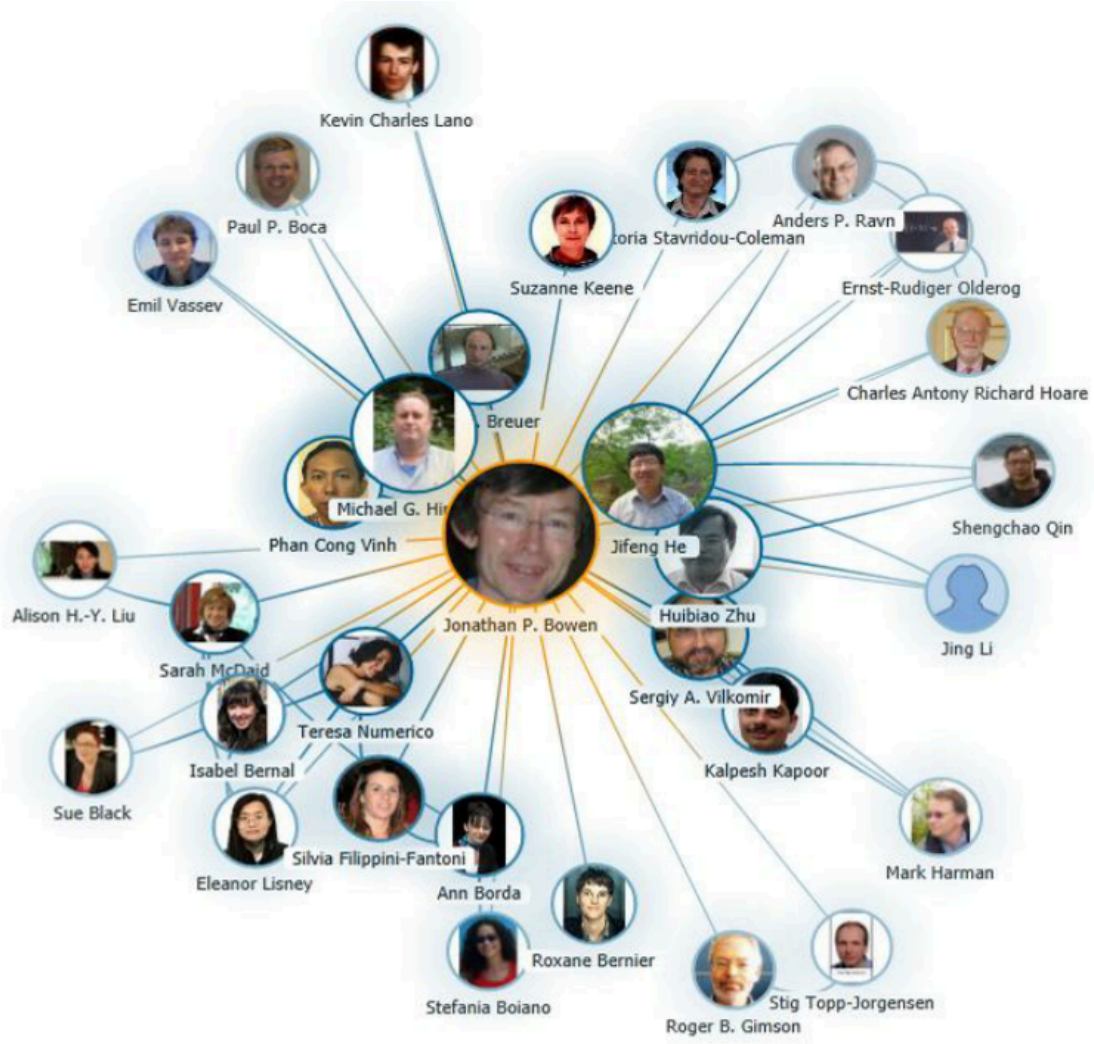
Projet
Analyse des Réseaux Sociaux

Coauthorships in network science



Hassouna Hiba

1ère année Mastère de rechercher Business computing(BC)

Partie 1 : Collecte des données

1. Identifier une source de données en ligne

réseau de coauteurs des scientifiques travaillant sur la théorie et l'expérimentation des réseaux, compilé par M. Newman en mai 2006. Ce réseau a été compilé à partir des bibliographies de deux articles de revue sur les réseaux : M. E. J. Newman, SIAM Review 45, 167-256 (2003) et S. Boccaletti et al., Physics Reports 424, 175-308 (2006), avec quelques références supplémentaires ajoutées manuellement.

lien : Network data (umich.edu)

2. Identifier les entités (noeuds) et les relations entre elles (liens)

le réseau non orienté avec 1589 sommets et 2742 arêtes. Chaque nœud représente un auteur ; les arêtes représentent les articles écrits conjointement par les auteurs. La valeur est le poids MEJ Newman. Il n'y a pas de boucles ni d'arêtes multiples incluses.

3. Identifier les informations additionnelles valables

Chaque sommet est représentée par (id:numéro de sommet,Label:Nom de l'auteur)

Le réseau est pondéré, avec des poids variables attribués comme décrit dans M.E.J. Newman, Phys. Rev. E 64, 016132 (2001). Les poids représentent la force de la collaboration scientifique.

calcul des poids

Dans l'article de M.E.J. Newman, Phys. Rev. E 64, 016132 (2001), les poids sont calculés en utilisant une méthode basée sur la théorie des graphes. Plus précisément, Newman propose une métrique appelée poids

MEJ (MEJ weight) qui est déterminée par la fréquence à laquelle deux auteurs apparaissent ensemble sur des articles scientifiques.

Le poids entre deux auteurs est calculé en utilisant la formule suivante :

$$w_{ij} = \sum_k \frac{n_{ik} \cdot n_{jk}}{(\sum_{k'} n_{ik'}) \cdot (\sum_{k'} n_{jk'})}$$

où w_{ij} est le poids entre les auteurs i et j, n_{ik} est le nombre d'articles auxquels l'auteur i a contribué avec l'auteur k, et n_{jk} est le nombre d'articles auxquels l'auteur j a contribué avec l'auteur k. Les termes $\sum_{k'} n_{ik'}$ et $\sum_{k'} n_{jk'}$ représentent respectivement le nombre total d'articles auxquels l'auteur i et l'auteur j ont contribué.

4. Obtenir les données à partir de la source de données

Le fichier netscience.gml est téléchargé à partir de lien : [Network data (umich.edu)]

Le fichier netscience.gml contient un réseau de coauteurs de scientifiques travaillant sur la théorie et l'expérimentation des réseaux

5. Construire un réseau à partir des données

★ Chargement des données :

```
#Charger le fichier dans un graphe
G = nx.read_gml('/content/netscience.gml', label="id", destringizer=int)
```

```
Informations sur le graphe :
Nombre de nœuds : 1589
Nombre d'arêtes : 2742

Le graphe est dirigé : False
Le graphe est vide : False
Le graphe est bipartie : False
 Le graphe est biconnecté : False
Le graphe est Connecté : False
```

```
Liste des nœuds avec leurs étiquettes
ID: 0 | Label: ABRAMSON, G
ID: 1 | Label: KUPERMAN, M
ID: 2 | Label: ACEBRON, J
ID: 3 | Label: BONILLA, L
ID: 4 | Label: PEREZVICENTE, C
ID: 5 | Label: RITORT, F
ID: 6 | Label: SPIGLER, R
ID: 7 | Label: ADAMIC, L
ID: 8 | Label: ADAR, E
ID: 9 | Label: HUBERMAN, B
ID: 10 | Label: LUKOSE, R
ID: 11 | Label: PUNIYANI, A
ID: 12 | Label: AERTSEN, A
ID: 13 | Label: GERSTEIN, G
ID: 14 | Label: HABIB, M
ID: 15 | Label: PALM, G
ID: 16 | Label: AFRAIMOVICH, V
ID: 17 | Label: VERICHEV, N
ID: 18 | Label: RABINOVICH, M
ID: 19 | Label: AGRAWAL, H
ID: 20 | Label: AHUJA, R
ID: 21 | Label: MAGNANTI, T
ID: 22 | Label: ORLIN, J
```

```
Liaison entre SALWINSKI, L et BARON, M avec un poids de 0.2
Liaison entre YANG, K et HUANG, L avec un poids de 0.5
Liaison entre YANG, K et YANG, L avec un poids de 0.5
Liaison entre HUANG, L et YANG, L avec un poids de 0.5
Liaison entre YAN, G et ZHOU, T avec un poids de 0.25
Liaison entre YAN, G et FU, Z avec un poids de 0.25
Liaison entre ZHOU, T et FU, Z avec un poids de 0.25
Liaison entre YAOUM, Y et LAUMANN, E avec un poids de 1
Liaison entre YEHIA, A et JEANDUPREUX, D avec un poids de 0.333333
Liaison entre YEHIA, A et ALONSO, F avec un poids de 0.333333
Liaison entre YEHIA, A et GUEVARA, M avec un poids de 0.333333
Liaison entre JEANDUPREUX, D et ALONSO, F avec un poids de 0.333333
Liaison entre JEANDUPREUX, D et GUEVARA, M avec un poids de 0.333333
Liaison entre ALONSO, F et GUEVARA, M avec un poids de 0.333333
Liaison entre YOOK, S et TU, Y avec un poids de 0.333333
Liaison entre SAGER, J et CSARDI, G avec un poids de 0.333333
Liaison entre SAGER, J et HAGA, P avec un poids de 0.333333
Liaison entre CSARDI, G et HAGA, P avec un poids de 0.333333
Liaison entre YUSONG, T et LINGJIANG, K avec un poids de 0.333333
Liaison entre YUSONG, T et MUREN, L avec un poids de 0.333333
Liaison entre LINGJIANG, K et MUREN, L avec un poids de 0.333333
Liaison entre ZAKS, M et PARK, E avec un poids de 0.333333
Liaison entre ZASLAVER, A et MAYO, A avec un poids de 0.142857
```

★ Visualisation de réseau

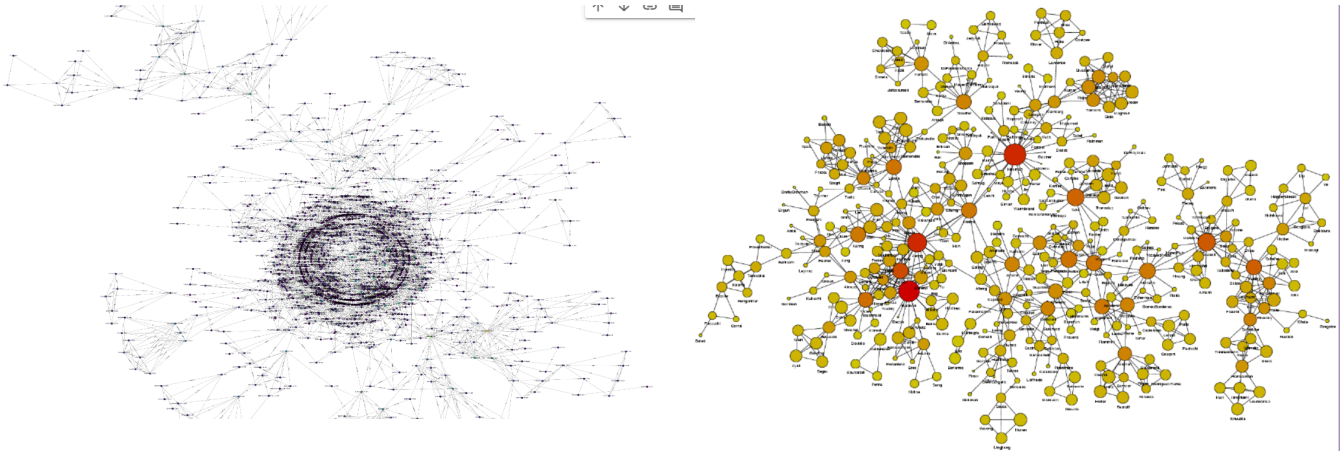


figure 1 le graphique complet du réseau NetScience

Partie 2 : Analyse du réseau

❖ Distribution des degrés: Le degré d'un sommet est le nombre d'arêtes incidentes sur le sommet.La figure 2 montre la distribution du degré de collaboration entre auteurs.

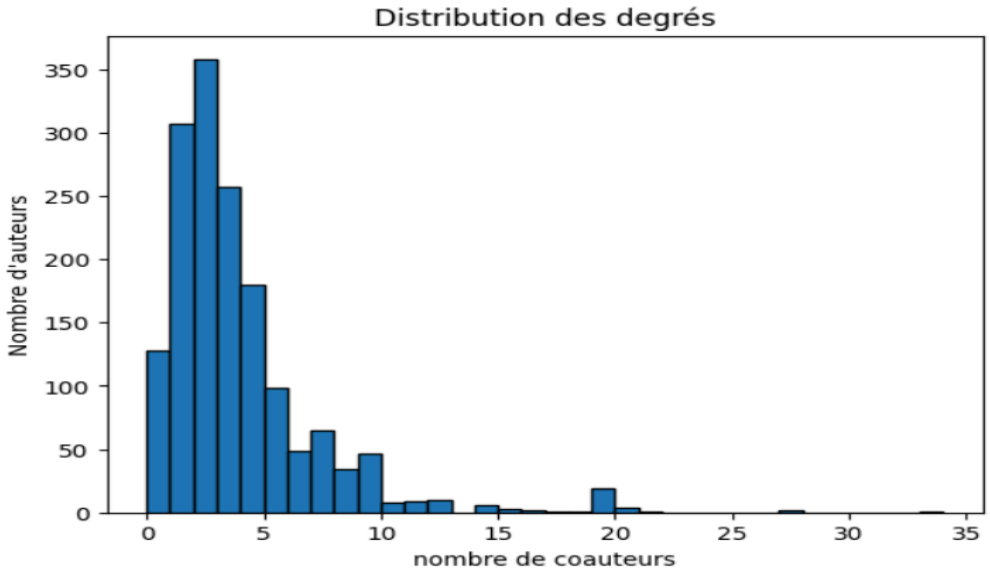


figure 2 Distribution des degrés

degrés minimale est égale à 0 et degrés maximale est égale à 34

Cela montre la présence des nœuds isolés.

- ❖ Distribution des poids des arrêts

Le Tableau 1 montre la distribution des poids des arêtes dans le réseau NetScience, la valeur la plus basse est 0.0526 et la valeur la plus élevée est 4.7500.

Table 1: Distribution des poids des aretes dans le reseau NetScie
Le poids minimale 0.0526316 et le poids maximale est 4.75.

Poids	Frequence
0.0526316	187
0.111111	135
3.5	2
3.58333	1
3.83333	1
4.225	1
4.75	1
Poids minimale	0.0526316

Table 1 Distribution de poids

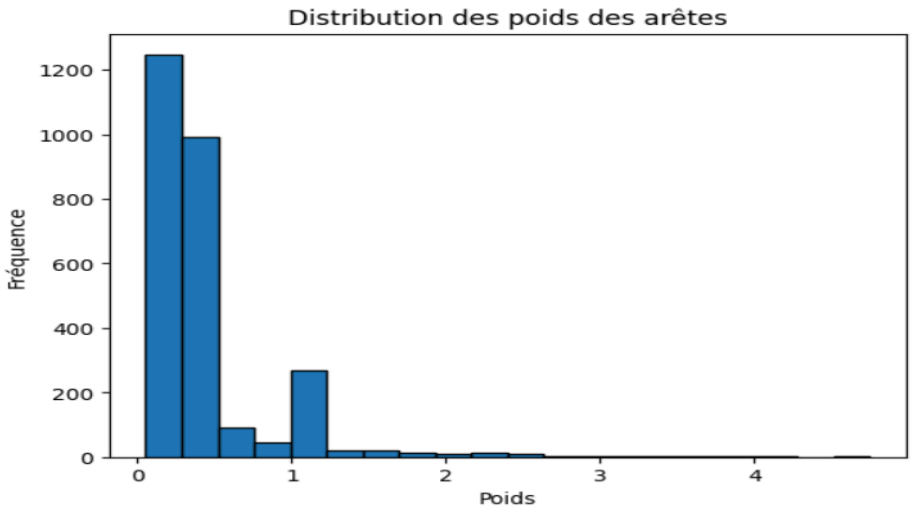


Figure 3 la distribution des poids des arêtes

- Densité

La densité est le quotient entre le nombre de connexions d'un réseau donné et le nombre maximal possible de connexions dans le même réseau. Il est évident qu'un réseau complet a une densité maximale. Cependant, la densité n'est pas la meilleure mesure car elle dépend de la taille du réseau, c'est-à-dire du nombre de connexions possibles.

```
Densité du graphe: 0.0021733168683312383
```

La densité du réseau NetScience est de 0.0021 ⇒ environ 0.21% des paires de nœuds possibles sont reliées par des arêtes dans le graphe

- Coefficient de clustering

Le coefficient de clustering est un moyen de mesurer à quel point les nœuds d'un graphe sont regroupés en clusters ou en communautés.

```
Coefficient de clustering moyen: 0.6377905695067805
```

- les voisins d'un noeud

```
Les voisins du DURAN, O ( 911 ) sont : [(909, 'KLEVECZ, R'), (910, 'BOLEN, J')]
```

- Trouver les chercheurs que je peux collaborer avec

```
Voisin: KLEVECZ, R - Voisins des voisins avec poids maximal: ['BOLEN, J', 'DURAN, O']
Voisin: BOLEN, J - Voisins des voisins avec poids maximal: ['KLEVECZ, R', 'DURAN, O']
```

les auteurs que DURAN, O peut collaborer avec sont:

Le Voisin : KLEVECZ, R - Voisins des voisins avec poids maximal: ['BOLEN, J', 'DURAN, O']

Le Voisin : BOLEN, J - Voisins des voisins avec poids maximal: ['KLEVECZ, R', 'DURAN, O']

- Centralité

	Statistique	Valeur
0	degree_centrality	{0: 0.0012594458438287153, 1: 0.00188916876574307, 2: 0.0025188916876574307, 3
1	betweenness_centrality	{0: 0.0, 1: 1.5872033318572343e-06, 2: 0.0, 3: 0.0, 4: 0.0, 5: 0.0, 6: 0.0, 7:
2	closeness_centrality	{0: 0.0014168765743073047, 1: 0.00188916876574307, 2: 0.0025188916876574307, 3

+ La centralité de proximité(Closeness Centrality) peut être interprétée dans les réseaux sociaux comme la facilité pour un nœud d'atteindre les autres nœuds dans l'ensemble du réseau.

+Le concept d'intermédiarité (betweenness) est la mesure dans laquelle un nœud se trouve entre d'autres nœuds dans le réseau social. Cette mesure prend en compte la connectivité des voisins du nœud. Elle reflète le nombre de nœuds auxquels un nœud est connecté indirectement via leurs liens directs

+ Les chercheurs en réseaux sociaux mesurent l'activité du réseau pour un nœud en utilisant le concept de degrés - le nombre de connexions directes qu'un nœud possède. Il peut être défini comme la quantité de liens qui se produisent sur un nœud (c'est-à-dire le nombre de liens qu'un nœud possède).

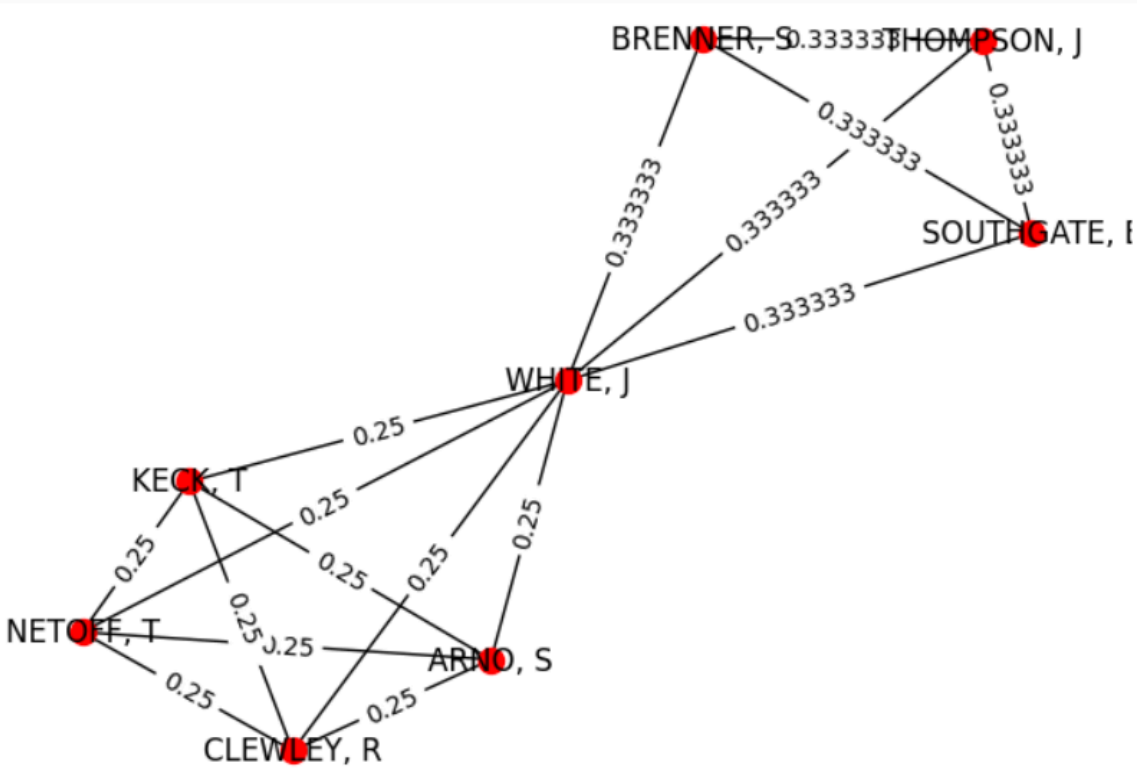
- ❖ L’analyse des composants connectés

```
Nombre de composantes connectées: 396
La plus grande composant connectée = {30, 31, 32, 33, 34, 44, 45, 46, 47, 48, 49, 50, 51, 52, 53, 54, 55, 56, 57, 58, 69, 70, 71, 72, 7
Nombre de nœuds de la plus grande composante connectée: 379
Nombre d'arêtes dans la plus grande composante connexe : 914
Nœuds isolés: [19, 26, 41, 89, 101, 110, 115, 125, 159, 168, 178, 204, 232, 236, 253, 257, 272, 295, 407, 420, 451, 504, 510, 536, 543,
Nombre des noeuds isolés 128
Le centre du sous graphe la plus large est : [78, 131, 203, 756, 757, 758, 759, 1123]
Le diamètre du sous graphe la plus large est : 17
Le rayon du sous graphe la plus large est : 9
```

figure 4 graphe de plus grand composant

➔ Diamètre et centre de chaque composante connectée

```
Composante 1 : 4 noeuds - Diamètre : 2, Centre : [1], nombre de noeud : 4, nombre des arrêts : 4
Composante 2 : 5 noeuds - Diamètre : 1, Centre : [2, 3, 4, 5, 6], nombre de noeud : 5, nombre des arrêts : 10
Composante 3 : 8 noeuds - Diamètre : 3, Centre : [7, 9, 10, 11], nombre de noeud : 8, nombre des arrêts : 11
Composante 4 : 8 noeuds - Diamètre : 2, Centre : [12], nombre de noeud : 8, nombre des arrêts : 16
Composante 5 : 3 noeuds - Diamètre : 1, Centre : [16, 17, 18], nombre de noeud : 3, nombre des arrêts : 3
Composante 6 : 1 noeuds - Diamètre : 0, Centre : [19], nombre de noeud : 1, nombre des arrêts : 0
Composante 7 : 3 noeuds - Diamètre : 1, Centre : [20, 21, 22], nombre de noeud : 3, nombre des arrêts : 3
Composante 8 : 7 noeuds - Diamètre : 3, Centre : [201, 202, 24, 25], nombre de noeud : 7, nombre des arrêts : 12
Composante 9 : 1 noeuds - Diamètre : 0, Centre : [26], nombre de noeud : 1, nombre des arrêts : 0
Composante 10 : 3 noeuds - Diamètre : 1, Centre : [27, 28, 29], nombre de noeud : 3, nombre des arrêts : 3
Composante 11 : 379 noeuds - Diamètre : 17, Centre : [78, 131, 203, 756, 757, 758, 759, 1123], nombre de noeud : 379, nombre des arrêts
Composante 12 : 6 noeuds - Diamètre : 1, Centre : [35, 36, 37, 38, 39, 40], nombre de noeud : 6, nombre des arrêts : 15
Composante 13 : 1 noeuds - Diamètre : 0, Centre : [41], nombre de noeud : 1, nombre des arrêts : 0
Composante 14 : 2 noeuds - Diamètre : 1, Centre : [42, 43], nombre de noeud : 2, nombre des arrêts : 1
Composante 15 : 3 noeuds - Diamètre : 1, Centre : [59, 60, 61], nombre de noeud : 3, nombre des arrêts : 3
Composante 16 : 31 noeuds - Diamètre : 4, Centre : [62], nombre de noeud : 31, nombre des arrêts : 97
Composante 17 : 3 noeuds - Diamètre : 1, Centre : [66, 67, 68], nombre de noeud : 3, nombre des arrêts : 3
Composante 18 : 57 noeuds - Diamètre : 7, Centre : [523, 1356, 742, 746], nombre de noeud : 57, nombre des arrêts : 149
```




```
Nombre de noeuds dans le composant 281: 8
Nombre d'arêtes dans le composant 281: 16
```

➔ plus court chemin

```
Plus court chemin entre NETOFF, T et THOMPSON, J : ['NETOFF, T', 'WHITE, J', 'THOMPSON, J']
```

➔ Liste des voisinages de tous les noeuds

```
The neighbors of node 2 are [3, 4, 5, 6]
The neighbors of node 0 are [1, 1084]
The neighbors of node 1 are [0, 946, 1084]
The neighbors of node 2 are [3, 4, 5, 6]
The neighbors of node 3 are [2, 4, 5, 6]
The neighbors of node 4 are [2, 3, 5, 6]
The neighbors of node 5 are [2, 3, 4, 6]
The neighbors of node 6 are [2, 3, 4, 5]
The neighbors of node 7 are [8, 9, 10, 11]
The neighbors of node 8 are [7]
The neighbors of node 9 are [7, 10, 11, 1424, 1425, 1532]
The neighbors of node 10 are [7, 9, 11]
The neighbors of node 11 are [7, 10, 9]
The neighbors of node 12 are [13, 14, 15, 1047, 1048, 1049, 1050]
The neighbors of node 13 are [12, 14, 15]
The neighbors of node 14 are [12, 13, 15]
```

partie 3: Identification des communautés

une communauté (ou cluster) désigne un groupe de nœuds (ou d'individus) étroitement liés les uns aux autres au sein du réseau. Ces communautés se distinguent souvent par leur densité interne élevée de liens et leur relative faible connectivité avec les nœuds en dehors de la communauté.

bibliothèque CDlib : est une bibliothèque Python dédiée à la détection de communautés dans les réseaux complexes. Elle fournit une gamme d'algorithmes de détection de communautés et d'évaluation de partitions communautaires.

Dans cette partie, On identifie, évalue et valide l'échantillon du réseau choisi par l'implémentation des algorithmes de détections de la communautés , tel que :

- ☑ Propagation des labels
- ☑ Louvain
- ☑ Infomap

1. Définition des algorithmes
    - Propagation des labels

Cette approche repose sur le principe que les nœuds appartenant à la même communauté ont tendance à partager des étiquettes similaires ou des propriétés similaires. Voici comment la propagation des labels est appliquée dans la détection de communautés :

- Initialisation des étiquettes : Chaque nœud du graphe se voit attribuer une étiquette initiale, souvent basée sur des informations telles que les attributs des nœuds ou des étiquettes préalablement connues.
- Propagation des étiquettes : Les étiquettes sont propagées à travers le graphe en plusieurs itérations. À chaque itération, les nœuds mettent à jour leurs étiquettes en se basant sur les étiquettes de leurs voisins. Typiquement, un nœud adoptera l'étiquette majoritaire parmi ses voisins.
- Convergence : Le processus de propagation continue jusqu'à ce qu'un critère de convergence soit atteint. Cela peut être défini par une stabilité dans les étiquettes attribuées aux nœuds, c'est-à-dire lorsque les étiquettes ne changent plus ou changent très peu entre deux itérations successives.
- Définition des communautés : Une fois que le processus de propagation des étiquettes a convergé, les nœuds partageant la même étiquette sont regroupés dans la même communauté. Les étiquettes finales des nœuds définissent ainsi la structure des communautés dans le graphe.

    - Louvain

L'algorithme Louvain est un algorithme de détection de communautés dans les réseaux, largement utilisé pour sa simplicité et son efficacité. Il a été développé par Vincent D. Blondel, Jean-Loup Guillaume, Renaud Lambiotte et Étienne Lefebvre à l'Université catholique de

Louvain en Belgique, d'où son nom.

Voici un aperçu de son fonctionnement :

1. Initialisation : Chaque nœud du graphe est initialement attribué à une communauté distincte.
2. Optimisation : L'algorithme vise à maximiser une mesure de la modularité du réseau, qui mesure la qualité de la division en communautés. Il le fait en deux étapes :
    - Étape 1 : Pour chaque nœud, il explore la possibilité de le déplacer dans la communauté de l'un de ses voisins afin d'augmenter la modularité. Cette opération est répétée pour tous les nœuds du réseau.
    - Étape 2 : Les communautés identifiées à l'étape précédente sont agrégées pour former des super-nœuds, et le processus est répété jusqu'à ce qu'une solution stable soit atteinte.
3. Évaluation de la modularité : Une fois que le processus d'optimisation est terminé, la qualité de la partition en communautés est évaluée en calculant la modularité du réseau. Plus la modularité est élevée, meilleure est la division en communautés.
4. Itération : Les étapes d'optimisation et d'évaluation de la modularité sont répétées jusqu'à ce qu'une configuration stable soit atteinte, c'est-à-dire jusqu'à ce qu'aucun déplacement de nœud ne conduise à une amélioration significative de la modularité.

- Infomap

L'algorithme Infomap est un autre algorithme populaire pour la détection de communautés dans les réseaux. Il a été développé par Martin Rosvall et Carl T. Bergstrom à l'Université de Washington. Cet algorithme est basé sur la théorie de l'information et est particulièrement efficace pour détecter les structures hiérarchiques dans les réseaux.

Voici un aperçu de son fonctionnement :

1. Optimisation de la compression de l'information : L'objectif de l'algorithme Infomap est de trouver une partition du réseau qui minimise la longueur moyenne de la description nécessaire pour représenter les chemins de marche aléatoire sur le réseau. En d'autres termes, il cherche à identifier les communautés qui compressent le mieux l'information dans le réseau.

2. Marche aléatoire : L'algorithme utilise une approche de marche aléatoire pour explorer le réseau. Il simule un grand nombre de trajectoires de marche aléatoire à partir de différents nœuds du réseau, en attribuant un coût à chaque transition entre les nœuds.
3. Optimisation par algorithme de descente de gradient : L'algorithme utilise un algorithme de descente de gradient pour optimiser la longueur moyenne de la description de l'information. Il ajuste itérativement la partition du réseau pour minimiser cette longueur, en déplaçant les nœuds d'une communauté à une autre lorsque cela améliore la compression de l'information.
4. Détection de communautés : Une fois que l'algorithme a convergé vers une partition optimale du réseau, les communautés sont identifiées en regroupant les nœuds qui appartiennent à la même partition.

2. Résultat

Algorithme	Nombre de communautés
Louvain	405
Infomap	442
Label Propagation	467

En utilisant le coefficient de Jaccard, on peut comparer de manière quantitative les résultats de différents algorithmes de détection de communautés, ce qui permet de mieux évaluer leur performance et leur efficacité dans la détection des structures de communautés dans un réseau donné.

```
Indice de similarité de Jaccard entre Louvain et Infomap: 0.8574561403508771
Indice de similarité de Jaccard entre Louvain et Propagation des labels: 0.7942386831275721
Indice de similarité de Jaccard entre Infomap et Propagation des labels: 0.8107569721115537
```

- Pour l'indice de similarité de Jaccard entre Louvain et Infomap, la valeur est d'environ 0.86. Cela signifie que 86% des nœuds sont affectés aux mêmes communautés par ces deux algorithmes.
- Pour l'indice de similarité de Jaccard entre Louvain et la Propagation des labels, la valeur est d'environ 0.80. Cela signifie que 80% des nœuds sont affectés aux mêmes communautés par ces deux algorithmes.

- Pour l'indice de similarité de Jaccard entre Infomap et la Propagation des labels, la valeur est d'environ 0.82. Cela signifie que 82% des nœuds sont affectés aux mêmes communautés par ces deux algorithmes.

Partie 4 : Prédiction des liens

La prédiction des liens consiste à anticiper la formation de nouvelles connexions entre les nœuds d'un réseau donné.