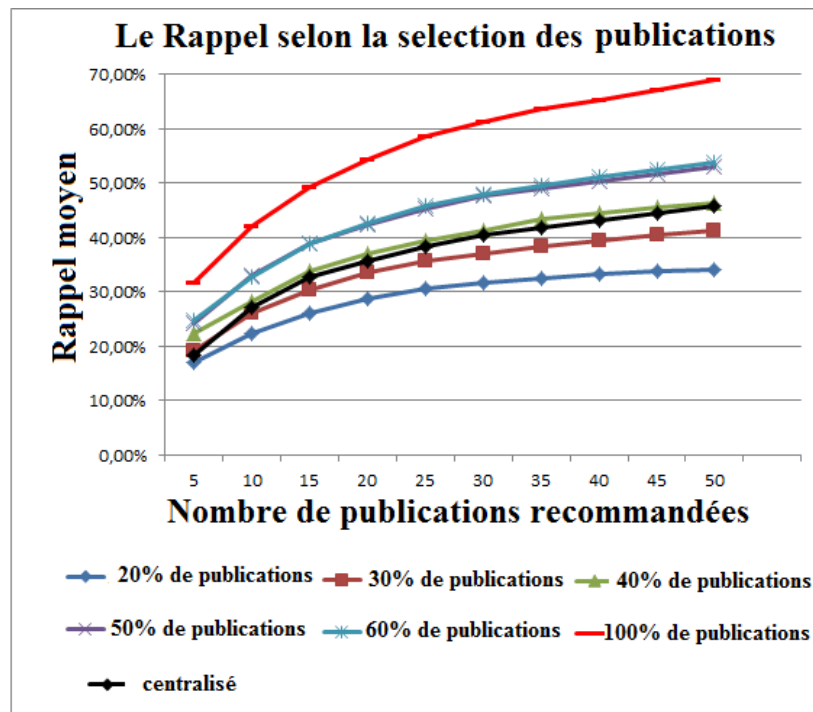


FIGURE 4.10 – Le rappel moyen en fonction du pourcentage de publications sélectionnées par chaque ami

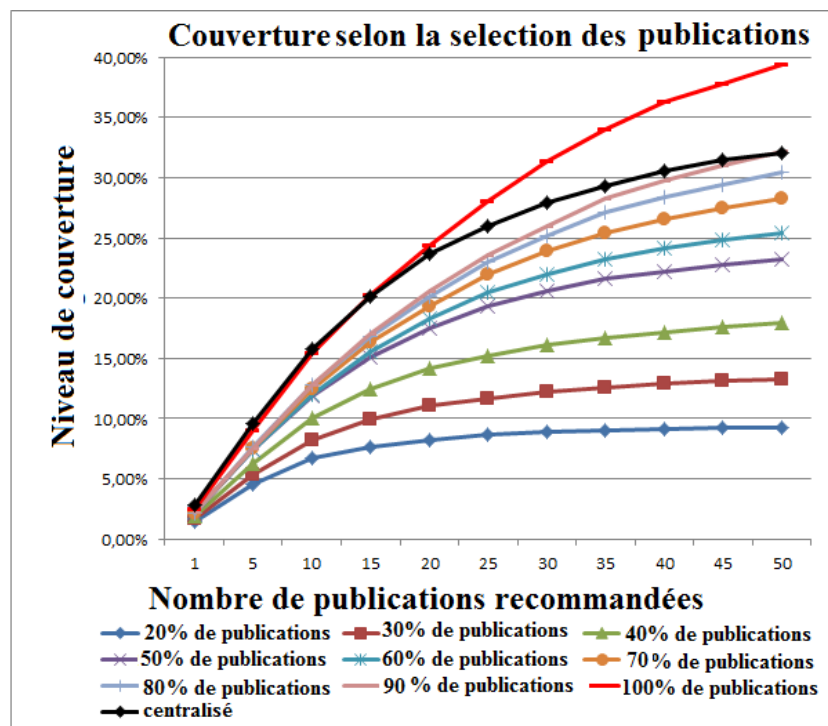


FIGURE 4.11 – Niveau de couverture en fonction du pourcentage de publications sélectionnées par chaque ami

Conclusion générale et perspectives

L'ÉMERGENCE de sites de réseautage social a attiré l'intérêt des utilisateurs, car ils leur ont permis de communiquer efficacement et facilement avec des amis de la vie réelle ou des personnages célèbres qui étaient difficiles à atteindre. En outre, ces plateformes ont permis une liberté d'expression sans précédent, permettant aux utilisateurs d'interagir avec des événements via une interface utilisateur simple et attrayante. Au cours des années, ces sites sont en concurrence pour attirer plus d'utilisateurs en développant leurs fonctionnalités et s'impliquant dans tous les domaines de la vie.

L'évolution du nombre d'utilisateurs et la diversité des informations qu'ils partagent ont contribué à la constitution d'un trésor de données stockées par les fournisseurs de ce type de système. Ces derniers n'ont pas manqué l'occasion d'exploiter ce trésor pour atteindre des gains financiers. Cet investissement des informations des utilisateurs a conduit à plusieurs incidents dans lesquels la vie privée des utilisateurs est menacée : des informations sensibles sont partagées avec des parties indésirables telles que les entreprises de publicité et les gouvernements. Pour combler ce problème d'exploitation des données, plusieurs travaux de recherche ont été lancés pour concevoir des réseaux sociaux décentralisés offrant les mêmes services sans se baser sur une entité centrale de stockage et de contrôle.

Les architectures pair à pair sont la solution triviale pour cette vision. Elles ont atteint un succès acceptable dans le partage de fichiers, la diffusion multimédia, les applications de messagerie et les logiciels de télétravail. Plusieurs idées ont été proposées pour implémenter les fonctionnalités des réseaux sociaux sur une architecture pair à pair.

Notre thèse est inscrite dans ce cadre. Nous avons étudié les travaux connexes et nous avons conclu que ces propositions ont prouvé la possibilité de décentraliser les réseaux sociaux en fournissant des fonctions de base telles que la publication, l'interaction et la messagerie. Mais les systèmes proposés sont restés principalement dans le milieu académique et n'ont pas réussi à attirer les utilisateurs et les industriels.

On a constaté les limites des performances des machines des utilisateurs qui sont proposées comme alternative aux serveurs. Les systèmes proposés n'arrivent pas à assurer un niveau de disponibilité similaire aux systèmes centralisés.

Dans ce contexte nous avons proposé notre première contribution. Dans un schéma de stockage sur les machines des utilisateurs, nous avons adopté un schéma de réplication symétrique dont les machines des utilisateurs forment des groupes de réplication

pour stocker la même liste de profils. Pour assurer la cohérence des profils stockés, nous avons proposé un nouvel algorithme de mise à jour asynchrone basé sur le journal des évènements. Ce journal est le moyen de communication avec lequel les machines d'un même groupe de réplication échangent les publications et les réactions des utilisateurs. Ce mécanisme permet de réduire le coût des opérations de mise à jour (*temps et surcharge*). Pour l'équilibrage de charge entre les machines, nous avons proposé un mécanisme de recherche et de routage basé sur une priorité spécifique à chaque profil stocké.

Nous avons évalué notre solution à travers la simulation. Les résultats ont démontré sa capacité d'équilibrer la charge. En plus, on a observé que la disponibilité est assurée si les machines du groupe de réplication couvrent plus de temps de connectivité. La taille du groupe est peu importante. Ce constat démontre l'importance de la stratégie de création des groupes de réplication.

La stratégie selon laquelle les machines sont sélectionnées pour créer un groupe de réplication n'a pas été abordée dans cette thèse. Elle peut faire l'objet d'un futur travail. Nous proposons que plusieurs critères doivent être considérés : temps de connexion, puissance, bande passante et la confiance. Ces critères doivent être considérés aussi dans la stratégie d'équilibrage de charge.

Le deuxième reproche aux réseaux sociaux P2P est le manque de fonctionnalités. La majorité des architectures proposées se sont limitées à l'implémentation des fonctions de base telles que la publication d'un contenu et les réactions avec le contenu des amis. Nous avons proposé notre deuxième contribution (*P2PCF*) pour enrichir les fonctionnalités supportées par le modèle décentralisé.

Nous utilisons une approche de filtrage collaboratif pour recommander du contenu dans les réseaux sociaux P2P. P2PCF permet de préserver la confidentialité par l'interdiction d'une sauvegarde centralisée des interactions des utilisateurs au sein du réseau social. Notre approche suppose que la matrice de notation est distribuée et gérée par les pairs, de telle sorte que chaque pair ne voit que les interactions faites par ses amis sur son mur. Les recommandations sont ensuite calculées localement au sein de chaque pair avant d'être renvoyées au demandeur. Un résumé de nos principales contributions est présenté ci-dessous :

- Nous introduisons un nouveau paradigme de préservation de la vie privée pour la recommandation de filtrage collaboratif dans les réseaux sociaux P2P.
- Nous montrons que des recommandations pertinentes peuvent être efficacement compilées à partir de pairs, lorsque les bonnes métriques de similarité sont utilisées.
- Nous proposons une solution pour faire face aux problèmes inhérents au démarrage à froid tant pour les nouveaux utilisateurs que pour les publications.

Notre évaluation est menée sur un ensemble de données réelles provenant de Twitter. Les expériences ont montré l'efficacité de notre proposition en termes de rappel et de couverture. En effet, nos résultats montrent une augmentation de 50% dans le rappel et 23% dans le niveau de couverture de P2PCF par rapport au système centralisé de base, lorsque tous les amis contribuent au processus de recommandation. Nous avons

également étudié l'effet du nombre d'amis interférant et du nombre de publications retournées par utilisateur sur la qualité des recommandations. Nous avons trouvé que l'implication de 50% des amis seulement permet à P2PCF de dépasser le système de recommandation centralisé.

EN guise de perspectives, nous souhaitons nous concentrer sur l'effet de l'assouplissement de certaines contraintes sur la qualité de la recommandation. Cela inclut, par exemple, l'échange de parties des notes (*interactions*) locales avec des amis de confiance, ce qui pourrait conduire à des recommandations plus précises et personnalisées.

En plus des améliorations des architectures P2P proposées dans la littérature, il existe d'autres pistes à explorer. La première piste peut être la décentralisation de certaines fonctionnalités dans le cadre d'un réseau social centralisé. On peut citer : la messagerie, la diffusion multimédia, les jeux en ligne. Une deuxième perspective est la concentration des réseaux sociaux P2P dans des domaines spécifiques tels que les réseaux sociaux locaux, la collaboration scientifique ou la gestion des catastrophes. Une troisième possibilité émergente est l'exploitation du potentiel de la technologie Block-Chain qui a prouvé sa capacité pour remplacer une entité centrale.

Bibliographie

- [1] <https://misbar.com/>.
- [2] <https://factcheck.afp.com/>.
- [3] <https://fatabyyano.net/>.
- [4] <https://about.facebook.com/company-info/>.
- [5] <https://about.twitter.com/fr/company>.
- [6] *The blockstack project*, <https://www.stacks.co/>.
- [7] *Diaspora*, <https://joindiaspora.com/>.
- [8] *The federation*, <https://the-federation.info/>.
- [9] *Ask.fm owners considered shutting down social network*, <https://www.bbc.co.uk/news/newsbeat-31249209> (2015).
- [10] <http://www.alexa.com/topsites> (2016).
- [11] Sofiane Abbar, Yelena Mejova, and Ingmar Weber, *You tweet what you eat : Studying food consumption through twitter*, Proceedings of the 33rd Annual ACM Conference on Human Factors in Computing Systems, 2015, pp. 3197–3206.
- [12] Sofiane Abbar, Rade Stanojevic, and Mohamed Mokbel, *Stad : Spatio-temporal adjustment of traffic-oblivious travel-time estimation*, 2020 21st IEEE International Conference on Mobile Data Management (MDM), IEEE, 2020, pp. 79–88.
- [13] Mumtaz Ahmad and Abdessamad Imine, *Decentralized collaborative editing platform*, 2015 16th IEEE International Conference on Mobile Data Management, vol. 1, IEEE, 2015, pp. 323–326.
- [14] Luca Maria Aiello and Giancarlo Ruffo, *Lotusnet : Tunable privacy for distributed online social network services*, Computer Communications **35** (2012), no. 1, 75–88.
- [15] Mourad Amad, Abdelmalek Boudries, and Lyes Badis, *Alm based services on hpm p2p architecture*, The Second International Conference on Computational Science and Engineering, KualaLumpur,Malaisie, 2017.
- [16] ———, *Application layer multicast based services on hierarchical peer to peer architecture*, Applied Mechanics and Materials, vol. 892, Trans Tech Publ, 2019, pp. 64–71.

- [17] Mourad Amad, Ahmed Meddahi, Djamil Aissani, and Zonghua Zhang, *Hpm : A novel hierarchical peer-to-peer model for lookup acceleration with provision of physical proximity*, Journal of Network and Computer Applications **35** (2012), no. 6, 1818–1830.
- [18] Jonathan Anderson, Claudia Diaz, Joseph Bonneau, and Frank Stajano, *Privacy-enabling social networking over untrusted networks*, Proceedings of the 2nd ACM workshop on Online social networks, 2009, pp. 1–6.
- [19] Gil Appel, Lauren Grewal, Rhonda Hadi, and Andrew T Stephen, *The future of social media in marketing*, Journal of the Academy of Marketing Science **48** (2020), no. 1, 79–95.
- [20] Randy Baden, Adam Bender, Neil Spring, Bobby Bhattacharjee, and Daniel Starin, *Persona : an online social network with user-defined privacy*, Proceedings of the ACM SIGCOMM 2009 conference on Data communication, 2009, pp. 135–146.
- [21] Lyes Badis, Mourad Amad, and Djamil Aïssani, *Asynchronous model for updating replicated profiles in decentralized social network*, 2017 International Conference on Mathematics and Information Technology (ICMIT), IEEE, 2017, pp. 227–233.
- [22] ———, *Asocial recommender system with privacy preserving*, Doctoriales Recherche Opérationnelle, Bejaia University, 2018.
- [23] ———, *A log based update of replicated profiles in decentralized social networks*, Journal of Digital Information Management **16** (2018), no. 5, 230–245.
- [24] Lyes Badis, Mourad Amad, Djamil Aïssani, and Sofiane Abbar, *P2pcf : A collaborative filtering based recommender system for peer to peer social networks*, Journal of High Speed Networks **27** (2021), no. 1, 13–31.
- [25] Lyes Badis, Mourad Amad, Djamil Aïssani, Kahina Bedjguelal, and Aldja Benkerrou, *Routil : P2p routing protocol based on interest links*, 2016 International Conference on Advanced Aspects of Software Engineering (ICAASE), IEEE, 2016, pp. 1–5.
- [26] Jason Baker, Chris Bond, James C Corbett, JJ Furman, Andrey Khorlin, James Larson, Jean-Michel Leon, Yawei Li, Alexander Lloyd, and Vadim Yushprakh, *Megastore : Providing scalable, highly available storage for interactive services*, (2011).
- [27] John Arundel Barnes, *Class and committees in a norwegian island parish*, Human relations **7** (1954), no. 1, 39–58.
- [28] Doug Beaver, Sanjeev Kumar, Harry C Li, Jason Sobel, Peter Vajgel, et al., *Finding a needle in haystack : Facebook’s photo storage.*, OSDI, vol. 10, 2010, pp. 1–8.
- [29] Jose Benitez, Laura Ruiz, Ana Castillo, and Javier Llorens, *How corporate social responsibility activities influence employer reputation : The role of social media capability*, Decision Support Systems **129** (2020), 113223.
- [30] Sebastian Biedermann, Nikolaos P Karvelas, Stefan Katzenbeisser, Thorsten Strufe, and Andreas Peter, *Proofbook : An online social network based on proof-of-work and friend-propagation*, International Conference on Current Trends in Theory and Practice of Informatics, Springer, 2014, pp. 114–125.

- [31] Mohamed Reda Bouadjenek, Hakim Hacid, and Mokrane Bouzeghoub, *Social networks and information retrieval, how are they converging? a survey, a taxonomy and an analysis of social information retrieval approaches and platforms*, Information Systems **56** (2016), 1–18.
- [32] Danah M Boyd and Nicole B Ellison, *Social network sites : Definition, history, and scholarship*, Journal of computer-mediated Communication **13** (2007), no. 1, 210–230.
- [33] Sonja Buchegger, Doris Schiöberg, Le-Hung Vu, and Anwitaman Datta, *Peerson : P2p social networking : early experiences and insights*, Proceedings of the Second ACM EuroSys Workshop on Social Network Systems, 2009, pp. 46–52.
- [34] Ronald S Burt, *Structural holes*, Harvard university press, 1992.
- [35] David Camacho, Ángel Panizo-LLedot, Gema Bello-Orgaz, Antonio Gonzalez-Pardo, and Erik Cambria, *The four dimensions of social network analysis : An overview of research methods, applications, and software tools*, Information Fusion **63** (2020), 88–120.
- [36] Dominique Cardon, *Pourquoi l'internet n'a-t-il pas changé la politique?*, <https://www.lemonde.fr/blog/internetactu/2011/08/19/dominique-cardon-pourquoi-linternet-na-t-il-pas-change-la-politique/> (2011).
- [37] CareerBuilder, *Number of employers using social media to screen candidates at all-time high, finds latest careerbuilder study*, <http://press.careerbuilder.com/2017-06-15-Number-of-Employers-Using-Social-Media-to-Screen-Candidates-at-All-Time-High-Finds-Latest-CareerBuilder-Study> (2017).
- [38] AEO Carosia, Guilherme Palermo Coelho, and AEA Silva, *Analyzing the brazilian financial market through portuguese sentiment analysis in social media*, Applied Artificial Intelligence **34** (2020), no. 1, 1–19.
- [39] Fred Cavazzan, *Panorama des médias sociaux 2012*, <https://fredcavazza.net/2012/02/20/panorama-des-medias-sociaux-2012/> (2012).
- [40] Koyel Chakraborty, Siddhartha Bhattacharyya, and Rajib Bag, *A survey of sentiment analysis from social media data*, IEEE Transactions on Computational Social Systems **7** (2020), no. 2, 450–464.
- [41] Fay Chang, Jeffrey Dean, Sanjay Ghemawat, Wilson C Hsieh, Deborah A Wallach, Mike Burrows, Tushar Chandra, Andrew Fikes, and Robert E Gruber, *Bigtable : A distributed storage system for structured data*, ACM Transactions on Computer Systems (TOCS) **26** (2008), no. 2, 1–26.
- [42] Chaochao Chen, Ziqi Liu, Peilin Zhao, Jun Zhou, and Xiaolong Li, *Privacy preserving point-of-interest recommendation using decentralized matrix factorization*, Proceedings of the AAAI Conference on Artificial Intelligence, vol. 32, 2018.
- [43] Gobinda G Chowdhury, *Introduction to modern information retrieval*, Facet publishing, 2010.
- [44] Shihabur Rahman Chowdhury, Arup Raton Roy, Maheen Shaikh, and Khuzaima Daudjee, *A taxonomy of decentralized online social networks*, Peer-to-Peer Networking and Applications **8** (2015), no. 3, 367–383.

- [45] Kate Conger, *F.t.c. investigating twitter for potential privacy violations*, <https://www.nytimes.com/2020/08/03/technology/ftc-twitter-privacy-violations.html> (2020).
- [46] Marco Conti, Andrea De Salve, Barbara Guidi, Francesco Pitto, and Laura Ricci, *Trusted dynamic storage for dunbar-based p2p online social networks*, OTM Confederated International Conferences On the Move to Meaningful Internet Systems, Springer, 2014, pp. 400–417.
- [47] Marco Conti, Andrea De Salve, Barbara Guidi, and Laura Ricci, *Epidemic diffusion of social updates in dunbar-based dosn*, European Conference on Parallel Processing, Springer, 2014, pp. 311–322.
- [48] SYLVIE CORLAY-KNAFF, *Les 5 tendances du social média en 2016*, <https://www.webmarketing-com.com/2016/04/08/46793-5-tendances-social-media-2016> (2016).
- [49] Leucio Antonio Cutillo, Refik Molva, and Thorsten Strufe, *Safebook : A privacy-preserving online social network leveraging on real-life trust*, IEEE Communications Magazine **47** (2009), no. 12, 94–101.
- [50] Gianni D’Angelo, Francesco Palmieri, and Salvatore Rampone, *Detecting unfair recommendations in trust-based pervasive environments*, Information Sciences **486** (2019), 31–51.
- [51] Gianni D’Angelo, Raffaele Pilla, Jay B Dean, and Salvatore Rampone, *Toward a soft computing-based correlation between oxygen toxicity seizures and hyperoxic hyperventilation*, Soft Computing **22** (2018), no. 7, 2421–2427.
- [52] Gianni D’Angelo, Raffaele Pilla, Carlo Tascini, and Salvatore Rampone, *A proposal for distinguishing between bacterial and viral meningitis using genetic programming and decision trees*, Soft Computing **23** (2019), no. 22, 11775–11791.
- [53] Anwitaman Datta, Sonja Buchegger, Le-Hung Vu, Thorsten Strufe, and Krzysztof Rzadca, *Decentralized online social networks*, Handbook of social network technologies and applications, Springer, 2010, pp. 349–378.
- [54] Anwitaman Datta and Rajesh Sharma, *Godisco : selective gossip based dissemination of information in social community based overlays*, International Conference on Distributed Computing and Networking, Springer, 2011, pp. 227–238.
- [55] Andrea De Salve, Barbara Guidi, and Laura Ricci, *Evaluation of structural and temporal properties of ego networks for data availability in dosns*, Mobile Networks and Applications **23** (2018), no. 1, 155–166.
- [56] Andrea De Salve, Paolo Mori, and Laura Ricci, *A survey on privacy in decentralized online social networks*, Computer Science Review **27** (2018), 154–176.
- [57] Alain Degenne and Michel Forsé, *Les réseaux sociaux*, vol. 2, Armand Colin Paris, 2004.
- [58] María del Carmen Rodríguez-Hernández, Sergio Ilarri, Raquel Trillo, and Ramón Hermoso, *Context-aware recommendations using mobile p2p*, Proceedings of the 15th International Conference on Advances in Mobile Computing & Multimedia, 2017, pp. 82–91.
- [59] Eric Delcroix, *Qu’est ce que les réseaux sociaux*, <http://les-zed.com/qu-est-ce-que-les-reseaux-sociaux/> (2007).

- [60] Alan Demers, Dan Greene, Carl Hauser, Wes Irish, John Larson, Scott Shenker, Howard Sturgis, Dan Swinehart, and Doug Terry, *Epidemic algorithms for replicated database maintenance*, Proceedings of the sixth annual ACM Symposium on Principles of distributed computing, 1987, pp. 1–12.
- [61] Rachid Djerbi, Mourad Amad, and Rabah Imache, *A new model for communities' detection in dynamic social networks inspired from human families*, International Journal of Internet Technology and Secured Transactions **10** (2020), no. 1-2, 24–60.
- [62] Peter Sheridan Dodds, Roby Muhamad, and Duncan J Watts, *An experimental study of search in global social networks*, science **301** (2003), no. 5634, 827–829.
- [63] Yingtong Dou, Hao Yang, and Xiaolong Deng, *A survey of collaborative filtering algorithms for social recommender systems*, 2016 12th International conference on semantics, knowledge and grids (SKG), IEEE, 2016, pp. 40–46.
- [64] Fady Draidi, Esther Pacitti, Michelle Cart, and Hinde Lilia Bouziane, *Leveraging social and content-based recommendation in p2p systems*, AP2PS11 : 3rd International Conference on Advances in P2P Systems, 2011, pp. 13–18.
- [65] Fady Draidi, Esther Pacitti, and Bettina Kemme, *P2prec : a p2p recommendation system for large-scale data sharing*, Transactions on large-scale data-and knowledge-centered systems III, Springer, 2011, pp. 87–116.
- [66] Antoine Dupin, *Communiquer sur les réseaux sociaux : les méthodes et les outils indispensables pour vos stratégies de communication sur les médias sociaux*, Fyp éditions, 2010.
- [67] Erika Duriakova, Elias Z Tragos, Barry Smyth, Neil Hurley, Francisco J Peña, Panagiotis Symeonidis, James Geraci, and Aonghus Lawlor, *Pdmfrec : a decentralised matrix factorisation with tunable user-centric privacy*, Proceedings of the 13th ACM Conference on Recommender Systems, 2019, pp. 457–461.
- [68] Michael Dürr, Marco Maier, and Florian Dorfmeister, *Vegas—a secure and privacy-preserving peer-to-peer online social network*, 2012 International Conference on Privacy, Security, Risk and Trust and 2012 International Confernece on Social Computing, IEEE, 2012, pp. 868–874.
- [69] emarketer, *Average time spent per day with mobile messaging apps by adults in the united states 2015 to 2019*, <https://www.emarketer.com/chart/211665/average-time-spent-per-day-with-mobile-messaging-apps-by-us-adults-2015-2019> (2017).
- [70] Esra Erdin, Eric Klukovich, Gurhan Gunduz, and Mehmet Hadi Gunes, *Posn : A personal online social network*, IFIP International Information Security and Privacy Conference, Springer, 2015, pp. 51–66.
- [71] Wenqi Fan, Yao Ma, Qing Li, Yuan He, Eric Zhao, Jiliang Tang, and Dawei Yin, *Graph neural networks for social recommendation*, The World Wide Web Conference, 2019, pp. 417–426.
- [72] Komal Florio, Valerio Basile, Marco Polignano, Pierpaolo Basile, and Viviana Patti, *Time of your hate : The challenge of time in hate speech detection on social media*, Applied Sciences **10** (2020), no. 12, 4180.
- [73] Facebook for media, *Working to stop misinformation and false news*, <https://www.facebook.com/formedia/blog/working-to-stop-misinformation-and-false-news> (2017).

- [74] Friendica, <http://friendi.ca/>.
- [75] Laura Garton, Caroline Haythornthwaite, and Barry Wellman, *Studying online social networks*, Journal of computer-mediated communication **3** (1997), no. 1, JCMC313.
- [76] Jyotirmoy Gope and Sanjay Kumar Jain, *A survey on solving cold start problem in recommender systems*, 2017 International Conference on Computing, Communication and Automation (ICCCA), IEEE, 2017, pp. 133–138.
- [77] Vipul Goyal, Omkant Pandey, Amit Sahai, and Brent Waters, *Attribute-based encryption for fine-grained access control of encrypted data*, Proceedings of the 13th ACM conference on Computer and communications security, 2006, pp. 89–98.
- [78] Kalman Graffi, Christian Gross, Dominik Stingl, Daniel Hartung, Aleksandra Kovacevic, and Ralf Steinmetz, *Lifesocial. kom : A secure and p2p-based solution for online social networks*, 2011 IEEE Consumer Communications and Networking Conference (CCNC), IEEE, 2011, pp. 554–558.
- [79] Benjamin Greschbach, Gunnar Kreitz, and Sonja Buchegger, *The devil is in the metadata-new privacy challenges in decentralised online social networks*, 2012 IEEE international conference on pervasive computing and communications workshops, IEEE, 2012, pp. 333–339.
- [80] Barbara Guidi, *When blockchain meets online social networks*, Pervasive and Mobile Computing **62** (2020), 101131.
- [81] Barbara Guidi, Tobias Amft, Andrea De Salve, Kalman Graffi, and Laura Ricci, *Didusonet : A p2p architecture for distributed dunbar-based social networks*, Peer-to-Peer Networking and Applications **9** (2016), no. 6, 1177–1194.
- [82] Barbara Guidi, Marco Conti, Andrea Passarella, and Laura Ricci, *Managing social contents in decentralized online social networks : a survey*, Online Social Networks and Media **7** (2018), 12–29.
- [83] Barbara Guidi, Andrea Michienzi, and Giulio Rossetti, *Dynamic community analysis in decentralized online social networks*, European Conference on Parallel Processing, Springer, 2017, pp. 517–528.
- [84] Ido Guy, *Social recommender systems*, Recommender systems handbook, Springer, 2015, pp. 511–543.
- [85] Abdelhalim Hacini, Mourad Amad, and Semchedine Fouzi, *A scalable and hierarchical p2p architecture based on pancake graph for group communication*, Journal of High Speed Networks **23** (2017), no. 4, 287–309.
- [86] Peng Han, Bo Xie, Fan Yang, and Ruimin Shen, *A scalable p2p recommender system based on distributed collaborative filtering*, Expert systems with applications **27** (2004), no. 2, 203–210.
- [87] Waleed Hariri, Khairil Imran Ghauth, and C Eswaran, *A multimedia content recommender system using table of contents and content-based filtering*, Advanced Science Letters **24** (2018), no. 2, 1119–1123.
- [88] Md Sajjad Hosain and Ping Liu, *The role of social media on talent search and acquisition : Evidence from contemporary literature*, Journal of Intercultural Management **12** (2020), no. 1, 92–137.

- [89] Li Huang, Wenan Tan, and Yong Sun, *Collaborative recommendation algorithm based on probabilistic matrix factorization in probabilistic latent semantic analysis*, *Multimedia Tools and Applications* **78** (2019), no. 7, 8711–8722.
- [90] Daniel Lazaro Iglesias, Joan-Manuel Marques, Guillem Cabrera, Helena Rifa-Pous, and Albert Montane, *Hornet : Microblogging for a contributory social network*, *IEEE internet computing* **16** (2012), no. 3, 37–45.
- [91] Muhammad Imran, Carlos Castillo, Ji Lucas, Patrick Meier, and Sarah Vieweg, *Aidr : Artificial intelligence for disaster response*, *Proceedings of the 23rd international conference on world wide web*, 2014, pp. 159–162.
- [92] Sonia Jahid, Shirin Nilizadeh, Prateek Mittal, Nikita Borisov, and Apu Kapadia, *Decent : A decentralized architecture for enforcing privacy in online social networks*, 2012 IEEE International Conference on Pervasive Computing and Communications Workshops, IEEE, 2012, pp. 326–332.
- [93] Pijitra Jomsri, *A combination indexing for image social bookmarking system to improve search results*, *Journal of Digital Information Management* **14** (2016), no. 6.
- [94] Prateek Joshi and C-C Jay Kuo, *Security and privacy in online social networks : A survey*, 2011 IEEE international conference on multimedia and Expo, IEEE, 2011, pp. 1–6.
- [95] journaldunet, *Twitter devrait passer la barre des 500 millions d'utilisateurs fin février*, <https://www.journaldunet.com/ebusiness/le-net/1096954-twitter-devrait-passer-la-barre-des-500-millions-d-utilisateurs-fin-fevrier/> (2012).
- [96] Pierre St Juste, David Wolinsky, P Oscar Boykin, and Renato J Figueiredo, *Litter : A lightweight peer-to-peer microblogging service*, 2011 IEEE Third International Conference on Privacy, Security, Risk and Trust and 2011 IEEE Third International Conference on Social Computing, IEEE, 2011, pp. 900–903.
- [97] Pantelis Karamolegkos and Nikolaos Doulamis, *Multimedia file exchange and p2p social networking : Efficient allocation of users into communities of interest*, 2009 International Conference on Ultra Modern Telecommunications & Workshops, IEEE, 2009, pp. 1–4.
- [98] Wasiat Khan, Mustansar Ali Ghazanfar, Muhammad Awais Azam, Amin Karami, Khaled H Alyoubi, and Ahmed S Alfakeeh, *Stock market prediction using machine learning classifiers and social media, news*, *Journal of Ambient Intelligence and Humanized Computing* (2020), 1–24.
- [99] David Koll, Dieter Lechler, and Xiaoming Fu, *Socialgate : Managing large-scale social data on home gateways*, 2017 IEEE 25th International Conference on Network Protocols (ICNP), IEEE, 2017, pp. 1–6.
- [100] David Koll, Jun Li, and Xiaoming Fu, *Soup : an online social network by the people, for the people*, *Proceedings of the 15th International Middleware Conference*, 2014, pp. 193–204.
- [101] ———, *The good left undone : Advances and challenges in decentralizing online social networks*, *Computer Communications* **108** (2017), 36–51.
- [102] Nicolas Kourtellis, Joshua Finnis, Paul Anderson, Jeremy Blackburn, Cristian Borcea, and Adriana Iamnitchi, *Prometheus : User-controlled p2p social data*

- management for socially-aware applications*, ACM/IFIP/USENIX International Conference on Distributed Systems Platforms and Open Distributed Processing, Springer, 2010, pp. 212–231.
- [103] Salma Ktari, Mathieu Zoubert, Artur Hecker, and Houda Labiod, *Performance evaluation of replication strategies in dhts under churn*, Proceedings of the 6th international conference on Mobile and ubiquitous multimedia, 2007, pp. 90–97.
- [104] Avinash Lakshman and Prashant Malik, *Cassandra : a decentralized structured storage system*, ACM SIGOPS Operating Systems Review **44** (2010), no. 2, 35–40.
- [105] Emmanuel Lazega, *Réseaux sociaux et structures relationnelles*, Presses universitaires de France Paris, 1998.
- [106] Seung Yeop Lee and Sang Woo Lee, *Social media use and job performance in the workplace : The effects of facebook and kakaotalk use on job performance in south korea*, Sustainability **12** (2020), no. 10, 4052.
- [107] Alain Lefebvre, *Les réseaux sociaux : pivot de l'internet 2.0*, M21 Editions, 2005.
- [108] Vincent Lemieux, *Les réseaux d'acteurs sociaux*, Presses universitaires de France, 1999.
- [109] Amanda Lenhart, Kristen Purcell, Aaron Smith, and Kathryn Zickuhr, *Social media & mobile internet use among teens and young adults. millennials.*, Pew internet & American life project (2010).
- [110] Jeen-Su Lim, Phuoc Pham, and John H Heinrichs, *Impact of social media activity outcomes on brand equity*, Journal of Product & Brand Management **29** (2020), no. 7, 927–937.
- [111] Dongtao Liu, Amre Shakimov, Ramón Cáceres, Alexander Varshavsky, and Landon P Cox, *Confidant : Protecting osn data without locking it up*, ACM/IFIP/USENIX International Conference on Distributed Systems Platforms and Open Distributed Processing, Springer, 2011, pp. 61–80.
- [112] Alireza Mahdian, Richard Han, Qin Lv, and Shivakant Mishra, *Results from a practical deployment of the myzone decentralized p2p social network*, arXiv preprint arXiv :1305.0606 (2013).
- [113] Tahir Maqsood, Osman Khalid, Rizwana Irfan, Sajjad A Madani, and Samee U Khan, *Scalability issues in online social networks*, ACM Computing Surveys (CSUR) **49** (2016), no. 2, 1–42.
- [114] Ziryeb Marouf, *Les réseaux sociaux numériques d'entreprises : état des lieux et raisons d'agir*, Editions L'Harmattan, 2011.
- [115] Newton Masinde and Kalman Graffi, *Peer-to-peer-based social networks : A comprehensive survey*, SN Computer Science **1** (2020), no. 5, 1–51.
- [116] Ariadna Matamoros-Fernández and Johan Farkas, *Racism, hate speech, and social media : A systematic review and critique*, Television & New Media **22** (2021), no. 2, 205–224.
- [117] Vishal Mathur and Rekha Jain, *A study on social network mining*, Journal of Advanced Database Management & Systems **6** (2019), no. 2, 36–39.

- [118] Giuliano Mega, Alberto Montresor, and Gian Pietro Picco, *Efficient dissemination in decentralized social networks*, 2011 IEEE International Conference on Peer-to-Peer Computing, IEEE, 2011, pp. 338–347.
- [119] Stanley Milgram, *The small world problem*, *Psychology today* **2** (1967), no. 1, 60–67.
- [120] Neal Mohan and Robert Kyncl, *Additional changes to the youtube partner program (ypp) to better protect creators*, <https://blog.youtube/news-and-events/additional-changes-to-youtube-partner> (2018).
- [121] Raymond J Mooney and Loriene Roy, *Content-based book recommending using learning for text categorization*, Proceedings of the fifth ACM conference on Digital libraries, 2000, pp. 195–204.
- [122] Athicha Muthitacharoen, Robert Morris, Thomer M Gil, and Benjie Chen, *Ivy : A read/write peer-to-peer file system*, ACM SIGOPS Operating Systems Review **36** (2002), no. SI, 31–44.
- [123] Rammohan Narendula, Thanasis G Papaioannou, and Karl Aberer, *A decentralized online social network with efficient user-driven replication*, 2012 International Conference on Privacy, Security, Risk and Trust and 2012 International Conference on Social Computing, IEEE, 2012, pp. 166–175.
- [124] Jack Nicas Natasha Singer and Kate Conger, *Youtube said to be fined up to \$200 million for children's privacy violations*, <https://www.nytimes.com/2019/08/30/technology/youtube-childrens-privacy-fine.html> (2019).
- [125] Tien T Nguyen, Pik-Mai Hui, F Maxwell Harper, Loren Terveen, and Joseph A Konstan, *Exploring the filter bubble : the effect of using recommender systems on content diversity*, Proceedings of the 23rd international conference on World wide web, 2014, pp. 677–686.
- [126] Shirin Nilizadeh, Sonia Jahid, Prateek Mittal, Nikita Borisov, and Apu Kapadia, *Cachet : a decentralized architecture for privacy preserving social networking with caching*, Proceedings of the 8th international conference on Emerging networking experiments and technologies, 2012, pp. 337–348.
- [127] Official Journal of the European Union, *General data protection regulation*, <https://eur-lex.europa.eu/eli/reg/2016/679/oj> (2016).
- [128] Tim O'reilly, *What is web 2.0 : Design patterns and business models for the next generation of software*, Communications & strategies (2007), no. 1, 17.
- [129] Thomas Paul, Niklas Lochschmidt, Hani Salah, Anwitaman Datta, and Thorsten Strufe, *Lilliput : A storage service for lightweight peer-to-peer online social networks*, 2017 26th International Conference on Computer Communication and Networks (ICCCN), IEEE, 2017, pp. 1–10.
- [130] Famulari Antonino Paul Thomas and Strufe Thorsten, *A survey on decentralized online social networks*, Computer Networks **75** (2014), 437–452.
- [131] Michael J Pazzani, *A framework for collaborative, content-based and demographic filtering*, Artificial intelligence review **13** (1999), no. 5, 393–408.

- [132] Timothy Perfit and Burkhard Englert, *Megaphone : Fault tolerant, scalable, and trustworthy p2p microblogging*, 2010 Fifth international conference on internet and web applications and services, IEEE, 2010, pp. 469–477.
- [133] Chetna Priyadarshini, Ritesh Kumar Dubey, YLN Kumar, and Rajneesh Ranjan Jha, *Impact of a social media addiction on employees' wellbeing and work productivity*, The Qualitative Report **25** (2020), no. 1, 181–196.
- [134] Adithya Rao, Nemanja Spasojevic, Zhisheng Li, and Trevor DSouza, *Klout score : Measuring influence across multiple social networks*, 2015 IEEE International Conference on Big Data (Big Data), IEEE, 2015, pp. 2282–2289.
- [135] Renata Lopes Rosa, Eduardo Lucio Lasmar Junior, and Demóstenes Zegarra Rodríguez, *A recommendation system for shared-use mobility service through data extracted from online social networks*, Journal of Communications Software and Systems **14** (2018), no. 4, 359–366.
- [136] Antony Rowstron and Peter Druschel, *Pastry : Scalable, decentralized object location, and routing for large-scale peer-to-peer systems*, IFIP/ACM International Conference on Distributed Systems Platforms and Open Distributed Processing, Springer, 2001, pp. 329–350.
- [137] Arunima Roy, Katerina Nikolitch, Rachel McGinn, Safiya Jinah, William Klement, and Zachary A Kaminsky, *A machine learning approach predicts future risk to suicidal ideation from social media data*, NPJ digital medicine **3** (2020), no. 1, 1–12.
- [138] Daniel Sandler and Dan S Wallach, *Birds of a fethr : open, decentralized micro-publishing.*, IPTPS, 2009, p. 1.
- [139] Christophe Olivier Schneble, Bernice Simone Elger, and David Shaw, *The cambridge analytica affair and internet-mediated research*, EMBO reports **19** (2018), no. 8, e46579.
- [140] Erick Schonfeld, *Yang decides to shut down yahoo 360-nobody notices*, <https://techcrunch.com/2007/10/23/yang-decides-to-shut-down-yahoo-360%E2%80%94nobody-notices/?guccounter=1> (2007).
- [141] David A Schweidel and Wendy W Moe, *Listening in on social media : A joint model of sentiment and venue format choice*, Journal of marketing research **51** (2014), no. 4, 387–402.
- [142] Seok-Won Seong, Jiwon Seo, Matthew Nasielski, Debangsu Sengupta, Sudheendra Hangal, Seng Keat Teh, Ruven Chu, Ben Dodson, and Monica S Lam, *Prpl : a decentralized social networking infrastructure*, Proceedings of the 1st ACM Workshop on Mobile Cloud Computing & Services : Social Networks and Beyond, 2010, pp. 1–8.
- [143] Nashid Shahriar, Shihabur Rahman Chowdhury, Mahfuza Sharmin, Reaz Ahmed, Raouf Boutaba, and Bertrand Mathieu, *Ensuring beta-availability in p2p social networks*, 2013 IEEE 33rd International Conference on Distributed Computing Systems Workshops, IEEE, 2013, pp. 150–155.
- [144] Amre Shakimov, Harold Lim, Ramón Cáceres, Landon P Cox, Kevin Li, Dongtao Liu, and Alexander Varshavsky, *Vis-a-vis : Privacy-preserving online social networking via virtual individual servers*, 2011 Third International Conference on

- Communication Systems and Networks (COMSNETS 2011), IEEE, 2011, pp. 1–10.
- [145] Qusai Shambour and Jie Lu, *A trust-semantic fusion-based recommendation approach for e-business applications*, Decision Support Systems **54** (2012), no. 1, 768–780.
- [146] Rajesh Sharma and Anwitaman Datta, *Supernova : Super-peers based architecture for decentralized online social networks*, 2012 fourth international conference on communication systems and networks (COMSNETS 2012), IEEE, 2012, pp. 1–10.
- [147] Arunan Sivanathan, Hassan Habibi Gharakheili, and Vijay Sivaraman, *Managing iot cyber-security using programmable telemetry and machine learning*, IEEE Transactions on Network and Service Management **17** (2020), no. 1, 60–74.
- [148] statista, *Most popular social networks worldwide as of january 2021, ranked by number of active users*, <https://www.statista.com/statistics/272014/global-social-networks-ranked-by-number-of-users/> (2021).
- [149] Patrick Stuedi, Iqbal Mohamed, Mahesh Balakrishnan, Z Morley Mao, Venugopalan Ramasubramanian, Doug Terry, and Ted Wobber, *Contrail : Decentralized and privacy-preserving social networks on smartphones*, IEEE internet computing **18** (2013), no. 5, 44–51.
- [150] Udeep Tandukar and Julita Vassileva, *Selective propagation of social data in decentralized online social network*, International Conference on User Modeling, Adaptation, and Personalization, Springer, 2011, pp. 213–224.
- [151] Jiliang Tang, Huiji Gao, Xia Hu, and Huan Liu, *Exploiting homophily effect for trust prediction*, Proceedings of the sixth ACM international conference on Web search and data mining, 2013, pp. 53–62.
- [152] Jiliang Tang, Xia Hu, and Huan Liu, *Social recommendation : a review*, Social Network Analysis and Mining **3** (2013), no. 4, 1113–1133.
- [153] Ying Tang and Khe Foon Hew, *Does mobile instant messaging facilitate social presence in online communication? a two-stage study of higher education students*, International Journal of Educational Technology in Higher Education **17** (2020), 1–17.
- [154] Florian Tegeler, David Koll, and Xiaoming Fu, *Gemstone : empowering decentralized social networking with high data availability*, 2011 IEEE Global Telecommunications Conference-GLOBECOM 2011, IEEE, 2011, pp. 1–6.
- [155] Mike Thelwall, *Social network sites : Users and uses*, Advances in computers **76** (2009), 19–73.
- [156] Philippe Torloting, *Enjeux et perspectives des réseaux sociaux*, Institut supérieur du commerce de paris (2006).
- [157] Dinh Nguyen Tran, Frank Chiang, and Jinyang Li, *Friendstore : cooperative online backup using trusted nodes*, Proceedings of the 1st Workshop on Social Network Systems, 2008, pp. 37–42.
- [158] Amund Tveit, *Peer-to-peer based recommendations for mobile commerce*, Proceedings of the 1st international workshop on Mobile commerce, 2001, pp. 26–29.
- [159] Patrick Van Eecke and Maarten Truyens, *Privacy and social networks*, Computer Law & Security Review **26** (2010), no. 5, 535–546.

- [160] Donghui Wang, Yanchun Liang, Dong Xu, Xiaoyue Feng, and Renchu Guan, *A content-based recommender system for computer science publications*, Knowledge-Based Systems **157** (2018), 1–9.
- [161] Xinhua Wang, Xinxin Yang, Lei Guo, Yu Han, Fangai Liu, and Baozhong Gao, *Exploiting social review-enhanced convolutional matrix factorization for social recommendation*, IEEE Access **7** (2019), 82826–82837.
- [162] Stéphane Weiss, Pascal Urso, and Pascal Molli, *Logoot : A scalable optimistic replication algorithm for collaborative editing on p2p networks*, 2009 29th IEEE International Conference on Distributed Computing Systems, IEEE, 2009, pp. 404–412.
- [163] Stephane Weiss, Pascal Urso, and Pascal Molli, *Logoot-undo : Distributed collaborative editing system on p2p networks*, IEEE transactions on parallel and distributed systems **21** (2010), no. 8, 1162–1174.
- [164] Jianshu Weng, Ee-Peng Lim, Jing Jiang, and Qi He, *Twiterrank : finding topic-sensitive influential twitterers*, Proceedings of the third ACM international conference on Web search and data mining, 2010, pp. 261–270.
- [165] Christo Wilson, Troy Steinbauer, Gang Wang, Alessandra Sala, Haitao Zheng, and Ben Y Zhao, *Privacy, availability and economics in the polaris mobile social network*, Proceedings of the 12th Workshop on Mobile Computing Systems and Applications, 2011, pp. 42–47.
- [166] Tawaziwa Wushe and Jacob Shenje, *The relationship between social media usage in the workplace and employee productivity in the public sector : Case study of government departments in harare*, SA Journal of Human Resource Management **17** (2019), no. 1, 1–10.
- [167] Guandong Xu, Zhiang Wu, Yanchun Zhang, and Jie Cao, *Social networking meets recommender systems : survey*, International Journal of Social Network Mining **2** (2015), no. 1, 64–100.
- [168] Tianyin Xu, Yang Chen, Jin Zhao, and Xiaoming Fu, *Cuckoo : towards decentralized, socio-aware online microblogging services and data measurements*, Proceedings of the 2nd ACM International Workshop on Hot Topics in Planet-scale Measurement, 2010, pp. 1–6.
- [169] Liming Yao, Zhongwen Xu, Xiaoyang Zhou, and Benjamin Lev, *Synergies between association rules and collaborative filtering in recommender system : An application to auto industry*, Data Science and Digital Business, Springer, 2019, pp. 65–80.
- [170] Xinzhi Zhang and Wenshu Li, *From social media with news : Journalists’ social media use for sourcing and verification*, Journalism Practice **14** (2020), no. 10, 1193–1210.
- [171] Youliang Zhong, Lan Du, and Jian Yang, *Learning social relationship strength via matrix co-factorization with multiple kernels*, International Conference on Web Information Systems Engineering, Springer, 2013, pp. 15–28.
- [172] Youliang Zhong, Weiliang Zhao, and Jian Yang, *Personal-hosting restful web services for social network based recommendation*, International Conference on Service-Oriented Computing, Springer, 2011, pp. 661–668.

- [173] M. Zimmer, *Mark zuckerberg's theory of privacy*, <http://wapo.st/1gJOqEu> (2014).

Contributions de l'auteur (7) :— Publications (3) :

- ◆ **BADIS Lyes**, AMAD Mourad , et AÏSSANI Djamil. A Log Based Update of Replicated Profiles in Decentralized Social Networks. *Journal of Digital Information Management*, 2018, vol. 16, no 5, p. 231. DOI : 10.6025/jdim/2018/16/5/230-245 .
- ◆ **Badis Lyes**, Amad Mourad, AÏSSANI Djamil, Abbar Sofiane, P2PCF : A collaborative filtering based recommender system for peer to peer social networks, *Journal of High Speed Networks*, vol. 27, no. 1, pp. 13-31, 2021. DOI : 10.3233/JHS-210649 .
- ◆ AMAD Mourad, BOUDRIES Abdelmalek, et **BADIS Lyes**. Application Layer Multicast Based Services on Hierarchical Peer to Peer Architecture. *Applied Mechanics and Materials. Trans Tech Publications Ltd*, 2019. p. 64-71.

— Communications internationales (3) :

- ◆ **BADIS Lyes**, AMAD Mourad, et AÏSSANI Djamil. Asynchronous model for updating replicated profiles in decentralized social network. In *International Conference on Mathematics and Information Technology (ICMIT)*. IEEE, 2017. p. 227-233.
- ◆ **BADIS Lyes**, AMAD Mourad, AÏSSANI Djamil, Bedjguelal Kahina and Benkerrou Aldja. ROUTIL : P2P routing protocol based on interest links. In *International Conference on Advanced Aspects of Software Engineering (ICAASE)*. IEEE, 2016. p. 1-5.
- ◆ Mourad Amad, Abdelmalek Boudries and **Lyes Badis**, ALM Based Services on HPM P2P Architecture, *The Second International Conference on Computational Science and Engineering 2017*, 29/30 novembre 2017, Kuala Lumpur, Malaisie.

— Communications nationales (1) :

- ◆ **BADIS Lyes**, AMAD Mourad, et AÏSSANI Djamil, A social recommender system with privacy preserving, *Doctoriales Recherche Opérationnelle*, Université de Béjaïa, Mercredi 12 et jeudi 13 Décembre 2018.

Résumé: Les sites de réseaux sociaux sont très performants et très répandus. Cependant, la possibilité de détourner les données des utilisateurs vers des utilisations en dehors des accords déclarés sont de plus en plus préoccupants. La recherche est orientée vers les architectures peer-to-peer afin de fournir les mêmes services sans avoir besoin d'une autorité centrale de stockage et de contrôle. Plusieurs propositions adoptent ce type d'architecture et y installent les fonctions de base des réseaux sociaux, mais ces actions sont loin de devenir des concurrents des sites centralisés actuels. Cela est dû au manque de fonctionnalités, car la plupart d'entre elles ne fournissent que des opérations de base telles que la publication et les commentaires sans fonctions avancées telles que la recherche et les systèmes de recommandation. D'autre part, les architectures proposées souffrent de lacunes en termes de performances (*disponibilité et temps de réponse*), en raison de leur dépendance aux machines des utilisateurs comme alternative aux serveurs centraux. Cette thèse vise à améliorer ces architectures décentralisées où nous apportons deux contributions. Dans la première contribution, nous proposons un nouveau modèle de mise à jour des profils des utilisateurs stockés sur les machines des utilisateurs au moyen du journal des événements normalement utilisé dans les bases de données. Cette contribution vise à réduire le coût des opérations de mise à jour, car on n'enregistre à chaque fois que les derniers changements intervenus sur les profils des utilisateurs. Dans la deuxième contribution, nous introduisons P2PCF : un nouveau système de recommandation basé sur le filtrage collaboratif, une technique qui repose sur le stockage centralisé des interactions des utilisateurs, mais nous l'avons modifiée pour correspondre à la nature décentralisée des réseaux sociaux pair à pair.

Mots clés: réseaux sociaux, pair à pair, système de recommandation, P2PCF, journal des événements, mise à jour, cohérence, réplication.

Abstract: Social networking sites are among the most successful and widespread sites, whether considering the number of users or the areas in which they are present. However, concerns are growing about the fate of user data and the possibility of diverting it to uses outside of the announced agreements. This is why research is geared towards peer-to-peer architectures in order to provide the same services without the need for a central authority to store data and control the entire system. Several proposals have been published that adopt this type of architecture and install basic functions of social networks, but these actions are still far from becoming competitors of current sites based on a central model. This is mainly due to the lack of available features, as most of them only provide basic operations like posting and commenting without advanced features like search tools and recommender systems. On the other hand, the proposed architectures suffer from performance shortcomings, such as availability and response speed, due to their reliance on user machines as an alternative to large central servers. This thesis aims to improve these decentralized architectures where we make two contributions. In the first contribution, we propose a new model for updating user profiles stored on user machines by means of the event log normally used in databases. This contribution aims to reduce the cost of updating operations, because only the latest changes to user profiles are recorded. In the second contribution, we introduce P2PCF: a new recommender system based on collaborative filtering, a technique that relies on centralized storage of user interactions, but we modified it to match the decentralized nature of peer-to-peer social networks.

Keywords: social networks, peer to peer, recommendation system, P2PCF, event log, update, consistency, replication.

ملخص: تعتبر مواقع التواصل الاجتماعي من أنجح المواقع وأوسعها انتشارا سواء بالنظر لعدد مستخدميها أو للمجالات التي تتواجد فيها. لكن المخاوف تتزايد حول مصير معطيات المستخدمين وامكانية تحويلها لاستخدامات خارج الاتفاقيات المعلنة. هذا ما جعل الأبحاث تتوجه نحو أنظمة الند للند من أجل توفير نفس الخدمات دون الحاجة لجهة مركزية تخزن المعطيات وتتحكم في النظام بأكمله. تم نشر عدة مقترحات تتبنى هذا النوع من الأنظمة وتركب عليها الوظائف الأساسية للشبكات الاجتماعية غير ان هذه الاعمال لا تزال بعيدة من أن تصبح منافسة للمواقع الحالية المبنية على نموذج مركزي. يرجع ذلك اساسا الى فقر الميزات المتاحة حيث لا توفر اغلبها سوى العمليات الأساسية كالنشر والتعليق دون الوظائف المتقدمة كأدوات البحث وأنظمة التوصية. من جهة اخرى تعاني الأنظمة المقترحة من نقائص في الأداء كالتوفر وسرعة الاستجابة بسبب اعتمادها على أجهزة المستخدمين كبديل للخوادم المركزية الكبيرة. يأتي هذا العمل من أجل تحسين هذه الأنظمة اللامركزية حيث نقدم مساهمتين. في المساهمة الاولى نقترح نموذج جديد لتحسين معطيات المستخدمين الموزعة على اجهزة المستخدمين عن طريق سجل الاحداث المستخدم عادة في قواعد البيانات. تهدف هذه المساهمة للتخفيف من تكلفة عمليات التحسين حيث نكتفي في كل مرة بتسجيل التعديلات الاخيرة التي حدثت على ملفات المستخدمين. في المساهمة الثانية نقدم نظام توصية مبنى على الفلترة التعاونية وهي تقنية تعتمد على تخزين مركزي لتفاعلات المستخدمين لكن قمنا بتعديلها لتناسب مع الطبيعة اللامركزية للشبكات الاجتماعية المبنية للند للند.

الكلمات المفتاحية: الشبكات الاجتماعية, الند للند, نظام التوصية, P2PCF, سجل الاحداث, تحسين, التناسق, التكرار