REPUBLIQUE ALGERIENNE DEMECRATIQUE ET POPULAIRE

MINISTERE DE L'ENSEIGNEMENT SUPERIEUR

ET DE LA RECHERCHE

UNIVERSITE SAAD DAHLEB DE BLIDA

FACULTE DES SCIENCES

DEPARTEMENT D'INFORMATIQUE

Mémoire de fin d'étude

Pour l'obtention d'un diplôme de master en informatique

Spécialité : système informatique et réseaux

Thème

Application des algorithmes de machine Learning pour la prédiction de lien dans les réseaux sociaux

Réalisé par :

Hamzaoui Bouchra

Kara Lydia

Encadré par :

Mr. MERAZKA Mustapha

Organisme d'accueil : Centre de Recherche sur l'Information Scientifique et Technique

Promoteur : Mr.DERRAR HaceneUSDB

Présidente : Mme.BOUSTIA NarimeneUSDB

Examinateur : Mr.BALA Mahfoud USDB

PROMOTION 2018-2019

Remerciement

Nous remercions dieu pour nous avoir donné santé, courage et patience afin de nous aider à réaliser ce travail.

Au terme de ce modeste travail nous tentons à remercier chaleureusement et respectivement tous ceux qui ont contribué de près ou de loin à la réalisation de ce modeste projet de fin d'étude, à savoir notre encadreur Mr .Merazka Mustapha et tout l'équipe de CERIST pour avoir accueillis.

Nous tentons aussi à remercier particulièrement notre promoteur Mr Derrar Hacene de nous avoir suivis et guidés tout au long de réalisation de notre travail.

On remercie vivement mesdames et messieurs les membres de jury d'avoir accepté d'évaluer notre modeste travail.

Dédicace

***J**e dédie ce travail qui n'aura jamais pu voir le jour sans les soutiens indéfectibles et sans limite de ma grande mère qu'elle était la base de mon existence et le bonheur infini.*

***À** mes chers parents et de mes tantes et mon cher oncle qui ne cessent de me donner avec amour le nécessaire pour que je puisse arriver à ce que je suis aujourd'hui.*

***A** mon binôme Lydia qui a était m'accompagnante dans les années de mon cursus d'étude.*

***A** tous mes amies pour son soutien moral en particulier Yasmin, manel et salima.*

Bouchra

Dédicace

***J**e dédie ce modeste travail à ma chère grand-mère qui est la joie, le bonheur et la base de la famille.*

***A**ux deux étoiles de ma vie ceux que je vois jour et nuit, je parle bien sure de * ma mère et mon père * que Dieu les protège, ceux qui ont était toujours là pour moi, c'est grâce à eux que je suis ce que je suis.*

***A** ma chère sœur et mon frère qui sont été toujours me soutient*

***A** mon petit cœur et mon bébé d'amour Anes*

***A** mon binôme Bouchra qui a m'accompagné dans ce travail et dans les années de mon cursus d'étude.*

***A** mes meilleurs amis qui ont été toujours là pour moi en particulier Fatma Zohra, Yasmine.*

***A** tout Personne qui m'aime et me respecte en particulier Melissa et Mēga.*

Et

***A** tous les membres de ma famille.*

Lydia

Résumé

La prédiction des liens, dont l'objectif est de comprendre en profondeur la structure des réseaux, est l'un des sujets de recherche les plus en vogue dans le domaine de l'analyse des réseaux sociaux.

L'objectif de ce travail est de proposer une approche qui facilite la prédiction des liens, manquants ou futurs dans un réseau, basée sur une approche issue du domaine de l'apprentissage automatique. La solution que nous proposons dans ce mémoire se base, dans un premier temps, sur l'extraction des caractéristiques des nœuds et des liens et les combine, par la suite avec les caractéristiques de la structure du graphe afin de générer à la fin des caractéristiques optimales qui seront utilisés pour prédire les liens manquants ou futurs.

Les résultats de cette approche, dont les performances ont été comparées avec d'autres algorithmes de prédiction de liens, ont été testés sur différents types de réseaux.

Mots clés : prédiction de lien, apprentissage automatique, liens manquants/futurs, extraction des caractéristiques, caractéristiques optimales.

Abstract

The prediction of links, which aims to deeply understand the structure of networks, is one of the most popular research topics in the field of social network analysis.

The goal of this work is to propose an approach that facilitates the prediction of missing or future links in a network, based on an approach from the field of machine learning. The solution that we propose in this memory is based, initially, on the extraction of the characteristics of the nodes and the links and combines them, with the characteristics of the structure of the graph in order to generate at the end of characteristics which will be used to predict missing or future links.

The results of this approach, whose performance has been compared with other link prediction algorithms, have been tested on different types of networks.

Key words: link prediction, machine learning, missing / future links, feature extraction, optimal characteristics.

ملخص

يعود التنبؤ بالروابط أحد أكثر الموضوعات شيوعا في مجال تحليل الشبكات الاجتماعية والتي تهدف إلى فهم بنية الشبكات بعمق.

الغرض من هذا العمل اقتراح نهج تنبؤ بالرابط يلاءم مجال التعلم الآلي الذي يسهل التنبؤ بالوصلات المفقودة أو المستقبلية في الشبكة. وبالتالي فان الحل المقترح يعمل عن طريق الاستناد إلى استخراج خصائص العقد والروابط ودمجها مع خصائص هيكل الرسم البياني للخروج في النهاية مع الخصائص المثلى التي ستستخدمها للتنبؤ بالروابط المفقودة/المستقبل.

يتم تقييم النهج المطور على أنواع مختلفة من الشبكات. تتم مقارنة أداء هذا النهج مع خوارزميات التنبؤ بالوصلة الأخرى.

Table des matières

Liste des figures

Liste des tableaux

Liste d’algorithme

Introduction générale

De nombreux systèmes réels sont modélisés par des graphes, dont les entités sont représentées par des sommets et leurs relations par des arêtes, on la retrouve dans de nombreux domaines et champs d'application et dans la structure de graphe tels que : les graphes sociologiques, les graphes de collaborations, les réseaux biologiques, tels que : la représentation des interactions protéine-protéine, les réseaux de neurones, les réseaux de gènes), les réseaux sportifs, dans le web pour modéliser les pages web connectés, dans les réseaux de transport, etc. ces graphes portent le nom de graphes de terrains ou de réseaux complexes.

L'étude et l'analyse de ce genre de graphes révèlent des informations sur les acteurs et expliquent la structure même du réseau. Parmi les nombreuses applications utilisant les graphes, comme structure de modélisation, on trouve la prédiction de liens qui est l'un des sujets pertinents pour l'analyse des réseaux sociaux. Ces derniers sont souvent constitués de liens manquants et des liens qui peuvent être créés dans le futur dont la prédiction de tels liens aidera à mieux comprendre leurs fonctionnements.

La prédiction de lien fait référence au processus d'exploration et détermine si un lien entre deux nœuds dans un réseau donné peut émerger dans le futur ou s'il est déjà présent mais caché. La prédiction de lien fait partie des classes de système de recommandation. A titre d'exemple, chercher à trouver ou prévoir un lien ou une recommandation entre des utilisateurs et des articles ou bien identifier un ensemble de nouvelles interactions en cherchant des liens cachés et d'extraire par la suite des informations manquantes dans un réseau.

La prédiction de lien a plusieurs défis. Le premier, est que les travaux sur les réseaux réels sont extrêmement rares, c'est-à-dire que les nœuds n'ont des connexions qu'avec une très petite fraction de tous les nœuds du réseau. Par exemple, dans le cas de Facebook, un utilisateur typique est connecté à environ 100 sur plus de 500 millions de nœuds du réseau, malheureusement c'est un moyen inutile pour la prédiction et ça ne permet pas de prédire aucun nouveau lien, car cela permet d'obtenir une précision prédictive presque parfaite (c'est-à-dire sur 500 millions de prédictions possibles, il ne fait que 100 erreurs).La question qui se pose est : à quelle mesure les liens du réseau social peuvent être modélisés en utilisant les caractéristiques intrinsèques du réseau lui-même ? De même, comment les caractéristiques des utilisateurs (âge, sexe, ville natale, par exemple) interagissent-elles avec la création de nouveaux liens ? Prenons l'exemple du réseau social Facebook. Plusieurs raisons peuvent exister pour que deux utilisateurs se connectent : il se peut qu'ils se soient rencontrés lors d'une soirée, après ils seront connectés sur Facebook. Puisqu'ils se sont rencontrés lors d'une fête, ils ont probablement le même âge et vivent probablement dans la même ville, de plus, ce lien pourrait également être évoqué par la structure du

réseau : deux personnes sont plus susceptibles de se rencontrer à la même soirée s'ils sont « proches » du réseau. Une telle paire de personnes a probablement des amis en commun, et voyager dans des cercles sociaux similaires. Ainsi, malgré le fait qu'ils soient devenus amis en raison de cet événement exogène (c'est-à-dire une fête), des indices dans leurs réseaux sociaux laissent supposer une forte probabilité d'amitié future. La question est donc de savoir comment les caractéristiques de la structure du réseau et des nœuds interagissent dans la création de nouveaux liens et aussi quelle est l'importance d'avoir des intérêts et des caractéristiques communs.

Le deuxième défi de la prédiction des liens, c'est qu'au lieu de développer une méthode qui combine les caractéristiques des nœuds (c'est-à-dire les informations du profil utilisateur) et des arêtes (c'est. À-dire les informations sur les interactions) avec la structure du réseau, il faut rechercher une approche commune, qui consiste à extraire simplement un ensemble de caractéristiques décrivant la structure du réseau (comme le degré de nœud, le nombre d'amis communs, la longueur du chemin le plus court) autour des deux nœuds d'intérêt et à le combiner avec les informations de profil de l'utilisateur.

L'objectif de ce projet de fin d'études est développé une approche permettant de prédire les interactions susceptibles de se produire, entre les nœuds existants, dans un avenir proche ou de prédire les interactions qui manquent dans un réseau donné. Cette prédiction consiste à retrouver une nouvelle structure plus complète dans différent types de réseaux, en particulier les réseaux sociaux qui sont hautement dynamique. En effet les réseaux sociaux évoluent et changent rapidement grâce aux ajouts de nouveaux liens qui signifient l'apparition de nouvelles interactions entre les nœuds du réseau. Nous considérons le problème classique de la prédiction de lien **[21]** où nous avons un réseau social à l'instant **t**, et nous cherchons à prédire avec précision les liens qui seront ajoutées au réseau pendant l'intervalle de temps**t** à un avenir **t+1**.Plus concrètement, nous considérons, à l'instant **t,** un grand réseau, de type Facebook, et pour chaque utilisateur, nous voudrions prédire quelles nouvelles arêtes (amitiés) cet utilisateur créera entre **t** et dans un temps futur **t+1**. A chaque utilisateur une liste des personnes avec lesquelles il est susceptible de créer de nouvelles connexions.

Notre travail consiste à concevoir et implémenter une méthode pour la prédiction des liens en intégrant un algorithme basé sur des techniques de Machine Learning. Notre approche est présentée dans ce mémoire qui est composé de quatre chapitres :

- Dans le premier chapitre, nous présentons les concepts fondamentaux des graphes, la notion liés aux réseaux complexe et leurs caractéristiques, la définition de la prédiction de lien, ses catégories, ses intérêts ainsi que leur domaine d'application.

- Dans le second chapitre, nous présentons une classification des méthodes de prédiction de lien et une description de chacune de ces méthodes.
- Dans le troisième chapitre, nous proposons une approche pour la prédiction de lien. Nous présentons, en premier lieu, une description générale de notre approche avec un organigramme qui définit les étapes de la méthode utilisée puis nous reprenons chaque étape de la méthode en l'expliquant en détail. Ensuite nous avons proposé un exemple illustratif pour mieux comprendre le déroulement des étapes.
- Dans le quatrième chapitre nous présentons les résultats des tests réalisés sur des réseaux synthétiques et des réseaux réels et l'évaluation de ces résultats.

Enfin nous terminions notre travail par une conclusion générale et des perspectives.

Chapitre 1

Introduction aux graphes et réseaux complexes

1. Introduction

Dans ce chapitre nous présentons les concepts fondamentaux des graphes ainsi que les différents types de réseaux complexes. Nous détaillons par la suite les notions de bases inhérentes au domaine de la prédiction des liens dans les réseaux sociaux.

2. Présentation des notions générale des graphes

Un graphe est un ensemble de sommets (points ou nœuds) noté **V** et d'arcs (liens orientées) ou d'arêtes (liens non orientées) noté **E**liant certains couples de points. La notation usuelle est la suivante : **|V|= N** (nombre de nœuds), **|E| = M** (nombre de liens). Les arêtes du graphe peuvent être pondérées grâce à une fonction du poids $\boldsymbol{\omega}: \boldsymbol{E} \rightarrow \boldsymbol{\mathcal{R}}^{+}$permettant de modéliser plus finement les interactions entre sommets. Nous obtenons ainsi un graphe pondéré **G = (V, E,$\boldsymbol{\omega}$)**. Le poids d'une arrêtes $\{\boldsymbol{i},\boldsymbol{j}\}$entre deux sommets **i** et **j** sera noté$\boldsymbol{\omega_{ij}}$.

Par convention, un poids nul est attribué dans le cas où l'arrêtes n'existe pas ($\boldsymbol{\omega_{ij}} = \mathbf{0}\ \boldsymbol{si}\{\boldsymbol{i},\boldsymbol{j}\} \notin \mathbf{E}$**).** Dans le cas d'un graphe non pondéré les poids des arrêtes de **E** sont fixé à **1**, ainsi dons ce cas particulier $\forall_{\boldsymbol{ij}} \in \boldsymbol{V}, \boldsymbol{\omega_{ij}} \in \{\mathbf{0},\mathbf{1}\}$.

- Le degré **d(v)** d'un sommet **v** ∈ $\boldsymbol{V}$est le nombre d'arrêtes incidente au sommet **v**. Il s'agit du nombre de sommets voisins de **v**. Nous définissons aussi le poids$\boldsymbol{\omega}$ **(i)** d'un sommet **i** comme la somme des poids de ces arrêtes incidentes :

$$\boldsymbol{\omega}(\mathbf{i}) = \sum_{\boldsymbol{j}\in\boldsymbol{V}} \boldsymbol{\omega_{ij}}$$

Notons que le poids d'un sommet coïncide avec la définition d'un sommet dans le cas des graphes non pondéré.

- Soit un graphe **G = (V, E)**,on appelle la distance d'un sommet **V** à un autre la longueur du plus court chemin/chaîne entre ces deux sommets, ou **1** s'il n'y a pas un tel chemin/chaîne. Le diamètre d'un graphe est la plus grande distance possible qui puisse exister entre deux de ses sommets.
- La densité d'un graphe est définie comme $\frac{\boldsymbol{2m}}{\boldsymbol{n(n-1)}}$soit le rapport entre le nombre d'arêtes et le nombre maximum d'arêtes possibles compte tenu du nombre de nœuds du graphe.
- On dit que deux sommets d'un graphe non-orienté sont voisins ou adjacents s'ils sont reliés par une arête. Dans un graphe **G** non orienté le voisinage d'un sommet **v** ∈ **V,** souvent noté $\boldsymbol{N_G}$**(v)**

Peut désigner l'ensemble de ces sommets voisins ou bien un sous-graphe associé. Dans un graphe orienté, on emploie généralement le terme de prédécesseur ou de successeur.

- Un graphe **G** est représenté par une matrice d'adjacence **A** tel que dans le cas ou **G** est pondéré non-orienté **A** est définie par :

$$A_{ij} = A_{ij} = \begin{cases} \mathbf{0}\ si\ \{i,j\} \notin E \\ w_{ij}\ \ si\ \{i,j\} \in \mathbf{E} \end{cases}$$

Dans le cas d'un graphe non pondéré $A_{ij} \in \{\mathbf{0},\mathbf{1}\}$(car nous fixons un poids w_{ij} égale à **1** pour toutes les arêtes de **E**.

3. Types des graphes

• Graphe complet

Un graphe orienté est dit complet s'il comporte un arc (v_i, v_j) et un arc (v_j, v) pour tout couple de sommets différents$v_i, v_j \in V$. De même, un graphe non-orienté est dit complets il comporte une arête $\{v_i, v_j\}$ pour toute paire de sommets différents$v_i, v_j \in V$. On note $K_n$un graphe complet d'ordre **n**.

• Sous graphe partiel

Un sous graphe de **G** est un graphe **G'** ayant pour sommets un sous-ensemble **V'** des sommets de **G** et pour arcs/arêtes un sous-ensemble de ceux joignant les sommets de **V'.**

• Sous graphe induit

Un graphe induit de **G** est un graphe **G'** ayant pour sommets un sous-ensemble **V'** des sommets de **G** et pour arcs/arêtes uniquement ceux de **G** joignant les sommets de **V'**.

• Graphe connexe

Un graphe**G** est dit connexe lorsqu'il existe une chaine entre deux sommets quelconques de **G**.

• Graphe aléatoire

Un graphe aléatoire est un graphe qui est généré par un processus aléatoire. Il est caractérisé par la distribution des degrés suivant une loi puissance, fort nombre de triangles et aussi par la densité de ces types des graphes. Elle dite petite si les degrés des sommets sont petits comparé à la taille du graphe.
Les graphes aléatoires sont des modèles pour étudier les grands graphes comme les graphes des réseaux sociaux, biologique, information et technologie etc.… **La figure 1.1** présente la structure d'un graphe.

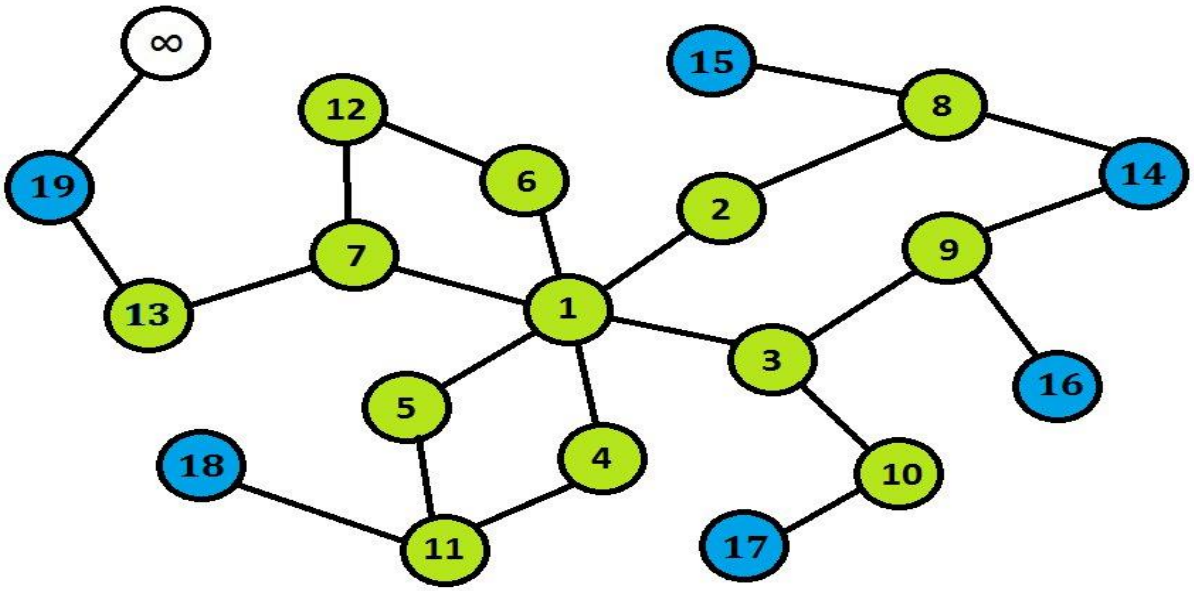


Figure 1.1 structure d'un graphe

4. Réseaux complexe

Les graphes sont utilisés généralement pour représenter n'importe quel type de réseau tel que les systèmes complexe du monde réel tels que : les réseaux biologiques, les réseaux sociaux, les réseaux cibles, le Word Wide Web(WWW) et les réseaux de communication.

Un réseau complexe est un graphe constitué des nœuds qui peuvent être (individus, organisation, objets) relié par des liens qui sont des interactions ou des relations. Il représente donc des données collectées qui correspondent à une vérité comme les réseaux biologiques (interactions protéine-protéine, réseaux de neurones, réseaux de gènes…etc.).

Dans les sections qui suivent, nous présentons quelque type de réseaux complexes.

4.1. Les réseaux sociaux

Le mot « réseau social » a été utilisé par le juge Barnes dans la classe et des comités dans une île norvégienne paroisse en1954 pour expliquer les relations humaines **[2,3]**. Le réseau social apparaît comme une structure sociale composée de différents nœuds du réseau. Chaque « nœud » signifie une personne ou une organisation. D'une manière générale, un réseau social est une carte de tous les nœuds et les connexions marquées comme le montre **la figure 1. 2**. Chaque nœud représente une entité unique. Ils peuvent être soit une personne ou un groupe. De nombreuses connexions / liens relient les nœuds entre eux. Ces liens peuvent être des relations familiales, des amis ainsi que des collègues, etc. De l'étude des graphes et de ses processus d'évolution dynamique, nous pouvons généralement trouver des informations précieuses qui peuvent nous aider à résoudre des problèmes pratiques dans notre monde réel **[4,5]**.

Un réseau social est défini comme un réseau d'interactions ou de relations, dans lequel les nœuds sont constitués d'acteurs, et les arêtes sont constituées des relations ou des interactions entre les acteurs **[1]**. La généralisation de l'espace des réseaux sociaux est celle des réseaux d'information, dans lesquels les nœuds peuvent comprendre des acteurs ou des entités, et les contours indiquent les relations qui les unissent. De toute évidence, le concept de réseau social ne se limite pas au cas particulier d'un réseau social basé sur Internet tel que Facebook.

Il existe deux catégories de réseaux sociaux. La première porte sur les réseaux sociaux humains et la deuxième catégorie porte sur les réseaux sociaux en ligne. Le problème des réseaux sociaux a souvent été étudié dans le domaine de la sociologie sous l'angle des interactions génériques entre tous les groupes d'acteurs. De telles interactions peuvent être dans n'importe quelle forme conventionnelle, que ce soit des interactions entre contacts différents, des interactions de télécommunication, des interactions de courrier électronique ou des interactions de courrier postal **[1]. La figure 1.2**, ci-dessous, présente une structure d'un réseau social.

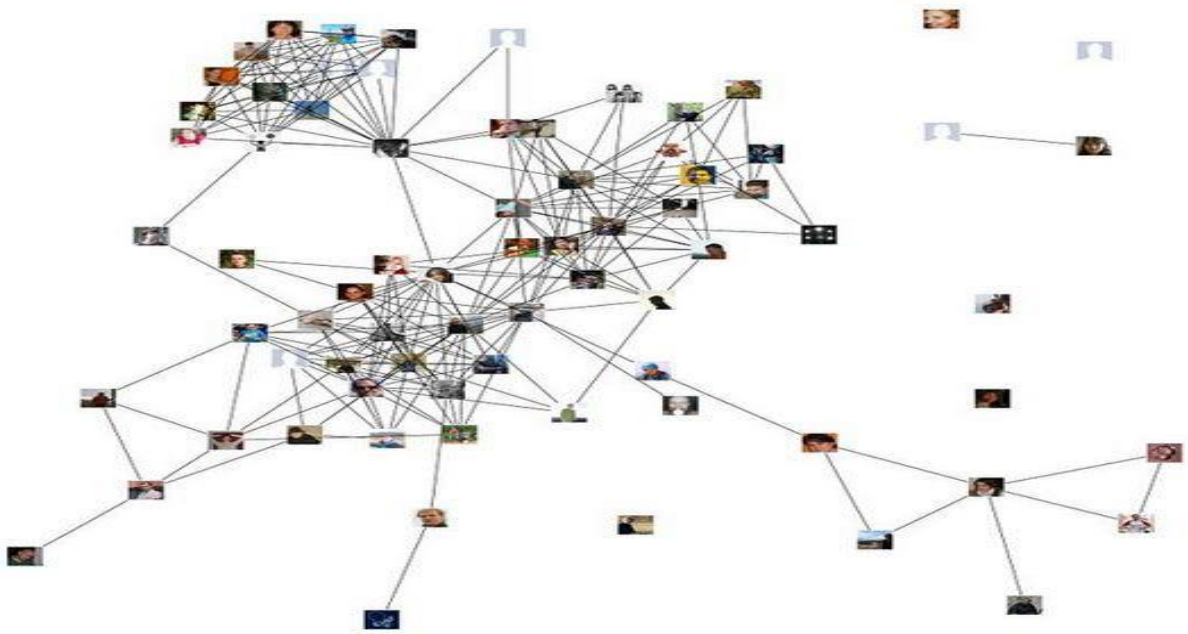


Figure 1.2 Structure d'un réseau social

Il existe plusieurs types de réseaux sociaux :

- **Réseaux sociaux signé :**

C'est un graphe constitués des liens entre les nœuds qui peuvent être positifs désigné par le signe «+», ou négatifs désigné par «-».dans les réseaux sociaux non dirigés, les signes sur les liens indiquent la relations entre les individus représentés par les nœuds tel que le lien positif indique la confiance, amitié, soutien ou

amour entre les utilisateurs, tandis que le lien négatif indique la méfiance, l'hostilité, l'aversion et l'opposition des utilisateurs, alors que dans un réseau social dirigé, le signe sur les liens indique le rang des personnes une telle enseigne sur un lien dirigé reflète le statut social ou le prestige des personnes représentés par des nœuds **[6].**

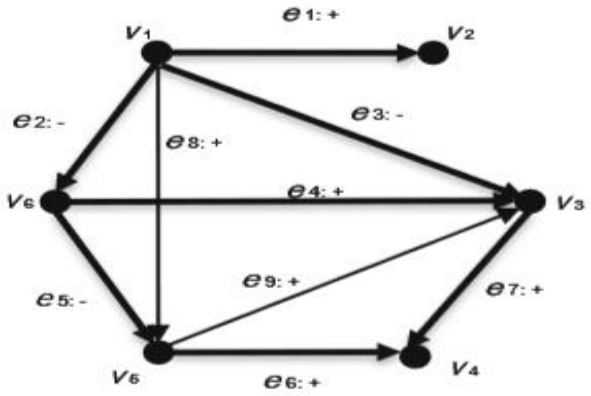


Figure 1.3 structure d'un réseau signé

- **Réseaux sociaux en ligne :**

Les réseaux sociaux en ligne (sur internet) se matérialisent par des sites sur lesquels chaque utilisateur (nœud) se crée un profil. Le profil est en quelque sorte la carte d'identité numérique sur le réseau social. Une fois inscrit, chaque membre peut publier du contenu, aussi bien sous forme de messages (liens), d'images, de vidéos… Les autres actions possibles sur les réseaux sociaux ont en général été : La recherche d'amis ou de proches, Le partage de contenu, Le postage de commentaire ou de mention « j'aime ».

- **Réseaux sociaux humain :**

Les réseaux sociaux humains existaient bien avant l'arrivée d'Internet,ils représentent un regroupement de personnes ou d'organisations qui échangent, communiquent et partagent leurs idées autour d'un sujet commun, par exemple, les clubs d'équitation, de tennis, de belote… sont par définition des réseaux sociaux**.**

4.2. Les réseaux biologiques

Les biologistes rencontrent des réseaux métaboliques, modélisant les processus de génération et de dégradation des matériaux et de l'énergie au sein d'organismes vivants **[19]**, des réseaux d'interaction entre protéines **[20]** ou encore des réseaux de régulation génétique **[21]**. Nous pouvons citer également les

réseaux alimentaires. Ces réseaux exhibent des propriétés de réseaux complexes, en particulier la distribution sans échelles d'après les études topologiques dont ils font l'objet **[22,23]**.

La Figure 1.4 montre un exemple de réseau de régulation génétique : Interaction de régulation entre des gènes, des protéines et des petites molécules.

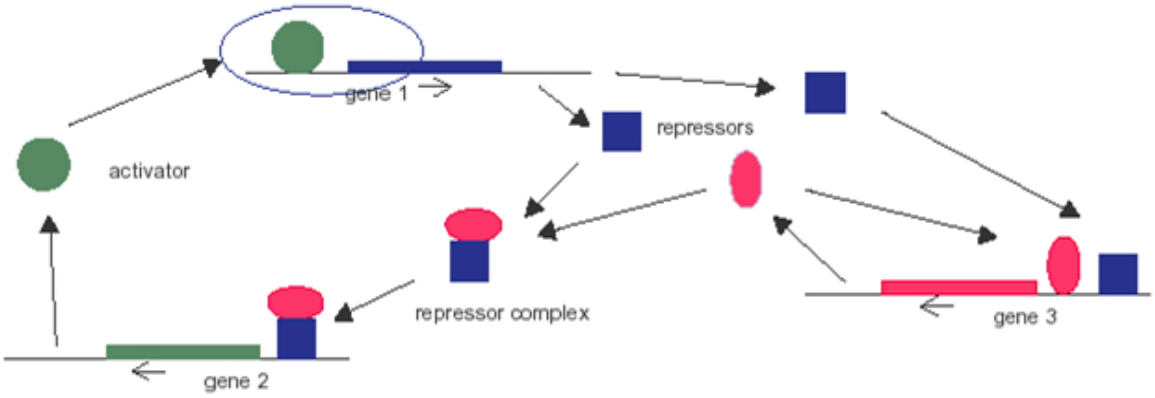


Figure 1.4 structure d'un réseau biologique

5. Conclusion

Dans ce premier chapitre, nous avons présenté les notions principales relatives aux graphes et aux différents types des réseaux complexes. Nous avons également défini la prédiction des liens ainsi que les ses concepts de bases. Dans le prochain second chapitre nous présentons l'état de l'art relatif à la prédiction des liens ainsi que les différentes méthodes utilisées dans ce domaine.

Chapitre 2

Etat de l’art

1. Introduction

La prédiction des liens est une tâche qui consiste à prédire les relations et les interactions dans un réseau. Les techniques d'apprentissage automatique sont proposées pour la prédiction de liens manquants ou futurs entre les nœuds de graphe. L'objectif principal est de prédire l'avenir des liens qui ne sont pas encore observées dans l'état actuel du réseau. En raison de son importance, la tâche de prédiction de lien a reçu une grande attention des chercheurs de diverses disciplines. Ainsi, un grand nombre de méthodes pour résoudre ce problème ont été proposées au cours des dernières années. Ces méthodes diffèrent selon les aspects considérés, y compris les modifications de l'évolution, le type ou la quantité d'informations traitée.

Dans ce chapitre nous effectuons une étude sur les approches de prédiction de lien et nous essayons de présenter les principales méthodes existées. Pour ce faire, nous avons choisi de faire une classification de prédiction de liens selon les deux principaux types d'approche à savoir : les approches heuristiques et des approches d'apprentissage automatique. A travers cette classification, nous effectuons une étude comparative des méthodes proposées qui sera illustrée dans un tableau récapitulatif qui présentera les défis et les avantages de ces méthodes.

1. Prédiction de lien

Les réseaux sont de plus en plus utilisés pour modéliser des systèmes complexes composés d'éléments en interaction, tels que : les réseaux sociaux, les réseaux biologiques qui ont été décrit dans les sections précédentes. Différentes études ont montré qu'il est possible de prédire de nouvelles relations entre les éléments présents dans la topologie d'un réseau. Cette thématique qui consiste à chercher de nouvelles relations dans les réseaux est appelée prédiction de lien. Elle vise à prédire le comportement de lien, c'est-à-dire si une relation entre deux éléments dans un réseau peut être créée ou si une relation entre eux est manquante en basant sur les relations actuellement observées. Beaucoup d'études et de recherches ce sont orientés vers ce domaine compte tenu de son champ d'application. Pour cette raison plusieurs méthodes ont été conçues et appliquées pour rechercher et de prédire des liens dans différents types de réseaux.

1.1. Intérêt de la prédiction de liens dans les réseaux sociaux

L'analyse des réseaux sociaux est devenue un sujet de recherche populaire en informatique. Il est bien connu que la prédiction sur les réseaux sociaux est un domaine complexe et difficile, car les réseaux sociaux, particulièrement, ceux en ligne se composent d'un grand nombre d'utilisateurs comptant des millions de nœuds ou plus, voire des milliards d'arêtes. De plus, les données des réseaux sociaux en ligne sont très dynamiques. Les activités sociales des utilisateurs dans ce type de réseau sont imprévisibles. Le

regroupement ou la sortie des utilisateurs, ainsi que l'émergence ou l'élimination des contours, peuvent survenir à tout moment **[17]**.

Les relations dans les réseaux sociaux présentent une grande diversité. Les différents types de systèmes ont différents types de relations, leur degré de force, leur sens de l'orientation, etc.

Si nous pouvons prédire avec précision les limites qui seront créées entre deux nœuds du réseau pendant un intervalle de temps allant de **t** à un temps futur donné **t'(t'> t) [18]**, nous pouvons comprendre comment un réseau social évolue et quel est la dynamique qui est derrière.

Plus important encore, étant donné que les liens du réseau représentent leur maintien et leur qualité reflétant les comportements sociaux d'individus et de communautés, la recherche de prédiction de lien peut donc être très utile pour l'évaluation quantitative et qualitative des relations humaines en cette ère d'informations où davantage de personnes participent à des communautés d'un réseau social en ligne ou humain par exemple les clubs sportifs. Ainsi, la prédiction de lien est une tâche importante dans l'analyse de réseau social c'est pour cela plusieurs algorithmes et méthodes de la prédiction de liens sont appliqués à une grande variété de réseaux **[17]**.

1.2. Domaine d'application de prédiction de lien

Les techniques de prédiction de liens ont trouvé un grand nombre d'applications dans des domaines très différents. Tout domaine dans lequel les entités interagissent de manière structurée peut potentiellement bénéficier de la prédiction de lien. Ces techniques sont utilisées pour améliorer la sélection des utilisateurs similaires dans les systèmes de recommandation qui adoptent une approche collaborative, ce qui permet d'obtenir de meilleurs résultats de recommandation **[26]**. Une application similaire est liée aux réseaux sociaux, qui sont devenus extrêmement populaires dans la société moderne, Les utilisateurs de ces systèmes s'attendent à disposer de mécanismes simples et efficaces leur permettant de se familiariser avec l'énorme quantité d'utilisateurs enregistrés.

La plupart des réseaux sociaux utilisent des techniques de prédiction de lien pour suggérer automatiquement des connaissances avec un haut degré de précision.

Dans le domaine de la biologie, des techniques de prédiction de liens sont appliquées pour trouver des interactions possibles entre des paires de protéines dans un réseau d'interaction protéine-protéine (réseau PPI) **[27]**.

Une autre application se trouve dans la prédiction de collaboration dans les réseaux de co-auteurs scientifiques. Les données de collaboration sont facilement accessibles, car certains sites d'indexation de journaux rendent publiques leurs collections. Les méthodes de prédiction de liens sont devenues un outil

permettant de mieux comprendre les domaines de recherche des réseaux de prédiction des groupes d'auteurs ou des groupes d'auteurs ou de collaboration potentiellement à l'avenir **[28]**.

1.3. Définition formelle de prédiction de liens

Etant donné un réseau G_t= **(V,**E_t**)** à un instant donné **t**, nous devons prévoir l'ensemble des nouveaux liens **E** qui apparaîtront probablement dans le réseau dans l'intervalle de temps **[t,**t'**]**, où t'**> t**. Le réseau $G_{t'}$ à l'instant t'peut être représenté par :

$$G_{t'} = (V, E_{t'}) \text{ où } E_{t'} = E_t \cup E$$

Il est important de noter que, dans le problème de la prédiction de liens, **V** reste statique avec l'heure. En revanche, **E** varie d'une heure à l'autre à mesure que de nouveaux liens sont ajoutés au réseau **[7]**. De nombreux ensembles de données peuvent naturellement être représentés sous forme de graphique où les nœuds **V** représentent les instances et les liens **E** représentent les relations entre ces instances **[8].**

Des liens peuvent être manquants entre deux nœuds liés ou des liens qui peuvent être créent à l'avenir **[8]**. L'objectif de la prédiction de lien est donc de prédire l'existence de ces liens futurs ou manquants entre les nœuds du graphe. **Figure 2.1** illustre la prédiction des liens entre plusieurs instants.

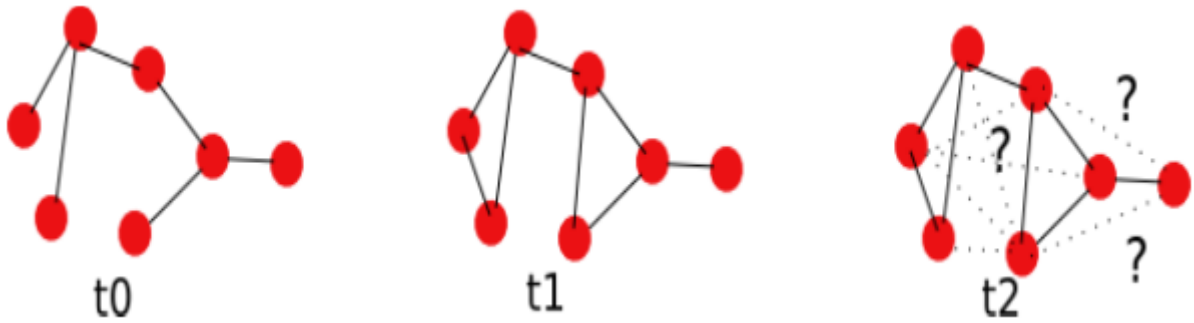


Figure 2.1 la prédiction de lien dans les instants t_1, t_2

1.4. Catégories des modèles de prédiction de lien

Nous citons dans ce qui suit, les deux principales catégories de modèles de prédiction de lien utilisées à savoir, la prédiction des liens manquants et la prédiction des liens futurs.

1.4.1. Prédiction des liens manquants

Considérons un graphe **G = (V, E),** où **V** est l'ensemble des nœuds et **E** est l'ensemble des arêtes et que chaque lien **e = (u, v) ∈ E** représente un lien entre **u** et **v**. Appelons le sous-graphe **G [k]** composé de tous liens qui sont disponibles (également appelé graphe de formation) et **G [k ']** composé de tous liens

manquants (appelé graphe de test) **[29]**. En d'autres termes, l'union des deux sous-graphes, **G [k]** et **G [k']** est égale à la courbe d'origine et l'intersection de ces deux sous-graphes est vide.

$$E_k \cup E_{k'} = E \text{ et } E_k \cap E_{k'} = \emptyset$$

Pour accéder au sous-graphe **G [k],** il faut donner une liste des arêtes non présentées dans **G [k]** qui sont prévus pour apparaître dans le **G [k]. La Figure 2.1** décrit une vue simple de prédiction de liens manquants dans un réseau social.

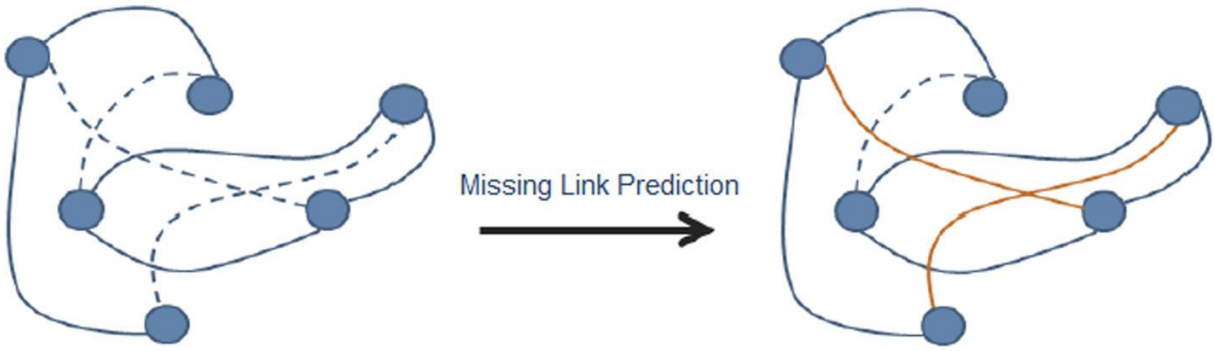


Figure 2.2 structure de graphe avant et après la prédiction des liens manquants

Cette figure montre un aperçu de chaîne manquante entrant et sortant. Liens pointillés dans le graphe de gauche sont manqués et les liens rouges dans le graphe de droite montre les liens prédisent.

1.4.2. Prédiction de liens futurs

La prédiction des liens futurs peut être regroupée dans les catégories suivantes : prédiction de lien périodique et non périodique. Le premier type de prédiction de lien considère la nature dynamique du graphe comme un élément clé en raison de la prédiction. D'autre part, ce dernier type n'est pas pour découvrir les changements dans le temps, mais il met l'accent sur l'état actuel du réseau. Le comportement de prédiction de lien futur est représenté dans **la Figure 2.2**.

- **Prédiction de liens périodiques**

Étant donné une série d'instantanés $\{\mathbf{g_1}, \mathbf{g_2}, .., \mathbf{g_t}\}$d'un graphe en évolution $\mathbf{g_t = (V, E_t)}$ dans laquelle chaque $\mathbf{e = (u, v) \in E_t}$, représentent un lien entre **u** et **v** qui a eu lieu à un moment donné **t [30, 31,32]**. Il faut chercher à prédire l'état de lien le plus probable dans la prochaine étape de temps $\mathbf{g_{t+1}}$. Dans presque toutes les méthodes analysées dans l'étude, il est supposé que les nœuds **V** restent les mêmes dans toutes les étapes du temps, mais les liens $\mathbf{E_t}$ changent pour chaque **t**. En outre, certains nouveaux liens ont été prédits. En d'autres termes, l'objectif est de prédire correctement le graphe d'état **[33].**

- **Prédiction de liens non périodiques**

Dans le type non périodique, au lieu d'avoir une série d'instantanés d'un graphe en pleine évolution, il y a un aperçu de l'état actuel du graphe $\mathbf{g_t}$. Plus formellement, soit $\mathbf{G = (V, E_t)}$, où **V** est l'ensemble des nœuds et **E** désigne ses liens**.** Considérons deux sous-graphes correspondant à l'état actuel, $\mathbf{g_t}$ et l'avenir $\mathbf{g_{t+1}}$

cette notion est définit comme suit : $\mathbf{E}_t \cup \mathbf{E}_{t+1} = \mathbf{E}$, $\mathbf{E}_t \cap \mathbf{E}_{t+1} = \emptyset$. En raison de l'état actuel, il faut chercher à prédire la prochaine étape de temps de graphe $\mathbf{g}_{t+1}$**[34,35]**.

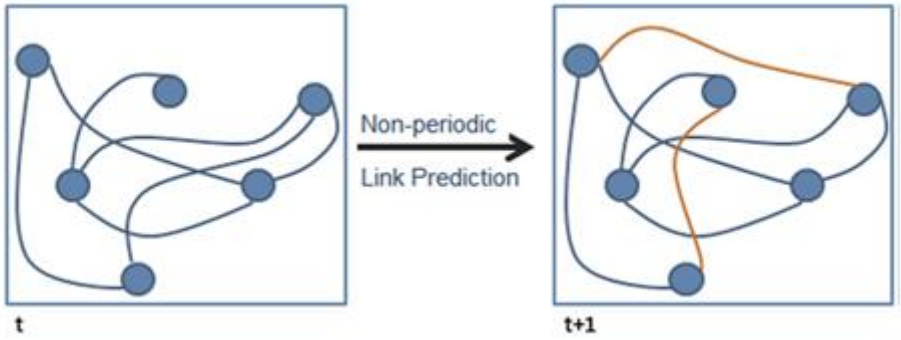


Figure 2.3 prédiction de liens non périodique

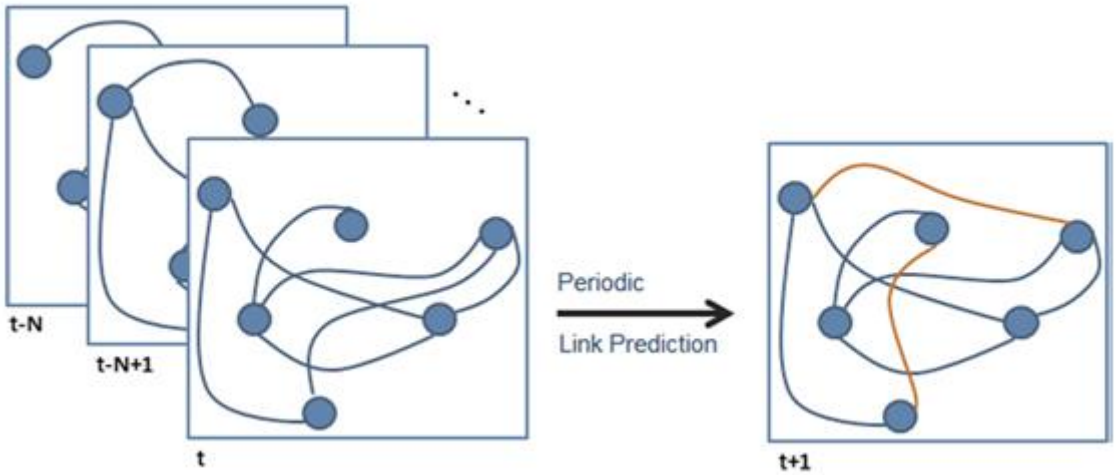


Figure 2.4 prédiction de lien périodique

La Figure 2.4 montre une prédiction de lien périodique, où les entrées sont l'instantané du graphe dans différents intervalles de temps et **la figure 2.3** montre la prédiction de lien non périodique où l'entrée est juste un instantané du graphe courant.

2. Classification des approches de prédiction de liens

Les approches de prédiction de lien peuvent être classées selon deux catégories : les approches basées sur l'apprentissage et des approches heuristiques **Figure 2.1**. Tout d'abord, dans l'approche heuristique, la phase de prédiction se fait directement après avoir déterminée les caractéristiques **[35, 36,35]**. Ce groupe d'algorithmes calculent un score de similarité entre une paire de nœuds **[36,37].**

Le deuxième est à base de modèles d'apprentissage qui extraient un modèle à partir des données d'entrée. Ces données d'entrée peuvent être un vecteur caractéristique prétraité donc ils obtiennent un modèle de données pour prédire les liens.

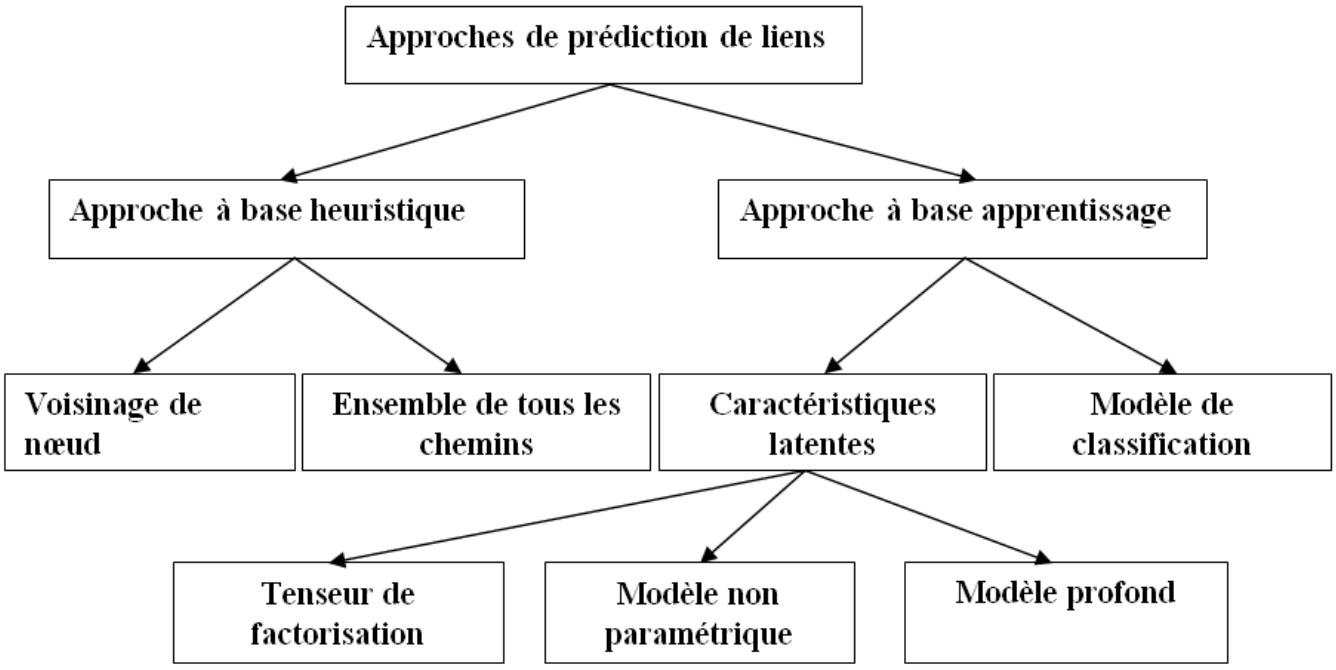


Figure 2.5. Classification des approches de prédiction de lien

Cette décomposition est due au fait que les algorithmes d'apprentissage extraient eux-mêmes un modèle à partir de données afin de prédire les liens futurs par contre d'autres méthodes de prédiction basée sur l'heuristique, prédire les liens grâce à des similitudes connues de la structure du graphe **[56].**

Dans la suite d'étude, chaque approche est décrite en détail. Nous commençons par l'approche heuristique et ces sous-catégories, puis nous traitons l'approche d'apprentissage automatique.

2.2. Approches à base heuristique

Une heuristique est une méthode de calcul qui fournit rapidement une solution réalisable, pas nécessairement optimale ou exacte, pour un problème d'optimisation difficile. C'est un concept utilisé entre autres en optimisation combinatoire, en théories des graphes, en théories de la complexité des algorithmes et en intelligence artificielle.

Cette approche inclut des méthodes qu'ils tentent de prédire les liens via des informations heuristiques**,** cette information capture les caractéristiques partagées ou les contextes de deux nœuds, en raison de ces informations capturées, l'approche heuristique est classifiée en deux classes qui sont : voisinage de nœud et ensemble de tous les chemins **[35, 29, 37,15].**

Les méthodes basées sur le voisinage prennent en compte les méthodes basées sur les indicateurs locaux et les chemins, sont appelées indicateurs globaux **[29,40].**

Ces techniques définissent une fonction **S (x, y)** qui attribue un score appelé similarité à chaque lien non observé pour chaque paire de nœuds **x** et **y**, et les **K** premiers liens avec le score le plus élevé sont prédits **[56,77]**.la fonction de similarité peut varier d'un réseau à l'autre, même du même domaine **[77].**
De plus, ils nous permettent de gérer efficacement le problème de prédiction de liens dans Des réseaux très dynamiques et changeant tels que les réseaux sociaux en ligne **[77].**

La supériorité de ces algorithmes n'est pas une connaissance de domaine nécessaire pour calculer le score de similarité. L'approche heuristique est également connue sous le nom d'approche par similarité **[56].**

2.2.1. Voisinage de nœud

Pour un nœud **x**, prenons **Γ(x)**l'ensemble des voisins de **x** dans un réseau.
Un certain nombre d'approches reposent sur l'idée que deux nœuds **x** et **y** sont plus susceptibles de former un lien à l'avenir si leurs ensembles de voisins **Γ(x)** et **Γ(y)** ont un grand chevauchement. Cette approche suit l'intuition naturelle selon laquelle ces nœuds **x** et **y** représentent des auteurs qui ont de nombreux collègues en commun et qui sont donc plus susceptibles d'entrer en contact eux-mêmes **[35].**
Il est attendu que les nœuds « similaires » soient plus susceptibles d'être un lien prédit.
En raison de la simplicité et du nombre réduit de paramètres, il est utilisé dans les études sur la prédiction de liens**.** Quatre indices de voisinage de nœud populaires sont expliqués ci-dessous : **[56]**

- **Voisins communs (CN) :**

La métrique CN (commun Neighbors) est l'une des mesures les plus répandues utilisées dans le problème de la prédiction de lien principalement en raison de sa simplicité **[78].** Pour deux nœuds **x** et **y**, le **CN** est défini comme le nombre de nœuds avec lesquels **x** et **y** ont une interaction directe **[79].** Un plus grand nombre de voisins communs facilite la création d'un lien entre **x** et **y**. Cette mesure est définie comme la formule suivante :

$$\textbf{Score}_{CN}(x, y) = |\Gamma(x) \cap \Gamma(y)|$$

Où **Γ(x)**et **Γ(y)** désigne l'ensemble voisin de nœuds **x** et **y**, respectivement.

Comme la métrique CN n'est pas normalisée, elle reflète généralement les similitudes relatives entre les paires de nœuds. Utilisation de cette méthode pour calculer la similarité pour toutes les paires possibles résulte en une technique de prédiction de lien local avec **O (vk2 (k + k))) = [O (VK3)]**temps de complexité. Par conséquent, certaines métriques basées sur les voisins considèrent comment normaliser la métrique CN de manière raisonnable **[79].**

- **L'indice de Jaccard (JA) :**

Cette métrique de similarité est principalement utilisée dans la récupération d'informations. Le coefficient Jaccard est un voisin normalisé :

$$\textbf{Score}_{JC}\ (x, y) = \frac{|\Gamma(x) \cap \Gamma(y)|}{\Gamma(x) \cup \Gamma(y)|}$$

En fait, il définit la probabilité qu'un voisin commun d'une paire de nœuds **x** et **y** soit sélectionné si la sélection est faite de manière aléatoire à partir de l'union des ensembles de voisins de **x** et **y**. Cependant, des résultats expérimentaux, **[35]** ont montré que la performance du coefficient de Jaccard est moins bonne que celle du nombre de voisins communs **[56].** La complexité algorithmique de cette méthode est :

$$\textbf{O (vk2 (2k + 2k)) = O (vk3)}$$

- **L'indice Adamique-Adar (AA) :**

Cette mesure de similarité, initialement proposée par LadaAdamic et Eytan Adar**[73],** était destiné à mesurer la similarité entre deux entités en fonction de leurs caractéristiques communes **[73],** chaque poids d'une entité est pénalisé logarithmique par sa fréquence d'apparition **[77].** Si nous prenons les voisins comme caractéristiques, cela peut être écrit ainsi :

$$\textbf{Score AA}\ (x, y) = \sum_{z \in \Gamma(x) \cap \Gamma(y)} \frac{1}{log|\Gamma(z)|}$$

Sur le plan conceptuel, Adamique / Adar **[73]** affine le comptage simple des voisins communs en attribuant plus de poids aux voisins moins connectés **[58].** D'après les résultats enregistrés pour la prédiction de lien existante, Adamique / Adar **[73]** fonctionne mieux que les deux métriques précédentes**.**

On peut facilement voir que cette méthode est une autre variante de la méthode des voisins communs où il existe une pénalisation pour chaque voisin non partagé. La complexité temporelle algorithmique de cette méthode est : **O (vk2 (2k + 2k)) = O (vk3)[77].**

- **L'indice de pièces jointes préférentiel (PA) :**

Cet indice est une conséquence directe du modèle bien connu de formation du réseau complexe **[74] [75].**

De nombreux degrés de nœuds de réseau réels suivent une distribution de puissance, ce qui résulte en une échelle de réseau qui ne pourrait pas être expliquée par des modèles de présentation de réseau précédents. **AlbertLaszloBarabásietRek[74]**a construit un modèle théorique basé sur l'observation que la probabilité de formation de lien entre deux nœuds augmente à mesure que le degré de ces nœuds augmente **[77].**

Cette information permet au concept de « s'enrichir », ce qui génère la distribution des degrés de la loi de puissance observée dans les réseaux sans échelle **[77].** La similarité entre deux nœuds, selon le modèle de Barabasi-Albert **[74],** peut être estimée comme suit :

$$\mathbf{S\ (x,\ y) = |\Gamma(x)\ ||\Gamma(y)|}$$

Cette mesure peut également être appliquée dans des contextes non locaux, car elle ne repose pas sur des voisins partagés. Cependant, la précision de ses prédictions est généralement médiocre lorsqu'elle est appliquée en tant que mesure globale. La complexité de la méthode **O (vk2)** est plus rapide que celle des méthodes basées sur des voisins partagés **[77].**

2.2.2. Ensemble de tous les chemins :

Un certain nombre de méthodes affinent la notion de distance du plus court chemin en considérant implicitement l'ensemble de tous les chemins entre deux nœuds **[35].**

Les chemins entre deux nœuds sont une autre heuristique qui peut être utilisée pour calculer les similitudes entre les paires de nœuds. Une brève introduction des quatre principaux indices mondiaux est donnée ci-dessous **[56].**

- **Katz**

Katz est une méthode déclarée parmi les approches basées sur les chemins qui compte tous les chemins entre deux nœuds **[80].** Les chemins sont exponentiels, amortis par leur longueur, ce qui peut donner plus de poids aux chemins les plus courts **[79].** L'expression mathématique est :

$$\sum_{l=1}^{\infty} \boldsymbol{\beta^l}.\,|\boldsymbol{path_{x,y}^l}| = \boldsymbol{\beta A + \beta^2 A^2 + \beta^3 A^3 + \cdots}$$

Où chemins $path_{x,y}^{l}$ est l'ensemble de tous les chemins de longueur **l** reliant **x** et **y**, et **β** un paramètre libre (c'est-à-dire le facteur d'amortissement) contrôlant les poids de chemin. La complexité de cette méthode est : **O (V_K + V_3 + v) [29].**

- **Temps de frappe**

Pour deux sommets, **x** et **y** dans un graphe, le temps de frappe, $\boldsymbol{H_{x,y}}$ définit le nombre attendu d'étapes requises pour une marche aléatoire commençant à **x** pour atteindre **y**. Un temps de frappe plus court montre que les nœuds sont similaires, ce qui permet de créer des liens. Pour un graphe non dirigé, cela peut être considéré comme **[56]** :

$$\boldsymbol{score_{HTugraph}(x,y) = H_{x,y} + H_{y,x}}$$

Il est facile de calculer la métrique du temps de frappe en effectuant quelques essais aléatoires. À la baisse, sa valeur peut avoir une variance élevée ; par conséquent, la prédiction par cette fonctionnalité peut être mauvaise. En raison de la nature sans échelle d'un réseau social, certains des sommets peuvent avoir une probabilité stationnaire très élevée (π) dans une marche aléatoire ; pour se protéger contre cela, le temps de frappe peut être normalisé en le multipliant par la probabilité stationnaire du nœud respectif, comme indiqué ci-dessous **[56]**

$$\boldsymbol{score_{NHTugraph}(x,y) = H_{x,y,\pi_y} + H_{y,x,\pi_x}}$$

- **Page Rank**

Le score de similarité entre deux sommets **x** et **y** peut être mesuré comme la probabilité stationnaire de **y** dans une marche aléatoire qui revient à **x** avec une probabilité **1-β** dans chaque étape, se déplacer vers un voisin aléatoire avec une probabilité **β** est un Page Rank pour la prédiction de lien **[72].**

$$\mathbf{Score_{RPR}(x, y) = (1-\beta)\,(I-\beta N)^{-1}}$$

- **SimRank :**

SimRank est défini de manière cohérente, en partant de l'hypothèse que deux nœuds sont similaires s'ils sont connectés à des nœuds similaires **[56].**

$$\boldsymbol{score_{SR(x,y)}} = \begin{cases} \mathbf{1} & \boldsymbol{if\ x = y} \\ \dfrac{\sum_{a\in\gamma(x)}\sum_{b\in\gamma(y)} \boldsymbol{simRank(a,b)}}{|\Gamma(x)||\Gamma(y)|} & \\ \boldsymbol{\gamma}. & \boldsymbol{autres} \end{cases}$$

Où **γ ∈ [0,1]** est le vecteur de décomposition. Le SimRank peut également être interprété par la marche aléatoire, c'est $\boldsymbol{score_{SR(x,y)}}$pour mesurer le moment auquel deux marcheurs aléatoires, partant respectivement des nœuds **x** à **y**, devraient se rencontrer à un certain nœud **[29].** Cependant, ce processus d'expansion récursif a une complexité **O (k^{2l})**. Étant donné que deux sommations imbriquées doivent être effectuées, la complexité de chaque paire de nœuds est de **0 (k^{2l+2})**. Ce score doit être calculé pour chaque paire de nœuds de sorte que la complexité temporelle algorithmique finale soit de **O (v^2k^{2l+2}) [77].**

Tableau 2 .1 représente les complexités des indices de similarités inclut dans cette catégorie d'approche.

Nom	**La complexité**
Voisins communs (CN) **[50]**	O (VK^3)
L'indice de Jaccard (JA) **[7, 27]**	O (vk^3).
L'indice Adamique-Adar (AA) **[44]**	O (vk^3).
L'indice de pièces jointes préférentiel (PA) **[45]**	O (vk^2).
Katz KI **[51]**	O(v^3)
Sim Rank **[48]**	O (v^2k^{2l+2})

Tableau 2 .1 Complexité et références pour les méthodes de prédiction de liens basées sur la similarité.

2.3. Approches à base apprentissage

L'apprentissage automatique (en anglais machine Learning) ou apprentissage statistique est un champ d'étude de l'intelligence artificielle qui se base sur des approches statistiques pour donner aux ordinateurs la capacité d'apprendre à partir de données. L'apprentissage automatique comporte généralement deux phases. La première, consiste à estimer un modèle à partir de données appelées observations qui sont disponibles et en nombre fini, lors de la phase de conception du système. L'estimation du modèle consiste à résoudre une tâche pratique telle qu'estimer une densité de probabilité, prédire des liens manquants ou futurs dans un réseau, c'est le cas du problème proposé. Cette phase dite « apprentissage » est généralement réalisée préalablement à l'utilisation pratique du modèle. La seconde phase correspond à la mise en production. Le modèle étant déterminé, de nouvelles données peuvent alors être soumises afin d'obtenir le résultat correspondant à la tâche souhaitée.
A ce niveau d'abstraction, on présente un modèle qui s'apprend avec une caractéristique, extrait du modèle et conduisent finalement à la prédiction de lien **[54].**

Conceptuellement, apprendre les modèles basés sur l'objectif vise à extraire la structure du graphe d'entrée et à prévoir les liens futurs en utilisant le modèle appris **[34]**. Cette approche est classée en deux types : les modèles de classification et les modèles basés sur les caractéristiques latentes. L'idée clé derrière ce regroupement est le type de modèle d'apprentissage.

La classification des modèles extrait un modèle à partir des données d'entrée et l'apprendre pour la prédiction des liens manquants ou futurs. Les données d'entrée sont les vecteurs de caractéristiques de prétraitement où chaque entrée est connue comme index de similarité, nœud externe ou information de lien **[34,78]**. Cependant, dans le modèle basé sur les caractéristiques latentes, ces vecteurs de caractéristiques de prétraitement peuvent être facultatifs. Clairement, le modèle basé sur les caractéristiques latentes, extraire les caractéristiques latentes du graphique d'entrée et apprendre un modèle. Pendant ce temps, ce modèle d'apprentissage peut utiliser des informations externes analogues aux données des réseaux sociaux ou vecteurs de caractéristiques de prétraitement. Les deux éléments suivants décrivent le modèle de classification et le modèle basé sur les caractéristiques latentes.

2.3.1. Modèle de classification

Toutes les méthodes dans cette catégorie sont un apprentissage supervisé. Après avoir identifié l'ensemble des caractéristiques qui sont essentielles à l'apprentissage supervisé **[34,38],** le problème de prédiction de lien est mis en correspondance avec une classification binaire **[39] (Figue 2.2**). Bien que dans la classification binaire, il est important de prévoir les deux classes en prédiction de lien, la question est de prédire la chaine manquante ou un lien futur. Dans un modèle de classification pour la prédiction de lien, les chercheurs ont utilisé le modèle supervisé, y compris la machine à vecteur de support **[40]**, Arbres de

décision **[41],** Perceptron multicouche **[34]**et la marche aléatoire supervisé **[38]**. Ils n'ont constaté que la marche aléatoire est plus performante dans la prédiction des liens futurs dans le modèle supervisé **[42]**. L'attribut de nœud et les caractéristiques topologiques sont principalement obtenus à partir de méthodes à la base heuristiques. En d'autres termes, au lieu de prédire directement le lien, il est utilisé comme une entrée de vecteur caractéristique pour l'apprentissage d'un modèle et d'autres se réfère aux fonctionnalités d'attribut de nœud telles que l'information sur le profil dans les réseaux sociaux. **La Figure 2.3** montre quelques-unes des propriétés du vecteur caractéristique. Bien que, juste avoir un vecteur caractéristique peut prédire un lien avec n'importe quel modèle de classification binaire, créant ainsi un vecteur de caractéristique précieuse a une tâche supplémentaire.

Nous présentons, dans ce qui suit, quelques algorithmes du modèle supervisé :

- **Marche aléatoire supervisé (supervised random walk SRW)**

  La marche aléatoire est un modèle mathématique d'un système possédant une dynamique discrète composée d'une succession de pas aléatoires. Le concept de la marche aléatoire se base qu'à chaque instant. Le futur du système dépend de son état présent, mais pas de son passé. Même le plus proche, il étudie souvent des marches au hasard sur des réseaux réguliers ou sur des graphes plus complexes. C'est l'exemple de la méthode utilisée pour faire la prédiction des liens manquants et futurs, **SRW** qui se décompose en unités élémentaires appelées pas, dont la longueur peut être elle-même constante, aléatoire ou fixée par le réseau ou le graphe sur lequel se circule. A chaque pas, il y a donc un éventail de possibilités pour choisir au hasard la direction et la grandeur du pas. Cet éventail de possibilités peut être discret (choix parmi un nombre fini de valeurs), ou continu.

  Dans le problème de prédiction de lien, le défi consiste à combiner efficacement les informations de la structure de réseau avec des données de nœud et d'attributs des liens qui restent largement ouvert. La marche aléatoire supervisé combine naturellement les informations de la structure du réseau avec les attributs de nœud et de lien, ces attributs sont utilisés pour guider la marche aléatoire dans le graphe, plus la tâche d'apprentissage est formulée dont le but est d'apprendre une fonction qui attribue des avantages aux liens du réseau, de sorte qu'une marche aléatoire a plus de chances de visiter les nœuds vers lesquels de nouveaux liens seront créés à l'avenir **[38]** .

- **Machine à vecteur**

Les machines à vecteurs de support ou séparateurs à vaste marge (en anglais support Vector machine, SVM) sont un ensemble de techniques d'apprentissage supervisé destinées à résoudre des problèmes de discrimination et de régression. Les SVM sont une généralisation des classificateurs linéaires.

Les SVM ont été appliqués à de très nombreux domaines (bioinformatique, recherche d'information, vision par ordinateur, finance, prédiction de liens…). Selon les données, la performance

des machines à vecteurs de support est de même ordre, ou même supérieure, à celle d'un réseau de neurones ou d'un modèle d'un mélange gaussien.

- **Le perceptron multicouche**

Le perceptron multicouche *(*multi layer perceptron MLP) est un type de réseau neuronal formel organisé en plusieurs couches au sein desquelles une information circule de la couche d'entrée vers la couche de sortie uniquement ; il s'agit donc d'un réseau à propagation directe (feedforward). Chaque couche est constituée d'un nombre variable de neurones, les neurones de la dernière couche (dite « de sortie ») étant les sorties du système global.

Dans la première version le perceptron était monocouche et n'avait qu'une seule sortie à laquelle toutes les entrées sont connectées.

- **Arbre de décision**

Un arbre de décision est un outil d'aide à la décision représentant un ensemble de choix sous la forme graphique d'un arbre. Les différentes décisions possibles sont situées aux extrémités des branches (les feuilles de l'arbre), et sont atteints en fonction de décisions prises à chaque étape.

Un avantage majeur des arbres de décision est qu'ils peuvent être calculés automatiquement à partir de bases de données par des algorithmes d'apprentissage supervisé. Ces algorithmes sélectionnent automatiquement les variables discriminantes à partir de données non-structurées et potentiellement volumineuses.

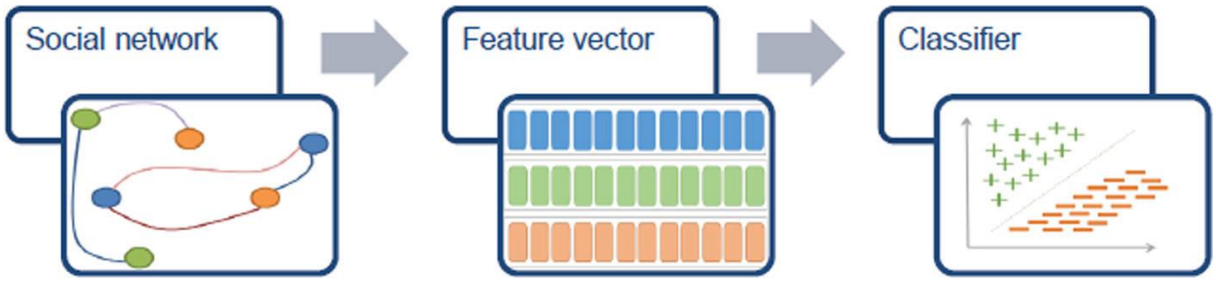


Figure 2.6 modèle de classification

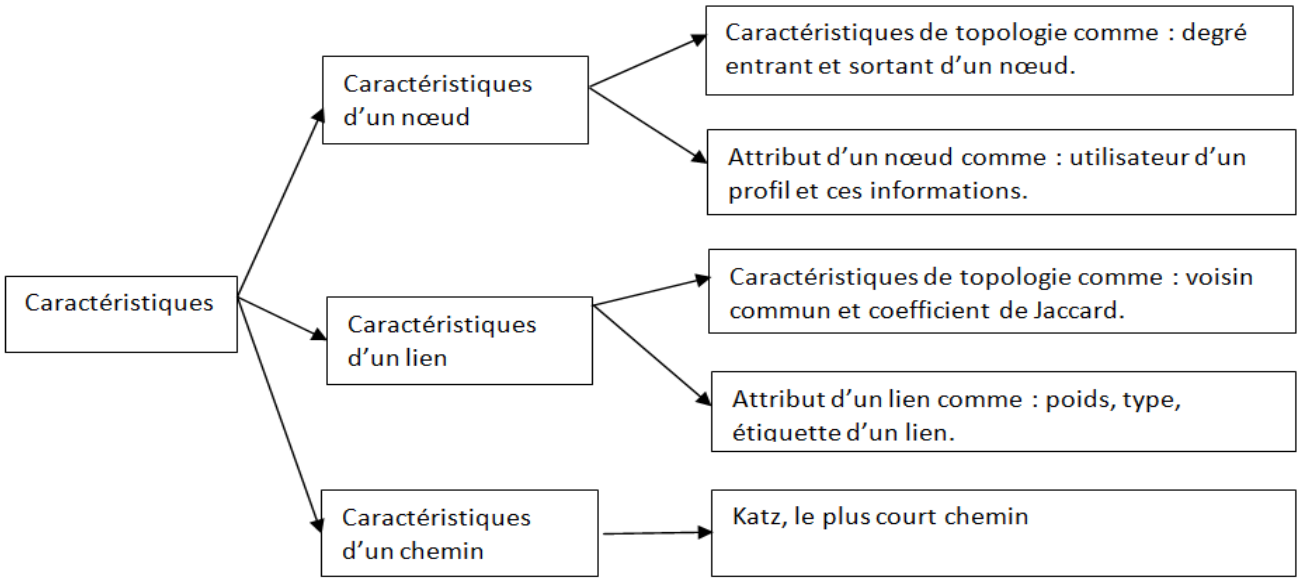


Figure 2.7 regroupement des caractéristiques les plus utilisé

2.3.2. Modèle basé sur les caractéristiques latentes

Une hypothèse du modèle à base de caractéristique latente (caractéristique latente c'est-à-dire qu'il ne peut pas être extrait directement) est de construire un modèle qui peut découvrir des caractéristiques latentes de la structure du graphe **[43, 31,44, 45].** Comme mentionné précédemment, il existe deux types d'informations qui peuvent être obtenues à partir du réseau **(Figure 2.3)**. Néanmoins, les chercheurs dans ce domaine croient que la structure du graphe et la combinaison de ces deux types d'informations ont des caractéristiques latentes, ce qui n'est pas évident, avec les techniques simples d'être atteint, en particulier lors de la capture des caractéristiques dynamiques ou en réseau hétérogène avec les données **[31,46,47]**. L'objectif de l'approche basée sur les caractéristiques latente est d'apprendre un modèle de liens observés tels que ceux qui peuvent prédire les valeurs des entrées non observées. La représentation latente de chaque nœud correspond à un point sur la surface d'une hyper sphère de l'unité. Dans la fonction latente-basedu modèle, chaque entité est associée à un vecteur $\boldsymbol{e_i} \in \mathbf{R}\ \mathbf{H^e}$, où $\mathbf{H_e}$ $\mathbf{N_e}$ ($\mathbf{N_e}$ est le nombre d'entités).

$$e_{ALI}=\begin{matrix}0.1\\0.9\end{matrix} e_{SINA}=\begin{matrix}0.15\\0.58\end{matrix} e_{MOHAMMED}=\begin{matrix}0.1\\0.8\end{matrix}$$

$$e_{SAM}=\begin{matrix}0.9\\0.5\end{matrix} e_{REZA}=\begin{matrix}0.92\\0.35\end{matrix} e_{BORNA}=\begin{matrix}0.82\\0.45\end{matrix}$$

Chaque lien est expliqué par les caractéristiques latentes des entités. Par exemple, comme la **figure 2.4** montrechaque nœud peut être modélisé par l'intermédiaire de vecteurs où le composant $\mathbf{e_{i1}}$correspond à la fonctionnalité latente de développeur compétent et $\mathbf{e_{i2}}$correspondre à être en bonne santé. Ainsi, Ali a une

connexion coéquipier, on peut conclure qu'il est plus sain, ou Sam a une connexion collègue Reza et Borna, de sorte que la fonction latente développeur compétent est plus élevée pour lui. Notez que, contrairement à cet exemple, les caractéristiques latentes qui sont les modèles inférés suivants sont généralement difficiles à interpréter. L'intuition clé derrière les modèles de caractéristiques latentes relationnelles est que les relations entre les entités peuvent être dérivées d'interactions de leurs caractéristiques latentes **[43, 48, 49, 45]**. Cependant, il existe plusieurs façons de modéliser ces interactions, et de nombreuses façons de tirer l'existence d'une relation d'eux. Avec la contrainte de dimensionnalité, la prédiction de lien est efficace dans le temps de calcul et les coûts de stockage **[43, 50, 47]**. De plus, la variation de la dimension de l'espace latent offre la possibilité de définir avec précision le compromis entre le coût de calcul et de la qualité de la solution. La dimension plus élevée conduit à une représentation d'espace latent plus précise de chaque nœud, mais donne également le coût de calcul plus élevée **[47]**. Plusieurs approches ont été présentées pour obtenir ces caractéristiques latentes. Les trois éléments suivants présenteront chaque approche en détail.

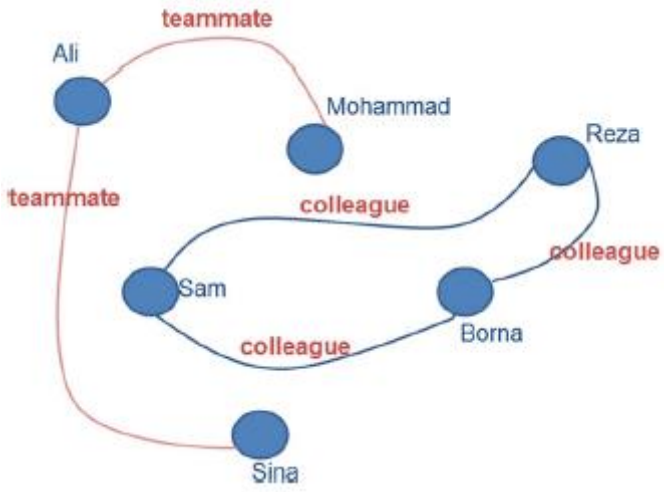


Figure 2.8 Exemples d'un réseau avec des caractéristiques latentes

- ## Tenseur de factorisation

Significativement, tenseur de factorisation est connu comme une approche pour les données structurées dans différents contextes d'apprentissage. Le succès de tenseur de factorisation dans le problème de prédiction de lien est dû à sa grande capacité à modéliser et d'analyser les données relationnelles **[51, 52,53]** méthodes à base de tenseur sont généralement en deux matrices **[54]** et trois ordres. Pour la prédiction de liens futurs, le troisième domaine est considéré comme un instantané horaire différent. Cette approche a une capacité raisonnable pour détecter la fonction latente pendant le temps. Plus formellement, étant donné une séquence de mots :

$$Z\,(i, j, t) = \begin{cases} \mathbf{1}\ \boldsymbol{si\ le\ noeud\ i\ a\ un\ lien\ avec\ le\ noeud\ j} \\ \mathbf{0}\ \boldsymbol{autrement} \end{cases}$$

Ce qui montre que le lien **i** à **j** est apparu au moment **t [56, 57, 51, 58, 47, 59]**. D'autre part, dans des réseaux hétérogènes avec des données multi-relationnelles, la troisième dimension montre différents types de liens. Il est le plus applicable dans des réseaux hétérogènes où les liens ont une forte dépendance **[53, 52, 59, 61, 62]**. Le troisième tenseur d'ordre est utilisé pour définir des données multi-relationnelles comme suit :

$$Z\ (i, j, t) = \begin{cases} 1\ si\ relation_k(noeud\ i, noeud\ j) = vrai \\ 0\ autrement \end{cases}$$

- ## Modèle non paramétrique

L'utilisation du modèle non paramétrique est un type de modèle de fonction latente. Dans ce modèle, les méthodes utilisent principalement des méthodes non paramétriques bayésienne pour découvrir les caractéristiques latentes discriminants et en déduisent automatiquement la dimension sociale inconnue **[69, 71, 47]**. Fondamentalement, chaque entité est décrite par un ensemble de fonctions binaires.

Les modèles non paramétriques permettent infère simultanée du nombre de fonctions latentes en même temps **[44]**. D'autre part, certains modèles sont non paramétriques basées sur un noyau. Ces modèles à base de noyau comprennent la régression du noyau, de calcul des similitudes entre la requête et tous les membres de l'ensemble de la formation **[44, 71]**. **[71**] a introduit un modèle non paramétrique de base. Ils ont montré que chaque entité est décrite par un ensemble de caractéristiques binaires et il n'y a pas de priorité pour chacun d'eux. La probabilité d'avoir un lien à partir d'une entité à une autre est entièrement déterminé par un effet combiné de toutes les interactions de caractéristiques appariées. S'il y a **K** caractéristiques, puis **Z** sera le **N × K** matrice binaire où chaque rangée correspond à une entité et chaque colonne correspond à une caractéristique de telle sorte que $Z_{ik} \equiv Z\ (i, k) = 1$ si la i ^{me} entité a une fonction **k** et $Z_{ik} = 0$ autrement. Le modèle comporte une matrice de poids réel de la valeur **(K × K).** Où $W_{kk'} \equiv W\ (k, k')$ est le poids qui influe sur la probabilité d'occurrence d'un lien à partir d'une entité **i**à**j**si l'entité **i** a une fonction **k** et l'entité **j** a une fonction **k'**.On suppose que les liens sont indépendamment conditionnés sur **Z** et **W,** et que seules les caractéristiques des entités **i etj** influencent la probabilité d'un lien entre ces entités. Cette notion définit la probabilité :

$$Pr\ (Y \mid Z, W) = \prod_{i,j} Pr\ (yi\ j \mid Zi\ Z\ j, W)$$

Où le produit se situe au-dessus de toutes les paires d'entités. Compte tenu de la matrice de fonction **Z** et la matrice de poids **W,** la probabilité qu'il y ait un lien à partir de l'entité je à l'entité j est donnée comme :

$$\Pr\ (y_{ij} = 1 \mid Z, W) = \sigma\left(Z_i\ W\ Z_J\right) = \sigma\ (\sum_{k,k'} Z_{ik} Z_{jK'}\ w_{kk'})$$

Où **σ (.)** est une fonction sigmoïde, qui transforme les valeurs de **(-∞, + ∞)** à **(0, 1)**. Les modèles non paramétriques ont unegrande capacité à explorer les modèles d'évolution particulièrement de fluctuations

saisonnières. Il est à noter que, cette efficacité est juste par rapport aux méthodes basées heuristiques et non à l'autre modèle basé sur la fonction latente **[44, 42,47]**. Parmi les modèles basés sur l'apprentissage, le modèle le plus rapide est non paramétrique. La raison étant qu'il n'a pas ou quelques paramètres. D'autre part, la plupart des méthodes utilisent la mise en œuvre LSH pour les rendre plus rapides. LSH ou localité hachage sensible est souvent utilisé dans la base de données pour les recherches de la table ou la récupération des éléments qui correspondent à la base de données [**44]**. Ce modèle est également connu sous le nom modèle de probabilité **[56].**

- **Modèle profond**

En raison du résultat significatif de l'approche d'apprentissage en profondeur dans la vision par ordinateur, la reconnaissance vocale et le traitement du langage naturel **[82, 84, 85]** Les chercheurs ont été motivés à utiliser un modèle profond dans la tâche de prédiction de lien **[83, 86, 88, 87, 12, 81].** Dans l'apprentissage en général, au plus le modèle profond est un ensemble d'algorithmes d'apprentissage machine qui effectue des tâches d'apprentissage à plusieurs niveaux, correspondant à différents niveaux d'abstraction. Il utilise généralement des réseaux de neurones artificiels. Les niveaux de ces modèles statistiques acquises correspondent à des niveaux distincts de concepts, où les concepts de niveau supérieur sont définis à partir de ceux de niveau inférieur, et les mêmes concepts de niveaux inférieurs peuvent aider à définir plusieurs concepts de niveau supérieur **[82,85].**

2.4. Tableau comparatif des approches :

Nous avons vu quelques méthodes de prédiction de liens d'apprentissage automatique et heuristique. **Le Tableau 2.2** récapitule les avantages et les défis des approches présentées précédemment.

Approche	**Avantage**	**Défis**	**Type**	**Type d'apprentissage**	**La description**
1. Voisinage de nœud	*L'idée de base pour d'autres Méthodes *Il ne nécessite pas des Connaissances spécifiques de domaine. *Indépendamment du type et de	***La non-détection des caractéristiques Latente. *Ne pas reconnaître les modèles d'évolution. *Le défaut dans les	Chaine manquant, les liens futurs.		Elle est basée sur une mesure de similarité, donner un score à un lien de test, plus le score est élevé la chance de prédiction

	La structure du réseau.	données multi-relationnelle.			augmente.
2. Ensemble de tous les chemins	*L'idée de base pour d'autres Méthodes *Il ne nécessite pas des Connaissances spécifiques de domaine. *Dans certain ensemble de données, il donne des résultats raisonnables.	*****Face à un grand réseau est Impuissant. *La non-détection des caractéristiques latentes et multi-relationnelle. *Ne pas reconnaître les Modèles d'évolution.	Chaine manquant, les liens futurs.		
3. Modèle de classification	Juste en ayant un vecteur caractéristique peut prédire avec tout modèle de classification binaire.	*on est besoin d'un vecteur Caractéristique pour l'avenir prévision du lien: Le manque d'attention à la formation de liens au fil du temps.	Chaine manquant, les liens futurs.	Supervisé	Carte à un problème de classification binaire. Il peut utiliser l'indice de proximité que l'entrée du vecteur caractéristique.
4. tenseur de factorisation	*****Résistant bruit. *capacité élevée dans la	*****Le coût élevé de calcul. *Il ne peut pas	Chaine manquant, les liens	Non supervisé	

	représentation des données multi-relationnelle. *Il peut se développer au tenseur d'ordre Supérieur facilement en raison de capturer des informations dynamiques données multi-relationnelles.	détecter la structure non linéaire.	futurs.		
5.modèle non paramétrique	*****Découvrez les caractéristiques latentes Automatiquement. *Capacité à explorer La caractéristique non linéaire.	*complexité. *Dans le réseau vaste et dynamique perdre la Performance.	Chaine manquant, les liens futurs.	Non supervisé, supervisé, semi-supervisé.	Le noyau approprié l'efficacité de la détection caractéristique non linéaire.

6. modèle profond	*Résistant bruit. *Capacité à explorer La caractéristique non linéaire. *La capacité à détecter des	***Dans certains réseaux de neurones en profondeur, il est moins compte tenu d'une structure graphique. *les modèles profonds sont nouveau nés dans la Prédiction de lien manquant.	Chaine manquant, les liens futurs.	Non supervisé	Machine Boltzmann restreinte est un bloc de construction du réseau de neurones profond.

Tableau 2.2 Avantages et défis des modèles de prédiction de lien.

3. Conclusion

Les études comparatives menées par les experts du domaine ont montré l'existence de nombreuses méthodes de prédiction de liens. Cette section a permis de formuler le problème de prédictions des liens, analyser les méthodes existées et les classifié pour pouvoir présenter les approches de prédictions de liens, notamment celles basée sur l'apprentissage automatique objet de notre travail. L'état de l'art, nous a permis de conclure que la technique de la marche aléatoire supervisée (supervisedrandomwalk) qui fait partie de la catégorie du modèle de classification est la plus efficace pour prédire les liens dans des réseaux dynamique. Ceci est due à la combinaison entre la structure du graphe et les caractéristiques de nœud, ce qui n'est pas évident à l'atteindre avec les techniques simples, donc cette approche permis de faire cette combinaison d'information de la structure du graphe, les caractéristiques des nœuds et des liens, et apprendre un modèle de liens observés ceux qui peuvent prédire les liens non observés.

Le prochain chapitre sera consacré à la description détaillée de notre approche et la mise en œuvre de l'algorithme proposé pour la prédiction des liens futurs.

Chapitre 3

La méthode de la marche aléatoire supervisée pour la prédiction de lien

1. Introduction

Nous avons présenté dans le chapitre précédent l'état de l'art sur lequel s'est appuyé notre travail. Les approches étudiées sont soit des approches à base heuristique soit à base apprentissage automatique.

Nous présentons dans ce chapitre notre approche basée sur le concept de marche aléatoire supervisée (supervisedrandomwalk) qui associe naturellement et de manière raisonnée la structure du réseau avec les caractéristiques (attributs, fonctionnalités) des nœuds et des liens du réseau selon un algorithme unifié de prédiction de lien.

2. Schéma et description général de notre approche

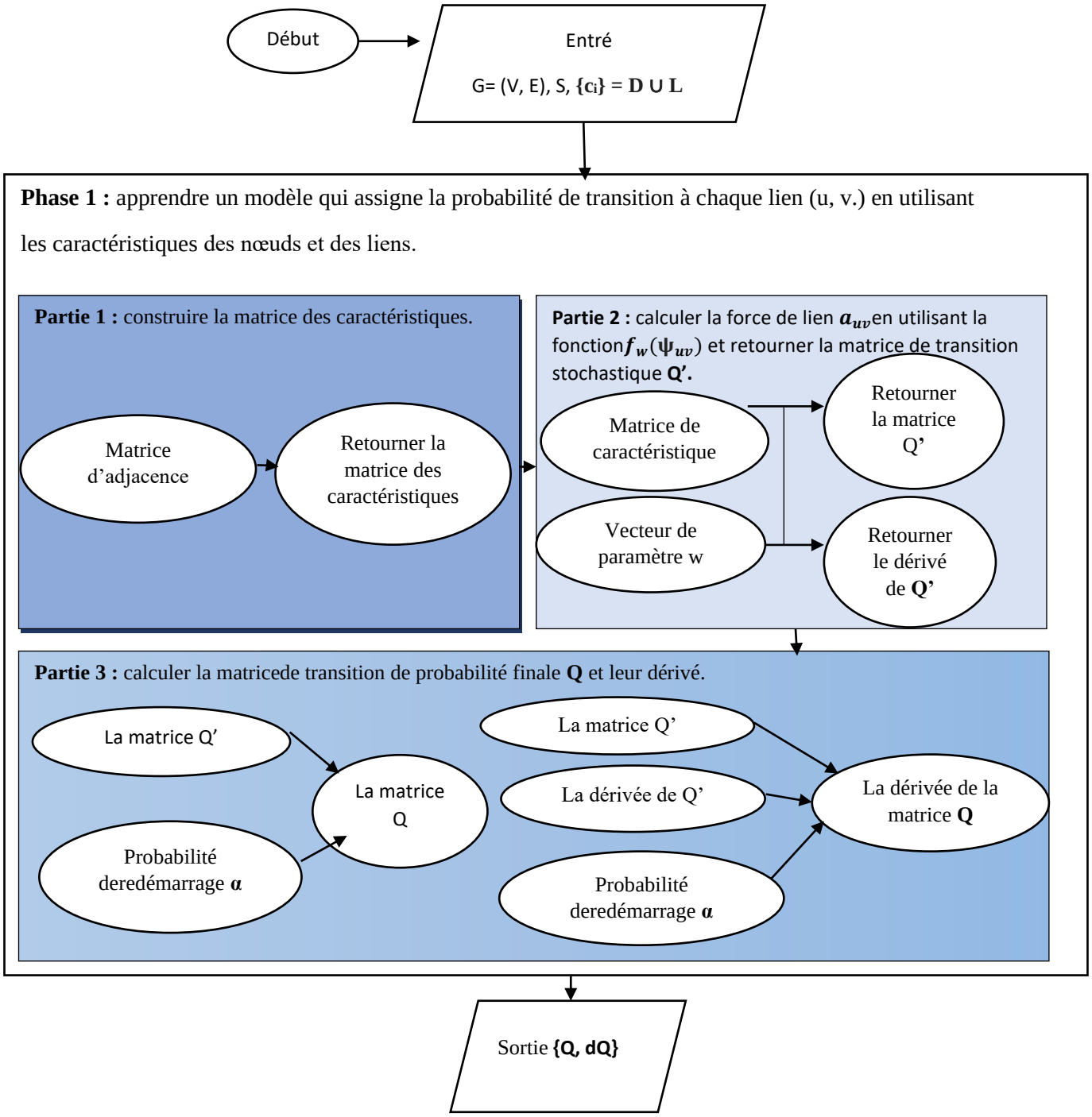
Pour combiner efficacement les informations de la structure de réseau avec des données de nœud et de liens, nous avons implémenté un algorithme basé sur la marche aléatoire supervisée qui combine les informations de la structure du réseau avec les attributs de nœud. Nous y parvenons en utilisant ces attributs pour guider une marche aléatoire sur le graphique. Nous formulons une tâche d'apprentissage supervisé dont l'objectif est d'apprendre une fonction qui attribue des forces aux liens du réseau, de sorte qu'une marche aléatoire a plus de chances de visiter les nœuds vers lesquels de nouveaux liens seront créés à l'avenir. Nous avons implémenté un algorithme efficace pour apprendre directement la fonction d'estimation de la force des liens. Cela peut se réaliser en suivant les taches suivantes :

Etape 1 : apprendre un modèle qui assigne la probabilité de transition à chaque lien $(\boldsymbol{u}, \boldsymbol{v})$ en utilisant les caractéristiques des nœuds et des liens.

Etape 2 : prédiction des liens futurs en minimisant l'ensemble de paramètre **w** de la fonction $\boldsymbol{f_w}$utilisé pour calculer la force des liens existés.

Dans l'organigramme suivant on a résumé toutes les étapes de notre algorithme basé sur la marche aléatoire supervisé.

2.1. Schéma général descriptif de notre méthode (SRW)



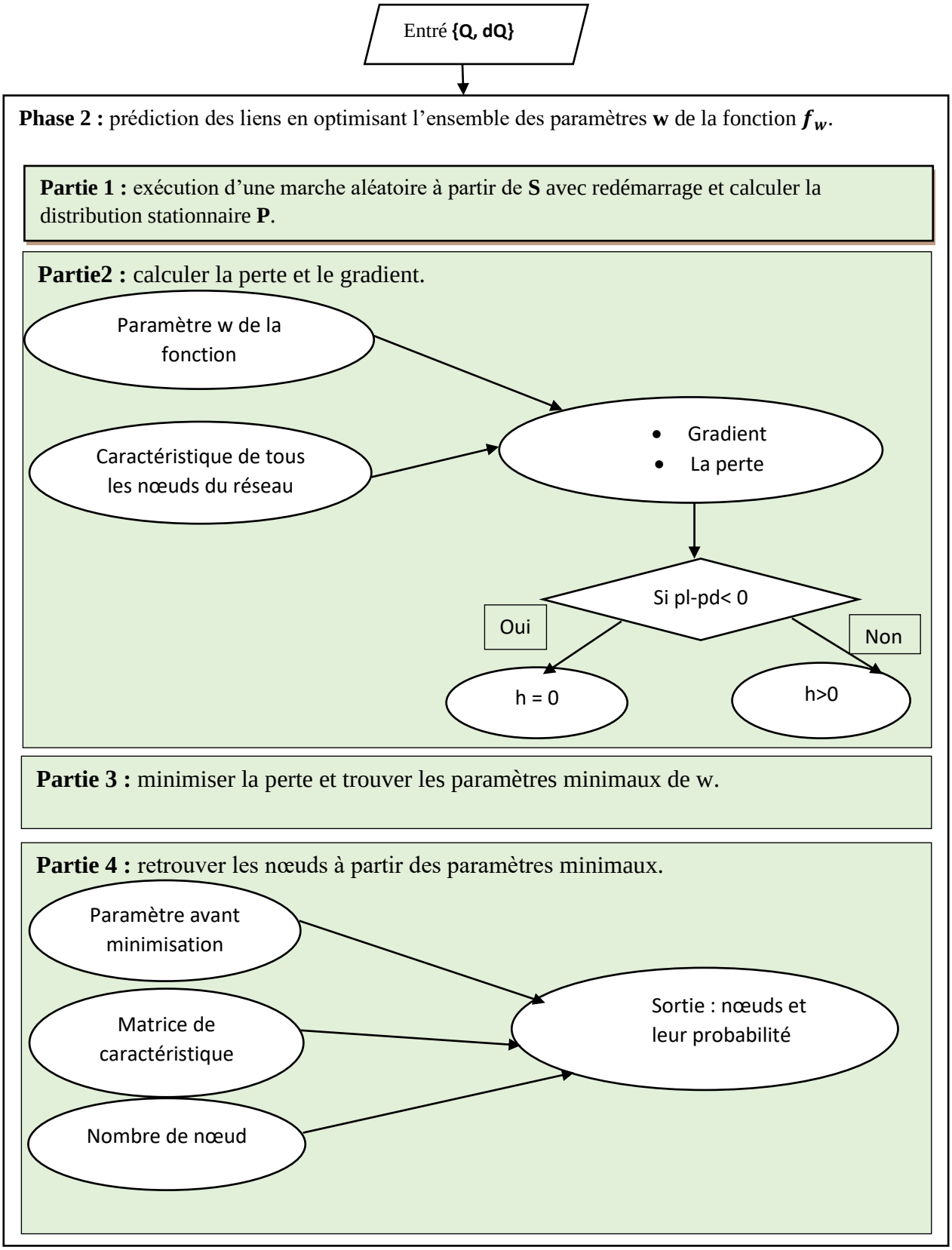


Figure 3.1 organigramme de notre approche de prédiction de lien

2.2. Description de la phase 1

Cette partie consiste à déterminer à chaque lien **(u, v)** la probabilité de transition en utilisant les caractéristiques des nœuds et des liens (on prend comme caractéristique voisin d'un nœud).

Tout d'abord, on a un graphe (orienté ou non orienté) **G (V, E)**,**S** et **C={D∪L}**comme entrée tel que **V** est l'ensemble des nœuds et **E** est l'ensemble des liens. Chaque nœud et chaque lien de **G** est décrit plus en détail avec un ensemble de caractéristiques. Le nœud **S** est le nœud de départ (ou un ensemble de nœuds). Nous étiquetons les nœuds auxquels **S** crée ultérieurement des liens en tant que nœuds de destination **D** =$\{ \boldsymbol{d_l}, \ldots, \boldsymbol{d_k}\}$, alors que nous appelons d'autres nœuds auxquels **S** ne crée pas de lien ultérieurement **L** =$\{\mathbf{l1}, \ldots, \mathbf{ln}\}$. Nous considérons les nœuds dans **D** comme positifs **(**les nœuds positifs sont des nœuds auxquels de nouveaux liens seront créées dans l'avenir**)** et les nœuds dans **L** comme des négative (les nœuds négatifs sont des nœuds auxquels ne créés pas des nouveaux liens à l'avenir). On appelle l'union des nœuds dans **D** et **L** des nœuds candidate avec un ensemble **C = {ci} = D ∪ L**.

Le réseau (graphe G non pondéré) est représenté par une matrice d'adjacence $\boldsymbol{A_{ij}} = \in \{\mathbf{0}, \mathbf{1}\}$(1 s'il existe un lien sinon 0 pas de lien). Elle est utilisée comme entrée dans notre méthode pour extraire la matrice de caractéristique. Nous supposons que chaque lien **(u, v)** a un vecteur de caractéristiques $\boldsymbol{\psi_{uv}}$correspondant qui décrit les nœuds **u** et **v** (par exemple, âge, sexe, ville natale). Et les attributs d'interaction (par exemple, quand lien a été créé, combien de messages échangés entre **u** et **v**, ou combien de photos où ils sont apparus ensemble).

Ensuite, pour chaque lien **(u, v)** dans **G**, nous calculons la force $\boldsymbol{a_{uv}}$ **=** $\boldsymbol{f_w}$ **(**$\boldsymbol{\psi_{uv}}$**)[38]**. La fonction $\boldsymbol{f_w}$ paramétrée par **w** prend le vecteur de caractéristique de $\boldsymbol{\psi_{uv}}$ en entrée et calcule la force de lien correspondante $\boldsymbol{a_{uv}}$que modélise la probabilité de transition de marche aléatoire de nœud vers l'autre. Ce processus est appliqué pour tous les liens jusqu'à l'obtention des forces des liens existés dans le graphe et construire par la suite la matrice de transition stochastique **Q'**, tel que

$$\boldsymbol{Q'_{uv}} = \begin{cases} \frac{\boldsymbol{a_{uv}}}{\sum_w \boldsymbol{a_{uv}}} & \boldsymbol{if}\ (\boldsymbol{u}, \boldsymbol{v}) \in \boldsymbol{E} \\ \mathbf{0} & \boldsymbol{otherwise} \end{cases} \ldots\ldots\ldots\ldots\ldots \textbf{(1)[38].}$$

Pour obtenir la matrice de probabilité de transition de marche aléatoire finale Q, nous incorporons également la probabilité de redémarrage alpha, c'est-à-dire la probabilité avec laquelle la marche aléatoire revient en arrière au nœud de départ**S**, et donc « redémarre » ainsi :

$$\boldsymbol{Q_{uv}} \textbf{= (1 − α)} \boldsymbol{Q'_{uv}} \textbf{+ α1(v = s)} \ldots\ldots\ldots\ldots\ldots\ldots\ldots \textbf{(2)[38].}$$

2.3. Description de la phase 2

Après avoir déterminé les forces des liens dans la partie précédente, la 2éme partie a pour but de prédire des liens en minimisant l'ensemble des paramètres w de la fonction $\boldsymbol{f_w}$ **(**$\boldsymbol{\psi_{uv}}$**)**.

Pour prédire de nouveaux liens de nœuds, la première force de tous les liens est calculée à l'aide de la commande $\boldsymbol{f_w}$ **(**$\boldsymbol{\psi_{uv}}$**)** mentionné dans la partie **1**. Ensuite, une marche aléatoire avec redémarrages est exécutée à partir de **s** le nœud de départ, elle est plus susceptible de traverser les liens de forte résistance et donc les nœuds connectés aux nœuds via des chemins de liens forts seront probablement visités par la marche aléatoire et auront donc un rang supérieur, cette dernière affecte à chaqu'un de ces nœud une probabilité $\boldsymbol{p_u}$(la distribution stationnaire de la marche aléatoire). Les nœuds sont classés par $\boldsymbol{p_u}$ et les nœuds les mieux classés sont ensuite prédits comme destinations des futurs liens.

Les données d'entrée contiennent des informations selon lesquelles le nœud source **s** créera des liens pour les nœuds **d ∈ D** et non pour les nœuds **l ∈L**. Donc, nous essayons de définir les paramètres **w** de la fonction $\boldsymbol{f_w}$ **(**$\boldsymbol{\psi_{uv}}$**)** pour qu'il attribue des poids aux liens $\boldsymbol{a_{uv}}$ de telle sorte que la marche aléatoire aura plus de chances de visiter les nœuds en **D** que **L**, c'est-à-dire. $\boldsymbol{pl < pd}$, pour chaque **d ∈ D** et **l ∈L**. Ainsi, nous définissons le problème de minimisation pour trouver l'ensemble minimal des paramètres **w** de la fonction de résistance du lien $\boldsymbol{f_w}$ **(**$\boldsymbol{\psi_{uv}}$**)** comme suit :

$$\mathbf{min}\, F(w) = ||w||^2 + \gamma \sum_{\mathbf{d}\in\mathbf{D},\mathbf{l}\in\mathbf{L}} \mathbf{h}(\mathbf{pl} - \mathbf{pd}) \ldots\ldots\ldots \mathbf{(3)[38].}$$

Où **p** est le vecteur des scores PageRank. Notez que les scores PageRank pi dépendent de la force des liens $\boldsymbol{a_{uv}}$et dépendent donc réellement de $\boldsymbol{f_w}$ **(**$\boldsymbol{\psi_{uv}}$**)** paramétré par **w**, **λ** est le paramètre de régularisation qui fait la différence entre la complexité (c'est-à-dire la norme de **w**) et l'ajustement du modèle (c'est-à-dire l'ampleur de la violation des contraintes). De plus, **h (·)** est une fonction de perte qui attribue une pénalité non négative en fonction de la différence des scores. Si$\boldsymbol{pl - pd} < 0$ alors **h (·) = 0** comme $\boldsymbol{pl} < pd$ et que la contrainte n'est pas violée, alors que pour $\boldsymbol{pl - pd} > 0$, **h (·)> 0** également.

Notre objectif est maintenant de minimiser l'équation **(3)** par rapport au vecteur de paramètre **w** pour minimiser **w** qu'on va l'utiliser pour trouver les nœuds **D**. Nous abordons cela en déduisant d'abord le gradient de **F (w)** par rapport à **w** qui est calculé de manière itérative, puis on a choisir d'utiliser une méthode d'optimisation basée sur un gradient pour trouver **w** qui minimise **F (w)**directement nommé **Fminunc**(Find minimum of unconstrained multivariable function)qui prend en paramètre la fonction objective (dans notre cas c'est l'équation (3)), le temps **t** et le vecteur de paramètre et retourner la perte minimal avec le vecteur de paramètre **w** minimal.

Finalement, après avoir déterminé le vecteur de paramètre **w** minimal, on va retourner les nœuds **D** dont lesquelles ils vont faire des liens à l'avenir.

2.4. Algorithme de notre approche

Algorithme :

Entrée : un graphe G= (V, E), S, Ci= {L∪ D}

1 : **A répéter jusqu'à l'obtention du paramètre w optimal et les nœuds D**

2 : **Etape 1** : apprendre un modèle qui assigne la probabilité de transition à chaque lien (u, v.) en utilisant

Les caractéristiques des nœuds et des liens.

3 : **1)** extraire la matrice des caractéristiques à partir de la matrice d'adjacence.

4 :**2)** Pour chaque lien (u.v) dans G

- ✓ Calculer la force $\boldsymbol{a_{uv} = f_w (\psi_{uv})}$, tel que $\boldsymbol{f_w}$est une fonction paramétrée par **w** prend En entrée le vecteur de caractéristique $\boldsymbol{\psi_{uv}}$ **.**
- ✓ Répéter le processus jusqu'à trouver toutes les forces des liens existant dans le graphe qui sont représentés par une matrice de transition stochastique**Q'.**

5:**3)** calculer la matrice de transition de probabilité finale **Q**

- ✓ En incorporant la matrice Q' et la probabilité de redémarrage alpha, c'est-à-dire la probabilité

avec laquelle la marche aléatoire revient en arrière au nœud de départ **S**, et donc « redémarre »

6 : **Etape 2** : prédiction des liens en minimisant l'ensemble des paramètres **w** de la fonction $\boldsymbol{f_w}$

7 :**1)** exécution d'une marche aléatoire à partir de **S** avec redémarrage et calculer la distribution stationnaire **P**.

8 :**2)** calculerlaperteet le gradient en utilisant le paramètre w les caractéristiques des nœuds et des liens.

9 :**3)** minimiser la perte et trouver les paramètres minimaux de **w**.

10:**4)** retrouver les nœuds**D** à partir des paramètres minimaux.

Algorithme : Algorithme de l'approche (supervisedrandomwalks)

3. Mis en œuvre de notre approche sur un graphe

3.1. Environnement de travail

- **GEPHI**

Est un logiciel libre d'analyse et de visualisation de réseaux, développé en java et fondé sur la plateforme NetBeans. Initialement développé par des étudiants de l'université de technologie de Compiègne (UTC), en France. Sa dernière version majeure, 0.9.0, a été lancée en décembre 2015, avec une mise à jour 0.9.1 en février 2016 et 0.9.2 en septembre 2017**[101].**

C'est un logiciel de génération des graphes aléatoires, dynamique qui montre toutes propriétés nécessaires : densité, degrés …etc.

- **MATLAB**

C'est un langage de programmation de quatrième génération émulé par un environnement de développement du même nom ; il est utilisé à des fins de calcul numérique. Développé par la société the Math Works, MATLAB permet de manipuler des matrices, d'afficher des courbes et des données, de mettre en œuvre des algorithmes, de créer des interfaces utilisateurs, et peut s'interfacer avec d'autres langages comme le C, C++ et JAVA**[102].**

3.2. Application de l'algorithme sur un graphe généré

On a généré un graphe non orienté et non pondéré de 8 nœuds et 8 liens pour appliquer notre méthode décrite auparavant et présenter les différentes étapes de l'algorithme.

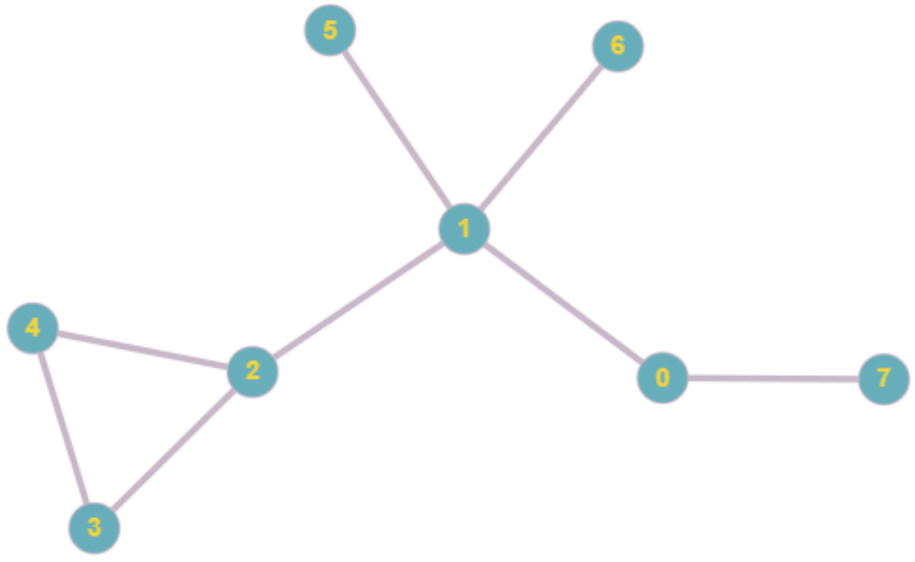


Figure 3.2 un graphe non orienté non pondéré

3.2.1. Représentation du graphe par une matrice d'adjacence :

Dans notre exemple on a un graphe non orienté, sa représentation matricielle est comme suit :

```
adj =

     0     1     0     0     0     0     0     1
     1     0     1     0     0     1     1     0
     0     1     0     1     1     0     0     0
     0     0     1     0     1     0     0     0
     0     0     1     1     0     0     0     0
     0     1     0     0     0     0     0     0
     0     1     0     0     0     0     0     0
     1     0     0     0     0     0     0     0
```

Figure 3.3 matrices d'adjacence de graphe généré.

S'il existe un lien entre deux nœuds, la valeur est égale à 1 sinon 0.

3.2.2. Représentation la matrice de caractéristique

Extraire la matrice de caractéristique à partir de la matrice d'adjacence, cette matrice attribue à chaque lien existé une valeur qui est considéré comme une caractéristique de ce lien, ces valeurs obtenues à partir des valeurs de la matrice d'adjacence.

Par exemple le lien entre les nœuds 2 et 1 est caractérisé par la valeur 0.0667 et le lien entre les nœuds 3 et 2 est caractérisé par la valeur 0.0333.

```
         0         0         0         0         0         0         0         0
    0.0667         0         0         0         0         0         0         0
         0    0.0333         0         0         0         0         0         0
         0         0    0.1667         0         0         0         0         0
         0         0    0.1667    0.2000         0         0         0         0
         0    0.1000         0         0         0         0         0         0
         0    0.1000         0         0         0         0         0         0
    0.1667         0         0         0         0         0         0         0
```

Figure 3.4 matrice de caractéristique de graphe généré

3.2.3. Calcul des forces des liens en les représentant dans la matrice Q'

Le calcul des forces des liens se fait en utilisant la matrice des caractéristiques et le vecteur **W**.

On a le paramètre **W** de la fonction de force, **W = rand (1, m)** tel que **m** est le nombre de caractéristique et **m = length (W),** donc de début on va générer le paramètre **W** avec la matrice de caractéristique, on peut calculer la force de toutes les liens existés dans le graphe en prennent en considération le nombre des nœuds **n** dans le graphe **G**.

0.5000	0.5692	0.5000	0.5000	0.5000	0.5000	0.5000	0.5692
0.6045	0.5000	0.5692	0.5000	0.5000	0.5692	0.5692	0.5000
0.5000	0.6002	0.5000	0.5692	0.5692	0.5000	0.5000	0.5000
0.5000	0.5000	0.6240	0.5000	0.5692	0.5000	0.5000	0.5000
0.5000	0.5000	0.6240	0.6282	0.5000	0.5000	0.5000	0.5000
0.5000	0.6089	0.5000	0.5000	0.5000	0.5000	0.5000	0.5000
0.5000	0.6089	0.5000	0.5000	0.5000	0.5000	0.5000	0.5000
0.6175	0.5000	0.5000	0.5000	0.5000	0.5000	0.5000	0.5000

Figure 3.5 matrice des forces des liens.

3.2.4. Calcul de la matrice final Q

Construire la matrice **Q** final se fait à partir de la matrice **Q'** et la probabilité de redémarrage **α.**

Alpha est la probabilité de redémarrage de la marche aléatoire de déplacement d'un nœud vers un autre.

1-**α**est la probabilité de revenir au nœud source S pour redémarrer le processus de la marche aléatoire.

Pour **α= 0.2,** le résultat obtenu est illustré dans **la figure 3.6**.

0.2967	0.1100	0.0967	0.0967	0.0967	0.0967	0.0967	0.1100
0.3122	0.0928	0.1056	0.0928	0.0928	0.1056	0.1056	0.0928
0.2944	0.1133	0.0944	0.1074	0.1074	0.0944	0.0944	0.0944
0.2954	0.0954	0.1190	0.0954	0.1086	0.0954	0.0954	0.0954
0.2941	0.0941	0.1174	0.1182	0.0941	0.0941	0.0941	0.0941
0.2974	0.1185	0.0974	0.0974	0.0974	0.0974	0.0974	0.0974
0.2974	0.1185	0.0974	0.0974	0.0974	0.0974	0.0974	0.0974
0.3200	0.0971	0.0971	0.0971	0.0971	0.0971	0.0971	0.0971

Figure 3.6 la matrice finale

3.2.5. Calcul de la distribution stationnaire P de chaque nœud

A partir de la matrice **Q** on peut calculer la distribution stationnaire **P** de chaque nœud lors de l'exécution de la marche aléatoire d'un nœud vers l'autre en choisissant les liens qui possèdent les plus grandes valeurs de force.

0.125 0.125 0.180 0.200 0.220 0.125 0.125 0.125

3.2.6. Calcul de la perte et le gradient

Pour calculer la perte et le gradient, on utilise une fonction qui prend comme paramètre le vecteur **W**, la matrice de caractéristique, la probabilité de redémarrage **α**, la probabilité de régularisation **λ**, le vecteur **d** (vecteur des nœuds de destination futur) et le paramètre **b= 0.4**. On obtient le résultat suivant :

La perte =33.2935.

Gradient :

```
0.5570
1.0938
1.9150
```

3.2.7. Minimisation de la fonction de perte et calcul des paramètres optimaux de W

Pour minimiser la perte et calculer le vecteur des paramètres W minimale, nous avons utilisé la fonction **fminunc**qui minimise directement la perte et le vecteur W.

La perte et **W** sont minimisés

La perte =32

Wminimisé= 0 0 0.

3.2.8. Prédiction des lins futurs

A partir des paramètres minimaux **W** on a pu prédire les nœuds de destination **d**qui peuvent faire des liens au futur. Le résultat est représenté dans **la figure 3.6** sous forme d'un graphe.

Les nouveaux liens sont : (4, 5) (5, 6) (6, 7).

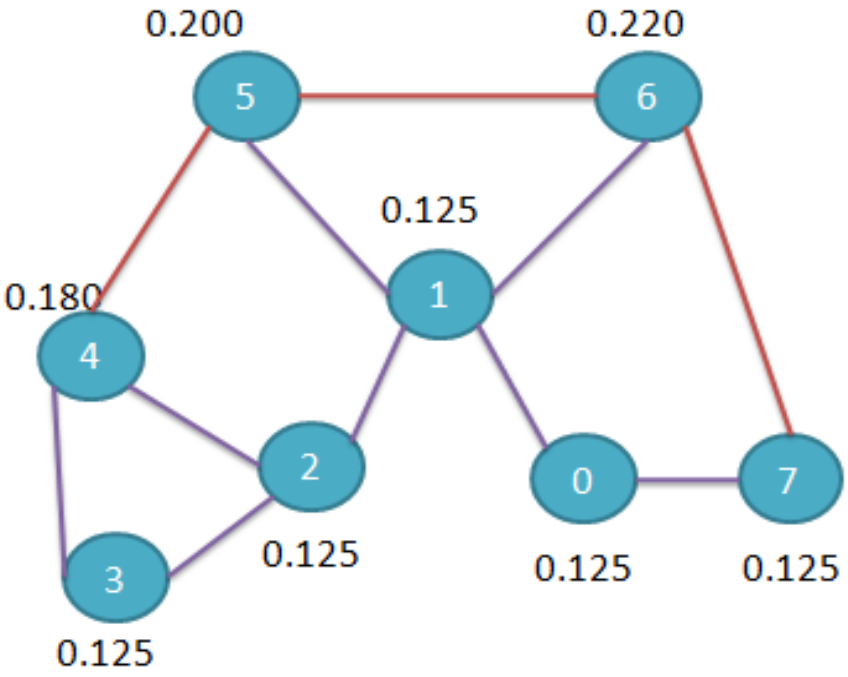


Figure 3.7 le graphe après la prédiction des liens

La figure 3.7montre le graphe après la prédiction des nouveaux liens futurs, on remarque l'existence de trois nouveaux liens après l'exécution de l'algorithme, on a construit la nouvelle matrice d'adjacence et générer le nouveau graphe.

0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0
0	0	0	0	1	0	0	0
0	0	0	0	0	1	0	0
0	0	0	0	0	0	1	0
0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0

Figure 3.8 matrice d'adjacence après la prédiction

4. Conclusion

Dans ce chapitre, nous avons présenté une approche de prédiction de lien dans les réseaux sociaux. La méthode proposée contient deux phases, dans la première phase, on a utilisé un modèle qui assigne la probabilité de transition à chaque lien (u, v) en utilisant la fonction de résistance paramétré par W, la deuxième phase consiste à prédire les liens futurs en minimisant le vecteur de paramètre W.

Nous avons montré les différentes étapes de l'approche proposée sur un exemple de réseau généré par le logiciel Gephi. Dans le chapitre suivant, on va tester notre approche sur un ensemble de réseaux réels et des réseaux synthétiques et évalué les résultats obtenus.

Chapitre 4

Expérimentation et évaluation des résultats

1. Introduction

Pour tester la fiabilité de notre algorithme de prédiction de liens, nous avons effectué plusieurs tests sur des réseaux artificiels et sur des réels connus ensuite nous avons évalué les résultats obtenus.

2. Application de l'approche proposée sur les réseaux synthétiques

Nous utilisons le logiciel Gephi pour la génération de réseaux artificiels. Nous calculons les meilleures valeurs des paramètres de l'algorithme proposé qui donnent une meilleure prédiction de liens. Nous avons choisi de générer des graphes aléatoires non orienté, sur ces graphes on va appliquer la méthode proposée avec les paramètres suivants :

- Le nombre des nœuds **n.**
- Le vecteur de paramètre **w= [1,-1]** de la fonction de résistance qui calcul la force $\boldsymbol{\alpha_{u,v}}$des liens.
- Le paramètre de redémarrage **α** pour la marche aléatoire
- Le paramètre de régularisation$\boldsymbol{\lambda}$**.**
- La fonction de perte **h ().**

Afin de trouver les meilleures valeurs du paramètre de redémarrage **α** de la marche aléatoire qui sert à donner le paramètre **w** de la fonction de résistance avec une meilleure minimisation, nous avons généré des différents graphes artificiels dans les 3 catégories suivantes : graphes faiblement denses (GFD), moyennement dense (GMD), fortement denses (GRD) qui sont résumés dans **le tableau 4.1** et le résultat de cette expérimentation est présenté dans **le tableau 4.2.**

Catégorie de graphe	Nombre de nœud	Nombre de liens	Densité (D)	Probabilité de connexion	Intervalle de densité
Faiblement Dense (GFD)	10	11	0.244	0.25	$0 < D \leq \frac{1}{3}$
	50	270	0.220		
	77	**772**	0.264		
Moyennement dense (GMD)	15	41	0.390	0.7	$\frac{1}{3} < D \leq \frac{2}{3}$
	34	234	0.417		
	70	976	0.404		
Fortement dense (GRD)	16	93	0.775	0.75	$D \geq \frac{2}{3}$
	40	596	0.764		
	400	59911	0.751		

Tableau 4.1 présentation des différentes catégories de graphes utilisés

λ	A	Faiblement dense (GFD)			Moyennement dense (GMD)			Fortement dense (GRD)		
		N=10	**N=50**	**N=77**	**N=15**	**N=34**	**N=70**	**N=16**	**N=40**	**N=400**
1	0	-0.03	-0.03	-0.03	0.043	0.043	0.043	0.30	0.30	0.32
1	0.1	-0.04	-0.03	-0.03	0.043	0.043	0.043	0.03	0.03	0.03
1	0.2	0	0	0	0	0	0	0	0	0
1	**0.3**	**-0.18**	**-0.18**	**-0.18**	0.057	0.18	0.057	-0.20	-0.22	-0.23
1	0.4	0	0	0	0.037	0.12	0.037	0.04	0.03	0.03
1	0.5	0.055	0.055	0.055	0	0	0	0.05	0.04	0.05
1	**0.6**	-0.11	-0.11	-0.11	0	0	0	**-0.30**	**-0.33**	**-0.33**
1	**0.7**	0	0	0	**-0.08**	**-0.08**	**-0.08**	0	0	0
1	0.8	0	0	0	0	0	0	0	0	0
1	0.9	0	0	0	0	0	0	0	0	0
1	1	-0.16	-0.16	-0.16	0	0	0	0	0	0

Tableau 4.2 minimisation du paramètre **w** obtenu par la variation du paramètre **α** dans les trois catégories de graphe.

Le tableau 4.2 montre la variation du paramètre $\boldsymbol{\alpha}\ \boldsymbol{\epsilon}[\mathbf{0},\mathbf{1}]$sur les 3 catégories des graphes artificiels faiblement dense, moyennement dense, fortement dense pour trouver le paramètre **w** le plus minimisé pour chaque catégorie.

Afin de faire cette expérimentation, nous avons utilisé le logiciel Gephi pour générer les graphes artificiels.

D'abord, nous avons générer trois graphes aléatoires non orienté pour chaque catégorie qui sont présentés dans le **tableau 4.1**. On a fixé un seuil pour catégoriser les graphes selon leur densité tel que les graphes faiblement denses ont une densité $\mathbf{0} < D \leq \frac{\mathbf{1}}{\mathbf{3}}$ avec une probabilité de connexion **0.25** Et les graphes moyennement denses ont une densité $\frac{\mathbf{1}}{\mathbf{3}} < D \leq \frac{\mathbf{2}}{\mathbf{3}}$ avec une probabilité de connexion **0.4** et les graphes fortement denses ont une densité $\mathbf{D} \geq \frac{\mathbf{2}}{\mathbf{3}}$avec une probabilité de connexion **0.75**.

Ensuite, nous avons construire la matrice d'adjacence pour chaque graphe pour pouvoir appliquer par la suite la méthode SRW qui prend en entrer le paramètre de régularisation$\boldsymbol{\lambda}$, dans notre modèle le sur-ajustement n'est pas un problème car le nombre de paramètre **w** est petit donc $\boldsymbol{\lambda} = \mathbf{1}$donne les meilleures performances.

Nous avons testé les différentes valeurs de paramètre de redémarrage $\boldsymbol{\alpha\epsilon}[\mathbf{0},\mathbf{1}]$tel que **α=**0, **α=** 0.1, **α** =0.2, **α** =0.3, **α** =0.4, **α** =0.5, **α** =0.6, **α** =0.7, **α=** 0.8, **α=** 0.9, **α** =1 sur les 3 catégories de graphes faiblement dense, moyennement dense et fortement dense :

Graphe faiblement dense (GFD) :

En utilisant les matrices d'adjacence des graphes de 10 nœuds, 50 nœuds et 77 nœuds, on va appliquer la méthode SRW sur ces graphes en générant comme entrer une valeur aléatoire de paramètre **w = rand (1, m)**tel que **m** est le nombre de caractéristique, dans notre modèle on a choisi **m=3,** donc on va avoir un vecteur de paramètre **w = rand (1, m)= [0.82, 0.75, 0.79].**On va minimiser ce vecteur de paramètre en utilisant la fonction de minimisation directe **fminuc** à chaque valeur de **α** et pour que le résultat soit plus clair on a fait la moyenne de ce vecteur et le présenté dans **le tableau 4.2 ,** on peut remarquer que si **α=0.5,** le paramètre **w >0** et si **α=0.2, α=0.4, α=0.7,α=0.8, α=0.9** le paramètre **w = 0**et pour **α=0.6** le paramètre **w** < 0 et la meilleure minimisation du paramètre **w = -0.18**est obtenu si **α=0.3,** ce résultat est mentionné en rouge donc on peut déduire que la meilleure valeur de **α** pour les graphes faiblement dense est égale à **0.3** pour trouver le vecteur de paramètre **w** le plus minimal possible utilisé par la fonction de résistance $\boldsymbol{f_w}$ **.**

Graphe moyennement dense (GMD) :

En utilisant les matrices d'adjacence des graphes de 15 nœuds, 34 nœuds et 70 nœuds on va appliquer la méthode SRW sur ces graphes en générant comme entrer une valeur aléatoire de paramètre **w = rand (1, m)**tel que **m** est le nombre de caractéristique, dans notre modèle on a choisi **m=3,** donc on va avoir un vecteur de paramètre **w = rand (1, m) = (0.82, 0.75, 0.79).** On va minimiser ce vecteur de paramètre en 1utilisant la fonction de minimisation directe **fminuc** à chaque valeur de **α** et pour que le résultat soit plus clair on a fait la moyenne de ce vecteur et le présenté dans **le tableau 4.2 ,** on peut remarquer que si **α=0.1,α=0.3,α=0.4**le paramètre **w >0** et si **α=0.2, α=0.5,α=0.6,α=0.8, α=0.9 ,α=1**le paramètre **w = 0** et la meilleure minimisation du paramètre **w = -0.08** est obtenu si **α=0.7,** ce résultat est mentionné en rouge donc on peut déduire que la meilleure valeur de **α** pour les graphes faiblement dense est égale à **0.7**pour trouver le vecteur de paramètre **w** le plus minimal possible utilisé par la fonction de résistance f_w

Graphes fortement denses (GRD) :

En utilisant les matrices d'adjacence des graphes de 16 nœuds, 40 nœuds et 400 nœuds on va appliquer la méthode SRW sur ces graphes en générant comme entrer une valeur aléatoire de paramètre **w = rand (1, m)**tel que **m** est le nombre de caractéristique, dans notre modèle on a choisi **m=3,** donc on va avoir un vecteur de paramètre **w = rand (1, m)=(0.82, 0.75, 0.79).**On va minimiser ce vecteur de paramètre en utilisant la fonction de minimisation directe **fminuc** à chaque valeur de **α** et pour que le résultat soit plus clair on a fait la moyenne de ce vecteur et le présenté dans **le tableau 4.2 ,** on peut remarquer que si**α=0, α=0.1** le paramètre **w >0**et si **α=0.2, α=0.7, α=0.8, α=0.9, α=1** le paramètre **w = 0** et la meilleure minimisation du paramètre **w = -0.3**est obtenu si **α=0.6,** ce résultat est mentionné en rouge donc on peut déduire que la meilleure valeur de **α** pour les graphes fortement dense est égale à **0.6**pour trouver le vecteur de paramètre **w** le plus minimal possible utilisé par la fonction de résistance f_w .

Le résultat obtenu des meilleures valeurs du paramètre **α** trouvés pour les graphes faiblement denses, moyennement denses, fortement denses afin de trouver le vecteur de paramètre **w** le plus minimal possible sont présentés dans **la figure 4.1**.

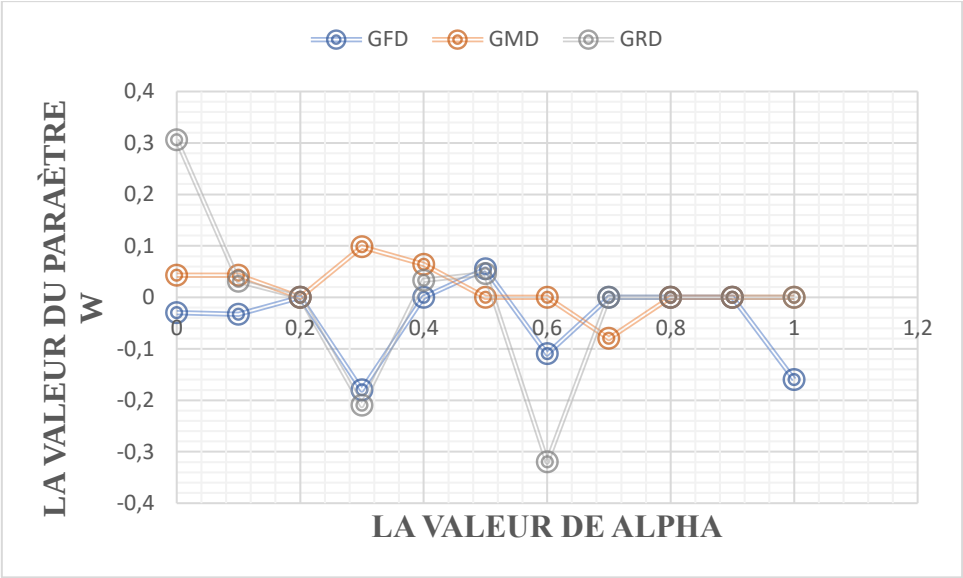


Figure 4.1 la représentation graphique des réseaux synthétiques

3. Application de l'approche proposé dans les réseaux réel

Les réseaux sociaux sont de deux types les réseaux sociaux humain comme le réseau High School, karaté club, les misérables et le réseau High Land tribus et les réseaux sociaux en ligne Face book. Nous avons appliqué la méthode proposée premièrement sur des réseaux sociaux humains après sur des réseaux sociaux en ligne.

Nous avons effectué plusieurs tests sur des réseaux réels connus représentés par le **Tableau 4.3** : HighSchoolnetwork, Karaté club, Les Misérables, High Land tribusetFacebook.

Réseaux	Nombre de nœuds	Nombre de liens	Référence
High School (HS)	70	366	**[97]**
Karaté club (KC)	34	78	**[99]**
Les misérables (MS)	77	254	**[95]**
High Land tribus (HL)	16	27	**[98]**
Facebook(FB)	2888	2981	**[96]**

Tableau 4.3 présentation des réseaux sociaux utilisés

3.1. Réseaux sociaux humains :

3.1.1. High school network

Premièrement, nous avons étudié un réseau social high school network, ce réseau est dirigé contient des amitiés entre des élèves dans un petit lycée de l'Illinois. Chaque élève a été interrogé une fois à l'automne de 1957 et au printemps de 1958. Cet ensemble de données regroupe les résultats des deux dates. Un nœud représente un élève et un lien entre deux élèves montre que l'élève de gauche a choisi l'élève de droite comme ami, le réseau se compose de 70 nœuds et 366 liens observé **[97].**

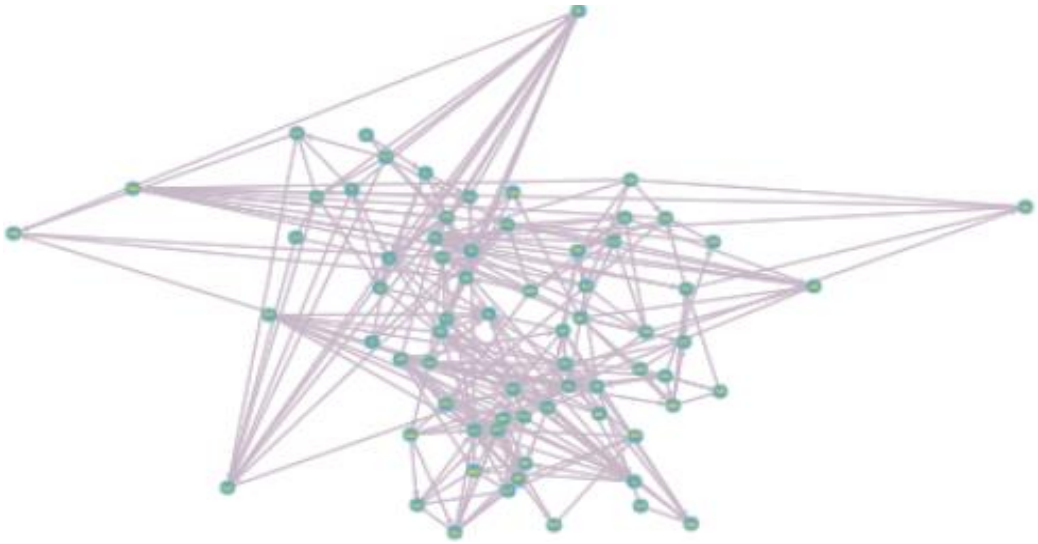


Figure 4.2 structure de réseau High school

En regardant **la figure 4.2**on peut voir qu'il existe des élèves de gauche qui ne sont pas relié avec les élèves de droit. En appliquant la méthode proposée, on va prédire les nouveaux liens (amitié) entre les nœuds (élèves). **La figure 4.3** montre les nouveaux liens obtenus.

Figure 4.3 les nouveaux liens prédits du réseau High school

Les liens prédits sont montrés dans **la figure 4.3** L'approche proposée a prédits 45 nouveaux liens .Les nouveaux liens sont les suivants :(2,3), (5,6), (7,8), (9,10), (10,11), (15,16), (16,17), (17,18), (20,21), (21,22), (22,23), (23,24), (24,25), (25,26), (26,27), (28,29), (29,30), (30,31), (31,32), (34,35), (35,36), (36,37), (37,38), (38,39), (39,40), (41,42), (42,43), (43,44), (44,45), (46,47), (47,48), (49,50), (50,51), (51,52), (52,53), (53,54), (54,55), (55,56), (56,57), (57,58), (59,60), (61,62), (63,64), (64,65), (68,69), (70,1).

3.1.2. Karaté club

Le deuxième réseau que nous avons étudié est un réseau social humain d'un club de karaté analysé par Wayne Zachary en 1977quia été observée sur une période de trois ans. Le réseau se compose de 34 membres d'un club de karaté en tant que nœuds et de 78 liens représentant l'amitié entre les membres du club. La figure suivante montre La structure de ce réseau **[99].**

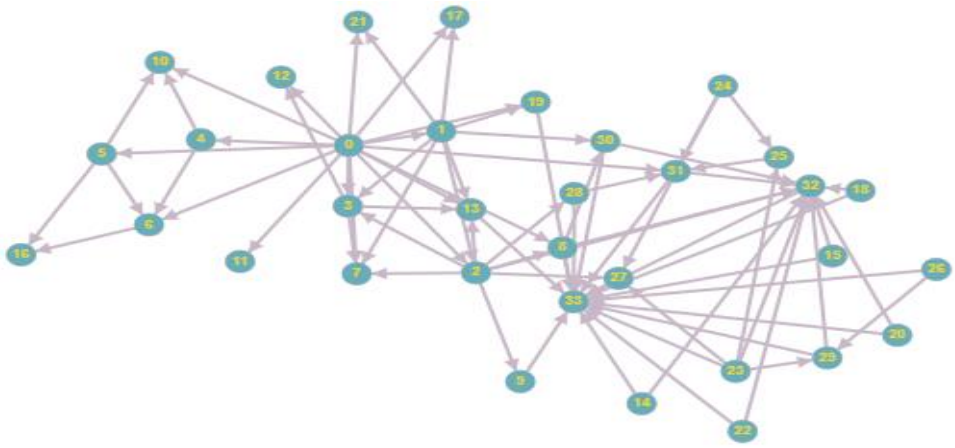


Figure 4.4 structure du réseau karaté club

En regardant la **figure 4.5** on peut voir qu'il existe des membres de club qui ne sont pas reliés malgré qu'ils soient des voisins donc ils peuvent être amis. En appliquant la méthode proposée, on va prédire les nouveaux liens (amitié) entre les nœuds (membre du club). **La figure 4.5** montre les nouveaux liens obtenus.

Figure 4.5 les nouveaux liens prédits du réseau karaté club

Les liens prédits sont montrés dans **la figure 4.5**. L'approche proposée a prédits 29 nouveaux liens. Les nouveaux liens sont les suivants :

(3,4),(4,5),(6,7),(7,8),(8,9),(9,10),(10,11),(11,12),(12,13),(13,14),(14,15),(15,16),(16,17),(17,18),(18,),(19,20),(21,22),(22,23),(23,24),(25,26),(26,27),(27,28),(28,29),(29,30),(30,31),(31,32),(32,33)et(33,0).

3.1.3. Réseau Les misérables

Ce réseau non dirigé contient des co-occurrentes de personnages dans le roman de Victor Hugo, Les Misérables. Les nœuds sont les personnages des misérables, et l'existence de lien entre deux nœuds signifie que les personnages représentés par ces nœuds sont apparus dans le même chapitre du livre. Ce réseau est composé de 77 personnages (nœuds) et 254 co-occurrentes (liens).

Afin de prédire les nœuds qui vont faire des nouveaux liens au futur c'est-à-dire les personnages qui vont être apparus dans le même chapitre, on peut dire qu'ils sont des voisins donc ces deniers sont suggérés comme les nœuds qui peuvent faire des liens à l'avenir **[95].**

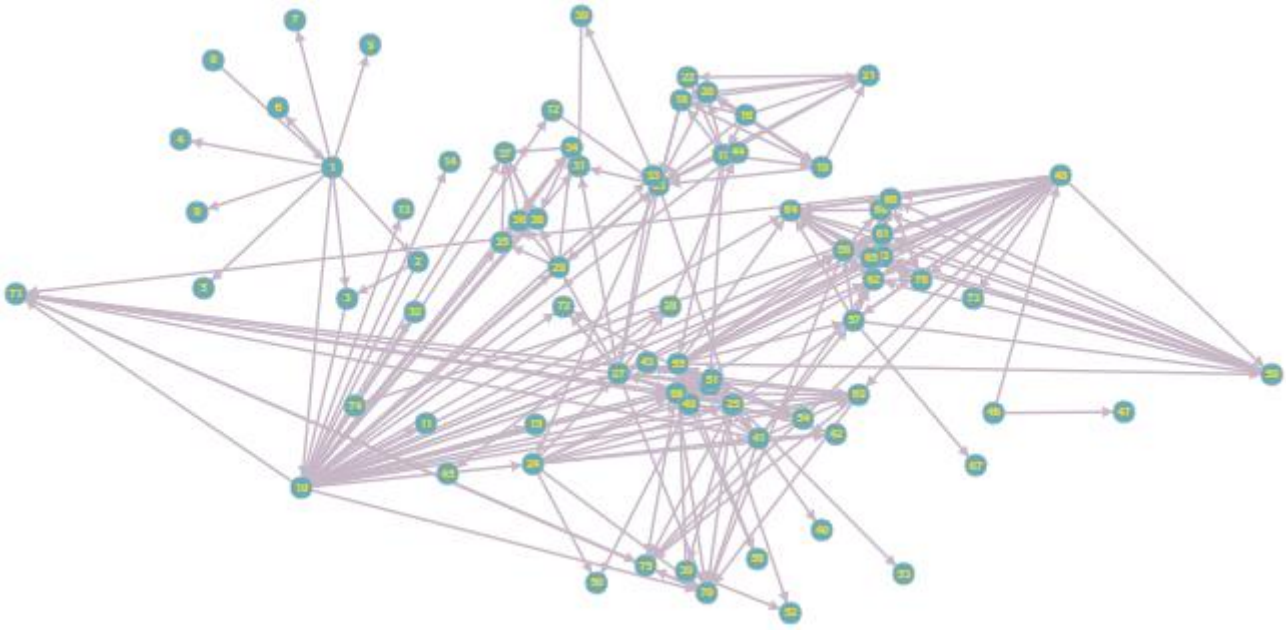


Figure 4.6 structure de réseau des misérables

En regardant la **figure 4.6** on peut voir qu'il existe des acteurs qui ne sont pas reliés malgré qu'ils soient des voisins donc ils peuvent apparaitre dans le même chapitre. En appliquant la méthode proposée, on va prédire les nouveaux liens (co-occurrentes) entre les nœuds (personnage).**La figure 4.7** montre les nouveaux liens obtenus.

Figure 4.7 Les nouveaux liens trouvés du réseau Les misérables

L'approche proposée à prédits 38 nouveaux liens. Les nouveaux liens sont les suivants :(4,5), (5,6), (6,7), (7,8), (8,9), (9,10), (9,10), (10,11), (12,13), (9,10), (10,11), (12,13), (13,14), (14,15), (15,16), (16,17), (29,30),(30,31),(32,33),(33,34),(34,35),(39,40),(40,41),(41,42),(43,44),(44,45),(45,46),(46,47),(48,49),(49,50),(51,52),(53,54),(54,55),(57,58),(67,68),(68,69),(72,73),(73,74),(74,75),(75,76),(76,77),(77,1).

3.1.4. Le réseau High Land tribus

Nous avons étudié un réseau social des humains, ceci est le réseau social signé des tribus de la structure d'alliance Gahuku Gama des hauts plateaux du centre, tiré de Kenneth Read (1954). Le réseau contient seize tribus connectées par amitié ("rova") et inimitié, tel que les tribus gauches choisissent les tribus droites comme amis **[98]**.

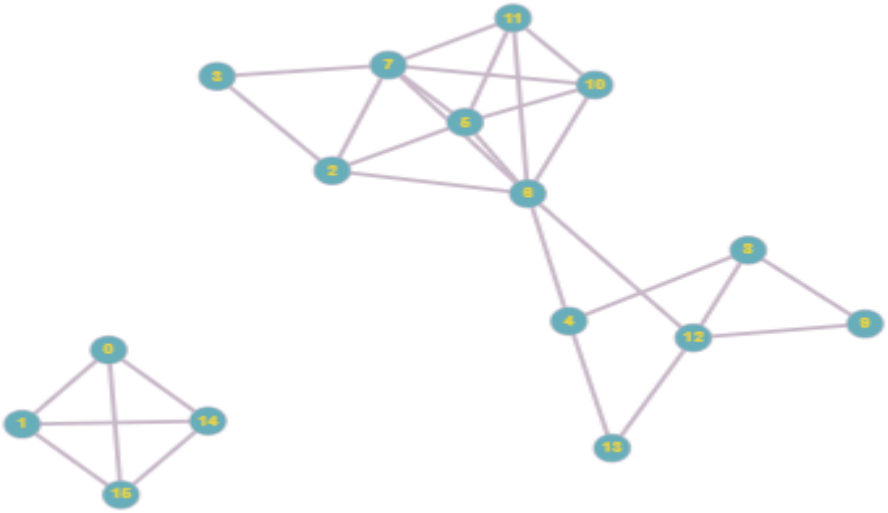


Figure 4.8 structure du réseau High Land tribus.

En effet, en regardant **la figure 4.8**, on peut distinguer qu'il y'a un sous réseau n'est pas attacher au réseau global High Land tribus qui est constituer de 4 nœuds (les nœuds 0,1, 14 et 16) et 6 liens. En appliquant notre méthode sur le réseau High Land tribus, on obtient les nouveaux liens prédits montré dans **la figure 4.9**.

L'approche proposée a prédits 5 liens. Les nouveaux liens sont les suivants : (2,3), (5,6), (9,9), (9,11), (11,13).

En remarquent que le sous réseau ne s'attache pas au réseau global c'est-à-dire les nœuds de sous réseau ne font pas des liens avec l'autre partie du réseau car ne sont pas des voisins, ce qui vérifier la validité de notre approche qui est basé sur le critère de voisinage.

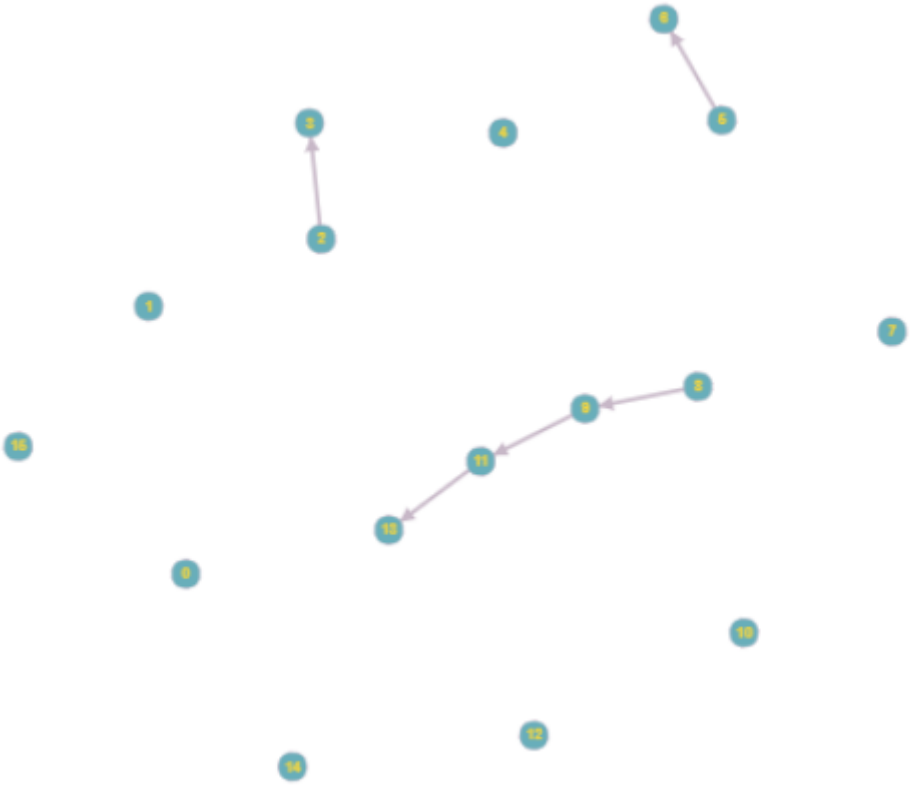


Figure 4.9 les nouveaux liens prédits du réseau High Land tribus

3.2. Réseaux sociaux en ligne

3.2.1. Réseau Facebook

Le premier réseau dans les réseaux sociaux en ligne est le réseau Facebook (NIPS), il est constitué de 2880 nœuds (utilisateurs) et 2981 liens (relation d'amitié). Ce réseau dirigé contient des amitiés d'utilisateur à utilisateur Face book. Un nœud représente un utilisateur et un lien indique que l'utilisateur représenté par le nœud de gauche est un ami de l'utilisateur représenté par le nœud de droite. **La figure 4.8** montre la structure du réseau Facebook (NIPS) **[96]**.

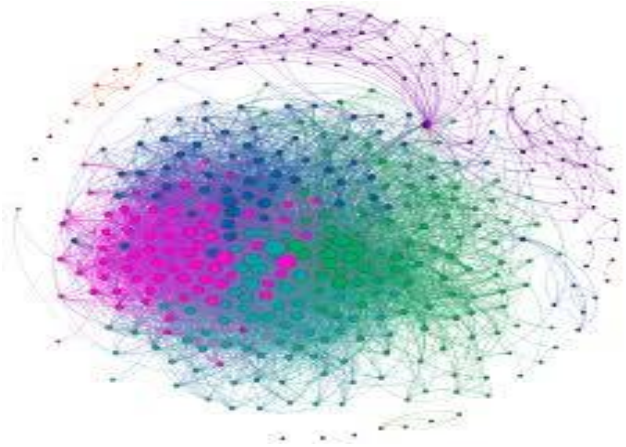


Figure 4.10 structure du réseau Face book (NIPS)

On peut voir qu'il existe des utilisateurs de gauche qui ne sont pas relié par les utilisateurs de droite, Le résultats de la méthode proposée est 2609 nouveaux liens.

Les résultats obtenus par l'algorithme proposé sur les réseaux réel considérés est résumé dans le **Tableau 4.4**

Réseaux	Nombre de nœuds	Nombre de liens	Nombre de liens prédits	Nombre de liens Totale
High school (HS)	70	366	45	411
Karaté club (KC)	34	78	29	107
Les misérables (MS)	77	254	38	292
High Land tribus (HL)	16	27	5	32
Face book (FB)	2888	2981	2609	5590

Tableau 4.4 présentation des réseaux sociaux utilisés après la prédiction de liens

La figure 4.11 montre le pourcentage d'augmentation du nombre de liens après l'exécution de l'algorithme proposé.

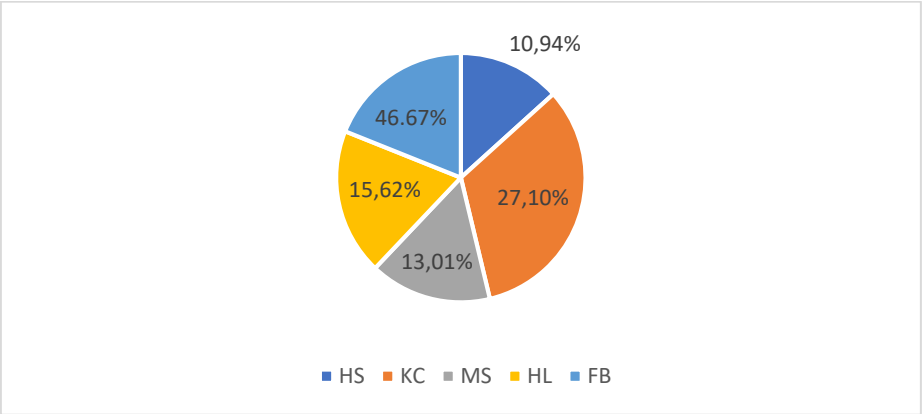


Figure 4.11 l'augmentation des réseaux sociaux après la prédiction de liens

4. Evaluation de l'approche proposé

Nous avons évalué la méthode proposée **SRW** selon le temps d'exécution, le score de précision et le score AUC ((Area Under Curve) qui servent à mesurer la qualité des résultats de prédiction obtenus par cette approche pour montrer sa performance.

4.1. Temps d'exécution

Le temps d'exécution a été calculé sur une machine avec les caractéristiques (soft et hard) suivantes :

Système d'exploitation	CPU	RAM	Disque dur	Implémentation de programme
Windows 10 64 bits	Intel i5-3120m 2.3GHZ	4GO	500 GO	Matlab2009b

Tableau 4.5 Caractéristique (Hard et Soft) de la machine

La figure 4.12montre le résultat obtenu après l'exécution de l'ensemble de graphes réels précédent.

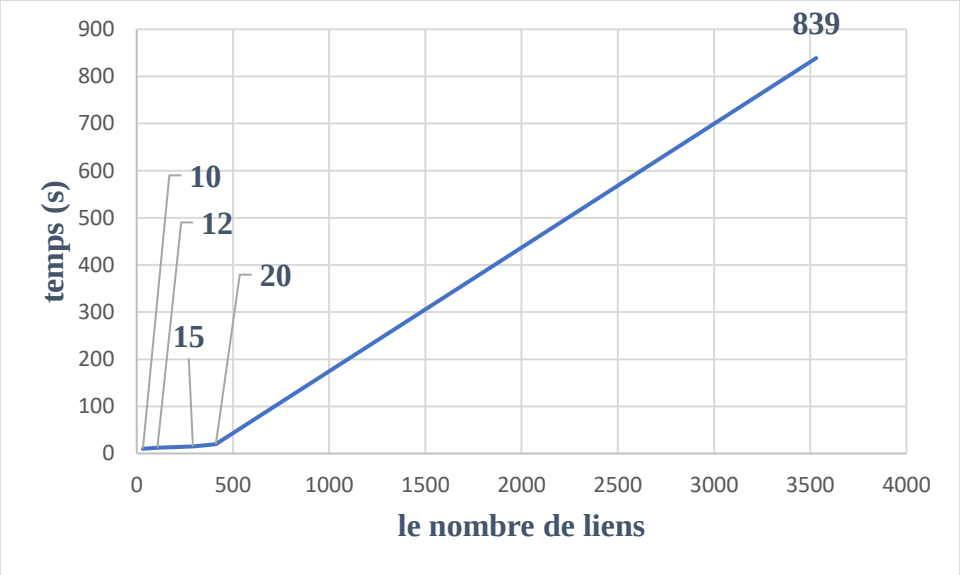


Figure 4.12 le temps d'exécution de l'approche **SRW** sur les graphes réels

L'axe horizontal est le nombre des liens dans les différents graphes réels présentés précédemment. L'axe vertical est le temps d'exécution en seconde et Le temps d'exécution de **SRW** reste dans l'intervalle [10s, 20s] pour les réseaux de 32, 292, 107 et 411 nœuds. Et comme le montre la figure, le temps d'exécution du dernier réseau 5590 liens augmente de manière significative de 839 secondes (13 min et 59s). Dans des graphes très denses, elle peut avoir une complexité et une consommation mémoires prohibitives.

4.2. Score AUC

Le score AUC (Area Under Curve) est un indice couramment utilisé pour estimer la qualité des résultats de prédiction. Le score AUC est le rapport entre nombre de liens avant la prédiction plus le nombre des liens prédits en le multipliant par la valeur **0.5** sur le nombre total des liens, alors la AUC est défini comme :

$$\mathbf{AUC} = \frac{n' + 0.5\, n''}{(n' + n'')}$$

n' **:** Le nombre des liens existés avant la prédiction.

n'' **:** Le nombre des liens prédits.

On peut distinguer de ces résultats qu'un score plus élevé de l'AUC représente généralement une meilleure qualité des résultats de prédiction. L'AUC la plus élevée est 1, ce qui indique un résultat correct. Le score de l'AUC d'une prédiction complètement aléatoire est de 0,5.

Le tableau 4.6 montre les scores AUC pour les différents réseaux réels présentés dans l'expérimentation.

Réseaux réels	HS	KC	MS	HL	FB
Score AUC	0.9452554745	0.8644859813	0.9349315068	0.921875	0.748568873

Tableau 4.6 Les scores AUC pour les différents réseaux réels.

On remarque que les scores AUC de cette approche SRW sont élevés qui représente une meilleure qualité des résultats de prédiction. Nous avons comparé le résultat trouvé par l'algorithme proposé en termes d'AUC pour le réseau Facebook avec plusieurs algorithmes de prédiction de liens. Le résultat est illustré par le **Tableau 4.7.**

Méthodes	**Score AUC**
Adamic-Adar **[38]**	0.60570
Common Friends **[38]**	0.59370
Degree**[38]**	0.56522
Nodefeatures**[38]**	0.60961
Network features**[38]**	0.59302
Node+network**[38]**	0.63711
Path features**[38]**	0.56213
Supervisedrandomwalks (SRW)	**0.748568873**

Tableau 4.7 comparaison du score AUC entre les approches sur le réseau Facebook

On peut voir que le score AUC de la méthode proposée SRW est plus élevé que les scores AUC des différentes autres approches, donc on peut déduire que cette approche est plus performante et elle est capable de prédire des liens significatifs.

4.3. Précision

La précision correspond au rapport du nombre des liens total après la prédiction, sur le nombre des liens ajoutés après la prédiction, nous l'avons calculé à l'aide de la formule : **Précision** $=\frac{n'+n''}{n''}$

Le tableau 4.8 montre les scores précision pour les différents réseaux réels présentés dans l'expérimentation.

Réseaux réels	HS	KC	MS	HL	FB
Précision	9.1333333333	3.6896551724	7.6842105263	6.4	6.42

Tableau 4.8 Les scores de précision pour les différents réseaux réels.

Le tableau 4.9 montre la comparaison des résultats des scores précisions de la méthode proposée SRW avec les scores de différentes autres méthodes.

Méthodes	**Score précision**
Adamic-Adar **[38]**	7.35
Common Friends **[38]**	7.35
Degree**[38]**	3.25
Nodefeatures**[38]**	2.38
Network features**[38]**	5.86
Node+network**[38]**	2.46
Path features**[38]**	5.86
Supervisedrandomwalks (SRW)	**6.42**

Tableau 4.9 montre la comparaison des scores de précision entre les approches sur le réseau Face book En général, la méthode Degree, nodefeutures, Node+network atteignent une précision inférieure par rapport aux autres, Tandis que la méthode proposée SRW, Adamic-Adar, Common Friends, Network features et Path features atteignent une précision plus élevée sur le réseau réel Face book, notre approche marque une bonne valeur de précision c'est-à-dire une bonne prédiction de liens. Les résultats obtenus permettent de constater que la méthode SRW est capable de prédire des liens significatifs.

5. Conclusion

Dans ce chapitre, nous avons présenté nos expérimentations avec les résultats obtenus, nous avons testé l'approche proposée sur un ensemble de réseaux synthétiques et réels afin de montrer comment l'approche a prédit les nouveaux liens et les meilleures valeurs de paramètres utilisés dans l'approche de prédiction des liens.

Conclusion et perspectives

La prédiction de liens est un domaine qui est encore dans une phase d'exploration et pour lequel il faudra encore attendre quelques années avant d'arriver à un stade de maturation. Bien que de nombreux algorithmes de prédiction de liens aient été développés récemment dans divers domaine (prédiction des protéines dans le domaine biologique, prédiction des liens dans les réseaux sociaux…etc.), la grande majorité des algorithmes existants ne combine pas la structure générale de graphe avec les caractéristiques des nœuds et des liens. L'état de l'art nous a permis de décrire et analyser quelques méthodes existantes pour la prédiction des liens à base apprentissage automatique, ce qui nous a permis de voir que la marche aléatoire supervisé qui est inclut dans le modèle de classification qui est à base apprentissage automatique est l'approche de prédiction de lien qui combine la structure générale du graphe avec les caractéristiques des nœuds et des liens.

Dans ce mémoire, nous proposons un algorithme à base de la marche aléatoire supervisé dans les réseaux sociaux. Notre travail consiste premièrement à apprendre un modèle qui assigne la probabilité de transition à chaque lien **(u, v)** existé dans le réseau qui est considérer comme la force de ces liens en utilisant les caractéristiques des nœuds et des liens en appliquant une fonction de résistance paramétré par le paramètre **w** qui est un vecteur de caractéristique, ensuite nous minimisant le vecteur de paramètre **w** tel que les scores de probabilité stationnaire des nœuds qui peuvent faire des liens au futurs soient supérieurs aux scores des nœuds qui peuvent pas crées des nœuds au futurs.

Enfin, Nous avons implémenté et testé la méthode proposée sur plusieurs réseaux sociaux utilisés comme benchmark dans le domaine de prédiction de liens. L'expérimentation de la méthode proposée, en termes de la mesure AUC, a donné des résultats satisfaisants.

La plupart des méthodes de prédiction de liens actuelles considèrent le problème de manière statique, ce qui signifie que le réseau dans lequel les nœuds sont présupposés reste inchangé. Cependant les réseaux de tous les types changent avec le temps et les réseaux sociaux en particulier sont très actifs. Bien que de nombreux mécanismes aient été appliqués pour résoudre ce problème, il existe encore un écart la réalité de la prédiction de liens en temps réels et les méthodes de prédiction de lien statiques.

Dans une autre perspective, la plupart des chercheurs sur la prédiction de lien se concentre actuellement sur les liens qui pourrait être crées dans le futur. Peu de chercheurs font des recherches dans l'inverse c'est-à-dire la prédiction de lien qui disparaîtra dans le futur.

References

[1]Aggarwal, C. C. (2011). An introduction to social network data analytics. In Social network data analytics (pp. 1-15). Springer, Boston, MA.

[2] Tang, F., Mao, C., Yu, J., & Chen, J. (2011, October). Notice of Retraction The implementation of information service based on social network systems. In The 5th International Conferenceon New Trends in Information Science and Service Science (Vol. 1, pp. 46-49). IEEE.

[3] Travers, J., & Milgram, S. (1969). An exploratorystudy of the small world problem. Sociometry, 32, 425-43.

[4] CLAUSET, A. & EAGLE, N. 2012. Persistence and periodicity in a dynamicproximity network. arXivpreprint arXiv:1211.7343.

[5] LESKOVEC, J., KLEINBERG, J. & FALOUTSOS, C. 2005. Graphs over time. Proceeding of the eleventh ACM SIGKDD international conference on Knowledgediscovery in data mining - KDD '05. ACM Press

[6]Gu, S., Chen, L., Li, B., Liu, W., & Chen, B. (2019). Link prediction on signed social networks based on latent space mapping. Applied Intelligence, 49(2), 703-722.

[7]Srinivas, V., & Mitra, P. (2016). Link prediction in social networks:role of power law distribution. Springer International Publishing.

[8]Guns, R. (2014). Link prediction. In Measuringscholarlyimpact(pp. 35-55). Springer, Cham.

[9]Tylenda, T., Angelova, R., &Bedathur, S. (2009, June). Towards time-awarelinkprediction in evolving social networks. In Proceedings of the 3rd workshop on social network mining and analysis (p. 9). ACM.

[11]Sarkar, P., Chakrabarti, D., & Jordan, M. (2012). Nonparametriclinkprediction in dynamic networks. arXivpreprint arXiv:1206.6394.

[12]Li, X., Du, N., Li, H., Li, K., Gao, J., & Zhang, A. (2014, April). A deeplearningapproach to linkprediction in dynamic networks. In Proceedings of the 2014 SIAM International Conference on Data Mining (pp. 289-297). Society for Industrial and AppliedMathematics.

[13]Dong, Y., Tang, J., Wu, S., Tian, J., Chawla, N. V., Rao, J., & Cao, H. (2012, December). Link prediction and recommendationacrossheterogeneous social networks. In 2012 IEEE 12th International conference on data mining (pp. 181-190). IEEE.

[14]Ge, L., & Zhang, A. (2012, April). Pseudo cold start linkpredictionwith multiple sources in social networks. In Proceedings of the 2012 SIAM International Conference on Data Mining (pp. 768-779). Society for Industrial and AppliedMathematics.

[15]Kuo, T. T., Yan, R., Huang, Y. Y., Kung, P. H., & Lin, S. D. (2013, August). Unsupervisedlinkpredictionusingaggregativestatistics on heterogeneous social networks. In Proceedings of the 19th ACM SIGKDD international conference on Knowledgediscovery and data mining (pp. 775-783). ACM.

[16] Raymond, R., & Kashima, H. (2010, September). Fast and scalable algorithms for semi-supervisedlinkprediction on static and dynamic graphs. In Joint europeanconference on machine learning and knowledgediscovery in databases (pp. 131-147). Springer, Berlin, Heidelberg.**s**

[17] Tang, F. (2017). Link-Prediction and its Application in Online Social Networks (Doctoral dissertation, Victoria University).

[18] Jagadishwari, V., &Umadevi, V. (2015). EmpiricalAnalysis of Traditional Link Prediction Methods. International Journal of Computer Applications, 121(2).

[19]Jeong, H., Tombor, B., Albert, R., Oltvai, Z. N., &Barabási, A. L. (2000). The large-scaleorganization of metabolic networks. Nature, 407(6804), 651.

[20]Ito, T., Chiba, T., Ozawa, R., Yoshida, M., Hattori, M., &Sakaki, Y. (2001). A comprehensivetwo-hybridanalysis to explore the yeastproteininteractome. Proceedings of the National Academy of Sciences, 98(8), 4569-4574.**[10]**Rümmele, N., Ichise, R., &Werthner, H. (2015, May). Exploringsupervisedmethods for temporal linkprediction in heterogeneous social networks. In Proceedings of the 24th International Conferenceon World Wide Web (pp. 1363-1368). ACM.

[21]Guelzim, N., Bottani, S., Bourgine, P., &Képès, F. (2002). Topological and causal structure of the yeasttranscriptionalregulatory network. Nature genetics, 31(1), 60.

[22]Albert, R. (2005). Scale-free networks in cellbiology. Journal of cell science, 118(21), 4947-4957.

[23]Zhu, X., Gerstein, M., & Snyder, M. (2007). Gettingconnected:analysis and principles of biological networks. Genes&development, 21(9), 1010-1024.

[24]Barabási, A. L., &Bonabeau, E. (2003). Scale-free networks. Scientific american, 288(5), 60-69.

[25]Clauset, A., Shalizi, C. R., & Newman, M. E. (2009). Power-law distributions in empirical data. SIAM review, 51(4), 661-703.

[26]Schafer, J. B., Frankowski, D., Herlocker, J., & Sen, S. (2007). Collaborative filteringrecommendersystems. In The adaptive web (pp. 291-324). Springer, Berlin, Heidelberg.

[27]Leininger, S., Urich, T., Schloter, M., Schwark, L., Qi, J., Nicol, G. W., ... &Schleper, C. (2006). Archaea predominateamongammonia-oxidizingprokaryotes in soils. Nature, 442(7104), 806.

[28] Pavlov, M., &Ichise, R. (2007). Finding experts by linkprediction in co-authorship networks. FEWS, 290, 42-55.

[29] Lü, L., & Zhou, T. (2011). Link prediction in complex networks: A survey. Physica A: statistical mechanics and its applications, 390(6), 1150-1170.

[30] Brandes, U., & Wagner, D. (2004). Analysis and visualization of social networks. In Graph drawing software (pp. 321-340). Springer, Berlin, Heidelberg.

[31] Steyvers, M., Miller, B., Hemmer, P., & Lee, M. D. (2009). The wisdom of crowds in the recollection of order information. In Advances in neural information processing systems (pp. 1785-1793).

[32] Tylenda, T., Angelova, R., &Bedathur, S. (2009, June). Towards time-aware link prediction in evolving social networks. In Proceedings of the 3rd workshop on social network mining and analysis (p. 9). ACM.

[33] Yang, Y., Chawla, N., Sun, Y., & Hani, J. (2012, December). Predicting links in multi-relational and heterogeneous networks. In 2012 IEEE 12th international conference on data mining (pp. 755-764). IEEE.

[34] Al Hasan, M., Chaoji, V., Salem, S., &Zaki, M. (2006, April). Link prediction using supervised learning. In SDM06: workshop on link analysis, counter-terrorism and security.

[35] Liben-Nowell, D., & Kleinberg, J. (2007). The link-prediction problem for social networks. Journal of the American society for information science and technology, 58(7), 1019-1031.

[36] Lichtenwalter, R. N., Lussier, J. T., & Chawla, N. V. (2010, July). New perspectives and methods in link prediction. In Proceedings of the 16th ACM SIGKDD international conference on Knowledge discovery and data mining (pp. 243-252). ACM.

[37] Sarkar, P., Chakrabarti, D., & Moore, A. W. (2011, June). Theoretical justification of popular link prediction heuristics. In Twenty-Second International Joint Conference on Artificial Intelligence.

[38] Backstrom, L., &Leskovec, J. (2011, February). Supervised random walks: predicting and recommending links in social networks. In Proceedings of the fourth ACM international conference on Web search and data mining (pp. 635-644). ACM.

[39] Lee, K., Agrawal, A., &Choudhary, A. (2013, August). Real-time disease surveillance using twitter data: demonstration on flu and cancer. In Proceedings of the 19th ACM SIGKDD international conference on Knowledge discovery and data mining (pp. 1474-1477). ACM.

[40] Bliss, C. A., Frank, M. R., Danforth, C. M., &Dodds, P. S. (2014). An evolutionary algorithm approach to link prediction in dynamic social networks. Journal of Computational Science, 5(5), 750-764.

[41] Wang, D., Pedreschi, D., Song, C., Giannotti, F., &Barabasi, A. L. (2011, August). Human mobility, social ties, and link prediction. In Proceedings of the 17th ACM SIGKDD international conference on Knowledge discovery and data mining (pp. 1100-1108). Acm.

[42] Nguyen-Thi, A. T., Nguyen, P. Q., Ngo, T. D., & Nguyen-Hoang, T. A. (2015). Transfer AdaBoost SVM for link prediction in newly signed social networks using explicit and PNR features. Procedia Computer Science, 60, 332-341.

[43] Sarkar, P., & Moore, A. W. (2006). Dynamic social network analysis using latent space models. In Advances in Neural Information Processing Systems (pp. 1145-1152).

[44] Sarkar, P., Chakrabarti, D., & Jordan, M. (2012). Nonparametric link prediction in dynamic networks. arXiv preprint arXiv:1206.6394.

[45] Sewell, D. K., & Chen, Y. (2016). Latent space models for dynamic networks with weighted edges. Social Networks, 44, 105-116.

[46] Bordes, A., Glorot, X., Weston, J., &Bengio, Y. (2014). A semantic matching energy function for learning with multi-relational data. Machine Learning, 94(2), 233-259.

[47] Zhu, L., Guo, D., Yin, J., VerSteeg, G., &Galstyan, A. (2016). Scalable temporal latent space inference for link prediction in dynamic social networks. IEEE Transactions on Knowledge and Data Engineering, 28(10), 2765-2777.

[48] Rastelli, R., Friel, N., &Raftery, A. E. (2016). Properties of latent variable network models. Network Science, 4(4), 407-432.

[49] Rahman, M., & Al Hasan, M. (2016, September). Link prediction in dynamic networks using graphlet. In Joint European Conference on Machine Learning and Knowledge Discovery in Databases (pp. 394-409). Springer, Cham.

[50] Nickel, M., Murphy, K., Tresp, V., &Gabrilovich, E. (2016). A review of relational machine learning for knowledge graphs. Proceedings of the IEEE, 104(1), 11-33.

[51] Dunlavy, D. M., Kolda, T. G., &Acar, E. (2011). Temporal link prediction using matrix and tensor factorizations. ACM Transactions on Knowledge Discovery from Data (TKDD), 5(2), 10.

[52] Ermiş, B., Acar, E., &Cemgil, A. T. (2012). Link prediction via generalized coupled tensor factorisation. arXiv preprint arXiv:1208.6231.

[53] Gao, S., Denoyer, L., &Gallinari, P. (2011, April). Link pattern prediction with tensor decomposition in multi-relational networks. In 2011 IEEE Symposium on Computational Intelligence and Data Mining (CIDM) (pp. 333-340). IEEE.

[54] Menon, A. K., &Elkan, C. (2011, September). Link prediction via matrix factorization. In Joint european conference on machine learning and knowledge discovery in databases (pp. 437-452). Springer, Berlin, Heidelberg.

[55] Haghani, S., &Keyvanpour, M. R. (2017). A systemic analysis of link prediction in social network. Artificial Intelligence Review, 1-35.

[56] Spiegel, S., Clausen, J., Albayrak, S., &Kunegis, J. (2011, May). Link prediction on evolving data using tensor factorization. In Pacific-Asia Conference on Knowledge Discovery and Data Mining (pp. 100-110). Springer, Berlin, Heidelberg.ss

[57] Yao, L., Sheng, Q. Z., Qin, Y., Wang, X., Shemshadi, A., & He, Q Context-aware point-fIiterest ecommendation using tensor factorization with social regularization. In Proceedings of the 38th ıternational ACM SIGIR conference on research and development in information retrieval (pp. 1007-010). ACM.

[58] Han, Y., &Moutarde, F. (2016). Analysis of large-scale traffic dynamics in an urban transportation network using non-negative tensor factorization. International Journal of Intelligent Transportation Systems Research, 14(1), 36-49.

[59] Nickel, M., &Tresp, V. (2013, September). Tensor factorization for multi-relational learning. In Joint European Conference on Machine Learning and Knowledge Discovery in Databases (pp. 617-621). Springer, Berlin, Heidelberg.

[60] London, B., Rekatsinas, T., Huang, B., &Getoor, L. (2013). Multi-relational learning using weighted tensor decomposition with modular loss. arXiv preprint arXiv:1303.1733.

[61] Nickel, M., Jiang, X., &Tresp, V. (2014). Reducing the rank in relational factorization models by including observable patterns. In Advances in Neural Information Processing Systems (pp. 1179-1187).

[62]Keyvanpour, M. R., & Moradi, S. S. (2014). A perturbation method based on singular value decomposition and feature selection for privacy preserving data mining. International Journal of Data Warehousing and Mining (IJDWM), 10(1), 55-76.

[63] Narita, A., Hayashi, K., Tomioka, R., & Kashima, H. (2012). Tensor factorization using auxiliary information. Data Mining and Knowledge Discovery, 25(2), 298-324.

[64]Yılmaz, K. Y., Cemgil, A. T., &Simsekli, U. (2011). Generalised coupled tensor factorisation. In Advances in neural information processing systems (pp. 2151-2159).

[65]Nakatsuji, M., Toda, H., Sawada, H., Zheng, J. G., &Hendler, J. A. (2016). Semantic sensitive tensor

factorization. Artificial Intelligence, 230, 224-245.

[66] Jiang, X., Tresp, V., Huang, Y., & Nickel, M. (2012). Link Prediction in Multi-relational Graphs using Additive Models. SeRSy, 919, 1-12.

[67]]Riedel, S., Yao, L., McCallum, A., & Marlin, B. M. (2013). Relation extraction with matrix factorization and universal schemas. In Proceedings of the 2013 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (pp. 74-84).

[68] Wang, C., Satuluri, V., & Parthasarathy, S. (2007, October). Local probabilistic models for link prediction. In Seventh IEEE international conference on data mining (ICDM 2007) (pp. 322-331). IEEE.

[69]Wang, C., Satuluri, V., & Parthasarathy, S. (2007, October). Local probabilistic models for link prediction. In Seventh IEEE international conference on data mining (ICDM 2007) (pp. 322-331). IEEE.

[70] Nguyen, C. H., &Mamitsuka, H. (2011, September). Kernels for link prediction with latent feature models. In Joint European Conference on Machine Learning and Knowledge Discovery in Databases (pp. 517-532). Springer, Berlin, Heidelberg.

[71]Feng, X., Zhao, J. C., & Xu, K. (2012). Link prediction in complex networks: a clustering perspective. The European Physical Journal B, 85(1), 3.

[72]Adamic, L. A., & Adar, E. (2003). Friends and neighbors on the web. Social networks, 25(3), 211-230.

[73]Barabási, A. L., & Albert, R. (1999). Emergence of scaling in random networks. science, 286(5439), 509-512.

[74]Mitzenmacher, M. (2004). A brief history of generative models for power law and lognormal distributions. Internet mathematics, 1(2), 226-251.

[75]Rossetti, G., Guidotti, R., Pennacchioli, D., Pedreschi, D., & Giannotti, F. (2015, August). Interaction prediction in dynamic networks exploiting community discovery. In 2015 IEEE/ACM International Conference on Advances in Social Networks Analysis and Mining (ASONAM) (pp. 553-558). IEEE.

[76]Martínez, V., Berzal, F., & Cubero, J. C. (2017). A survey of link prediction in complex networks. ACM Computing Surveys (CSUR), 49(4), 69.

[77]Newman, M. E. (2001). Clustering and preferential attachment in growing networks. Physical review E, 64(2), 025102.

[78]Wang, P., Xu, B., Wu, Y., & Zhou, X. (2015). Link prediction in social networks : the state-of-the-art. Science China Information Sciences, 58(1), 1-38.

[79]Dunlavy, D. M., Kolda, T. G., &Acar, E. (2011). Temporal link prediction using matrix and tensor factorizations. ACM Transactions on Knowledge Discovery from Data (TKDD), 5(2), 10.

[80]Fire, M., Tenenboim, L., Lesser, O., Puzis, R., Rokach, L., &Elovici, Y. (2011, October). Link

prediction in social networks using computationally efficient topological features. In 2011 IEEE Third International Conference on Privacy, Security, Risk and Trust and 2011 IEEE Third International Conference on Social Computing (pp. 73-80). IEEE.

[81]Zhai, S., & Zhang, Z. (2015, June). Dropout training of matrix factorization and autoencoder for link prediction in sparse graphs. In Proceedings of the 2015 SIAM International Conference on Data Mining (pp. 451-459). Society for Industrial and Applied Mathematics.

[82]Bengio, Y., Courville, A., & Vincent, P. (2013). Representation learning: A review and new perspectives. IEEE transactions on pattern analysis and machine intelligence, 35(8), 1798-1828.

[83]Socher, R., Chen, D., Manning, C. D., & Ng, A. (2013). Reasoning with neural tensor networks for knowledge base completion. In Advances in neural information processing systems (pp. 926-934).

[84]Li Deng, D. Y. (2014). Deep learning: methods and applications. Tech. rep., https://www. microsoft. com/en-us/research/publication/deep-learning-methods-and-applications.

[85]Goodfellow, I., Bengio, Y., & Courville, A. (2016). Deep Learning http://www. deeplearningbook. org.

[86]Liu, F., Liu, B., Sun, C., Liu, M., & Wang, X. (2013, November). Deep learning approaches for link prediction in social network services. In International Conference on Neural Information Processing (pp. 425-432). Springer, Berlin, Heidelberg.

[87] Li, K., Gao, J., Guo, S., Du, N., Li, X., & Zhang, A. (2014, December). Lrbm: A restricted boltzmannmachine based approach for representation learning on linked data. In 2014 IEEE International Conference on Data Mining (pp. 300-309). IEEE.

[88]Perozzi, B., Al-Rfou, R., &Skiena, S. (2014, August). Deepwalk: Online learning of social representations. In Proceedings of the 20th ACM SIGKDD international conference on Knowledge discovery and data mining (pp. 701-710). ACM.

[89] Li, X., Du, N., Li, H., Li, K., Gao, J., & Zhang, A. (2014, April). A deep learning approach to link prediction in dynamic networks. In Proceedings of the 2014 SIAM International Conference on Data Mining (pp. 289-297). Society for Industrial and Applied Mathematics.

[90] Grover, A., &Leskovec, J. (2016, August). node2vec: Scalable feature learning for networks. In Proceedings of the 22nd ACM SIGKDD international conference on Knowledge discovery and data mining (pp. 855-864). ACM.

[91]Hinton, G. E., Osindero, S., &Teh, Y. W. (2006). A fast learning algorithm for deep belief nets. Neural computation, 18(7), 1527-1554.

[92]Mikolov, T., Chen, K., Corrado, G., & Dean, J. (2013). Efficient estimation of word representations in vector space. arXiv preprint arXiv:1301.3781.

[93] Tang, J., Qu, M., Wang, M., Zhang, M., Yan, J., & Mei, Q. (2015, May). Line: Large scale information network embedding. In Proceedings of the 24th international conference on world wide web (pp. 1067-1077). International World Wide Web Conferences Steering Committee.

[94] Zhang, X., Chen, W., & Yan, H. (2016, November). TLINE: Scalable transductive network embedding. In Asia Information Retrieval Symposium (pp. 98-110). Springer, Cham.

[95] Backstrom, L., &Leskovec, J. (2011, February). Supervised random walks: predicting and recommending links in social networks. In Proceedings of the fourth ACM international conference on Web search and data mining (pp. 635-644). ACM.

[96] Knuth, D. E. (1993). The Stanford GraphBase: a platform for combinatorial computing (pp. 74-87). New York: AcM Press.

[97]Leskovec, J., &Mcauley, J. J. (2012). Learning to discover social circles in ego networks. In Advances in neural information processing systems (pp. 539-547).

[98] Coleman, J. S. (1964). Introduction to mathematical sociology. Introduction to mathematical sociology.

[99] Kenneth, E. (1954). Read. Cultures of the Central Highlands, New Guinea. Southwestern J. of Anthropology, 10(1), 1-43.

[100]Zachary, W. W. (1977). An information flow model for conflict and fission in small groups. Journal of anthropologicalresearch, 33(4), 452-473.

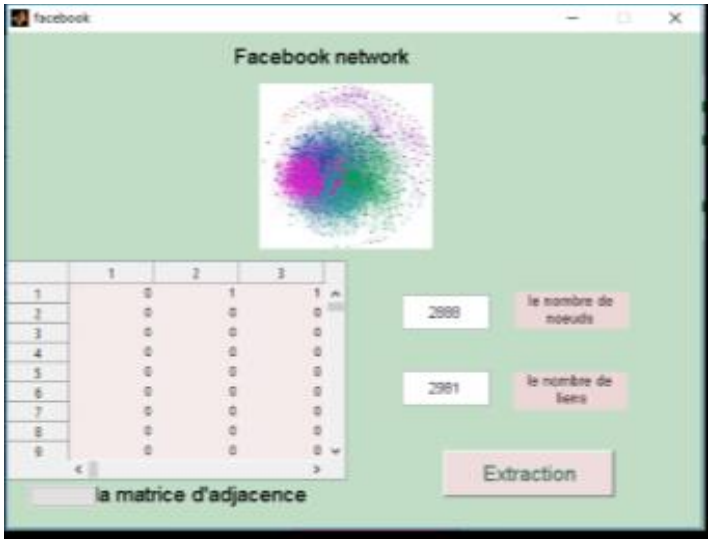
[101]Bastian, M., Heymann, S., &Jacomy, M. (2009, March). Gephi: an open source software for exploring and manipulating networks. In *Third international AAAI conference on weblogs and social media*.

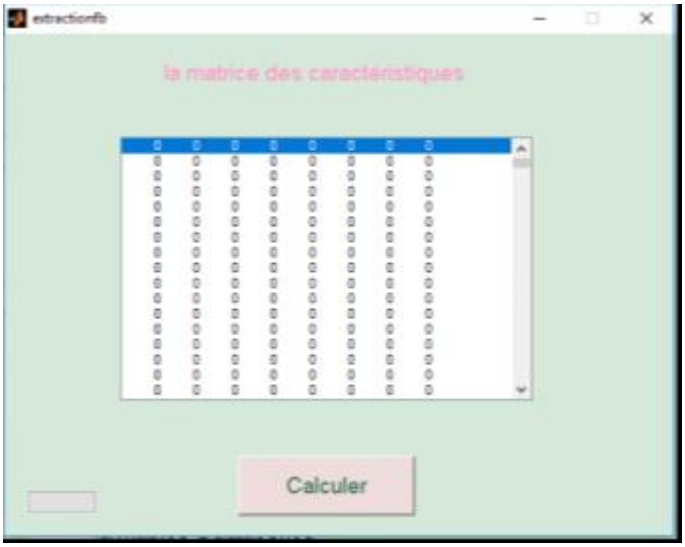
[102]https://fr.mathworks.com/company/aboutus.html.

Annexe

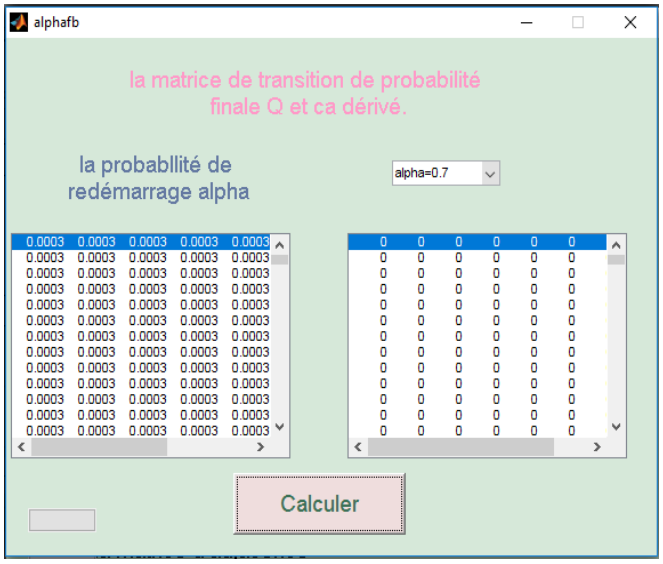
Les interfaces d'accueil :

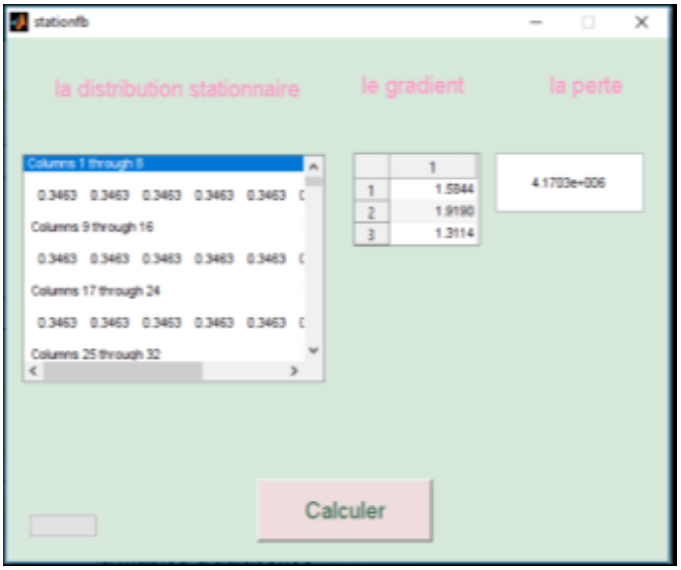
Les interfaces de différentes étapes de l'approche SRW du réseauFaceBook :

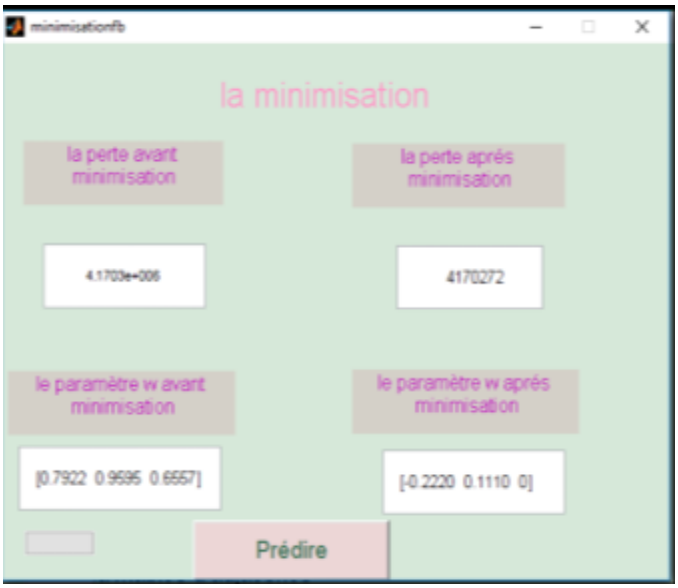



extractionfb
la matrice des caractéristiques
Calculer


forcefb
la matrice de transition et ca dérivé (les forces des liens)
le paramètre w avant minimisation
[0.7922 0.9595 0.6557]
Calculer

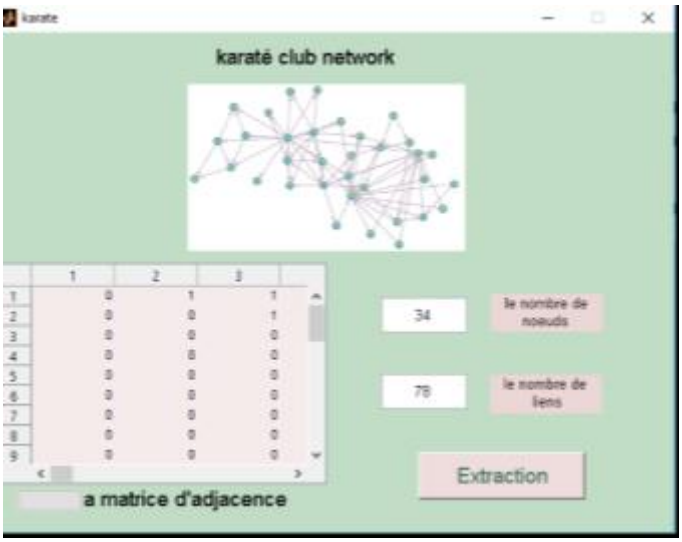

alphafb
la matrice de transition de probabilité finale Q et ca dérivé.
la probabllité de redémarrage alpha
alpha=0.7
Calculer

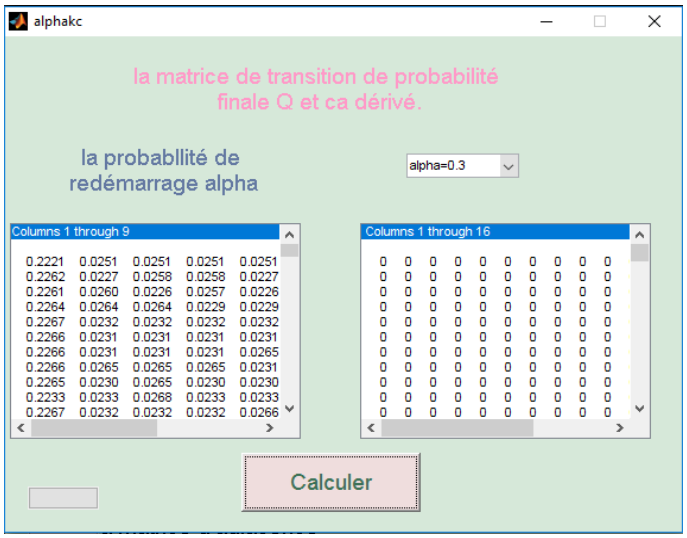

stationfb
la distribution stationnaire
le gradient
la perte
4.1703e+006
Calculer

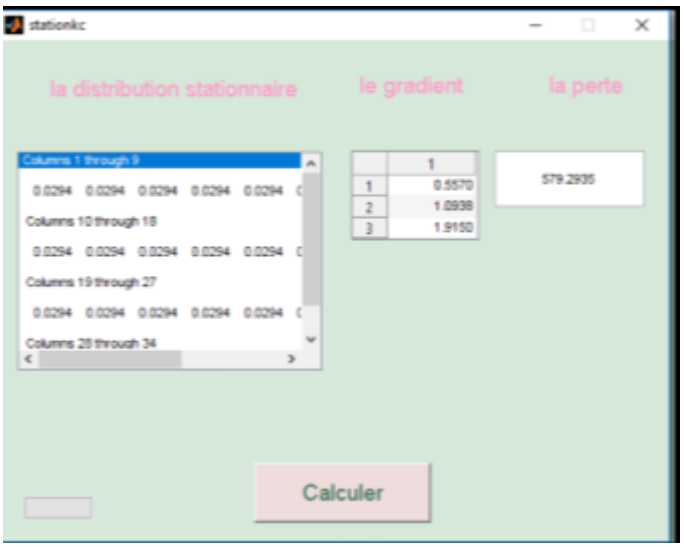

minimisationfb
la minimisation
la perte avant minimisation
la perte aprés minimisation
4.1703e+006
4170272
le paramètre w avant minimisation
le paramètre w aprés minimisation
[0.7922 0.9595 0.6557]
[-0.2220 0.1110 0]
Prédire

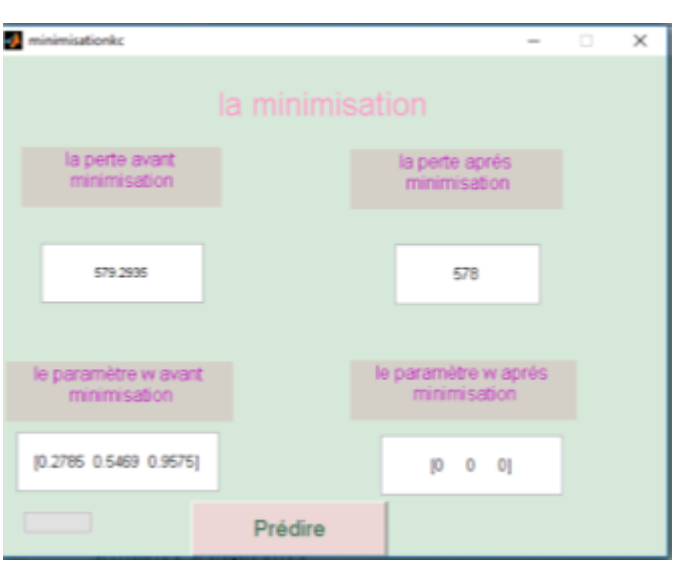

finalefb
les nouveaux liens prédits pour le réseau facebook
le nombre de liens prédit
2609
le nombre de liens totale
5590
Retour
la matrice d'adjacence aprés prédiction

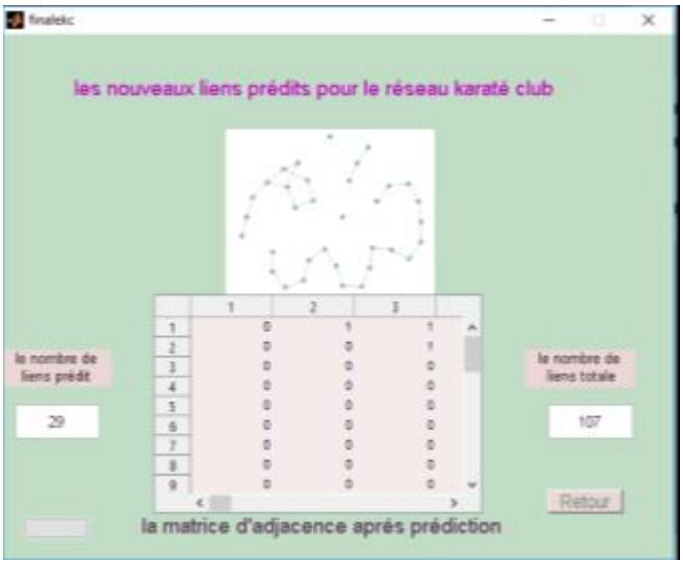
Les interfaces de différentes étapes de l'approche SRW du réseau karaté club :



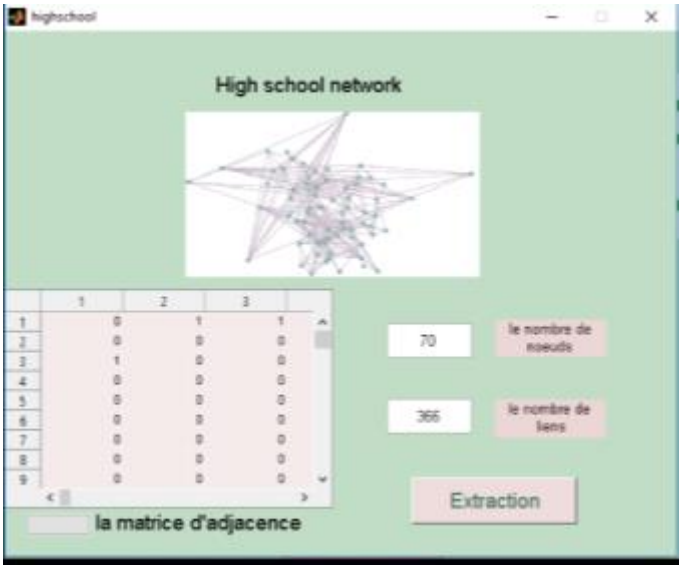


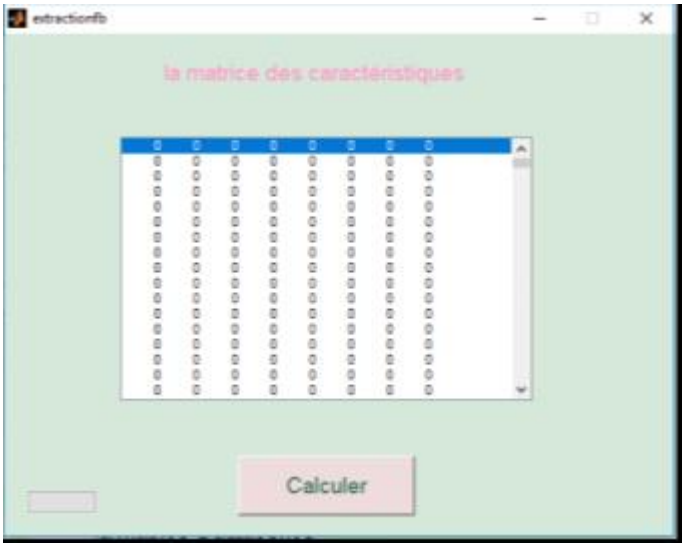


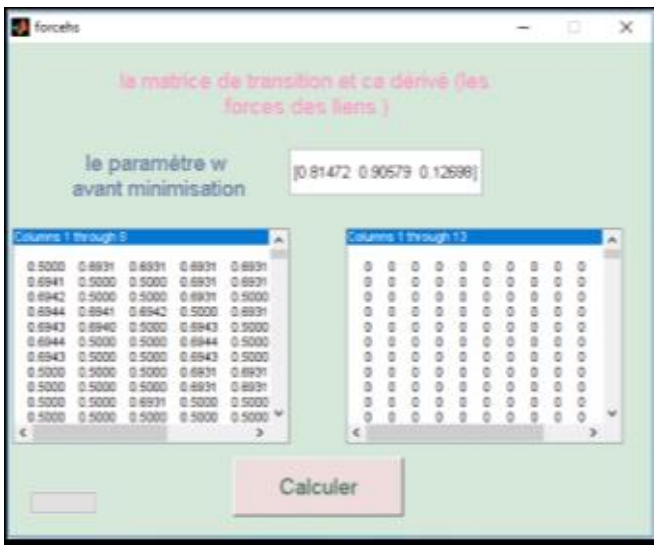


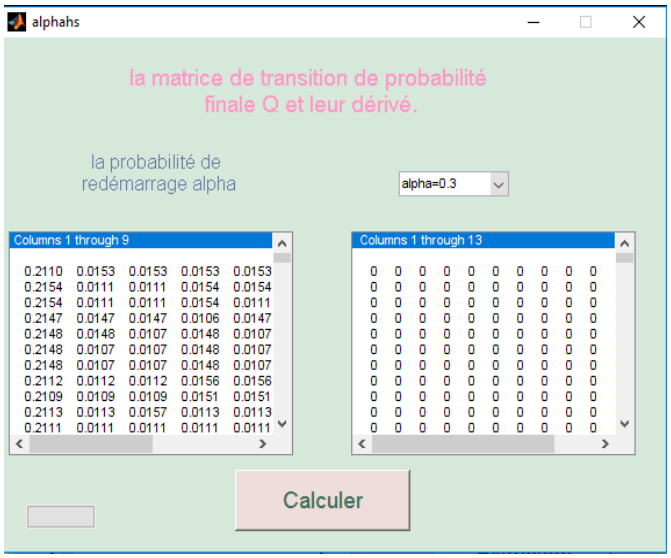


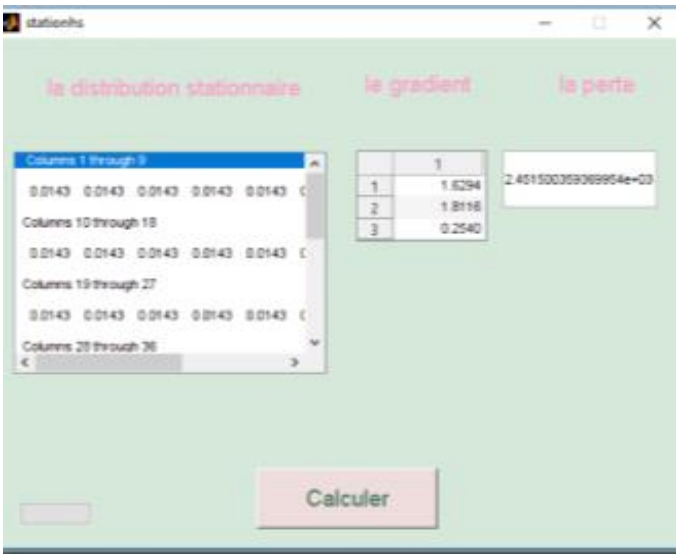
Les interfaces de différentes étapes de l'approche SRW du réseau High School :

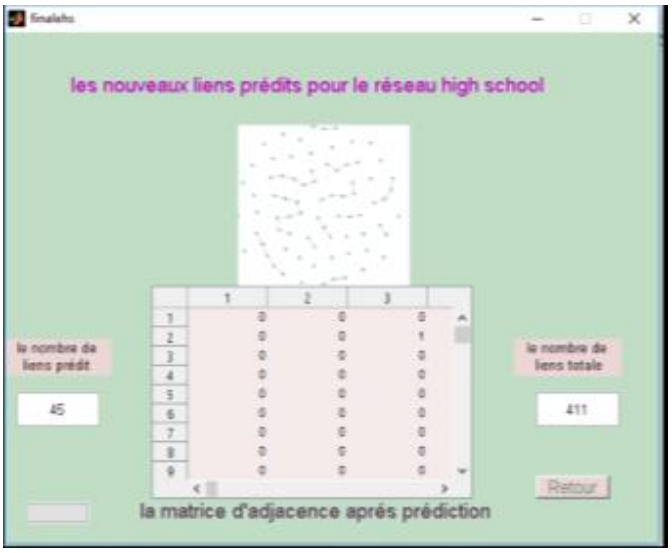




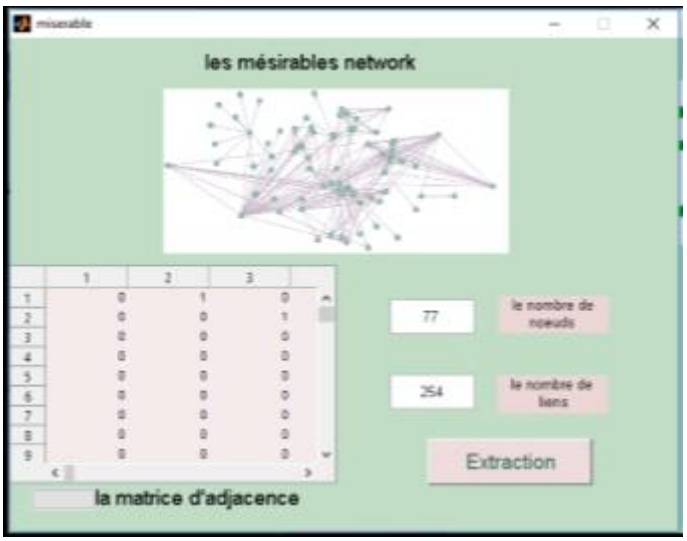


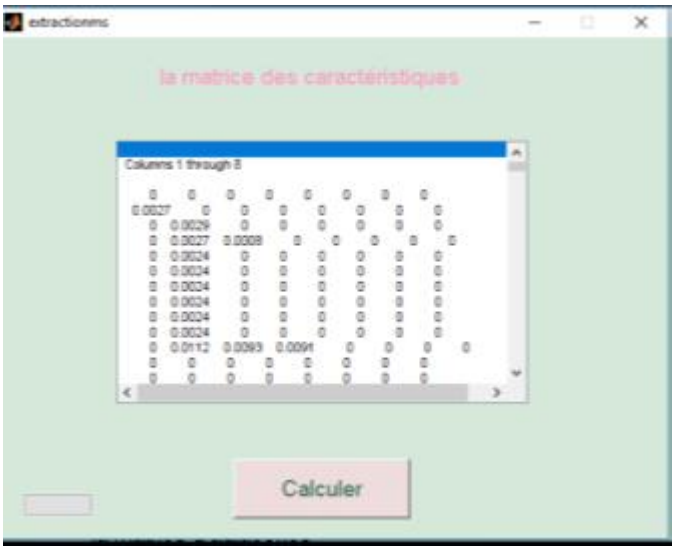


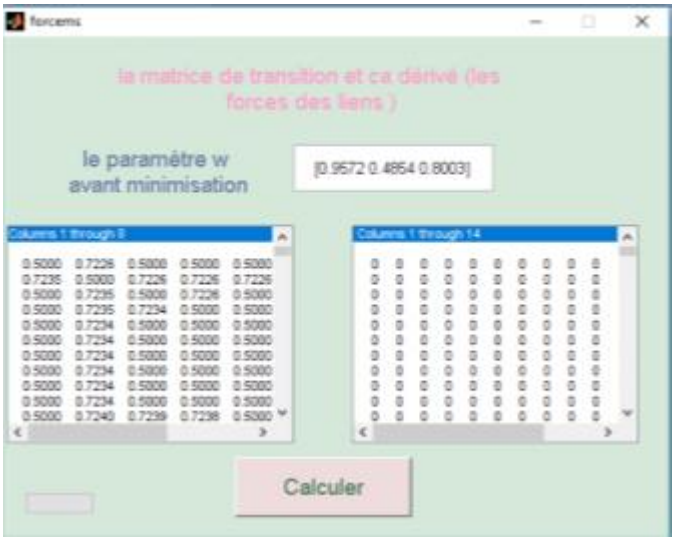


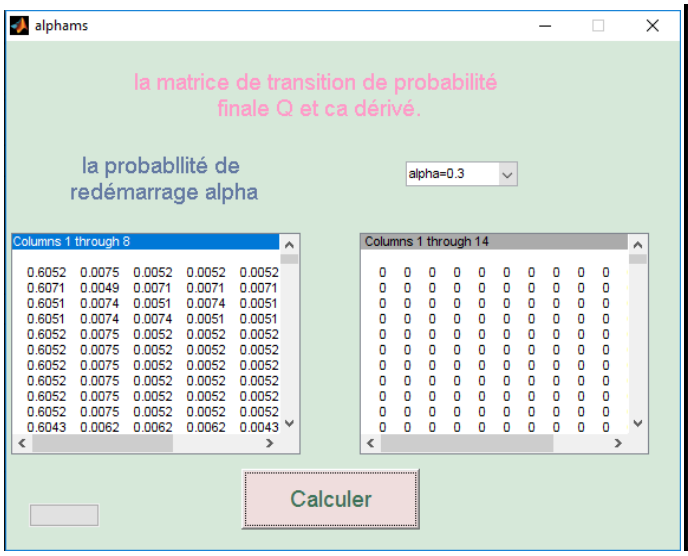


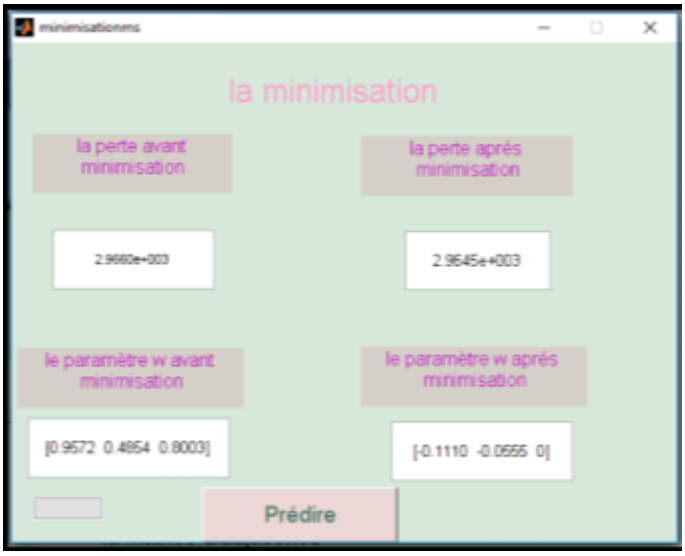
Les interfaces de différentes étapes de l'approche SRW du réseau les misérables :

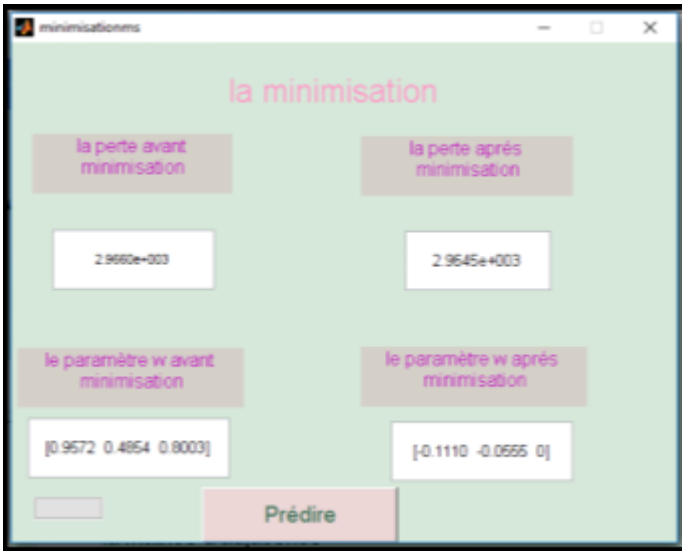



extractionnms
la matrice des caractéristiques
Columns 1 through 8
Calculer


forcenns
la matrice de transition et ca dérivé (les forces des liens)
le paramètre w avant minimisation
[0.9572 0.4854 0.8003]
Columns 1 through 8
Columns 1 through 14
Calculer

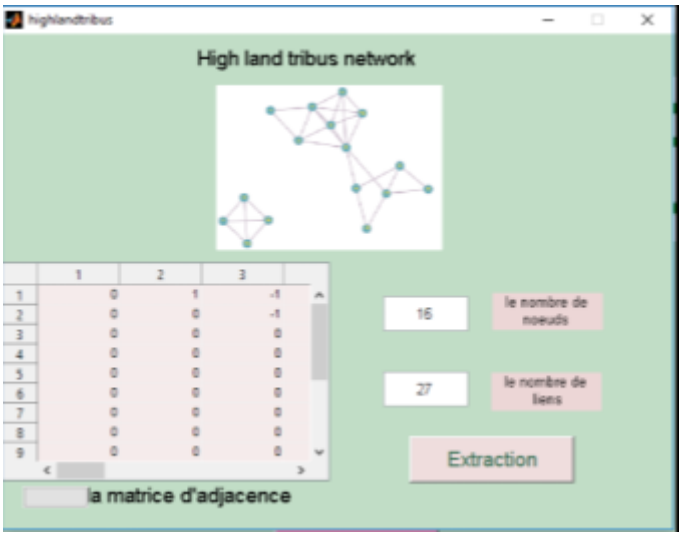

alphams
la matrice de transition de probabilité finale Q et ca dérivé.
la probabllité de redémarrage alpha
alpha=0.3
Columns 1 through 8
Columns 1 through 14
Calculer

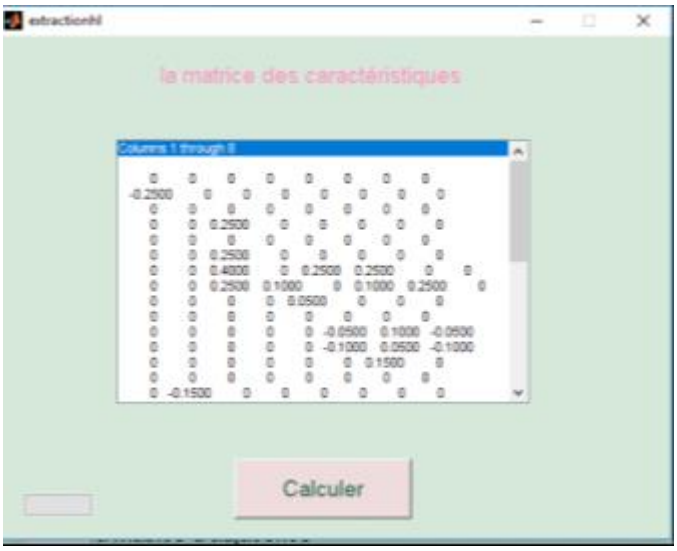

minimisationnms
la minimisation
la perte avant minimisation
la perte aprés minimisation
2.9660e+003
2.9645e+003
le paramètre w avant minimisation
le paramètre w aprés minimisation
[0.9572 0.4854 0.8003]
[-0.1110 -0.0555 0]
Prédire

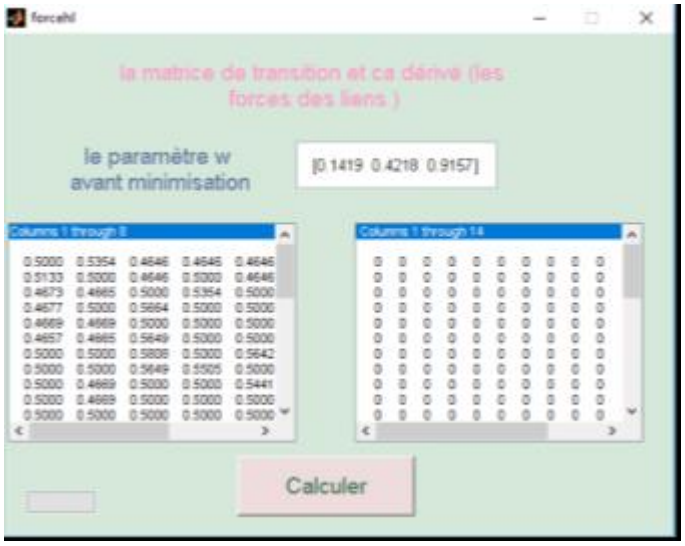

minimisationnms
la minimisation
la perte avant minimisation
la perte aprés minimisation
2.9660e+003
2.9645e+003
le paramètre w avant minimisation
le paramètre w aprés minimisation
[0.9572 0.4854 0.8003]
[-0.1110 -0.0555 0]
Prédire

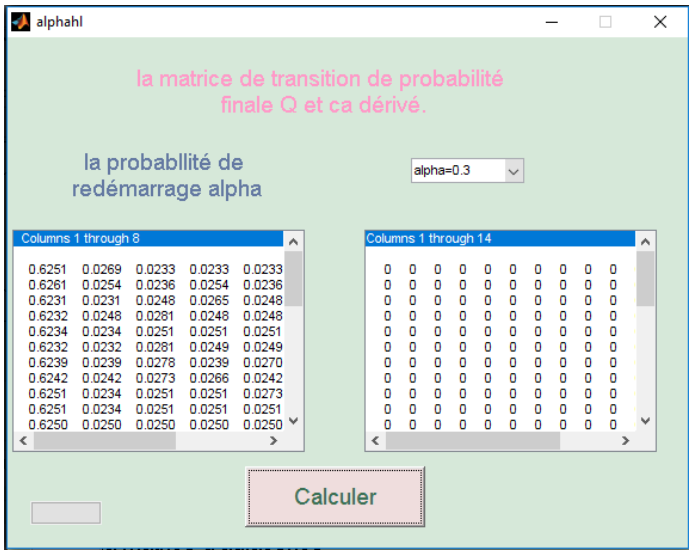

finalenns
les nouveaux liens prédits pour le réseau mésirable
le nombre de liens prédit
38
le nombre de liens totale
292
Retour
la matrice d'adjacence aprés prédiction

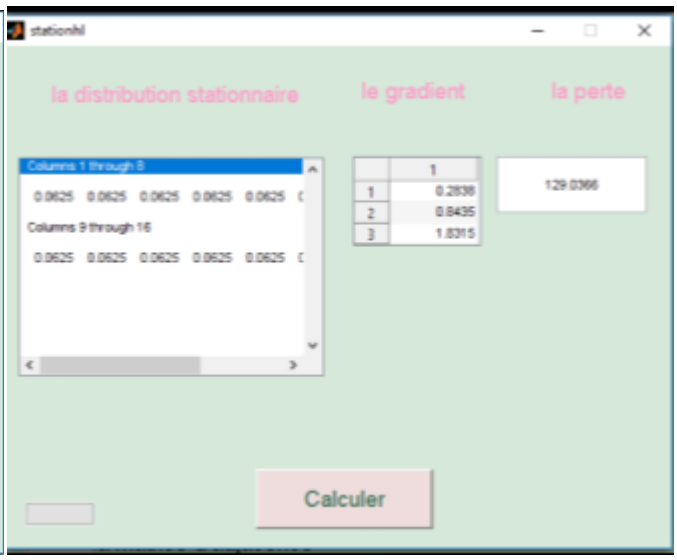
Les interfaces de différentes étapes de l'approche SRW du réseau les misérables :



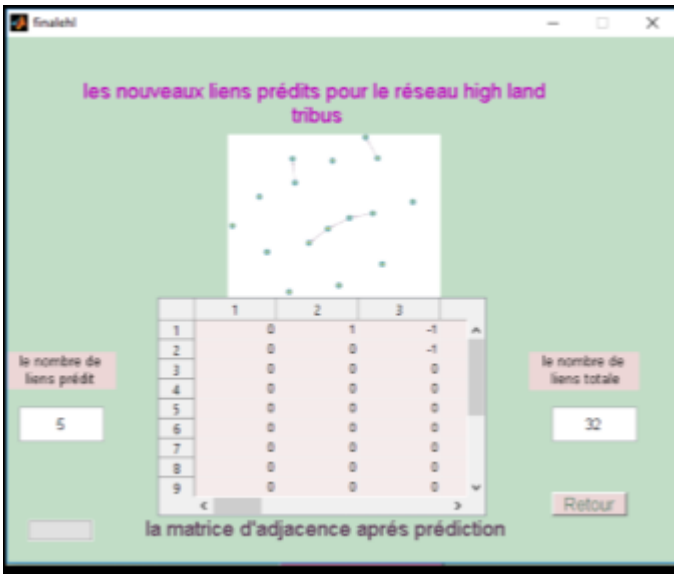









minimisationhl
la minimisation
la perte avant minimisation
la perte aprés minimisation
129.0366
128
le paramètre w avant minimisation
le paramètre w aprés minimisation
[0.1419 0.4218 0.9157]
[0 0 0]
Prédire


finalehl
les nouveaux liens prédits pour le réseau high land tribus
le nombre de liens prédit
5
le nombre de liens totale
32
Retour
la matrice d'adjacence aprés prédiction