

الجمهورية الجزائرية الديمقراطية الشعبية

République Algérienne Démocratique et Populaire

Ministère de l'enseignement supérieur et de la recherche scientifique

Université de 8 Mai 1945 – Guelma -

Faculté des Mathématiques, d'Informatique et des Sciences de la matière

Département d'Informatique



**Mémoire de Fin d'études Master**

**Filière :** Informatique

**Option :** Systèmes Informatiques

**Thème :**

---

---

**Analyse des réseaux sociaux et détection de  
communautés**

---

---

**Encadré Par :**

Mme Louafi Wafa

**Présenté par :**

Louafi Noor elHouda

**Juillet 2019**

# Remerciements

Avant tout nous remercions dieu le tout puissant qui m'a donné la force et le courage pour qu'on puisse accomplir ce modeste travail.

Un grand remerciement à **Mme Louafi Wafa** pour son encadrement et ses Conseils précieux qu'elle nous avons prodigués durant toutes l'année.

On tient à exprimer notre profond amour et immense gratitude à mes parents pour leur contribution, leur soutien et leur patience.

Enfin, je remercie tous les enseignants qui ont contribué à ma formation, et à Tous ceux qui ont participé de près ou de lion à la réalisation de ce travail.





Avant tout, je remercie mon dieu qui m'a donné la force et la patience pour que je puisse accomplir ce modeste travail.

Je tiens à exprimer mon profond amour et mon immense gratitude aux êtres les plus chères :

**Mes chers parents** que ce travail puisse être un témoignage de respect et attachement aux valeurs que vous nous avez inculqués,

# Résumé

Un réseau social est un ensemble d'acteurs sociaux reliés entre eux par des connexions représentant des interactions sociales.

L'analyse des réseaux sociaux, fondé principalement sur la théorie des graphes et la détection de communautés qui consiste à former des groupes de sorte que les nœuds au sein d'un même groupe soient connectés selon des critères bien précis.

La détection de ces communautés a été étudiée dans de nombreux domaines où les systèmes complexes sont souvent représentés sous forme de graphiques, ce problème est fortement lié à la relation entre les sommets, malgré il existe plusieurs algorithmes de détection de communautés dans la littérature mais jusqu'aujourd'hui le problème n'est pas encore résolu.

Notre travail consiste à implémenter quelques métriques d'analyse des réseaux sociaux et quelques algorithmes de l'état de l'art concernant la détection des communautés dans les réseaux sociaux et proposer un nouvel algorithme de détection de communautés dans les réseaux sociaux qui se base principalement sur le concept de voisinage.

Mots clés

Détection de communautés, Voisinage, Réseau social, Analyse des réseaux sociaux, Théorie de graphe.

# Table de matières

Liste des tableaux .....	7
Introduction générale.....	8
Chapitre 01 : Introduction aux réseaux sociaux.....	10
1 Introduction .....	10
2 Définition .....	10
3 Un peu d'histoire :.....	11
4 Exemples des réseaux sociaux : .....	11
4.1 Club de karaté du Zachary: (Zachary, 1977).....	11
4.2 Le réseau de page web :.....	12
4.3 Football américain .....	12
4.4 Le réseau de dauphins de Lusseau.....	13
5 Conclusion.....	14
Chapitre 02 : Théorie des graphes et Analyse des réseaux sociaux .....	15
1 Introduction .....	15
2 Théorie des graphes :.....	15
2.1 Nœud .....	15
2.2 Lien.....	16
2.3 Communauté.....	16
2.4 Distance géodésique .....	16
3 Quelques métriques d'analyse des réseaux sociaux .....	16
3.1 Diamètre .....	16
3.2 Rayon (radius) .....	17
3.3 Efficacité globale .....	17
3.4 Densité.....	17
3.5 Degré d'un nœud .....	17

## Sommaire

3.6	Betweenness Centrality .....	18
3.7	Closeness centrality .....	18
4	Conclusion.....	19
Chapitre 03 : La détection de communautés .....		20
1	Introduction .....	20
2	Définition d'une communauté.....	20
3	Classification des algorithmes de détection des communautés.....	21
3.1	Détection de communautés statiques.....	22
3.2	Détection de communautés dynamiques .....	22
3.3	Détection de communautés sans chevauchement .....	22
3.4	Détection de communautés avec chevauchement.....	22
4	Les algorithmes de détection des communautés .....	23
4.1	L'algorithme de Louvain (Vincent D Blondel, 2008) .....	23
4.2	L'algorithme de Girvan-Newman (Mark EJ Newman, 2004) .....	24
4.3	L'algorithme de k-clique(G. Palla, 2005).....	25
4.4	Fluid communities (Ferran Parés, 2017) .....	26
4.5	L'algorithme de Newman (Newman M. E., 2004).....	27
4.6	L'algorithme de Radicchi et al. (Radicchi, 2004) .....	27
4.7	L'algorithme de Label Propagation (Usha Nandini Raghavan, 2007) .....	27
4.8	L'algorithme de LICOD (Kanawati, 2011) .....	28
4.9	L'algorithme de Psorakis et al. (Ioannis Psorakis, 2011).....	29
4.10	L'algorithme de WalkTrap (Latapy, 2006).....	29
4.11	L'algorithme de Fortunato et al. (Santo Fortunato, 2004).....	30
4.12	L'algorithme de Markov CLustering (Dongen., 2000).....	30
4.13	L'algorithme de FastQ (Aaron Clauset, 2004).....	30
5	Conclusion.....	31
Chapitre 04 : Conception du système.....		32

## Sommaire

1	Introduction .....	32
2	Problématique.....	32
3	Objectif.....	32
4	Schéma et description générale de l’approche proposée.....	32
5	Conception détaillée de l’approche proposée.....	35
5.1	La première phase.....	35
5.1.1	Initialisation.....	35
5.1.2	Arrangement.....	35
5.2	La deuxième phase .....	36
5.2.1	Regroupement 01 .....	36
5.2.2	Regroupement 02 .....	36
5.2.3	Regroupement 03 .....	36
5.3	La troisième phase .....	36
5.3.1	La table de probabilité.....	37
5.3.2	Les communautés.....	37
6	L’algorithme de notre approche .....	37
6.1	L’algorithme de la première phase .....	37
6.2	L’algorithme de la deuxième phase.....	38
6.3	L’algorithme de la troisième phase .....	39
7	Exemples illustratifs.....	40
7.1	Exemple 01 .....	40
7.1.1	La première phase .....	40
7.1.2	La deuxième phase .....	41
7.1.3	La troisième phase.....	42
7.2	Exemple 02 .....	43
7.2.1	La première phase .....	44
7.2.2	La deuxième phase .....	44

## Sommaire

7.2.3	La troisième phase.....	45
8	Conclusion.....	46
Chapitre 05 : Implémentation du système.....		47
1	Introduction .....	47
2	Environnement de développement .....	47
2.1	Environnement matériel .....	47
2.2	Environnement logiciel.....	47
3	Langages de programmation .....	47
4	Architecture de notre application .....	50
5	Fonctionnalités du système .....	52
6	Présentation du système .....	52
7	Résultats et discussions .....	55
8	Les avantages et les limite de notre méthodes .....	56
9	Conclusion.....	57
Conclusion et perspectives .....		58
Bibliographie.....		59



## Liste des figures

Figure 1.1 Réseau social d'amitié.....	11
Figure 1.2 Réseau du Zachary Karaté Club. ....	12
Figure 1.3 Réseau de page web.....	12
Figure 1.4 Structure de communautés identifiée par Walktrap pour le réseau du Football américain. ....	13
Figure 1.5 Représentation du réseau dauphins de Lusseau.....	13
Figure 0.1 Représentation des nœuds.....	16
Figure 2.2 Représentation d'un lien entre deux acteurs.....	16
Figure 2.3 Représentation de la distance géodésique.....	16
Figure 2.4 Degré de chaque nœud.....	18
Figure 2.5 Betweenness Centrality.....	18
Figure 2.6 Closeness Centrality. ....	19
Figure 3.1 Représentation simple d'une communauté.....	21
Figure 3.2 Les approches de classification.....	21
Figure 3.3 Structure d'une communauté non-chevauchante.....	22
Figure 3.4 Structure d'une communauté chevauchante... ..	23
Figure 3.5 Visualisation des étapes de l'algorithme de Louvain.. ....	24
Figure 3.6 Exemple d'application de la méthode de palla et al.. ....	26
Figure 3.7 Trois structures de communautés trouvées par (Usha Nandini Raghavan, 2007) pour le réseau du club de karaté (Zachary, 1977)... ..	28
Figure 4.1 Conception générale de notre approche.....	34
Figure 4.2 Les métriques utilisées pour l'arrangement.....	41
Figure 4.3 Résultat de l'exemple 1.....	43
Figure 4.4 Graphe de l'exemple 2.....	44
Figure 4.5 Résultat de l'exemple 2.....	46
Figure 5.1 Le site d'installation de python.....	48
Figure 5.2 Interface d'anaconda.....	49
Figure 5.3 Architecture générale du système.....	51
Figure 5.4 Métriques d'ARS et algorithmes de détection de communautés.....	53
Figure 5.5 La saisie des données d'un réseau social.....	53
Figure 5.6 Les résultats obtenus après l'analyse des données. ....	54

## Sommaire

Figure 5.7 Les résultats de l'exécution des algorithmes de CD.....	55
Figure 5.8 Résultat d'exécution de notre algorithme.....	55

## Liste des tableaux

Tableau 1.1 L'ensemble des données des réseaux sociaux.....	14
Tableau 4.1 les communautés initiaux et leur arrangement.....	40
Tableau 4.2 Table de probabilité avec décision.....	42
Tableau 4.3 Résultat d'arrangement.....	44
Tableau 4.4 Table de probabilité avec décision.....	45
Tableau 5.1 Caractéristique du matériel.....	47
Tableau 5.2 Résultats de l'exécution des algorithmes sur le réseau des primates.....	56

### Introduction générale

Un réseau est un ensemble des nœuds reliant entre eux par des liens, cette définition est relatif à beaucoup de systèmes qui existe aujourd'hui dans notre monde tel que: réseaux sociaux, réseaux Web, réseaux biologiques, etc. L'analyse de ces réseaux trouve ses origines théoriques dans les travaux des mathématiciens sur les graphes, il a attiré beaucoup d'attentions parce qu'il est exécutable sur une large gamme d'applications dans diverses domaines citons, la détection des communautés dans les réseaux sociaux.

La détection de sous-ensembles de sommets plus densément connectés que d'autres, appelés des communautés, est un problème que l'on retrouve énormément dans la sociologie. Ces communautés jouent un rôle important dans l'organisation ou la structuration des réseaux sociaux, il s'agit de déterminer des classes dans un graphe.

Dans ce travail, on va analyser tout d'abord les données des réseaux sociaux et implémenter quelques algorithmes de détection de communautés sur ses réseaux et a base de ça on va proposer une nouvelle méthode de détection des communautés non supervisée chevauchantes dans les réseaux sociaux qui est simple, efficace et facile à mettre en œuvre.

Le principe de notre approche est simple, chaque nœud représente une partition avec tous ses voisins, une méthode de « regroupement des classes » est réalisée, jusqu'à ce qu'on peut plus regrouper, l'inclusion d'un nœud à une communauté ce fait par le calcul de probabilité de l'appartenance.

Notre algorithme se base principalement sur le voisinage puisqu'il est la base de la majorité des tous ces critères, notre idée est très simple mais aussi très efficace.

Pour réaliser ce travail on a tout d'abord commencé avec cette introduction générale puis on va détailler le mémoire en cinq chapitres dans ce qui suit:

**Le premier chapitre** est consacré à définir les réseaux sociaux et les réseaux sociaux numériques (réseautage social) passant par un peu d'histoire de la naissance du réseau social, comme une première partie , puis dans la seconde partie quelques exemples des réseaux sociaux seront abordés.

**Le deuxième chapitre** est consacré à présenter la théorie des graphes et les principaux métriques utilisés dans l'analyse les réseaux sociaux (ARS). Nous évoquerons l'analyse des réseaux sociaux et terme de graphe et l'analyse en termes de données.

## Introduction Générale

**Le troisième chapitre** est consacré à présenter la détection de communautés dans réseaux sociaux à travers quelques travaux de recherches sur les travaux connexes dans ce domaine.

**Le quatrième chapitre** est consacré à présenter la conception générale de notre approche, la description détaillée de notre algorithme, ainsi que l'illustration de quelques exemples.

**Le cinquième chapitre** est dédié à présenter les différents outils utilisés pour implémenter notre système et les interfaces de notre système.

Et nous terminons le mémoire par une conclusion et quelques perspectives.

## **Chapitre 01 : Introduction aux réseaux sociaux.**

### **1 Introduction**

Les réseaux sociaux sont sans doute vieux comme l'humanité, ils sont devenus très célèbres dans le domaine de recherches, la notion de réseau connaît un succès grandissant depuis quelques décennies. Un réseau social est basé sur le principe de communautés, le réseau sur internet est dit social lorsqu'il permet d'échanger les informations avec d'autres membres inscrits sur le même réseau, et il apparut sur internet en 1995 (S. Bhagwat, 2013), il est de plus en plus utilisé par de nombreuses personnes dans des différents domaines donc il a une grande popularité entre les personnes dans des pays différents, il permet aux différents individus d'un système d'interagir en communautés et de se regrouper selon des critères communs.

Dans ce chapitre nous allons tout d'abord définir ce qu'est un réseau social et un réseau social numérique (réseautage social) passant par un peu d'histoire de la naissance du réseau social, comme une première partie, puis dans la seconde partie quelques exemples des réseaux sociaux seront abordés, finissant par une conclusion.

### **2 Définition**

Un réseau social est un graphe composé des nœuds qui représentent les individus ou organisations, connectés par des liens représentant une relation "sociale" : appartenance à la même famille, échange de messages, goûts communs (Viennet, 2008)

Michel Forsé (Forcé, 2008) a défini les réseaux sociaux comme suit : "un réseau social est un ensemble de liens entre un ensemble d'acteurs. Cet ensemble peut être organisé (une entreprise, par exemple) ou non (comme un réseau d'amis) et ces relations peuvent être de nature fort diverse (pouvoir, échanges de cadeaux, conseil, etc.), spécialisées ou non, symétriques ou non Les acteurs sont le plus souvent des individus, mais il peut aussi s'agir de ménages, d'associations, etc."

D'après Nicole B. Ellison (ELLISON, 2007) Les réseaux sociaux numériques tels que MySpace, Facebook, Cyworld, et Bebo sont des services web qui permettent aux utilisateurs de créer un profil public ou semi-public dans un système lié et limité, articuler une liste d'autres utilisateurs avec lesquels ils partagent la connexion, et afficher puis parcourir leur liste de connexions et celles faites par d'autres dans le système.

Les réseaux sociaux sont des groupes de personnes, ou des communautés, qui partagent un intérêt, une perspective ou des antécédents communs (S. Bhagwat, 2013)

La figure 1.1 va nous présenter un exemple simple de réseau social d'amis

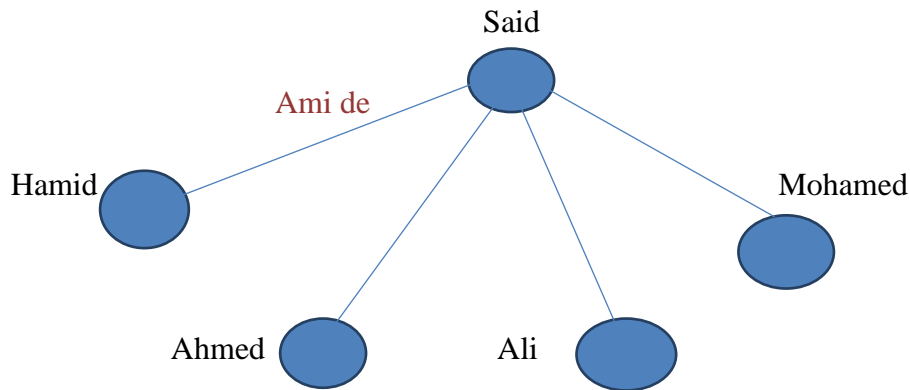


Figure 1.1 Réseau social d'amitié.

### 3 Un peu d'histoire :

La première personne à être représenté un réseau social est Jacob Levy Moreno au début des années 1930. (Barbié, 2010)

Le premier site de réseau social a été lancé en 1995. C'était classmates.com. L'objectif de ce site était l'école, le collège, le travail et l'armée, dans les premières années, les amis n'étaient pas visibles sur les sites mais classmates.com permettait aux personnes d'apprécier leur lycée ou collège et sur le réseau pour les autres qui étaient également affiliés. Mais les utilisateurs ne pouvaient créer des profils ou des amis qu'après des années plus tard.(S. Bhagwat, 2013)

### 4 Exemples des réseaux sociaux :

Les réseaux sociaux sont très populaires dans tous les domaines et il existe énormément des réseaux sociaux dans le monde actuel. Nous avons essayé de prendre des réseaux issus de cas réels, on va citer des exemples sur les réseaux sociaux les plus utilisés.

#### 4.1 Club de karaté du Zachary: (Zachary, 1977)

Le réseau de Zachary est un réseau social qui représente un club de karaté contenant 34 membres de l'université San Francisco aux Etats Unis représentés par des nœuds et 78 liens représentant les relations d'amitiés entre ces membres, la figure 1.2 va nous représenter le réseau d'amitié de ces membres.

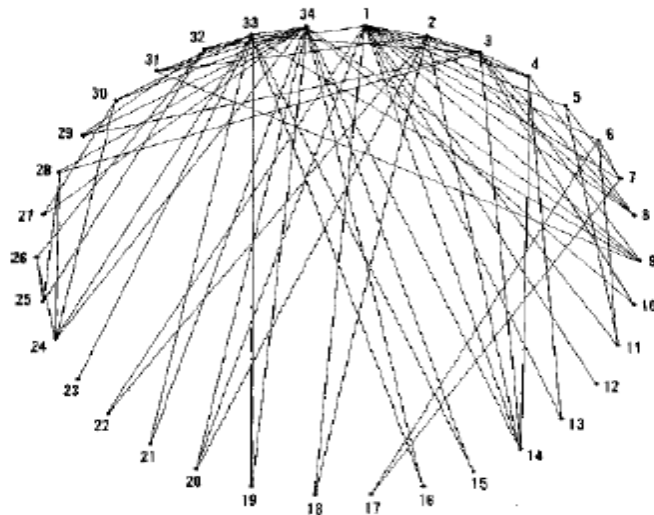


Figure 4.1 Réseau du Zachary Karaté Club (Zachary, 1977).

## 4.2 Le réseau de page web :

Les nœuds de ce graphe représentent les pages sur un site web et les arcs représentent les liens hypertextes entre eux (Mark EJ Newman, 2004), on regroupe les pages traitant le même sujet, ce réseau contient plusieurs millions de nœuds (Yacine Slimani).

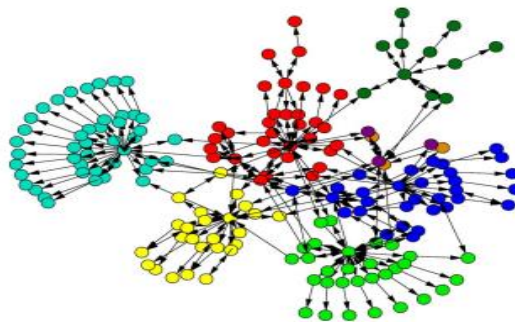


Figure 1.3 Réseau de page web (Mark EJ Newman, 2004).

## 4.3 Football américain

Le réseau de jeux de football américain (American football games), c'est un représentant du calendrier des matchs entre des équipes américaines de football durant l'année 2 000. Ce réseau est constitué de douze communautés, 115 nœuds et 613 liens.

Les sommets dans le graphique représentent les noms des collèges et les liens représentent les matchs de la saison régulière entre les deux équipes qu'ils connectent. (Talbi, 2013)



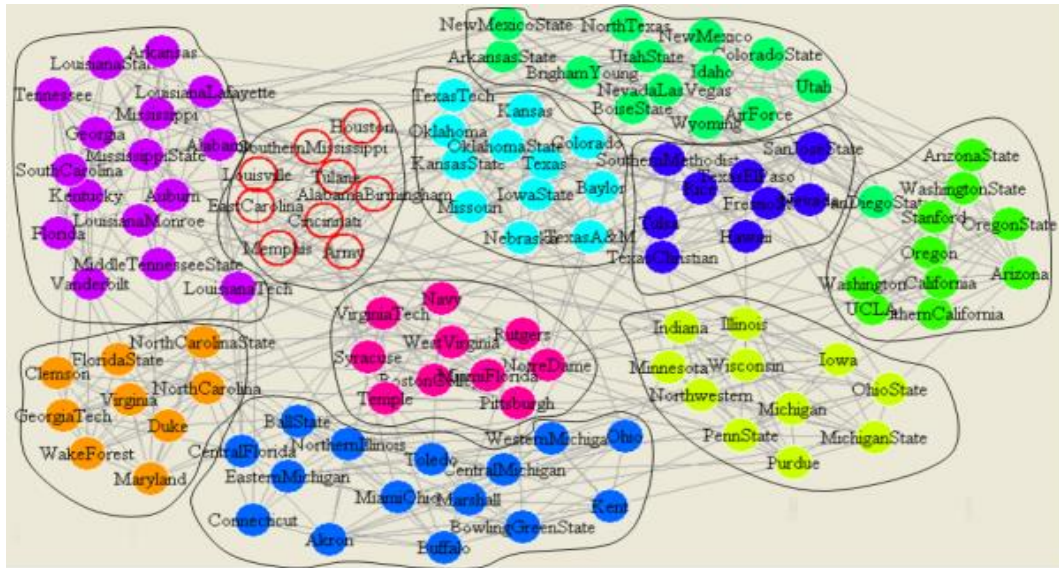


Figure 1.4 Structure de communautés identifiée par Walktrap pour le réseau du Football américain (Talbi, 2013).

#### 4.4 Le réseau de dauphins de Lusseau

Le dernier réseau réel étudié dans ce mémoire est bien le réseau de dauphins de Lusseau (David Lusseau, 2003) il contient 62 nœuds et 159 liens. Les nœuds représentent les dauphins et les liens ont été placés entre les animaux qui ont été vus ensemble plus souvent que prévu par hasard. Les dauphins se sont séparés en deux groupes, le réseau des dauphins de Lusseau et le club de karaté de Zachary sont souvent utilisés pour tester des algorithmes de détection de communautés. (Fortunato, 2010).

La figure suivante va nous présenter ce réseau social.

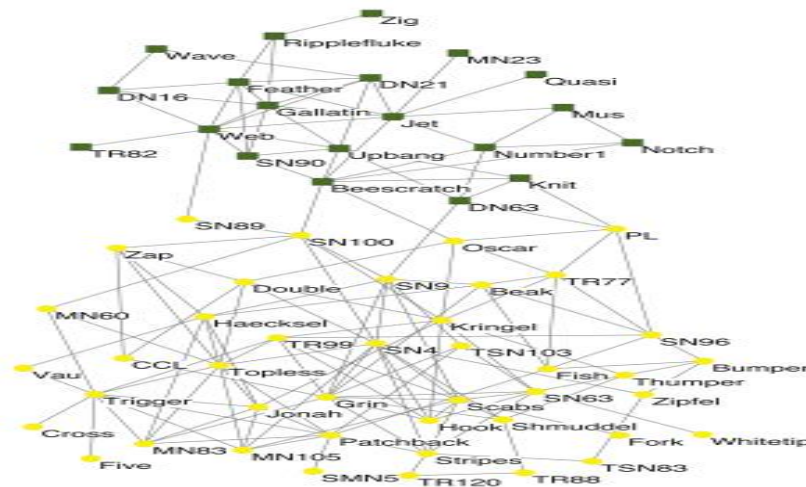


Figure 1.5 Représentation du réseau dauphins de Lusseau.(Fortunato, 2010)

Il existe plusieurs autres réseaux sociaux tels que :

Réseau de musiciens des JAZZ (GLEISER, 2003).

Un réseau d'Ego de Face book (Julian McAuley., 2012).

Les livres politiques(abdelhamid, 2015)...etc.

Le tableau 1.1 suivant nous représente l'ensemble des données des réseaux sociaux les plus connus :

Le réseau social	Nombre des nœuds	Nombre des liens
Club de karaté du Zachary	34	78
le réseau de page web	dynamique	dynamique
Football américain	115	613
le réseau de dauphins de Lusseau	62	159
Réseau de musiciens des JAZZ	198	2742
Un réseau d'Ego de Face book	4039	88234
Les livres politiques	105	441

Tableau 1.1 L'ensemble des données des réseaux sociaux.

## 5 Conclusion

D'après les définitions qu'on a cité précédemment un réseau social est un ensemble de nœuds et de liens où chaque nœud modélise un acteur et chaque lien une relation entre deux acteurs, donc on peut le représenté par un graphe et comme il se représente par un graphe donc l'analyse des réseaux sociaux se base sur les notions de la théorie des graphes (Scott, 1988) tels que le chemin et pour cette raison le chapitre suivant va être la théorie des graphes et l'analyse des réseaux sociaux.

## Chapitre 02 : Théorie des graphes et Analyse des réseaux sociaux

### 1 Introduction

L'analyse des réseaux sociaux (ou SNA : Social Network Analysis) c'est une étude basée sur plusieurs théories dont la théorie des graphes, il trouve ses origines théoriques dans les travaux des mathématiciens sur les graphes, d'après (Scott, 1988) donc on trouvera dans l'analyse des réseaux sociaux des terminologies de la théorie des graphes telles que le degré, le poids d'un lien, etc.

Un graphe est un ensemble de points, dont certaines paires sont reliées. (Flament, 2017)

Les graphes ont été largement utilisés dans l'analyse des réseaux sociaux comme moyens de représenter formellement les relations sociales (M.Denny, 2014), donc pour pouvoir commencer et utiliser un langage cohérent on va tout d'abord définir la terminologie de la théorie des graphes, et par la suite on va citer les principales métriques d'analyse des réseaux sociaux.

### 2 Théorie des graphes :

Les graphes ont été largement utilisés pour représenter formellement les relations sociales et de quantifier les propriétés structurelles sociales importantes.

La théorie des graphes a été largement utilisée en anthropologie sociale, psychologie sociale, communications, affaires, recherche organisationnelle et géographie.

Quand un graphe est utilisé comme modèle d'un réseau social, les arcs sont utilisés pour représenter les acteurs et les lignes reliant les points sont utilisés pour représenter les liens entre les acteurs, En ce sens, un graphe est un modèle de réseau social de la même manière qu'un train miniature est un modèle de système ferroviaire (S. Wasserman, 1995)

Nous introduisons ici des notions et notations principales de la théorie des graphes qui nous serviront dans la suite de ce chapitre à l'aide de (M.Denny, 2014)

#### 2.1 Nœud

Aussi appelé individu ou acteur c'est le sommet d'un graphe, il peut avoir des relations avec d'autres individus, Au niveau de la figure 2.1 A, B, C sont des nœuds (acteurs).

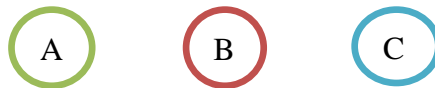


Figure 2.1 Représentation des nœuds.

## 2.2 Lien

C'est une relation particulière, bien spécifiée entre deux acteurs, les liens peuvent être Non-dirigés lorsque la relation signifie la même chose aux deux acteurs ou bien dirigés dans le cas contraire. Au niveau de la figure 2.2 le trait pointillé est un lien non-dirigé entre les deux acteurs A et B (acteurs).

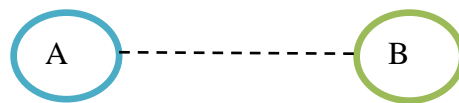


Figure 2.2 Représentation d'un lien entre deux acteurs.

## 2.3 Communauté

C'est un sous-ensemble d'acteurs qui partagent certaines caractéristiques communes, ayant beaucoup d'interactions entre elles et peu d'interactions avec l'extérieur. (Fortunato, 2010)

## 2.4 Distance géodésique

Désigne le plus petit nombre de liens à traverser pour arriver entre deux nœuds quelconques. Par exemple, dans la figure 2.3 décrit ci-dessous, la distance géodésique entre l'acteur A et l'acteur C est 2, alors que la distance entre l'acteur A et B est seulement 1

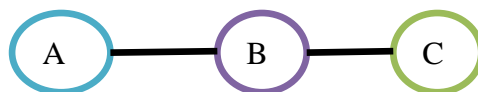


Figure 2.3 Représentation de la distance géodésique.

## 3 Quelques métriques d'analyse des réseaux sociaux

Nous allons représenter par la suite quelques métriques d'analyse des réseaux sociaux en termes de graphe (M.Denny, 2014)

### 3.1 Diamètre

La distance maximale possible existante entre deux sommets d'un graphe, le diamètre est défini par  $D(G) = \max_i (\max_j (\text{distance}(N_i, N_j)))$ .

Le diamètre du graphe représenté au niveau de la figure Fig.2.3 est 2.

### 3.2 Rayon (radius)

Contrairement au diamètre le rayon est la distance minimale à laquelle puisse se trouver un sommet de tous les autres, il définit implicitement quels nœuds du graphe sont des nœuds

«Centraux», et il est défini par  $R(G) = \min_i (\max_j (\text{distance}(N_i, N_j)))$ .

Le rayon du graphe représenté au niveau de la figure Fig.2.3 est 1.

### 3.3 Efficacité globale

C'est une mesure de taux où les nœuds peuvent être atteints les uns des autres dans le graphe.

Elle variera toujours entre 0 et 1.

Intuitivement, une efficacité globale élevée signifie que la distance entre les nœuds est généralement courte, lorsqu'elle atteint un nombre maximal qui est 1 cela signifie que chaque nœud peut être atteint en une seule étape, et une faible efficacité globale signifie que plusieurs étapes doivent être franchies entre les nœuds.

L'efficacité globale du graphe représenté au niveau de la figure Fig.2.3 est 0.833333333333.

### 3.4 Densité

C'est un paramètre mesure si le graphe a beaucoup d'arêtes ou peu, il est défini par le rapport entre les arêtes existantes et les arêtes potentiellement existantes.

La densité 0 correspond au graphe où tous les sommets sont isolés, et la densité 1 au graphe complet (Schaeffer, 2007).

La densité du graphe représenté au niveau de la figure Fig.2.3 est 0.666666666667.

### 3.5 Degré d'un nœud

Le degré d'un nœud est le nombre d'arêtes sur le nœud, il peut être calculé en tenant compte ou en ignorant les poids sur les bords.

La figure 2.4 décrit ci-dessous va nous représenter le degré de chaque nœud de la figure 2.3.

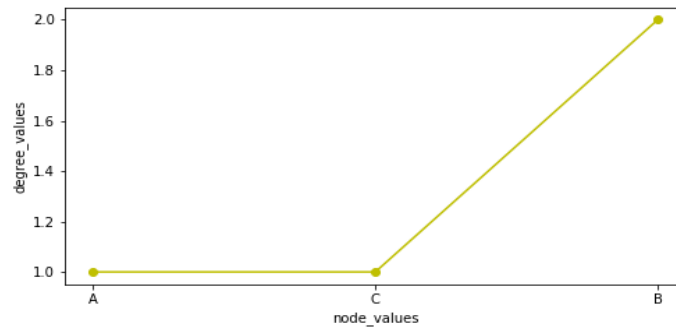


Figure 2.4 Degré de chaque nœud.

### 3.6 Betweenness Centrality

D'après (M.Denny, 2014) c'est la somme entre « les longueurs de chemin les plus courtes entre chaque ensemble d'altérations où le chemin passe par l'acteur pour lequel nous calculons la mesure divisée par les longueurs de chemin les plus courtes (pas nécessairement à travers l'acteur cible) entre ces acteurs ».

La figure 2.5 décrit ci-dessous va nous représenter graphiquement Betweenness Centrality du graphe de la figure 2.3.

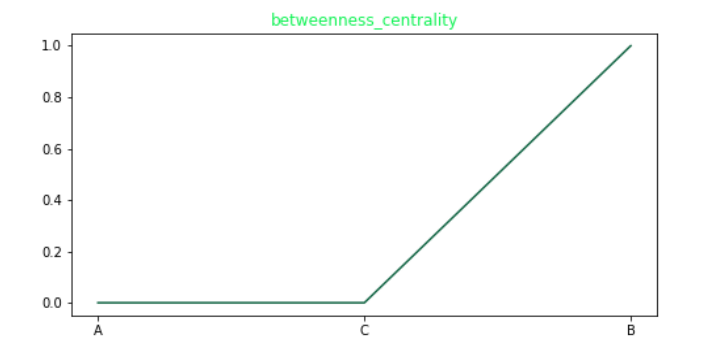


Figure 2.5 Betweenness Centrality.

### 3.7 Closeness centrality

C'est une catégorie d'algorithmes qui évaluent la centralité d'un nœud par sa proximité (distance) par rapport aux autres nœuds. Elle mesure le nombre d'étapes (liens) nécessaires à un acteur particulier pour accéder à tous les autres acteurs du réseau.

La figure 2.6 décrit ci-dessous va nous représenter graphiquement Closeness Centrality du graphe de la figure 2.3.

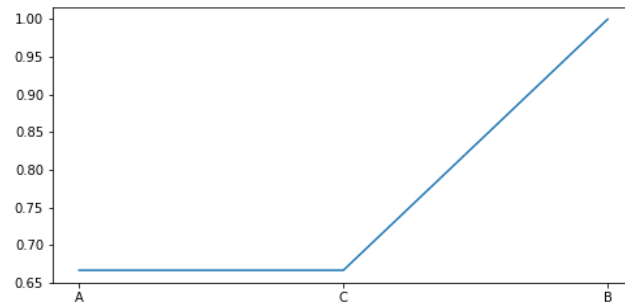


Figure 2.6 Closeness centrality.

#### 4 Conclusion

D'après ce chapitre l'analyse de réseaux sociaux s'appuie principalement sur les connaissances de la théorie des graphes, et comme la détection de communautés au niveau d'un réseau social est l'un des principaux défis de l'analyse des réseaux sociaux donc le prochain chapitre va entamer le sujet de la détection de communautés dans les réseaux sociaux.

## Chapitre 03 : La détection de communautés

### 1 Introduction

L'analyse des réseaux sociaux (ou SNA : *Social Network Analysis*). Le SNA trouve ses origines théoriques dans les travaux des mathématiciens sur les graphes, mais les premiers développements significatifs sont apparus en sciences sociales, la détection de communautés à l'intérieur d'un réseau est un sujet relativement récent mais il fait partie des principaux défis de l'analyse des réseaux sociaux, c'est un problème d'un intérêt considérable, Il a un très large éventail d'applications pas seulement en informatique, aussi en biologie et en sciences sociales, physique, etc. donc elle a attiré de nombreux chercheurs dans des domaines diversés. Dans ce chapitre, nous présentons l'état de l'art se rapportant à la détection de communautés on va tout d'abord expliquer brièvement ce qu'une communauté. Après on va retenir les travaux connexes les plus connus dans le domaine de la détection des communautés.

### 2 Définition d'une communauté

D'après S. Fortunato (Fortunato, 2010): Une communauté est un ensemble d'entités ayant beaucoup d'interactions entre elles et peu d'interactions avec l'extérieur.

Girvan et Newman définissent une communauté comme un ensemble d'entités qui ont plus de relations internes qu'externes (Newman M. G., 2002)

Radicchi et al. (Filippo Radicchi, 2004) Améliorent cette définition par la contrainte que chaque individu d'une communauté a plus de voisins à l'intérieur de sa communauté qu'à l'extérieur. Ils appellent ces structures les communautés fortes.

Généralement les réseaux sociaux sont constitués de parties plus denses représentant des personnes qui ont été connectées avec précision les unes aux autres qu'on appelle une communauté et d'autres parties plus dispersées.

La Figure 3.1 montre une représentation graphique simple avec deux communautés, délimitées par des cercles en pointillés.

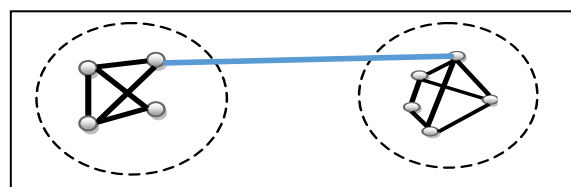




Figure 3.1. Représentation simple d'une communauté.

### 3 Classification des algorithmes de détection des communautés

La détection des communautés dans un réseau social n'est pas une tâche facile, elle peut être vue comme un problème de partition des nœuds de réseau en plusieurs groupes et chaque groupe définit une communauté tels que ces groupes de nœuds satisfasse efficacement le problème de la minimisation des liens inter-communautés, et la maximisation des liens intra-communautés (Cazabet, 2013). Elle a reçu beaucoup d'attention au cours de la dernière décennie et de nombreuses approches ont été proposées dont la plupart entre eux sont très coûteuses en calcul (Fortunato, 2010), généralement ces approches sont classifiées selon deux grands axes très importants (Cazabet, 2013):

- Statique ou bien Dynamique.
- Avec ou sans chevauchements.

Et à base de ça on peut conclure 4 approches :

- Statique sans chevauchements.
- Statique Avec chevauchements.
- Dynamique sans chevauchements.
- Dynamique Avec chevauchements.

La figure 3.2 suivante nous montre ces approches :

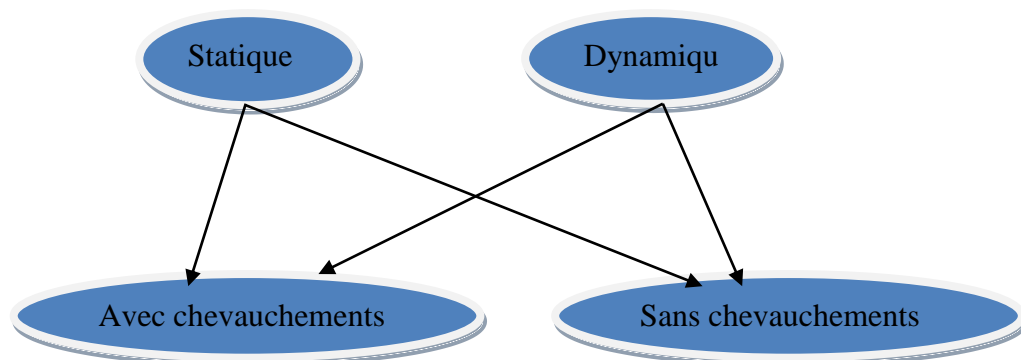


Figure 3.2 Les approches de classification.

### 3.1 Détection de communautés statiques

Un réseau statique c'est un réseau composé d'un ensemble de nœuds et d'un ensemble de liens entre eux, sans notion de temps ni d'ordre, pour les caractériser (Cazabet, 2013)

### 3.2 Détection de communautés dynamiques

Un réseau dynamique  $G$  est défini comme une séquence d'instantanés de réseau évoluant dans le temps:  $G = (G_0, G_1, G_2, \dots)$  (Nguyen, 2011), il est composé de différentes tranches temporelles, chacune représentée par une communauté distincte, contenant les liens entre les individus pour un intervalle de temps donné. Ces tranches temporelles sont séquentielles. Habituellement, les nœuds et leurs attributs sont les mêmes pour chaque tranche, tandis que les liens entre eux et les valeurs des attributs peuvent changer, La détection de communautés dynamiques est un domaine qui est encore dans une phase d'évolution. (Orman, 2013)

### 3.3 Détection de communautés sans chevauchement

Dans les communautés disjointes (sans chevauchement), chaque nœud ne peut appartenir qu'à une seule communauté (Attal, 2016) la figure 3.3 nous présente une communauté non-chevauchante où chaque nœud appartient à une seule communauté.

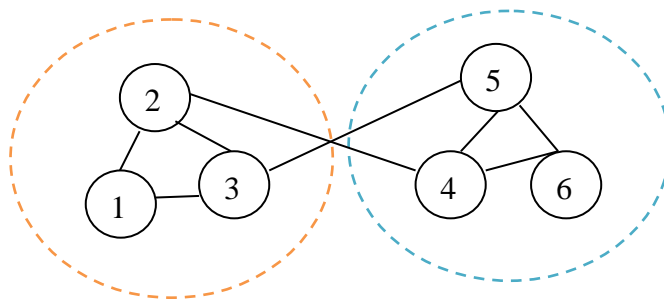


Figure 3.3 Structure d'une communauté non-chevauchante.

### 3.4 Détection de communautés avec chevauchement

Une communauté chevauchante est une communauté où certains nœuds de cette communauté sont partagés par plus d'une communauté (Attal, 2016), la figure 3.4 nous présente une communauté chevauchante où le nœud 4 est un nœud chevauchant mais les autres nœuds sont non chevauchants car ils appartiennent à une seule communauté.

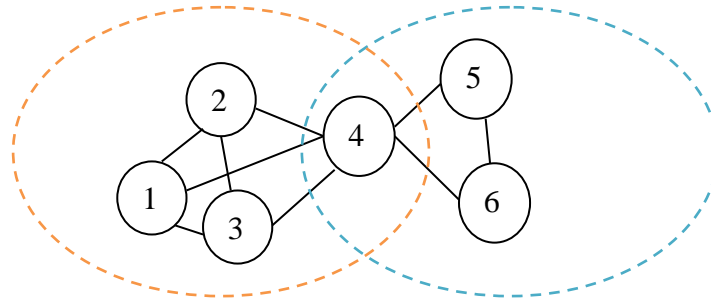


Figure 3.4 Structure d'une communauté chevauchante.

#### 4 Les algorithmes de détection des communautés

Il existe une pléthore d'algorithmes aux fondements théoriques divers abordant le problème de la détection des communautés dans les réseaux sociaux. Dans cette partie, on va présenter les algorithmes de détection les plus connus.

##### 4.1 L'algorithme de Louvain (Vincent D Blondel, 2008)

C'est une simple méthode hiérarchique de détection de communautés proposée en 2008 par Blondel et al. De l'université de Louvain, elle permet d'extraire la structure communautaire de grands réseaux, elle est basée sur l'optimisation de la modularité, il est démontré que l'algorithme de Louvain est le meilleur par rapport aux autres algorithmes de détection de communautés car il est efficace sur de grands graphes, le fait qu'il prenne en compte des graphes pondérés, et facile à interpréter son mode de calcul pour des non experts potentiellement utilisateurs contrairement à d'autres algorithmes. De plus, la qualité de communautés détectées est très bonne (Crampes, 2013).

Cet algorithme procède en deux phases qui se répètent de manière itérative.

Tout d'abord, Blondel et al considèrent chaque nœud comme une communauté différente. Donc, dans cette partition initiale, le nombre des communautés est égal au nombre des nœuds. Ensuite, pour chaque nœud  $i$ , on considère les voisins  $j$  de  $i$  et on évalue le gain de modularité qui aurait lieu en retirant  $i$  de sa communauté et en le plaçant dans la communauté de  $j$  pour laquelle ce gain est maximal et dans le cas d'une égalité, on utilise une règle de rupture, mais seulement si ce gain est positif. Si aucun gain positif n'est possible, on reste dans sa communauté d'origine. On répète ce processus séquentiellement pour tous les nœuds jusqu'à ce qu'aucune amélioration supplémentaire ne puisse être obtenue et que la première phase soit alors terminée.

Au niveau de la deuxième phase de l'algorithme on construit un nouveau réseau dont les communautés trouvées lors de la première phase sont considérés maintenant comme des nœuds, les poids des liens entre les nouveaux nœuds sont donnés par la somme des poids des liens entre les nœuds des deux communautés correspondantes. Les liens entre les nœuds d'une même communauté entraînent des boucles automatiques pour cette communauté dans le nouveau réseau. Une fois cette deuxième phase est terminée, il est alors possible d'appliquer de nouveau la première phase de l'algorithme sur le réseau pondéré obtenu et d'itérer.

Plusieurs tests ont confirmé que l'algorithme de Louvain avait une excellente précision malgré rien ne garantie que le maximum global de modularité soit atteint.

La complexité de cet algorithme est  $O(m)$

La figure 3.5 suivante nous représente une visualisation des étapes de l'algorithme de Louvain.

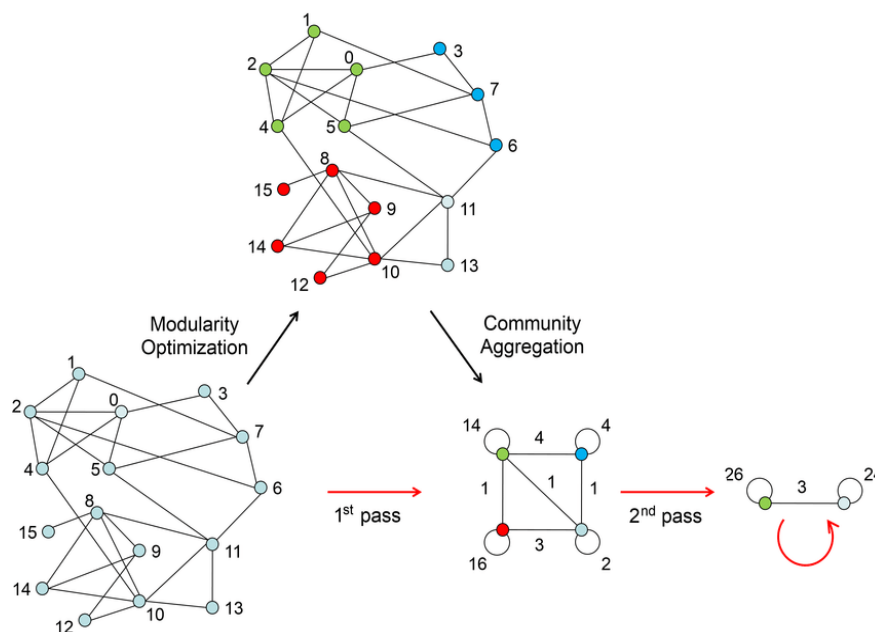


Figure 3.5 Visualisation des étapes de l'algorithme de Louvain (Vincent D Blondel, 2008)

#### 4.2 L'algorithme de Girvan-Newman (Mark EJ Newman, 2004)

Girvan et Newman (Mark EJ Newman, 2004) ont proposé un algorithme en 2004 qui détecte les communautés en supprimant de façon itératives les contours du graphique d'origine avec des scores élevés, l'algorithme supprime le bord le plus précieux, les composants connectés du réseau restant sont les communautés.

L'algorithme commence tout d'abord par le calcul des BETWEENNES de tous les bords existants du réseau comme une première étape, ensuite on supprime le bord avec l'intervalle le plus élevé, et après la suppression on recalcule les BETWEENNESS de nouvelles limites, et finalement on refait les étapes précédentes jusqu'à ce qu'il ne reste plus d'arêtes.

D'après Newman (Newman M. E., 2004) Le principal inconvénient de leur algorithme correspond aux exigences de calcul élevées qu'il fait. Dans sa forme la plus simple et la plus rapide, il tourne dans le pire des cas  $O(m^2 n)$  sur un réseau avec  $m$  arêtes et  $n$  sommets ou  $O(n^3)$  sur un réseau clairsemé

### 4.3 L'algorithme de k-clique(G. Palla, 2005)

On introduise ici une approche pour analyser les principales caractéristiques statistiques des ensembles imbriqués de communautés qui se chevauchent, on applique une technique efficace pour explorer les communautés qui se chevauchent à grande échelle. Les auteurs de cet algorithme constatent que les chevauchements sont importants et les distributions qu'ils introduisent révèlent des caractéristiques universelles des réseaux

C'est une approche qui est basée sur l'étude des cliques (une clique est un sous-ensemble de sommets qui sont en relation deux à deux) qui s'exécute en deux phases Dans la première phase, on développe un algorithme appelé MACH (Maximum Clique Heuristic), qui est une nouvelle approche permettant de calculer des cliques disjoints en utilisant une technique de branchement et de liaison heuristique. Dans la deuxième phase, les cliques de la première phase servent à guider la fusion des sommets individuels jusqu'à l'obtention d'une solution de bonne qualité. L'étape de fusion des communautés pour le papier proposé prend  $O(k)$ ,  $k$  étant le nombre de communautés.

La Figure 3.6 qui nous représente un exemple d'application où les nœuds de couleur rouge appartiennent à des communautés chevauchantes est extraite du site de Palla et al. (2005)

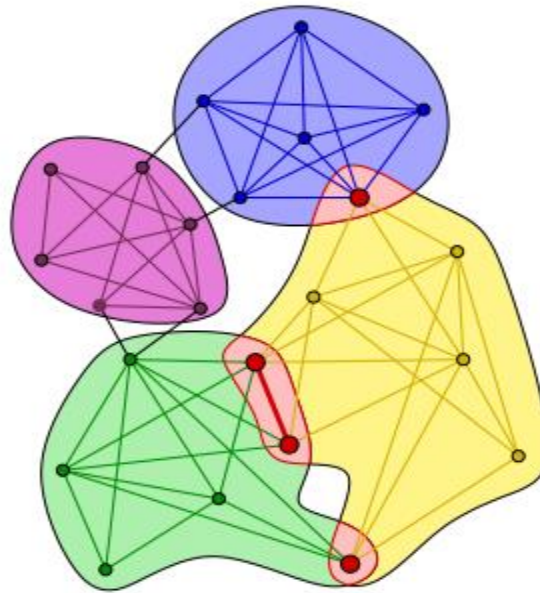


Figure 3.6 Exemple d'application de la méthode de Palla et al (G. Palla, 2005) .

Cette méthode permet la détection de communautés chevauchantes où un sommet peut appartenir à plusieurs  $k$ -cliques. La qualité de partitionnement est bonne mais il y a quelques inconvénients tels que la nécessité de préciser la taille des cliques (le paramètre  $k$ ) et la complexité algorithmique augmente de manière non linéaire en fonction du nombre de sommets ainsi que la recherche de motifs statiques qui n'est pas un trait absolu.

#### 4.4 Fluid communities (Ferran Parés, 2017)

Fluid Communities est un algorithme de détection de communautés asynchrone (chaque mise à jour de sommet est calculée en utilisant le dernier état partiel du graphe), c'est le premier algorithme de détection de communautés basé principalement sur la méthodologie de propagation et capable d'identifier un nombre variable de communautés en réseau. Il permet de détecter des communautés dans des graphiques synthétiques avec une précision proche des meilleurs algorithmes actuels.

Il est basé sur l'idée d'introduire un certain nombre de communautés (fluides) dans un graphe non complet (environnement non homogène), où les fluides vont se développer et se pousser mutuellement, influencés par la topologie de l'environnement jusqu'à atteindre un état stable.

Après une étude de la diversité des communautés trouvées par cette méthode on a trouvé qu'elles sont très différentes par rapport à celles trouvées par des méthodes alternatives, et la complexité de cet algorithme est linéaire  $O(E)$  et évolutivité.

#### **4.5 L'algorithme de Newman (Newman M. E., 2004)**

Newman a proposé un autre algorithme pour détecter la structure communautaire, il relève de la catégorie générale des méthodes de classification hiérarchique agglomérative, il se base principalement sur la notion de la modularité qui est une mesure de qualité d'une partition qui pour un graphe donné permet de vérifier de "meilleure" partition pour celle qui possède la plus grande modularité.

L'algorithme fonctionne sur des principes différents de ceux de Girvan et Newman (GN), mais donne des résultats qualitativement similaires.

La durée d'exécution la plus défavorable de l'algorithme est  $O((m + n) n)$  ou  $O(n^2)$  sur un graphe fragmenté.

#### **4.6 L'algorithme de Radicchi et al. (Radicchi, 2004)**

Dans cet algorithme on supprime à chaque itération le lien dont le coefficient de clustering est maximal. La définition de coefficient de clustering d'un lien est similaire à celle de coefficient de clustering d'un nœud. Il est donné par le nombre de circuits qui passent par le lien sur le nombre de circuits possibles. Le calcul de ce coefficient requiert des calculs locaux seulement. La complexité de calcul de l'algorithme est de l'ordre de  $O(n^2)$ .

#### **4.7 L'algorithme de Label Propagation (Usha Nandini Raghavan, 2007)**

L'idée principale de cet algorithme est:

Tout d'abord chaque sommet (X) du réseau se voit attribuer une étiquette(Label) unique (sa propre communauté), ensuite (X) détermine sa communauté en fonction des étiquettes de ses voisins. (X) fait partie de la communauté qui contient le plus grand nombre de nœuds voisins, Un processus itératif est exécuté afin que les groupes de sommets connectés puissent atteindre un consensus sur une étiquette donnant lieu à une communauté, à la fin du processus de propagation, les nœuds ayant les mêmes étiquettes sont regroupés en une seule communauté.

La figure 3.7 prise de (Usha Nandini Raghavan, 2007) représente trois structures communautaires différentes identifiées par l'algorithme sur le réseau de clubs de karaté de

Zachary (Zachary, 1977). Les communautés peuvent être identifiées par leurs nuances de couleurs grises.

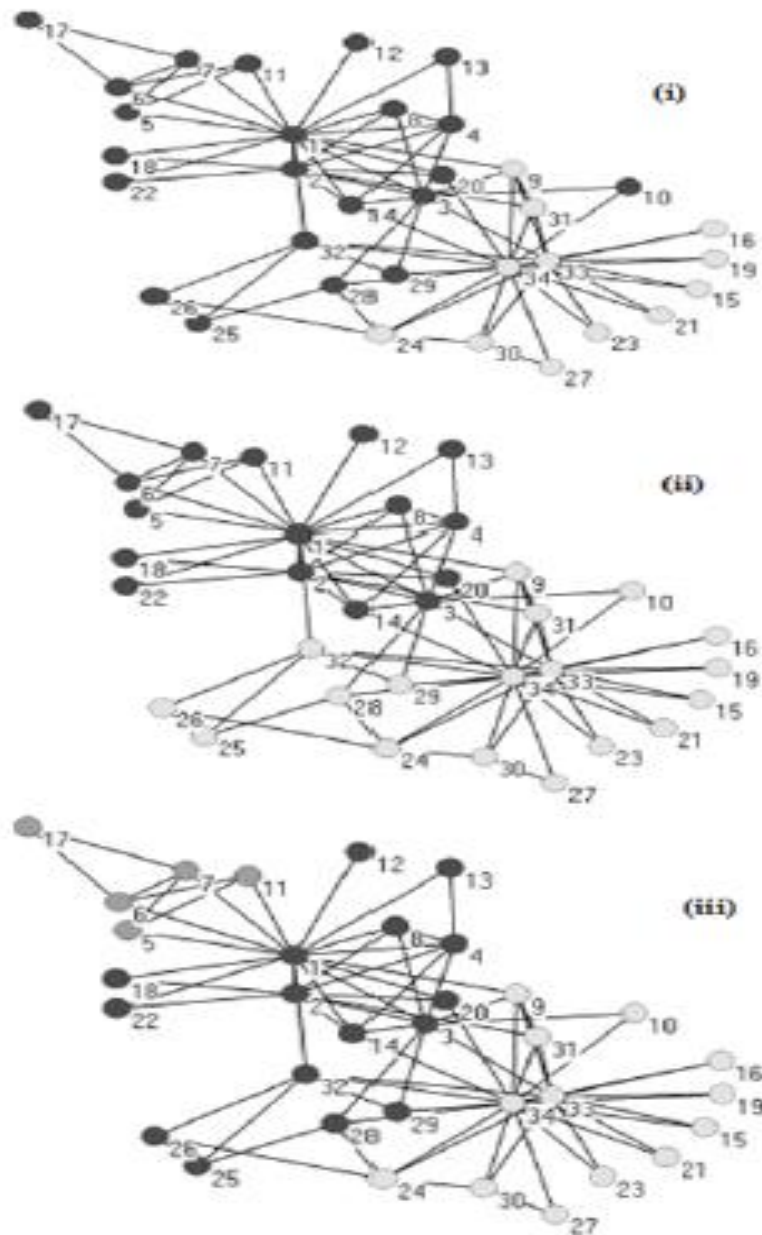


Figure 3.7 Trois structures de communautés trouvées par (Usha Nandini Raghavan, 2007) pour le réseau du club de karaté (Zachary, 1977).

#### 4.8 L'algorithme de LICOD (Kanawati, 2011)

C'est un algorithme pour construction de communautés basé sur l'idée qu'une communauté est animée par un ensemble de leaders suivis d'un ensemble nœuds,

Le suivi de différents leaders animant différentes communautés.



L'algorithme se devise en trois étapes :

Tout d'abord on identifie l'ensemble des Leaders  $L$  qui contient les nœuds ayant une centralité supérieur à celle de ces voisins. Après, l'ensemble des leaders est réduit en un ensemble de communautés de Leaders. Deux leaders sont regroupés s'ils ont un nombre de voisins communs élevé.

Les communautés sont triées en ordre décroissant, et le degré d'appartenance d'un nœud à une communauté appartient à l'ensemble, il est simplement donné par la distance minimale entre ce nœud et tout l'ensemble de communautés, on forme le vecteur de préférence pour chaque nœud.

On fusionne le vecteur de préférence d'un nœud avec ceux de ses voisins directs.

#### **4.9 L'algorithme de Psorakis et al. (Ioannis Psorakis, 2011)**

Psorakis et al. (Ioannis Psorakis, 2011) Proposent un modèle basé principalement sur la factorisation matricielle bayésienne non-négative (NMF). Dans ce travail, on présente une approche probabiliste de détection communautaire qui utilise un modèle bayésien de factorisation matricielle non négative pour extraire des modules qui se chevauchant d'un réseau. La complexité de cet algorithme est en  $O(Kn^2)$ .

Cet algorithme présente beaucoup d'avantages tels que la capacité excellente d'identification de module, ainsi que des solutions de chevauchement ou de partitionnement souple, où les communautés sont autorisées à partager des membres.

#### **4.10 L'algorithme de WalkTrap (Latapy, 2006)**

C'est une mesure de similarité entre les sommets basée sur des marches aléatoires pour établir une notion de distance entre les nœuds, Il calcule pour chaque paire de sommets  $(i,j)$  un poids correspondant à la probabilité qu'une marche aléatoire aille de  $i$  à  $j$  en  $t$  étapes. Il en déduit une distance sur les paires de nœuds. Cet algorithme appelé WALKTrap et suit ensuite une stratégie de partitionnement hiérarchique en fusionnant les communautés à faible distance.

Cet algorithme s'exécute dans le pire des cas avec le temps  $O(mn^2)$  et l'espace  $O(n^2)$ , et dans le cas du temps  $O(n^2 \log n)$  et l'espace  $O(n^2)$  dans la plupart des cas réels ( $n$  et  $m$  sont respectivement le nombre de sommets et des arêtes dans le graphe d'entrée). Des tests de

comparaison approfondis montrent que cet algorithme surpasse ceux proposés précédemment en ce qui concerne la qualité des structures de communauté obtenues et qu'il figure parmi les meilleurs en ce qui concerne le temps d'exécution.

#### **4.11 L'algorithme de Fortunato et al. (Santo Fortunato, 2004)**

C'est une nouvelle méthode de découverte des structures communautaires qui utilise la métrique de la centralité d'information pour identifier les arêtes inter-communautés, la centralité d'information est utilisée ici pour quantifier la pertinence de chacune des arêtes du réseau. La méthode consiste à rechercher et à supprimer les bords avec le score de centralité le plus élevé jusqu'à ce que le réseau se divise en composants.

La complexité de cet algorithme est de  $O(m^3n)$ .

#### **4.12 L'algorithme de Markov CLustering (Dongen., 2000)**

Le processus de cluster de Markov (processus MCL abrégé) définit une séquence de matrices stochastiques par alternance de deux opérateurs sur une matrice génératrice. C'est tout ce dont vous avez besoin pour regrouper les graphes, c'est un algorithme de cluster rapide et évolutif non supervisé pour graphes (également appelé réseaux) basé sur la simulation de flux (stochastique) dans des graphes.

Cet algorithme a une complexité proche du linéaire sur des graphes peu denses grâce à des techniques dites de «pruning », c'est-à-dire de suppression des faibles valeurs éloignées de la diagonale.

#### **4.13 L'algorithme de FastQ (Aaron Clauset, 2004)**

FastQ utilise la modularité pour guider un processus de recherche glouton qui est un processus agglomératif. Au début on place tous les sommets dans des classes dont ils sont les uniques représentants. On cherche ensuite les fusions successives qui produisent les meilleurs gains de modularité.

La complexité de FastQ selon ses auteurs est  $O(n^3 \cdot \log(n))$  dans le pire des cas, mais  $O(n \cdot \log(n))$  en pratique est le plus souvent.

## 5 Conclusion

La détection de communautés dans les réseaux sociaux est l'un des principaux défis d'analyse des réseaux sociaux.

Dans ce chapitre on a compris c'est quoi une communauté passant par la classification des algorithmes de détection des communautés, y compris l'ensemble des algorithmes de détection de communautés.

Dans ce qui suit, nous allons décrire notre proposition pour la détection de communautés dans les réseaux sociaux.

## Chapitre 04 : Conception du système

### 1 Introduction

Nous présentons, dans ce chapitre, une approche qui permet la détection de communautés chevauchantes dans les réseaux sociaux. Cette approche est basée sur le voisinage.

On va commencer tout d'abord par la problématique et les objectifs de notre travail, ensuite on va détailler les étapes de notre conception qui est basée sur l'analyse et la détection des communautés dans les réseaux sociaux en utilisant les principaux aspects de la théorie des graphes.

### 2 Problématique

D'après ce qu'on a vu dans le chapitre précédent, malgré il existe plusieurs algorithmes de détection des communautés dans les réseaux sociaux mais il existe toujours les problèmes de la détermination du nombre de classes et l'évaluation de la qualité des partitions, est-ce qu'il est possible de proposer une nouvelle approche de détection des communautés efficace, simple et facile à mettre en œuvre qui se base principalement sur les métriques d'analyses des réseaux sociaux particulièrement le voisinage?

### 3 Objectif

L'objectif de notre travail est de réaliser un environnement de détection des communautés dans les réseaux sociaux basant sur les principales métriques d'analyse des réseaux sociaux en terme de graphe et en terme des outils d'analyse des réseaux sociaux et d'implémenter quelques algorithmes de détection des communautés, pour pouvoir les comparer avec notre nouvelle approche.

Les buts à atteindre pour notre travail sont les suivants :

- Implémenter les principales métriques d'analyse des réseaux sociaux.
- Implémenter quelques algorithmes de détection de communautés.
- Proposer une nouvelle approche pour la détection des communautés dans les réseaux sociaux et donner un exemple illustratif pour notre proposition.

### 4 Schéma et description générale de l'approche proposée

Un graphe  $G(V, E)$  peut être représenté par un dessin où  $V$  (vertex) représente l'ensemble des sommets (nœuds),  $E$  l'ensemble des arêtes (lien) (Flament, 2017), on peut le représenter aussi

par une matrice dite d'adjacence qui indique les connexions entre les nœuds, un réseau social peut être représenté par un graphe dont les nœuds représentent les individus et les arrêtes représentent les relations entre eux (Sohail, 2018)

Notre but est d'utiliser d'une méthode basée sur le voisinage pour résoudre le problème de la détection de communautés dans les réseaux sociaux parce qu'il est la base de construction d'un réseau social. (Gauthier, 2015)

Pour répondre à ce problème, on a proposé une nouvelle méthode de détection des communautés locales ne prenant en compte que le voisinage d'un nœud donné et sans accéder à l'ensemble du réseau, il est principalement basé sur la notion de voisinage qui est la base du réseau social et qui affecte toutes les autres critères.

L'algorithme fonctionne sur quelques étapes itératives commençant par un cluster se chevauchant et convergeant vers des communautés chevauchantes ou disjointes. Notre approche de détection de communautés est non supervisée, elle permet à chaque nœud du réseau d'appartenir à une ou plusieurs communautés avec une probabilité, consiste d'abord à appliquer un ensemble d'opérateurs à chaque nœud pour détecter les communautés disjointes, ensuite à détecter les communautés chevauchantes à travers une succession d'opérations de regroupement des nœuds du réseau. L'algorithme que nous proposons détecte le chevauchement des nœuds en permettant à chaque nœud d'appartenir à n'importe quelle communauté avec un pourcentage de 0% à 100%.

La Figure 4.1 présente schématiquement notre approche. Elle comprend essentiellement les phases suivantes:

- Phase 01
  - Initialisation
  - Arrangement
- Phase 02
  - Regroupement des classes redondantes
  - Regroupement des classes où l'intersection est  $(n-1)$
- Phase 03
  - Construire la table de probabilité d'appartenir d'un nœud à une communauté.
  - Extraction des communautés disjointes et chevauchantes.

Le schéma suivant va nous détailler la conception générale de notre approche

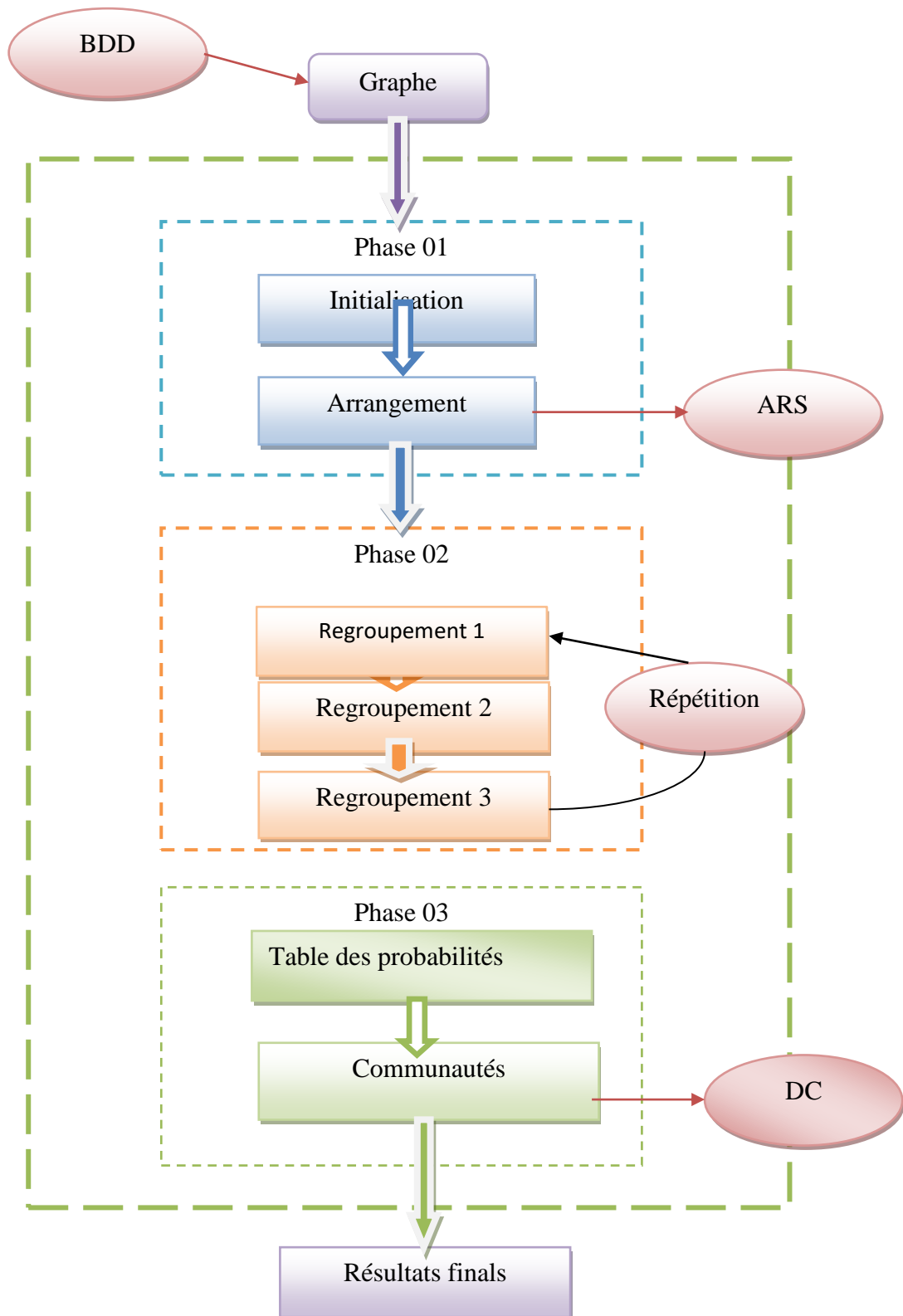


Figure 4.1 Conception générale de notre approche.

## 5 Conception détaillée de l'approche proposée

Dans notre méthode chaque nœud représente une classe avec tous ses voisins qui sont mis à jour itérativement selon le critère de voisinage. Des communautés disjointes sont découvertes lorsque la probabilité d'appartenir à une classe est forte. Une possibilité de trouver des classes chevauchantes dans le cas où la probabilité d'appartenir à plusieurs classes est équivalentes. Dans cette partie, on va détailler les étapes de conception de notre approche qu'on a vu dans la figure 4.1. Commençons par un graphe et se terminant par des communautés qui peuvent être chevauchantes ou disjointe, notre approche est non supervisée.

On considère un graphe  $G(V, E)$ , où  $V$  est l'ensemble des sommets et  $E \subseteq V \times V$  est l'ensemble des arêtes. On notera  $N$  le nombre de sommets de  $G$ , avec  $N = |V|$  et le nombre de voisinage du nœud  $N_i$  est  $V_i$ .

Notre algorithme comporte plusieurs étapes qui seront soigneusement expliqué dans ce qui suit.

### 5.1 La première phase

Cette phase consiste à initialiser les classes pour chaque nœud du réseau. Pour cela nous avons affecté chaque nœud à une classe avec tous ses voisins.

Le nombre de classe est  $N$  (nombre des nœuds), chaque classe est nommée par le nom de son nœud principal ajoutant le  $C$ . Cette phase consiste à appliquer deux étapes sur les nœuds : l'initialisation et l'arrangement.

#### 5.1.1 Initialisation

Dans cette première partie, on va créer des classes initiales qui sont  $N$  classes chevauchantes contenant  $[(2 * E) + N]$  nœuds, chaque classe contiens un nœud avec tous ces voisins.

- ✚ Le nombre total des nœuds dans tous les classes est le nombre des nœuds  $N$  plus le nombre des liens fois deux.
- ✚ Le nombre total des classes initial est le nombre des nœuds  $N$

#### 5.1.2 Arrangement

Dans cette étape la modification ne se fait pas sur les nœuds et les classes mais sur l'ordre des classes parce qu'il est très important et influe directement sur les résultats comme dans le cas de Louvain et de plusieurs autres algorithmes de détection des communautés, c'est pourquoi

on a les classé avec un ordre bien précise selon les métriques d'analyse des réseau sociaux qu'on à étudier aussi dans ce travail, commençant l'ordre par le degré de centralité, en cas d'égalité, on fait l'ordre en fonction de la betweenness-centralité qui est utilisé dans l'algorithme de Girven et Newman, et si on tombe dans le cas d'égalité, on effectue l'ordre en fonction de la closeness-centralité.

## **5.2 La deuxième phase**

Cette phase consiste à regrouper les classes selon leurs homogénéités, elle consiste à appliquer trois étapes itératifs sur les nœuds : regroupement des classes identiques, regroupement des classes inclus et regroupement des classes où la différence est un seul nœud.

### **5.2.1 Regroupement 01**

Dans cette partie on va regrouper les classes identique c'est-à-dire les classes où quelque soit  $N_i$  appartient à  $C_j$   $N_i$  appartient aussi à  $C_k$ , le regroupement se fait deux par deux selon une boucle qui suivre l'ordre qu'on à fait la phase d'arrangement, l'un des deux classes sera supprimer et l'autre prends les nœuds avec la répétition.

### **5.2.2 Regroupement 02**

Cette phase consiste à regrouper les classes incluses à 100% dans une autre classe, ajoutant les nœuds qui existe dans la classe inclus à la classe conteneur et supprimant la classe inclus.

### **5.2.3 Regroupement 03**

Cette phase consiste à regrouper les classes ayant le nombre de nœuds  $N$  et  $M$  (sans redondance) où l'intersection de ces deux partitions est  $N-1$  ou  $M-1$ , en ajoutant les nœuds qui existe dans la petite classe à la grande classe et supprimant la petite classe.

Répétez cette phase jusqu'à ce qu'aucun regroupement ou suppression des classes ne soit possible.

Le résultat de cette phase est un nombre des communautés où chaque communauté contient des nœuds répétant plusieurs fois.

## **5.3 La troisième phase**

C'est la phase final qui faite l'organisation pour obtenir les communautés qui peut être chevauchants ou disjoints; elle consiste à appliquer deux étapes sur les nœuds: Construire la



table de probabilité d'appartenir d'un nœud à une communauté et extraction des communautés disjointes et chevauchantes.

### 5.3.1 La table de probabilité

Cette étape consiste à faire un tableau des nœuds avec leurs redondances dans chaque communauté pour calculer la probabilité d'appartenir d'un nœud à une communauté.

### 5.3.2 Les communautés.

Avec la table de probabilité de l'étape précédente, on peut décider chaque nœuds appartiens à quelle communauté, dans le cas d'égalité entre les probabilités, les nœuds est appartiens à les deux communautés, on parle de chevauchement.

## 6 L'algorithme de notre approche

*Entrées* :  $G = (V, E)$  : le graphe initial

*Sorties* :  $C_1, C_2, C_3, \dots, C_k$  : des communautés.

### Fonctions :

Créer (C) : une fonction qui sert à créer des communautés.

Ajouter (C) : une fonction qui sert à ajouter un nœud à une communauté.

Supprimer (C) : une fonction qui supprimer des communautés

Regrouper (Ci, Cj) : une fonction qui regrouper deux communautés dans la première.

Inclus (Ci, Cj) : expression logique qui prend la valeur vrai si tous les nœuds de Ci appartiens à Cj.

Inclu(Ci,Cj): expression logique qui prend la valeur vrai si seulement un nœud de Ci n'appartiens pas à Cj.

### 6.1 L'algorithme de la première phase

Début :

```

Pour chaque nœud Ni faire
  Créer (Ci, Ni)
Fin pour
Pour chaque lien Nij faire
  Ajouter (Ci, Nj)

```

Ajouter (Cj, Ni)  
 Fin pour  
 nbC=N  
 Fin.

## 6.2 L'algorithme de la deuxième phase

Debut :

I=1

J=2

Tant que (i<=nbC) faire

Pour chaque Communauté Cj

Si (Ci ==Cj) alors :

Regrouper ( Ci,Cj)

Supprimer ( Cj)

nbC= nbC-1

Fin si

Fin pour

I=i+1

Fin tant que

I=1

J=2

Tant que (i<=nbC) faire

Pour chaque Communauté Cj

Si Inclus (Ci, Cj) alors :

Regrouper ( Ci,Cj)

Supprimer ( Cj)

nbC= nbC-1

Fin si

Fin pour

I=i+1

Fin tant que

I=1

J=2

Tant que (i<=nbC) faire

Pour chaque Communauté  $C_j$

Si  $\text{Inclu}(C_i, C_j)$  alors :

Regrouper ( $C_i, C_j$ )

Supprimer ( $C_j$ )

$\text{nbC} = \text{nbC} - 1$

Fin si

Fin pour

$I = i + 1$

Fin tant que

Fin

### 6.3 L'algorithme de la troisième phase

#### Fonctions :

$\text{REd}(N_i, C_i)$  : une fonction qui calcule le nombre de répétition de nœud dans une communauté.

$\text{Prob}(N_i, C_i)$  : une fonction qui calcule la probabilité d'appartenir d'un nœud à une communauté.

$\text{compar}(C_i, C_j)$  : une fonction qui compare les Probabilités d'une nœuds dans les communautés et retourne  $\text{COM}[]$  qui est une liste des communautés qui ont la valeur la plus grande.

$\text{Supprim}(N_i)$  : une fonction qui supprimer un nœud dans une communauté.

Debut :

Pour chaque Communauté  $C_i$

Pour chaque nœud  $N_i$

$\text{Red}(N_i, C_j)$

$\text{Prob}(N_i, C_j)$

Fin pour

Fin pour

Pour chaque nœud  $N_i$

$\text{Compar}(C_1, C_2, \dots, C_{\text{nbC}}, N_i)$ .

Si ( $C_k < > \text{com}[]$ ) alors

```

    Suprim (Ni)
  Fin si
Fin pour
Pour chaque Communauté Ci
  Si vide(Ci) alors
    Supprimer ( Nj)
  Fin Si
Fin pour
Return Communautés.
Fin.

```

## 7 Exemples illustratifs

On va utiliser deux exemples l'un donne des communautés disjointes et l'autre donne des communautés chevauchantes pour montrer l'efficacité de notre approche dans les deux cas.

### 7.1 Exemple 01

Notre premier exemple est le graphe de Réseau d'interaction des primates, Cet ensemble de données contient des interactions enregistrées au sein d'un groupe de primates, dans lesquels un bord existe entre deux primates s'ils ont été vus ensemble sur le même fleuve depuis un nombre suffisant de fois.

Ce graphe est constitué de vingt nœuds et dix-neuf liens et il est connexe et non orienté.

Tous les schémas qu'on va l'utiliser au cours de ces exemples sont de notre application.

#### 7.1.1 La première phase

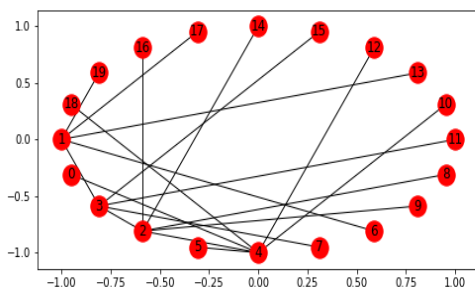
Comme déjà dit initialement, chaque sommet est affecté à une classe avec tous ses voisins, par ordre quelconque, dans notre cas le nombre de communauté est  $n=20$ .

Dans le tableau suivant on va représenter les communautés initiaux et l'ordre des communautés après l'arrangement qui fait d'après l'analyse de réseaux qu'on a fait et qui sont représentés avec les figures de notre système.

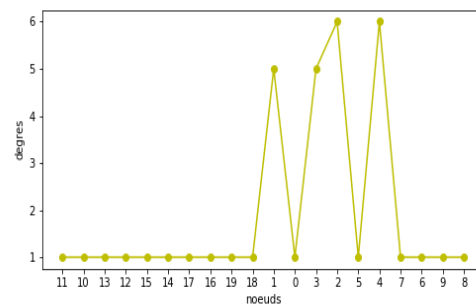
Initialisation		Arrangement	
Les nœuds Ni	Les communautés Ci	Ni avec ordre	Les communautés
0	0 4	2	2 3 4 8 9 14 16
1	1 3 6 13 17 19	4	4 2 0 5 10 12 18
2	2 3 4 8 9 14 16	3	3 2 1 7 11 15

3	3 2 1 7 11 15	1	1 3 6 13 17 19
4	4 2 0 5 10 12 18	11	11 3
5	5 4	12	12 4
6	6 1	10	10 4
7	7 3	7	7 3
8	8 2	16	16 2
9	9 2	8	8 2
10	10 4	9	9 2
11	11 3	14	14 2
12	12 4	15	15 3
13	13 1	0	0 4
14	14 2	5	5 4
15	15 3	18	18 4
16	16 2	13	13 1
17	17 1	6	6 1
18	18 4	17	17 1
19	19 1	19	19 1

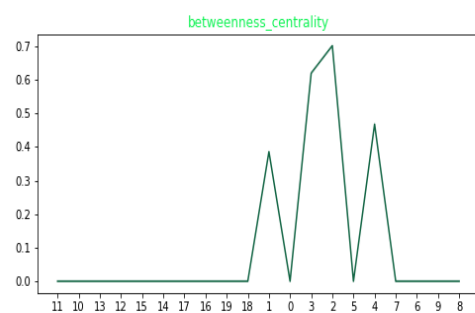
Tableau 4.1 les communautés initiaux et leur arrangement.



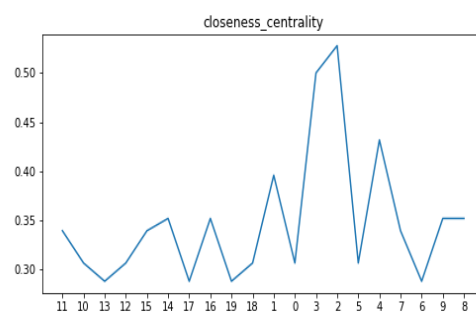
La représentation graphique du réseau



Le degré de chaque noeud



betweenness centrality de chaque noeud



closeness centrality de chaque noeud

Figure 4.2 Les métriques utilisées pour l'arrangement.

### 7.1.2 La deuxième phase

#### Regroupement 01

Il y a aucun changement dans cette étape car il n'existe aucunes deux communautés identiques.

**Résultats :**

C2= 2 3 4 8 9 14 16  
 C4= 4 2 0 5 10 12 18  
 C3= 3 2 1 7 11 15  
 C1= 1 3 6 13 17 19  
 C11=11 3  
 C12= 12 4  
 C10= 10 4  
 C7= 7 3  
 C16= 16 2  
 C8= 8 2  
 C9= 9 2  
 C14= 14 2  
 C15= 15 3  
 C0= 0 4  
 C5= 5 4  
 C18= 18 4  
 C13= 13 1  
 C6= 6 1  
 C17= 17 1  
 C19= 19 1

**Regroupement 02**

Il ya plusieurs changement dans cette étape car il ya beaucoup des communautés inclus dans d'autres exemple la communauté C12 inclus dans la communauté C4.

**Résultat :**

C1= 1 1 1 1 3 6 6 13 13 17 17 19 19  
 C2= 2 2 2 2 3 4 8 8 9 9 14 14 16 16  
 C4= 0 0 2 4 4 4 4 4 5 5 10 10 12 12 18 18  
 C3= 1 2 3 3 3 3 7 7 11 11 15 15

**Regroupement 03**

Il y a aucun changement dans cette étape car il n'existe aucunes deux communautés où l'intersection est seulement un nœud.

**7.1.3 La troisième phase**

Ni	Redondances	Prob C1	Prob C2	Prob C3	Prob C4	Décision
0	2	0/2	0/2	0/2	2/2	C4

1	6	5/6	0/6	1/6	0/6	C1
2	6	5/6	0/6	1/6	0/6	C1
3	6	1/6	1/6	4/6	0/6	C3
4	6	0/6	1/6	0/6	5/6	C4
5	2	0/2	0/2	0/2	2/2	C4
6	2	2/2	0/2	0/2	0/2	C1
7	2	0/2	0/2	2/2	0/2	C3
8	2	0/2	2/2	0/2	0/2	C2
9	2	0/2	2/2	0/2	0/2	C2
10	2	0/2	0/2	0/2	2/2	C4
11	2	0/2	0/2	2/2	0/2	C3
12	2	0/2	0/2	0/2	2/2	C4
13	2	2/2	0/2	0/2	0/2	C1
14	2	0/2	2/2	0/2	0/2	C2
15	2	0/2	0/2	2/2	0/2	C3
16	2	0/2	2/2	0/2	0/2	C2
17	2	2/2	0/2	0/2	0/2	C1
18	2	0/2	0/2	0/2	2/2	C4
19	2	2/2	0/2	0/2	0/2	C1

Tableau 4.2 Table de probabilité avec décision.

Dans ce graphe nous n'avons pas trouvé des chevauchements

Donc il existe quatre communautés disjointes C1, C2, C3, C4

C1= 1 6 13 17 19

C2= 2 8 9 14 16

C4= 0 4 5 10 12 18

C3= 3 7 11 15

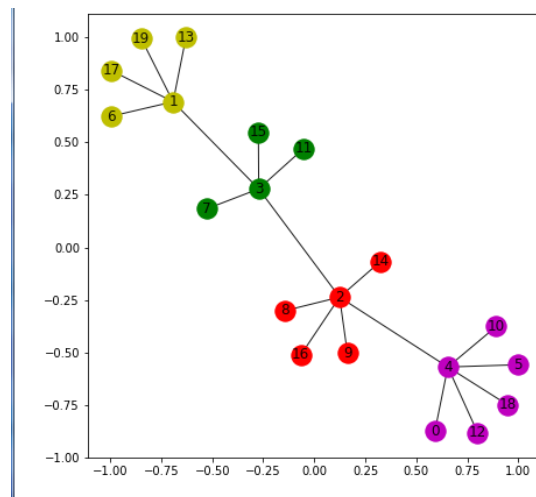


Figure 4.3 Résultat de l'exemple 1.

## 7.2 Exemple 02

Notre deuxième exemple est un graphe artificiel constitué de neuf nœuds et quatorze liens et il est connexe et non orienté.

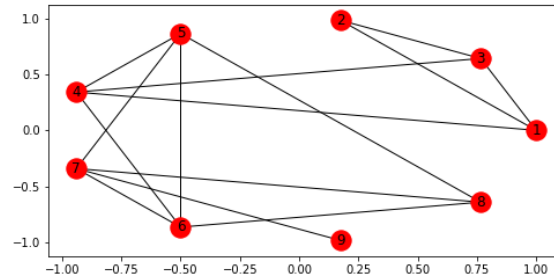


Figure 4.4 Graphe de l'exemple 2.

### 7.2.1 La première phase

Dans le tableau suivant on va résumer les résultats de la première phase. Le tableau suivant nous résume les résultats de l'arrangement des classes

TABLEAU 4.1 Résultat de l'arrangement des classes de l'exemple

Initialisation	Degré	betweness	Closeness
C1 = [1, 2, 3, 4]	C4 = [1, 3, 4, 5, 6]	C4 = [1, 3, 4, 5, 6]	C4 = [1, 3, 4, 5, 6]
C2 = [1, 2, 3]	C5 = [4, 5, 6, 7, 8]	C7 = [5, 6, 7, 8,9]	C7 = [5, 6, 7, 8,9]
C3 = [1, 2, 3, 4]	C6 = [4, 5, 6, 7, 8]	C6 = [4, 5, 6, 7, 8]	C6 = [4, 5, 6, 7, 8]
C4 = [1, 3, 4, 5, 6]	C7 = [5, 6, 7, 8,9]	C5 = [4, 5, 6, 7, 8]	C5 = [4, 5, 6, 7, 8]
C5 = [4, 5, 6, 7, 8]	C1 = [1, 2, 3, 4]	C1 = [1, 2, 3, 4]	C1 = [1, 2, 3, 4]
C6 = [4, 5, 6, 7, 8]	C3 = [1, 2, 3, 4]	C3 = [1, 2, 3, 4]	C3 = [1, 2, 3, 4]
C7 = [5, 6, 7, 8,9]	C8 = [5, 6, 7, 8]	C8 = [5, 6, 7, 8]	C8 = [5, 6, 7, 8]
C8 = [5, 6, 7, 8]	C2 = [1, 2, 3]	C2 = [1, 2, 3]	C2 = [1, 2, 3]
C9 = [7, 9]	C9 = [7, 9]	C9 = [7, 9]	C9 = [7, 9]

Tableau 4.3 Résultat d'arrangement.

### 7.2.2 La deuxième phase

#### Regroupement 01

Il ya quelques changement dans cette étape, exemple regroupement des deux communautés C5 et C6.

**Résultats :**



C4 = [1, 3, 4, 5, 6]

C7 = [5, 6, 7, 8,9]

C6 = [4, 5, 6, 7, 8,4, 5, 6, 7, 8]

C1 = [1, 2, 3, 4,1, 2, 3, 4]

C8 = [5, 6, 7, 8]

C2 = [1, 2, 3]

C9 = [7, 9]

### Regroupement 02

Il ya des changements dans cette étape exemple la communauté C2 inclus dans la communauté C1.

#### Résultat :

C4 = [1, 3, 4, 5, 6]

C7 = [5,5, 6,6, 7, 7, 7, 8, 8, 9, 9]

C6 = [4, 4,5, 5,6, 6, 7, 7,8, 8]

C1 = [1, 1, 1, 2, 2,2, 3, 3,3, 4, 4]

### Regroupement 03

Il y a des changements dans cette étape car il existe des communautés où l'intersection est manque seulement un nœud, exemple regroupement des la communauté C1 et C4.

#### Résultat :

C4 = [1, 1, 1, 1, 2, 2,2, 3, 3,3, 3, 4, 4, 5,6]

C7 = [4, 4,5, 5, 5, 5, 6, 6, 6, 6, 7, 7, 7, 7, 7, 8, 8, 8, 8, 9, 9]

### 7.2.3 La troisième phase

Ni	Redondances	Prob C4	Prob C7	Décision
1	4	4/4	0/4	C4
2	3	3/3	0/3	C4
3	4	4/4	0/4	C4
4	4	2/4	2/4	C4, C7
5	5	1/5	4/5	C7
6	5	1/5	4/5	C7
7	5	0/5	0/2	C7
8	4	0/4	4/4	C7
9	2	0/2	2/2	C7

Tableau 4.4 Table de probabilité avec décision.

Dans ce graphe nous avons trouvé un chevauchement où le nœud « 4 » appartient au deux communautés C4 et C7.

Donc il existe deux communautés chevauchantes C4 et C7.

C4= 1, 2, 3, 4

C3= 4, 5, 6, 7, 8, 9

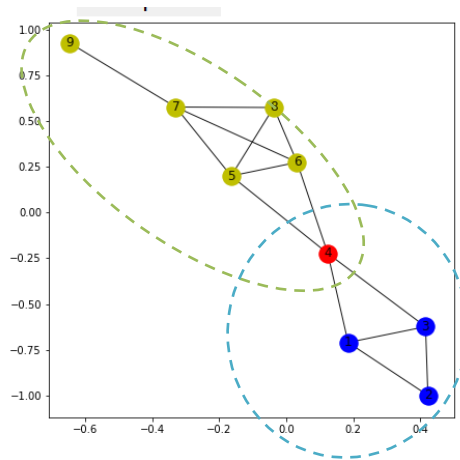


Figure 4.5 Résultat de l'exemple 2.

## 8 Conclusion

Dans ce chapitre, nous avons présenté une nouvelle méthode non supervisée de détection de communautés chevauchantes dans les réseaux sociaux avec deux exemples illustratifs détaillés où nous avons montré que notre méthode est capable de détecter des recouvrements facilement, aussi les communautés trouvées par notre méthode restant stables dans plusieurs exécutions sur le même réseau à cause de l'arrangement qu'on a fait.

Dans ce qui suit on va implémenter les principales métriques d'analyse des réseaux sociaux et les principaux algorithmes de détection de communautés ainsi que quelques exemples de notre approche.

## Chapitre 05 : Implémentation du système

### 1 Introduction

Dans cette partie, nous allons montrer l'implémentation de notre système qui a concrétisé l'étape de la conception. Nous présentons, en premier lieu, les environnements de développement matériel et logiciel et les outils utilisés pour implémenter notre application.

.Ensuite, nous illustrons l'utilisation de notre système à travers une présentation de quelques réseaux sociaux réalisées et une expérimentation de détection des communautés.

### 2 Environnement de développement

Pour développer n'importe quelle application, nous avons besoin d'un environnement matériel et un environnement logiciel.

#### 2.1 Environnement matériel

Notre système a été développé sur une machine dont la configuration est la suivante:

Matériel	Caractéristiques
PC	<p><b>Processeur</b> : AMD E2-9000e RADEON R2, 4 COMPUTE CORES 2C+2G 1.50 GHz</p> <p>Mémoire Vive (Ram) : 4 Go</p> <p><b>Disque Dur</b> : 500 Go.</p>

Tableau 5.1 Caractéristique du matériel.


#### 2.2 Environnement logiciel

Notre application a été développée sous un système d'exploitation : Windows 7 édition intégrale. Win32, où nous avons installé l'Anaconda sous Python 2.7.

### 3 Langages de programmation

On a utilisé principalement le langage Python 2.7 pour le codage de notre système.

Notre système à besoin de manipuler des graphes donc on à besoin de plusieurs bibliothèques, l'installation du langage de programmation avec la plateforme et les packages qu'on à besoin n'est pas une tâche facile, on a perdu beaucoup de temps lors de l'installation correcte de tous les packages.

 Python est un langage de programmation interprété de haut niveau, multi-paradigme et multiplateformes. Il favorise la programmation impérative structurée, fonctionnelle et orientée objet, impérative et styles procéduraux. Il possède une grande bibliothèque standard complète, créé par Guido van Rossum et sa première publication en 1991, il intègre un système de types dynamiques et une gestion automatique de la mémoire, il est également apprécié par certains pédagogues qui y trouvent un langage où la syntaxe, clairement séparée des mécanismes de bas niveau, permet une initiation aisée aux concepts de base de la programmation. (Peirce, 2007)


**Installation :** python est installé gratuitement dans le site : <https://www.python.org/downloads/>



Figure 5.1 Le site d'installation de python.

L'installation du python est juste la première étape mais le plus important et le plus difficile c'est l'installation de la plateforme avec les packages.

Au début on a installé **PyCharm** mais ça marche pas avec les graphes après des recherches on a changé à **Anaconda**.

 **ANACONDA** Est une distribution libre et open source et gratuit de Continuum Analytics (www.continuum.io), est une pile scientifique pratiquement complète pour Python.

Il comprend à la fois l'interpréteur Python principal et les bibliothèques standards, ainsi que la plupart des modules nécessaires à l'analyse de données. (Sheppard, 2012)

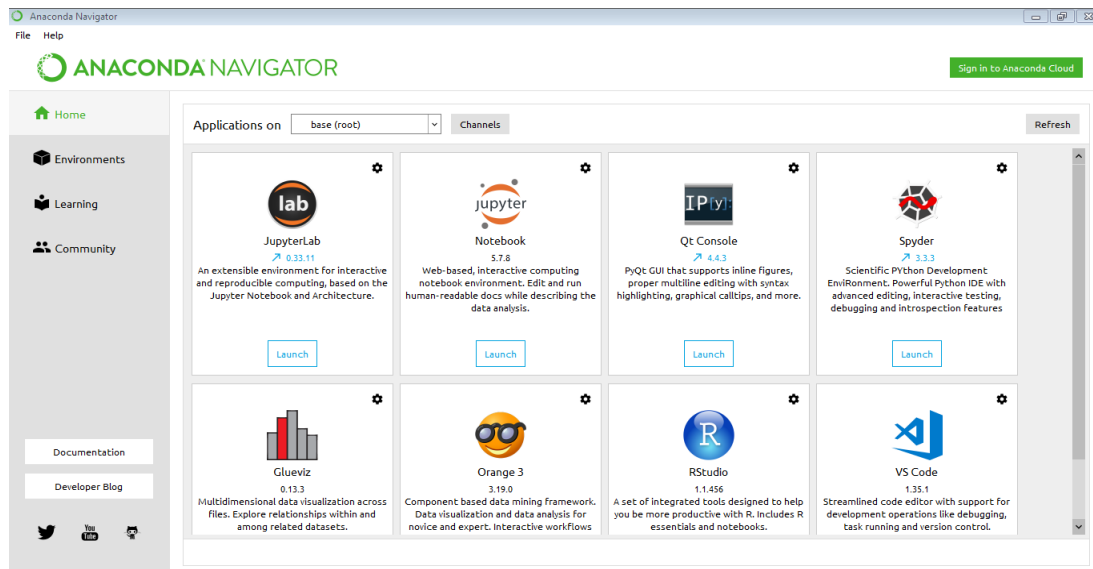


Figure 5.2 Interface d'anaconda.



C'est un IDE multiplateforme open source pour la science des données, spécialisé dans les applications scientifiques de Python plutôt que dans le développement d'applications à usage général, Spyder est presque certainement l'IDE à utiliser lors du démarrage de Python, et il est toujours relativement simple de migrer vers un IDE sophistiqué si nécessaire (Sheppard, 2012)

**NetworkX** C'est un logiciel en langage python permet de créer, manipuler et étudier la structure de fonctions de réseaux dynamiques et complexes, il est nécessaire pour l'analyse et la visualisation des réseaux sociaux et plusieurs autres caractéristiques tels que la possibilité de multi-graphes, analyse de réseau, Mesure productive pour les graphes classiques - graphes aléatoires et réseaux artificiels. (Jokar, 2016)

### Les bibliothèques :

Pour implémenter notre système on a besoin de quelques bibliothèques, et pour les installer il faut écrire (`conda-install-c anaconda le_nom_de_la_bibliothèque`), on trouve toutes les bibliothèques existantes dans le site suivant : <https://anaconda.org>

Les packages qu'on a téléchargé sont les suivant:

**Networkx** : conda install -c anaconda networkx.

**Pycairo** : conda install -c anaconda pycairo.

**Python-louvain** conda install -c anaconda python-louvain.

**Pyqt5** : conda install -c inso pyqt4.

**Numpy** : conda install -c anaconda Numpy.

**Pandas** : conda install -c anaconda Pandas.

L'installation peut prendre beaucoup de temps surtout dans le cas où la connexion est lourde.

#### **4 Architecture de notre application**

Notre application se compose de trois parties principales : les métriques d'analyse des réseaux sociaux, les algorithmes de détection des communautés les plus connus et l'implémentation de notre algorithme.

L'architecture de notre application est présentée ci-dessous au niveau de la figure 4.1.

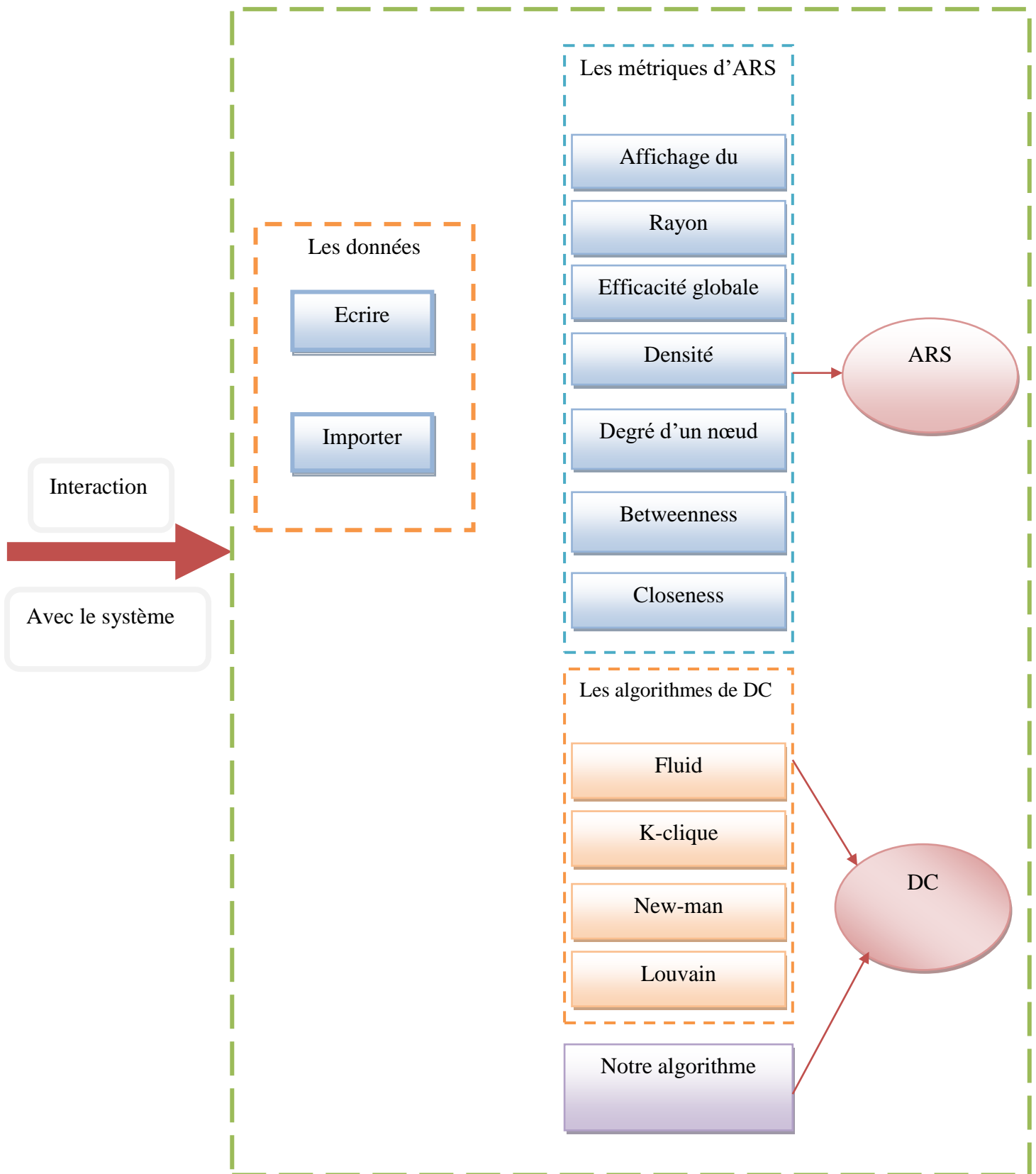


Figure 5.3 Architecture générale du système.

## 5 Fonctionnalités du système

On a implémenté quelques métriques d'analyse des réseaux sociaux (partie d'analyse des réseaux sociaux) et quelque algorithme de détection des communautés ainsi que l'implémentation de notre algorithme (partie de détection de communautés)

L'utilisateur peut choisir entre les métriques d'ARS ou bien les algorithmes de détection des communautés.

L'utilisateur peut soit importer ses données en cliquant sur le bouton d'importation des données ou bien les écrire manuellement.

Il peut sauvegarder les données qu'il écrit.

Les données doivent être sous forme d'un fichier texte.

En ce qui concerne l'analyse des données l'utilisateur peut choisir entre l'affichage du graphe, le rayon, le diamètre, la densité, le degré de chaque nœud, l'efficacité globale, betweenness centrality ou bien closeness centrality.

En ce qui concerne la détection des communautés l'utilisateur peut choisir entre l'algorithme de Fluid, k-clique, new-man, Louvain ou bien l'exécution de notre approche.

En haut de la page l'utilisateur trouvera un bouton fichier qui contient les fonctionnalités suivante (Nouveau, Ouvrir, Enregistrez- sous, quitter) et un autre bouton Edit qui contient Édition contient (Copier, Coller, Couper).

## 6 Présentation du système

Dans cette section, nous allons présenter quelques modules de notre système, il est divisé en trois modules principaux : module pour quelques métriques d'ARS, module pour quelques algorithmes de détection de communautés, et le dernier module pour l'implémentation de notre algorithme sur quelques réseaux sociaux.

La figure 5.4 suivante nous montre l'interface principale des métriques d'ARS et des algorithmes de détection de communauté avec un espace pour l'écriture ou bien l'importation de l'ensemble des données qu'on veut traiter.





Figure 5.4 Les métriques d’ARS et des algorithmes de détection de communautés.

La figure 5.5 nous montre le choix des métriques d’ARS et des algorithmes de détection des communautés avec un ensemble de données d’un réseau social.



Figure 5.5 La saisie des données d’un réseau social.

Après l'introduction et l'exécution des données, les résultats des métriques (le diamètre de graphe et le rayon, l'efficacité globale, la densité, le degré, betweenness centrality, et la closeness centrality, avec une représentation graphique du graph) vont apparaître au niveau de la figure 5.6.

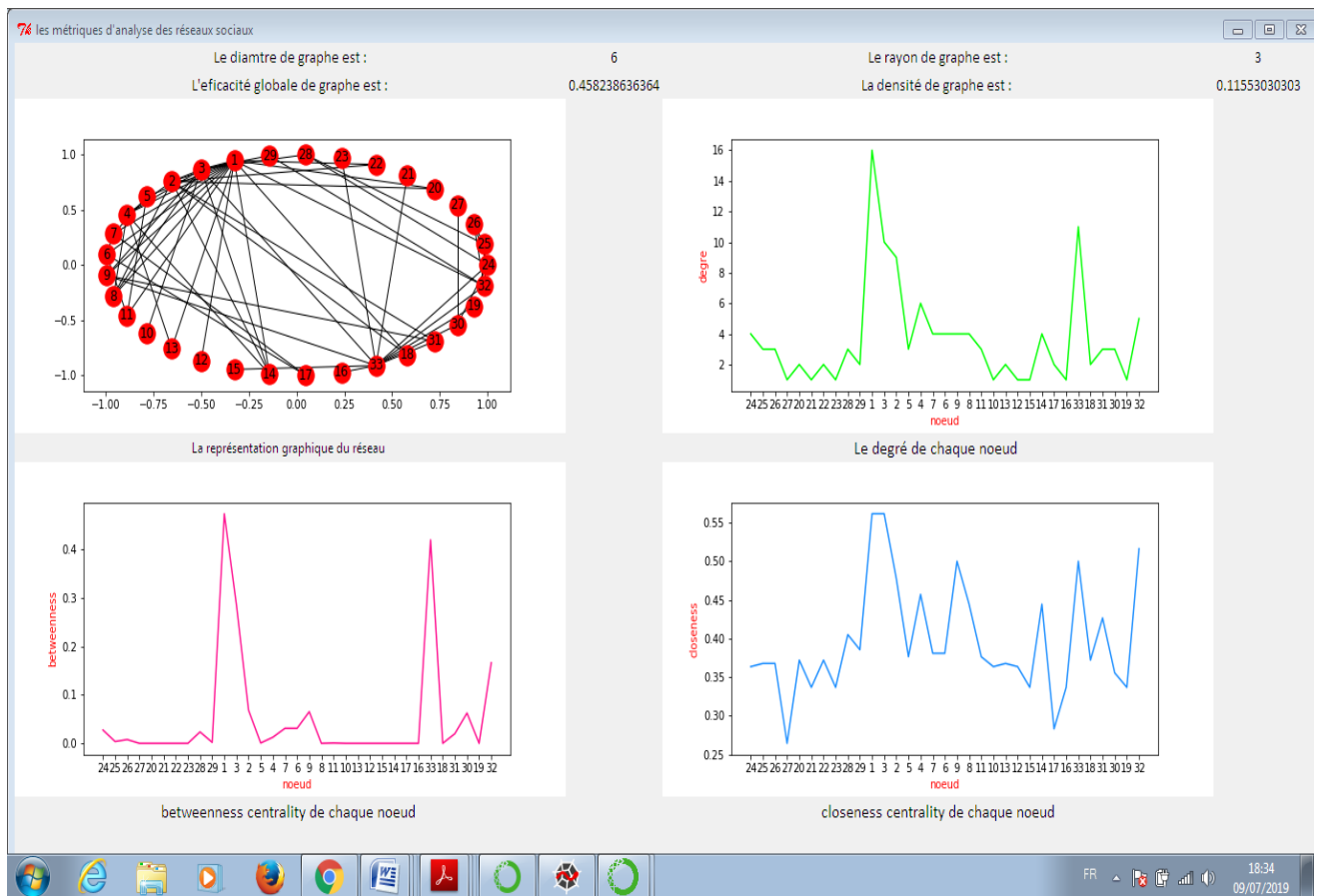


Figure 5.6 Les résultats obtenus après l'analyse des données.

Les résultats des algorithmes de détection de communauté (Fluid, K-clique, New-man, Louvain) après l'introduction et l'exécution des données d'un réseau social vont apparaître au niveau de la figure 5.7.

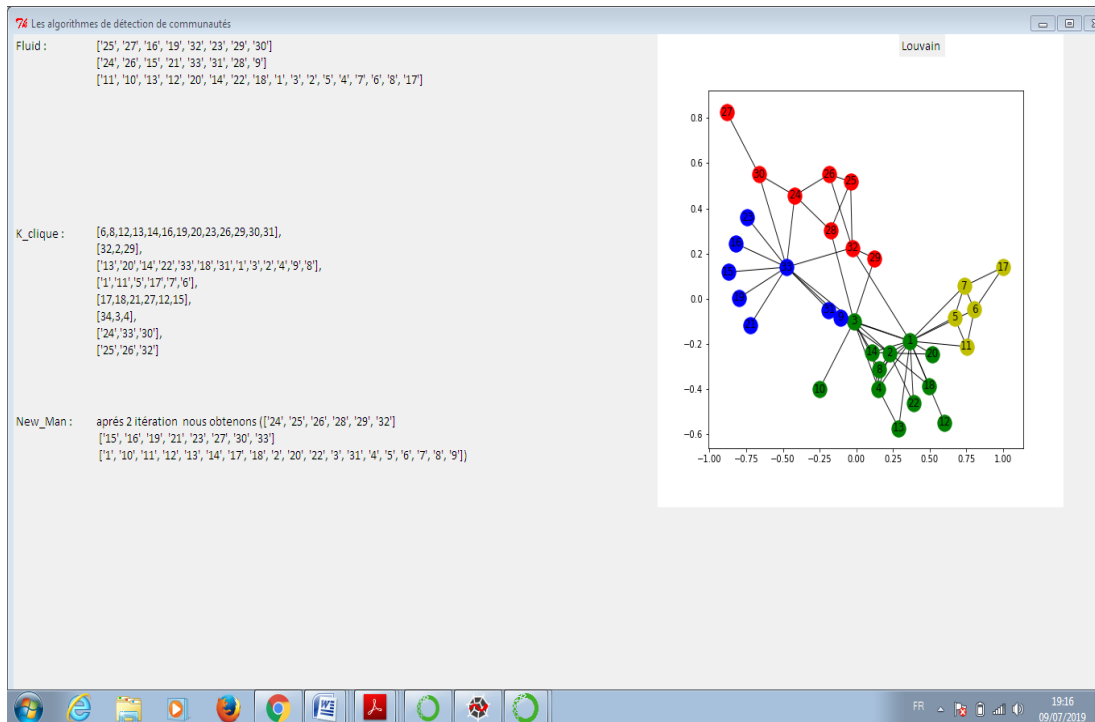


Figure 5.7 Les résultats de l'exécution des algorithmes de détection des communautés.

Le dernier imprimé écran FIGURE 5.8 nous montre l'exécution des données d'un réseau social avec notre approche

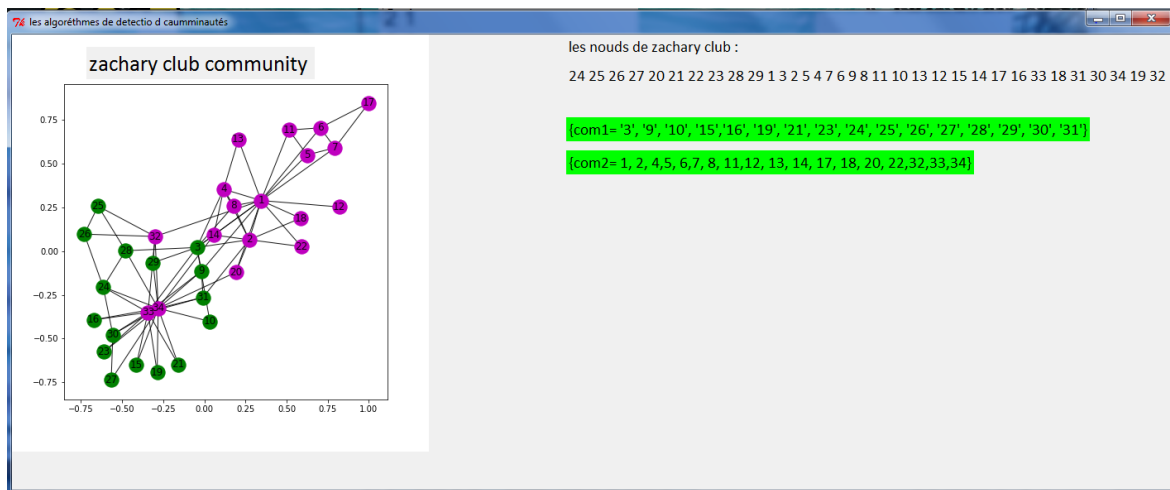


Figure 5.8 Résultat d'exécution de notre algorithme.

## 7 Résultats et discussions

Notre méthode est montrée leur efficacité dans tous les graphes qu'on a testés soit dans le cas de chevauchement ou bien dans les communautés disjointes.

Le tableau suivant fait une petite comparaison entre les résultats de notre approche avec les approches qu'on à implémenter prenant comme exemple le graphe de Réseau d'interaction des primates

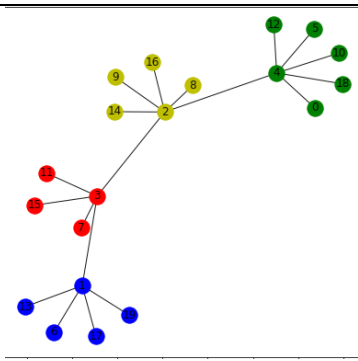
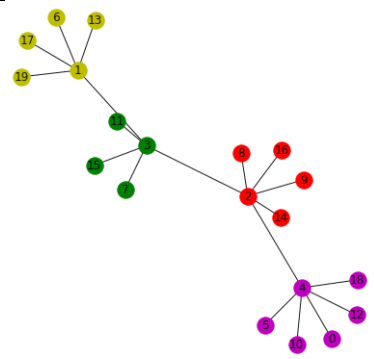
L'approche	Les communautés	N° de C
k-clique	<pre> i['11', '15', '14', '16', '3', '2', '7', '9', '8'] ['19', '1', '13', '17', '6'] ['10', '12', '18', '0', '5', '4']                     </pre>	3
Louvain		4
Newman et girvan	<p>après 2 itération nous obtenons (['1', '11', '13', '15', '17', '19', '3', '6', '7']                      ['0', '10', '12', '18', '4', '5']                      ['14', '16', '2', '8', '9'])</p>	3
Fluide de communautés	<pre> i['11', '15', '14', '16', '3', '2', '7', '9', '8'] ['19', '1', '13', '17', '6'] ['10', '12', '18', '0', '5', '4']                     </pre>	3
Notre approche		4

Tableau 5.2 Résultats de l'exécution des algorithmes sur le réseau des primates

### 8 Les avantages et les limite de notre méthodes

Notre méthode est simple et facile à comprendre, elle est capable de détecter les recouvrements facilement ce qui est très important dans notre domaine qui est les réseaux sociaux.

Il n'y a aucune valeur du paramètre requis à fixer, ce qui nous a évités de tomber dans le problème de la recherche du meilleur choix de ces paramètres, elle base seulement sur les voisins du nœud.

Les communautés trouvées par notre méthode restant stables dans plusieurs exécutions sur le même réseau à cause de l'arrangement qu'on a fait.

La mise en œuvre de la méthode proposée est très facile, donc on peut l'essayé avec les réseaux dynamique.

Notre algorithme est facile à implémenter.

Mais on à pas encore tester sur des grands réseaux.

## **9 Conclusion**

Dans ce chapitre, nous avons tout d'abord présenté les outils utilisés lors de l'implémentation de notre système, et exposé les résultats de notre système mettant l'accent sur leurs avantages et leurs inconvénients.

## Conclusion et perspectives

Ce mémoire avait comme objectif la détection de communautés dans les réseaux sociaux. Pour cela, nous intéressants dans l'état de l'art aux réseaux sociaux, les théories des graphes et les méthodes de détection de communautés qui existe dans la littérature.

Nous avons présenté une approche de détection de communautés chevauchantes dans les réseaux sociaux. En effet, à travers la méthode proposée les groupes de nœuds fortement liés entre eux, et faiblement liés avec le reste des réseaux sont identifiés. Notre approche permet le chevauchement entre les communautés, L'algorithme proposé est basé sur le voisinage et accepte le recouvrement donc chaque nœud peut appartenir à une ou plusieurs communautés avec une probabilité d'inclusion à cette communauté de 0 à 100%. Selon une valeur de probabilité qu'on a calculé.

Nous avons implémenté et testé la méthode proposée sur plusieurs réseaux sociaux utilisés comme le réseau d'interaction des primates, Zachary club et d'autres réseaux réelles et artificielles et les résultats sont très acceptables

Aussi notre méthode est simple et facile à comprendre, elle est capable de détecter les recouvrements facilement, il n'y a aucune valeur du paramètre requis à fixer, les communautés trouvées par notre méthode restant stables dans plusieurs exécutions sur le même réseau et la mise en œuvre de la méthode proposée est très facile.

Notre travail offre plusieurs perspectives :

La première, découle du fait que la méthode proposée traite les graphes non-orientés, nous souhaitons étendre cette approche pour appliquer sur les réseaux bayésiens qui sont des graphes orientés.

La deuxième perspective est liée au Base de données, on veut appliquer notre algorithme sur des BDD avec nombre important des nœuds.

Et enfin, on veut développer notre algorithme pour détecter aussi les communautés dans les réseaux dynamiques.

**Bibliographie**

Aaron Clauset, M. E. (2004). Finding community structure in very large networks. *Physical review E* .

abdelhamid, n. m. (2015). fouille et apprentissage automatique dans les réseaux sociaux dynamique. Doctoral dissertation, Université Mohamed Khider-Biskra.

Attal, J. P. (2016). Détection de communautés chevauchantes utilisant la propagation de labels robuste et fonction d'appartenance. *hal.archives-ouvertes.fr* .

Barbié, O. (2010). *Convergences entre économie et sociologie autour du concept de réseau social*. Lulu. com.

Cazabet, R. (2013). Détection de communautés dynamiques dans des réseaux temporels. Université Paul Sabatier-Toulouse III.

Crampes, M. &. (2013). Partition et recouvrement de communautés dans les graphes bipartis, unipartis et orientés.

David Lusseau, K. S. (2003). The bottlenose dolphin community. *Behavioral Ecology and Sociobiology*,.

Dongen., S. v. (2000). Graph clustering by flow simulation. Doctoral dissertation.

ELLISON, N. B. (2007). Social network sites : Definition, history, and scholarship. 210 - 230.

Ferran Parés, D. G.-G. (2017). Fluid Communities: A Competitive, Scalable and Diverse Community Detection Algorithm. *In International Conference on Complex Networks and their Applications* .

Filippo Radicchi, C. C. (2004). Defining and identifying communities in networks. *the national academy of sciences* .

Flament, C. (2017). *Théorie des graphes et structures sociales* . Walter de Gruyter GmbH & Co KG.

Forcé, M. (2008). définir et analyse les réseaux sociaux. *Information sociales* , 10 - 19.

Fortunato, S. (2010). Community detection in graphs. *Complex Networks and systems Lagrange Laboratory, ISI Foundation, Viale S. Severo* , 1013.

## Références Bibliographiques

G. Palla, A. B. (2005). Uncovering the overlapping community structure of complex networks in nature and society. 2007.

Gauthier, L. A. (2015). Inférence de liens signés dans les réseaux sociaux, par apprentissage à partir d'interactions utilisateur.

GLEISER, P. M. (2003). COMMUNITY STRUCTURE IN JAZZ.

Ioannis Psorakis, S. R. (2011). Overlapping community detection using bayesian non-negative matrix factorization. *Physical Review E* .

Jokar, N. H. (2016). The review of social networks analysis tools. Bulletin de la Société Royale des Sciences de Liège.

Julian McAuley., & J. (2012). Learning to Discover Social Circles in Ego Networks.

Kanawati, R. (2011). Licod: Leaders identification for community detection in complex networks. *IEEE Third International Conference on Privacy, Security, Risk and Trust and 2011 IEEE Third International Conference on Social Computing* .

Latapy, P. P. (2006). Computing communities in large networks using random walks.

M.Denny. (2014). Social Network Analysis.

Mark EJ Newman, M. G. (2004). Finding and evaluating community structure in networks. *Physical Review* , 026113.

Newman, M. E. (2004). Fast algorithm for detecting community structure in networks.

Newman, M. G. (2002). Community structure in social and biological networks. *Proceedings of the National Academy of Sciences* , 12 (99).

Nguyen, N. P. (2011). Overlapping Communities in Dynamic Networks: Their Detection and Mobile Applications. *Proceedings of the 17th annual international conference on Mobile computing and networking* .

Orman, G. K. (2013). Une méthode pour caractériser les communautés des réseaux dynamiques à attributs. *arXiv preprint arXiv* .



## Références Bibliographiques

Peirce, J. W. (2007). PsychoPy—psychophysics software in Python. *Journal of neuroscience methods*.

Radicchi, F. C. (2004). Defining and identifying communities in networks.

S. Bhagwat, A. (2013). Development of Social Networking Sites and Their Role in Business with Special Reference to Facebook.

S. Wasserman, K. F. (1995). *Social Network Analysis: Methods and Applications*.

Santo Fortunato, V. L. (2004). Method to find community structures based on information centrality. *Physical review E* .

Schaeffer, S. E. (2007). Graph clustering. *Computer science review* .

Scott, J. (1988). Social Network Analysis *Sociology*. *journals.sagepub.com* .

Sheppard, K. (2012). Introduction to python for econometrics, statistics and data analysis. University of Oxford, version, 2.

Sohail, M. &. (2018). *A Graph Theory Based Method to Extract Social Structure in the Society*. Intelligent Technologies and Applications: First International Conference, .

Talbi, M. (2013). Une nouvelle approche de détection de communautés dans les réseaux sociaux. Doctoral dissertation, Université du Québec en Outaouais.

Usha Nandini Raghavan, R. A. (2007). Near linear time algorithm to detect community structures in large-scale networks. *Physical review E* .

Viennet, E. (2008). Recherche de communautés dans les grands réseaux sociaux. [pdfs.semanticscholar.org: AAFD](https://pdfs.semanticscholar.org/AAFD).

Vincent D Blondel, J.-L. G. (2008). Fast unfolding of communities in large Network. *Journal of statistical mechanics: theory and experiment* .

Yacine Slimani, A. D. (n.d.). Découverte de communautés dans les réseaux complexes. [hal.archives-ouvertes.fr](https://hal.archives-ouvertes.fr) .

Zachary. (1977). An information flow model for conflict and fission in small groups. *Journal of anthropological research* , 33 (4), 452-473.

## Références Bibliographiques