

**Département de géomatique appliquée  
Faculté des lettres et sciences humaines  
Université de Sherbrooke**

**Application de réseaux de neurones sur les graphes pour  
Le post-traitement de détection du réseau routier**

Samir Wali

Essai présenté pour l'obtention du grade de Maître en sciences géographiques (M. Sc.),  
cheminement géo-développement durable

Eté 2023

© Samir Wali, 2023

Directeur d'essai : Samuel Foucher (Département de géomatique appliquée, Université de Sherbrooke)

Membre du jury : Mathieu Turgeon-Pelchat, (Centre canadien de cartographie et d'observation de la Terre)

## Résumé

Les réseaux de neurones convolutifs sont couramment utilisés pour extraire des informations à partir d'imagerie très haute résolution, comme les réseaux routiers. Cependant, lors de l'étape d'inférence, la prédiction du réseau routier est souvent caractérisée par une segmentation discontinue. Cette situation conduit à des erreurs et une qualité de cartographie réduite. Ce projet vise à résoudre les problèmes de segmentation discontinue dans la prédiction des réseaux routiers et d'explorer l'utilisation de réseaux de neurones sur des graphes (GNN) pour traiter directement la donnée vectorielle du réseau routier. Contrairement aux CNN, qui traitent les données sous forme de matrices, les GNN sont conçus pour travailler avec des données structurées telles que les graphes. En utilisant cette approche, il est possible de capturer des informations topologiques et géométriques sur le réseau routier, ce qui pourrait améliorer la qualité de la prédiction. Ce travail vise donc à exploiter les performances des GNN pour la prédiction des liens de réseaux routiers. Les résultats de cette étude pourraient permettre de mieux comprendre les avantages et les limites de cette approche pour ce type de tâche et ouvrir la voie à de nouvelles méthodes pour améliorer la qualité de la cartographie des réseaux routiers.

**Mots clés :** réseaux de neurones convolutifs, imagerie très haute résolution, segmentation discontinue, réseaux routiers, graphes, réseaux de neurones sur des graphes, données vectorielles, topologie, géométrie, prédiction.

## **Remerciements**

Je voudrais prendre un moment pour exprimer toute ma gratitude envers mes parents pour leur soutien inébranlable tout au long de mes études. Sans leur encouragement et leur persévérance, je ne serais jamais parvenu à atteindre la maîtrise et à réaliser mes rêves académiques. Leur confiance en moi et leur conviction que je pouvais réussir, même dans les moments les plus difficiles, m'ont donné la force de continuer et de persévérer. Bien que je sois loin d'eux en ce moment, je leur suis reconnaissant chaque jour pour tout ce qu'ils ont fait pour moi. Je suis fier d'avoir des parents si merveilleux et aimants, et je leur promets de continuer à travailler dur et à poursuivre mes objectifs avec la même passion et le même engagement qu'ils m'ont transmis.

Enfin, un grand merci à mon directeur, Samuel Foucher, pour son aide tout au long de ce projet. Ses commentaires et ses corrections m'ont permis de réfléchir plus en profondeur à mes idées et d'affiner mon écriture. Merci pour m'avoir initié au monde fascinant des graphes et des réseaux de neurones sur les graphes. En commençant ce projet, je connaissais à peine ces domaines, mais grâce à son expertise et à son soutien, j'ai pu en apprendre davantage et explorer de nouvelles idées passionnantes. Votre enseignement m'a permis de développer ma compréhension des graphes et des algorithmes de réseaux de neurones, ainsi que d'enrichir ma formation académique et mon développement personnel.

## Table des matières

Liste des figures .....	6
Liste des tableaux .....	7
Liste des abréviations .....	8
1 Introduction.....	9
1.1 Mise en contexte.....	9
1.2 Problématique.....	9
1.3 Objectifs.....	10
2 Cadre théorique.....	11
2.1 Généralités sur les graphes .....	11
2.1.1 Types des graphes .....	12
2.1.2 Théorie d'application des graphes.....	13
2.2 Les graphes en géomatique.....	14
2.2.1 Analyse des réseaux spatiaux .....	14
2.2.2 Représentation d'un modèle de données .....	15
2.3 Réseaux de neurones sur des graphes (GNN) .....	18
2.3.1 Tâches d'apprentissage avec les GNN .....	18
2.3.2 Conception et fonctionnement des GNN .....	20
3 Matériel et Méthodes .....	21
3.1 Site d'étude .....	21
3.2 Données .....	21
3.3 Méthodologie.....	23
3.3.1 Traitement et exploration des données.....	25
3.3.2 Création des graphes .....	25
3.3.2.1 Exploitation des attributs géométriques et radiométriques.....	25
3.3.2.2 Représentation des couches géographiques sous forme de graphes .....	27
3.3.2.3 Création du graphe dual .....	28

3.3.3	Application des réseaux de neurones sur des graphes.....	29
3.3.3.1	Mise en place du modèle d'apprentissage .....	30
3.3.3.2	Entraînement du modèle .....	31
3.3.3.3	Principe de génération des liens négatives.....	32
3.3.3.4	Evaluation du modèle .....	33
4	Résultats.....	34
4.1	Résultats des graphes.....	34
4.2	Résultat de l'application des réseaux de neurones .....	39
4.2.1	Résultat de l'entraînement sur le réseau routier canvec .....	39
4.2.2	Résultat de l'évaluation sur la détection des routes .....	41
5	Interprétation et discussion des résultats .....	44
6	Conclusion .....	46
7	Documentation.....	47
8	Listes des références .....	47

## Liste des figures

<b>Figure 1:</b>	Exemple de discontinuités dans les routes détectées.....	10
<b>Figure 2:</b>	Quelques exemples de graphes (Heinold, 2018) .....	11
<b>Figure 3:</b>	Matrice d'adjacence d'un graphe (Heinold, 2018) .....	11
<b>Figure 4:</b>	Exemple d'un graphe connexe (à gauche) et un graphe déconnecté (à droite) (Heinold, 2018).....	12
<b>Figure 5:</b>	Exemple d'un graphe non dirigé (à gauche) et dirigé (à droite) (Renton, 2021) ...	12
<b>Figure 6:</b>	Exemple d'un graphe homogène (à gauche) et un graphe hétérogène (à droite) (Elinas, 2019) .....	13
<b>Figure 7:</b>	Exemple de représentation d'un réseau de transport (Wilson, 1979) .....	16
<b>Figure 8:</b>	Topologie d'un modèle de données de réseau (Rodrigue, 2005).....	17
<b>Figure 9:</b>	Classification des nœuds (Masui, 2022).....	18
<b>Figure 10:</b>	Prédiction des liens (Masui, 2022) .....	19
<b>Figure 11:</b>	Classification des graphes (Masui, 2022).....	19

<b>Figure 12:</b> Exemple d'un GNN avec plusieurs couches de convolution de graphes (Wu et al., 2020).....	20
<b>Figure 13:</b> Localisation de la zone d'étude .....	21
<b>Figure 14:</b> Organigramme méthodologique .....	23
<b>Figure 15:</b> Aperçu des résultats des détections sur l'image satellitaire de Planetscope .....	27
<b>Figure 16:</b> Aperçu du réseau routiers Canvec sur l'image satellitaire de Planetscope .....	27
<b>Figure 17:</b> Architecture du modèle d'apprentissage (Masui, 2022) .....	30
<b>Figure 18:</b> Résumé de la division des arêtes (Masui, 2022).....	32
<b>Figure 19:</b> Graphe simple des détections de routes.....	34
<b>Figure 20:</b> Graphe simple de réseaux routier de Canvec .....	34
<b>Figure 21:</b> Aperçu sur le graphe composé des trois graphes simples .....	35
<b>Figure 22:</b> Exemple d'interaction entre deux types de nœuds.....	36
<b>Figure 23:</b> Graphe dual des détections des routes .....	37
<b>Figure 24:</b> Graphe dual de réseaux routier de Canvec .....	38
<b>Figure 25:</b> Les courbes ROC et DET obtenues en testant le modèle .....	40
<b>Figure 26:</b> La matrice de confusion obtenu en testant le modèle.....	40
<b>Figure 27:</b> Les courbes ROC et DET obtenus en évaluant le modèle.....	42
<b>Figure 28:</b> La matrice de confusion obtenue en évaluant le modèle.....	42
<b>Figure 29:</b> Aperçu sur le graphe des détection et les liens négatives créés (en rouge) .....	43
<b>Figure 30:</b> Aperçu sur le graphe des détection résultant de la prédiction des liens .....	44
<b>Figure 31:</b> Quelques exemples sur les liens générés après la prédiction des liens ( les liens bleus et les nœuds rouges représentent les résultats de détection avant l'apprentissage, alors que les nœuds noirs et les liens en vert foncé représentent les résultats après l'apprentissage) .....	44

## Liste des tableaux

<b>Tableau 1:</b> Tableau des données .....	22
<b>Tableau 2:</b> Quelques librairies utilisées dans le projet.....	24
<b>Tableau 3:</b> Attributs géométriques.....	26

## Liste des abréviations

3D : Trois Dimensions.

AUC : Aire sous la courbe (« Area Under the Curve »).

CCCOT : Centre canadien de cartographie et d'observation de la Terre.

CNN : Réseau de Neurones Connectés (« Convolutional Neural Network »).

EPSG : Groupe européen d'enquête pétrolière (« European Petroleum Survey Group »).

ESRI : Institut de recherche sur les systèmes environnementaux (« Environmental Systems Research Institute »).

GNN : Réseau de neurones sur les graphes (« Graph Neural Network »).

GPKG : Géopackage (« Geopackage »).

NAD83 : Système de référence nord-américain de 1983 (« North American Datum of 1983 »).

UTM : Mercator transverse universel (« Universal Transverse Mercator »).

WGS84 : Système géodésique mondial (« World Geodetic System 1984 »).



# **1 Introduction**

## **1.1 Mise en contexte**

Avoir des cartes routières à jour est essentiel pour fournir de nombreux services importants. Par exemple, une ville a besoin de cartes routières précises pour acheminer les véhicules d'urgence, tandis qu'un système de navigation a besoin des mêmes informations afin de fournir les meilleures directions à ses utilisateurs. Étant donné que de nouvelles routes sont construites fréquemment, la mise à jour des cartes routières reste un grand défi et un problème important (Mnih et al., 2010).

L'extraction des routes à partir de l'imagerie satellite haute résolution ou des images aériennes a été un sujet de recherche brûlant dans le domaine du traitement des images de télédétection au cours des dernières décennies (Yan et al., 2022). Et vu que de très vastes zones doivent être prises en compte, le processus de mise à jour est coûteux et prend du temps. Pour cette raison, la détection automatique des routes dans l'imagerie aérienne à haute résolution a attiré beaucoup d'attention dans la communauté de la télédétection (Mnih et al., 2010). Au cours des dernières années, l'apprentissage profond a obtenu d'excellents résultats dans la segmentation sémantique, la détection d'objets et d'autres domaines, en particulier des méthodes d'extraction des réseaux routiers (Yan et al., 2022).

Les réseaux de neurones convolutifs (CNN) sont un type de modèle d'apprentissage profond qui ont été largement utilisés dans plusieurs tâches telles que la classification d'images, la détection d'objets et la segmentation sémantique. Dans le cadre de l'extraction routière, les CNN peuvent être utilisés pour identifier et extraire automatiquement les réseaux routiers à partir d'images satellitaires ou aériennes (Wei et al., 2019).

Jusqu'à présent, des améliorations remarquables ont été apportées dans l'utilisation des réseaux de neurones, mais cependant, lors de l'inférence, les CNN rencontrent quelques problèmes dans l'extraction du réseau routier.

## **1.2 Problématique**

Un problème courant dans l'extraction de routes à l'aide de CNN est la discontinuité des routes. Cela peut se produire lorsque les routes sont fragmentées ou séparées en plusieurs segments, ce qui entraîne des difficultés à relier les segments pour former un réseau routier complet. Cela peut être particulièrement difficile dans les zones rurales où les routes sont moins bien définies,

dans l'imagerie satellite, où la résolution peut ne pas être suffisante pour distinguer clairement les routes des autres éléments de l'image.



**Figure 1:** Exemple de discontinuités dans les routes détectées

Pour résoudre ce problème, Il existe différentes stratégies pour pallier à ces problèmes (fonction de perte, modèles spécialisés et post-traitements des données) néanmoins les résultats restent mitigés. Une des approches consiste à utiliser une étape de post-traitement pour connecter les segments de route extraits en un réseau routier complet. Cela peut impliquer l'utilisation d'algorithmes basés sur des graphes pour identifier les connexions les plus probables entre les segments de route.

### **1.3 Objectifs**

L'objectif principal de cet essai consiste à l'application d'architectures de type GNN à partir des données vectorielles existantes, de réseaux routiers, afin de valider et d'inférer les portions manquantes (prédiction de liens) d'un graphe.

Le premier objectif spécifique est de représenter les données sous forme de graphes incluant les attributs des nœuds, attributs des liens et graphes hétérogènes.

Le second objectif spécifique est d'explorer la formation des attributs du graphe et d'agréger différentes sources d'information afin de l'enrichir.

Le troisième objectif spécifique est de traiter les données de détection pour nettoyer le bruit et les morceaux manquants.

## 2 Cadre théorique

### 2.1 Généralités sur les graphes

Un graphe est une structure de données constituée d'un ensemble de sommets (ou nœuds) et d'un ensemble d'arêtes qui relient ces sommets. Les sommets représentent les objets ou les entités dans un graphe. Ils sont souvent représentés sous forme de points ou de cercles dans la représentation visuelle. Chaque sommet est identifié par un identifiant unique, tel qu'un chiffre ou une lettre. Les arêtes représentent les connexions ou les relations entre les sommets. Ils sont souvent représentés sous forme de lignes ou de flèches dans la représentation visuelle (Renton, 2021). Les graphes peuvent être représentés de différentes manières, par exemple avec une matrice d'adjacence ou une liste d'adjacence (Pelle, 2005).

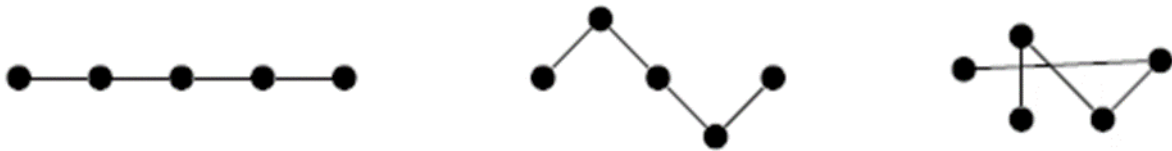
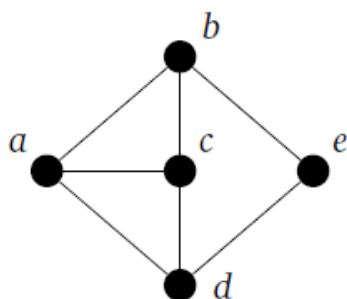


Figure 2: Quelques exemples de graphes (Heinold, 2018)



	<i>a</i>	<i>b</i>	<i>c</i>	<i>d</i>	<i>e</i>
<i>a</i>	0	1	1	1	0
<i>b</i>	1	0	1	0	1
<i>c</i>	1	1	0	1	0
<i>d</i>	1	0	1	0	1
<i>e</i>	0	1	0	1	0

Figure 3: Matrice d'adjacence d'un graphe (Heinold, 2018)

La connexité dans la théorie des graphes fait référence à la propriété d'un graphe où il existe un chemin entre deux sommets (Pelle, 2005). Un graphe est dit connexe s'il existe un chemin entre deux sommets du graphe, et déconnecté s'il n'y en a pas. Un composant connexe est un sous-graphe d'un graphe déconnecté dans lequel il existe un chemin entre deux sommets quelconques du sous-graphe (Wilson, 1979).



**Figure 4:** Exemple d'un graphe connexe (à gauche) et un graphe déconnecté (à droite)  
(Heinold, 2018)

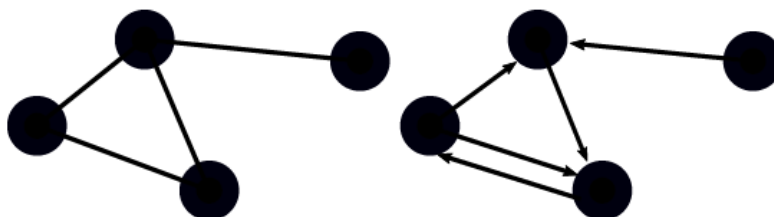
Les morceaux d'un graphe déconnecté sont appelés ses composants. Par exemple, dans le graphe déconnecté ci-dessus à droite, les trois composants sont le triangle, les deux sommets de droite reliés par une arête, et le sommet unique au sommet (Heinold, 2018).

Les graphes peuvent être utilisés pour modéliser une grande variété de phénomènes du monde réel, tels que les réseaux de routes, les connexions sociales et les dépendances de données et des différentes autres applications tels que la recherche du chemin le plus court entre deux points dans un réseau de transport ou la recherche de communautés d'objets liés dans un réseau (Béthune et al., 2020).

### 2.1.1 Types des graphes

Dans la théorie des graphes, on peut trouver différents types des graphes dont les principaux sont : les graphes dirigés et non dirigés, et les graphes hétérogènes et homogènes (Wu, 2021).

Un graphe dirigé est un type de graphe dans lequel les arêtes ont une direction et sont représentées par des flèches. Chaque arête d'un graphe orienté a un sommet de départ spécifique (la queue) et un sommet de fin spécifique (la tête). Alors que dans un graphe non dirigé, les arêtes n'ont pas de direction et sont représentées par des lignes. Dans un graphe non dirigé, il n'y a pas de distinction entre les sommets de début et de fin d'une arête, et les arêtes peuvent être parcourues dans les deux sens (Renton, 2021).



**Figure 5:** Exemple d'un graphe non dirigé (à gauche) et dirigé (à droite) (Renton, 2021)

Un graphe homogène est un graphe dans lequel tous les nœuds ont le même type ou les mêmes propriétés, ce qui les rend de nature uniforme. Alors qu'un graphe hétérogène est un graphe dans lequel les nœuds ont des types ou des propriétés différentes, ce qui les rend de nature diverse. Le traitement des graphes hétérogènes est plus complexe car la structure du graphe est plus diversifiée et les algorithmes doivent prendre en compte les différents types de nœuds et de relations, ce qui nécessite souvent des algorithmes plus sophistiqués pour gérer les diverses données et relations (Pancino et al., 2022).



**Figure 6:** Exemple d'un graphe homogène (à gauche) et un graphe hétérogène (à droite)  
(Elinas, 2019)

### 2.1.2 Théorie d'application des graphes

La théorie de l'application des graphes est l'étude des modèles mathématiques et des algorithmes utilisés pour représenter et analyser divers phénomènes du monde réel sous forme de graphes. On peut trouver plusieurs applications courantes de la théorie des graphes :

- Analyse de réseau : qui permet d'étudier la structure et les propriétés des réseaux du monde réel, tels que les réseaux sociaux, de transport et informatiques. Certaines applications courantes incluent l'identification des nœuds les plus importants d'un réseau (Zhao et al., 2020), la prédiction des liens manquants ou futurs dans un réseau en fonction de la structure actuelle du réseau (Zhang et al., 2018) et l'étude de la croissance et l'évolution des réseaux dans le temps (Dorogovtsev et al., 2002).
- Problèmes d'optimisation : où le but est de trouver la meilleure solution parmi plusieurs alternatives. Certaines applications courantes de la théorie des graphes dans les problèmes d'optimisation comprennent l'identification du chemin le plus court entre deux nœuds d'un réseau, l'identification du débit maximal d'un réseau tels que le transport et la recherche de la coupe minimale d'un réseau, qui sépare le réseau en deux ensembles disjoints de nœuds (Karger, 1995).

- La reconnaissance de formes : où le but est d'identifier les modèles et les anomalies dans les données représentées sous forme de graphes. Certaines applications courantes incluent l'identification d'une correspondance entre deux graphes avec des structures similaires tels que la reconnaissance d'images (Conte et al., 2003), le regroupement d'objets similaires en clusters en fonction des relations entre eux et la classification des objets (Zhang et al., 2009).

## **2.2 Les graphes en géomatique**

Actuellement, les graphes se retrouvent dans de nombreuses applications, où ils permettent de représenter de nombreux types de données telles que les vecteurs, les matrices ou encore les tenseurs (Kaloga, 2021). La théorie des graphes a été largement utilisée pour étudier les propriétés des réseaux spatiaux et la résilience de ces réseaux (Schintler et al., 2007).

En géomatique, les graphes sont utilisés pour représenter et analyser des données spatiales où peuvent être utilisés pour modéliser des entités géographiques telles que des routes, des bâtiments et des rivières, ainsi que les relations entre ces entités (Malinverni et al., 2020). Par exemple, un réseau routier peut être représenté sous forme de graphes, avec des routes comme tronçons et des intersections comme nœuds. Cette représentation permet l'analyse de la connectivité du réseau et le calcul efficace des itinéraires. Les graphes peuvent également être utilisés dans la modélisation 3D du terrain et dans la représentation de données provenant d'appareils de télédétection tels que le Lidar et la photogrammétrie (Malinverni et al., 2020).

### **2.2.1 Analyse des réseaux spatiaux**

La théorie des graphes se concentre sur la modélisation des réseaux d'éléments connectés et les connexions entre les éléments peuvent être identifiées grâce aux réseaux. Les algorithmes des graphes ont été largement utilisés dans les données géo-spatiales et en géomatique, qui est le domaine qui traite de la collecte, de l'analyse et de la gestion des données géographiques, pour représenter et analyser les relations spatiales (Zhu et al., 2022). On peut trouver plusieurs domaines d'applications où les graphes sont utilisés en géomatique :

- Analyse de réseaux : les graphes peuvent être utilisés pour représenter les réseaux de transport, tels que les routes et les chemins de fer, et pour analyser différents aspects comme les mouvements.
- Modélisation des données spatiales : les graphes peuvent être utilisés pour représenter les relations spatiales entre différentes caractéristiques, telles que les bâtiments et les parcelles de terrain, et pour analyser la distribution spatiale de ces caractéristiques.

- Analyse d'image : les graphes peuvent être utilisés pour représenter les relations entre les pixels d'une image et pour analyser les motifs et les structures de l'image.
- Gestion des données SIG : les graphes peuvent être utilisés pour représenter les relations entre différentes couches de données SIG, telles que les images satellites, les modèles numériques d'élévation et les cartes d'utilisation des sols, et pour analyser les relations entre ces couches afin d'extraire des informations (Boeglin et al., 2000).

Par exemple, en analyse d'image, les chercheurs décomposent souvent les grandes images en pixels et effectuent une analyse basée sur l'adjacence des pixels. Le nombre de pixels d'une image haute résolution pouvant être très important et pour les analyser, on peut utiliser la théorie des graphes pour représenter l'image. Cela implique de représenter les données d'image sous forme d'un graphe, où chaque pixel ou groupe de pixels est représenté comme un nœud, et les relations entre les pixels sont représentées sous forme d'arêtes. Des algorithmes basés sur des graphes peuvent ensuite être appliqués pour analyser les données d'image et extraire des informations utiles, telles que l'identification d'objets, la détection de modèles et l'extraction de caractéristiques. Cette approche est utile dans une variété d'applications en géomatiques, y compris la télédétection et les systèmes d'information géographique. Cette représentation permet aussi la capture spatiale efficace d'une image avec des coordonnées cartésiennes pour l'analyse et de créer un graphe tout en capturant les coordonnées et permet de capturer les propriétés spatiales du réseau (Schintler et al., 2007).

D'autre part, pour les données vectorielles comme la géographie des transports, la plupart des réseaux ont une base spatiale évidente, à savoir les réseaux routiers, de transit et ferroviaires, qui tendent à être définis davantage par leurs liens que par leurs nœuds. Par conséquent, tous les réseaux de transport peuvent être représentés par la théorie des graphes d'une manière ou d'une autre (Rodrigue et al., 2005).

### **2.2.2 Représentation d'un modèle de données**

La plupart des modèles de données SIG conventionnels séparent les informations en couches, chacune représentant une classe différente d'éléments géographiques symbolisés sous forme de points, de lignes et de polygones dans la majorité des cas. Dans les années 1990, ESRI (un important fournisseur de logiciels SIG) a développé le modèle de données shapefile, qui est devenu un des formats le plus courant pour stocker des informations vectorielles, notamment des points, des poly-lignes (une ligne continue composée de plusieurs segments) et des polygones. Ce format est devenu largement utilisé pour créer, stocker et analyser de

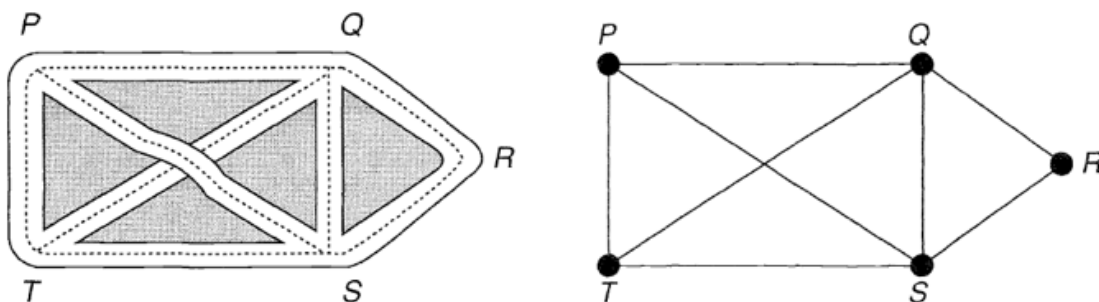
l'information sur un réseau. Pour les réseaux routiers urbains, les informations peuvent être extraites de photographies aériennes ou de cartes topographiques (Rodrigue, 2005).

La théorie des graphes a développé une représentation topologique et mathématique de la nature et de la structure des réseaux de transport. Cependant, la théorie des graphes peut être étendue à l'analyse de réseaux de transport réels et complexes en les encodant dans un système d'information. Au cours de ce processus, une représentation numérique du réseau est créée, qui peut ensuite être utilisée à diverses fins, telles que la gestion des livraisons ou la planification de la construction d'infrastructures de transport. Cette représentation numérique est très complexe car les données de transport sont souvent multimodales, peuvent couvrir plusieurs juridictions locales, nationales et internationales et ont des vues logiques différentes selon l'utilisateur particulier (Rodrigue, 2005).

Les graphes peuvent être utilisés dans la généralisation des réseaux de deux manières différentes : pour dériver des mesures quantitatives des propriétés topologiques ou métriques des arcs et des nœuds, ou pour identifier et représenter les informations topologiques importantes (Thomson et al., 1995).

Il est possible de convertir un fichier « Shape-file » en un graphe, où chaque entité du fichier de « Shape-file » (par exemple, un polygone ou une ligne) est représentée par un nœud, et les relations entre ces entités sont représentées par des arêtes. Par exemple, un fichier contenant un réseau routier peut être converti en un graphe, où chaque segment de route est un nœud et les intersections entre les segments de route sont représentés par des arêtes. Ces arêtes peuvent également contenir des informations telles que la distance ou le temps de trajet entre les nœuds.

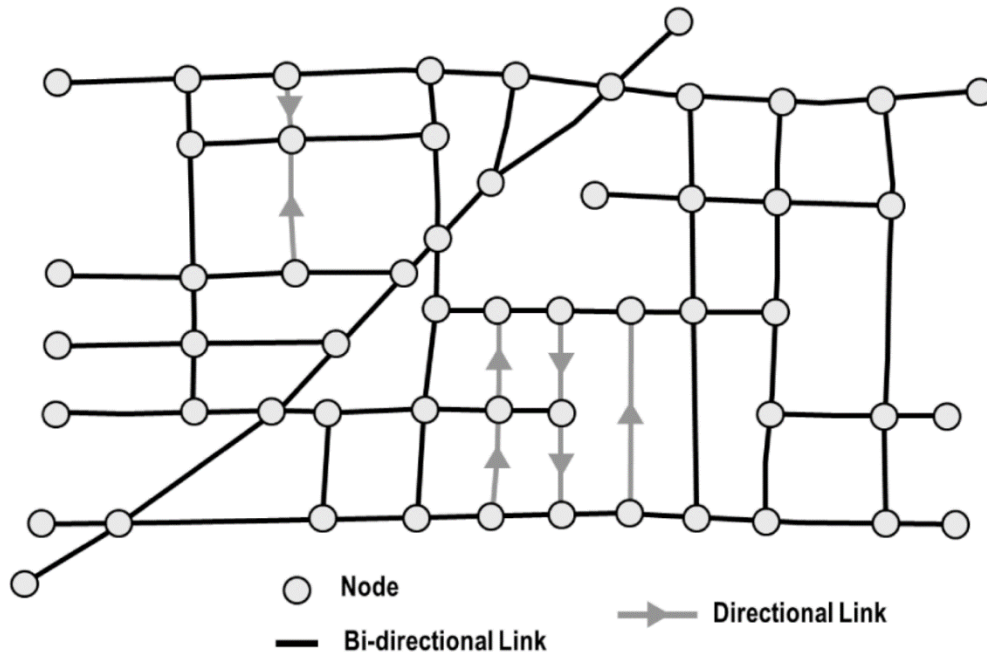
Ce type de transformation permet d'appliquer des algorithmes de graphes sur des données spatiales, et également d'utiliser l'analyse de graphes pour découvrir des modèles et des relations spatiales, tels que l'identification de lieux centraux ou l'analyse de l'accessibilité de zones.



**Figure 7:** Exemple de représentation d'un réseau de transport (Wilson, 1979)



La figure 7, à gauche, représente une partie d'une carte routière. Cette même partie peut être représentée schématiquement au moyen de points et de lignes (à droite). Les points P, Q, R, S et T sont appelés sommets, les lignes sont appelées arêtes et l'ensemble du diagramme est appelé un graphe. A noter que l'intersection des lignes PS et QT n'est pas un sommet, puisqu'elle ne correspond pas au croisement des deux routes (Wilson, 1979).



**Figure 8:** Topologie d'un modèle de données de réseau (Rodrigue, 2005)

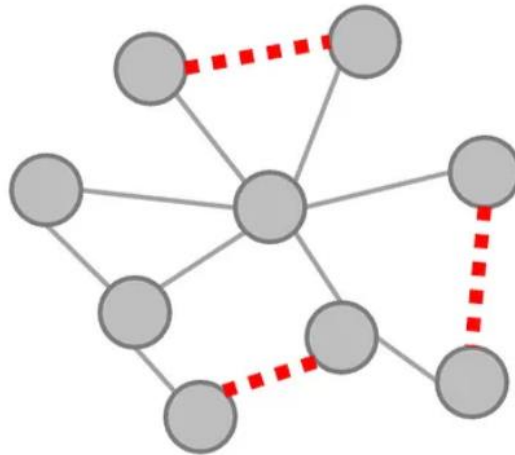
Ce graphe ci-dessus (figure 8) représente par exemple la topologie de base d'un réseau de transport urbain composé de nœuds reliés. Il a été encodé dans un modèle de données réseau pour représenter la réalité au plus près, tant topologiquement que géographiquement. Chaque nœud a été encodé avec la connectivité qu'il permet, par exemple si un virage à gauche est possible ou non (bien que cet attribut ne soit pas affiché ici). De plus, une direction a été codée dans chaque liaison (directionnelle ou bidirectionnelle) pour représenter les sens uniques. Géographiquement, chaque nœud est situé à une coordonnée qui correspond à l'intersection réelle qu'il représente. De plus, les liens entre chaque nœud ont été décomposés en plusieurs segments pour respecter la précision de positionnement de la route qu'ils représentent (Rodrigue, 2005).

Deux tables fondamentales sont requises dans la représentation topologique d'un modèle de données de réseau pouvant être stocké dans une base de données :

- Tableau des nœuds : Cette table contient au moins trois champs, un pour stocker un identifiant unique et les autres pour stocker les coordonnées X et Y du nœud. Bien que

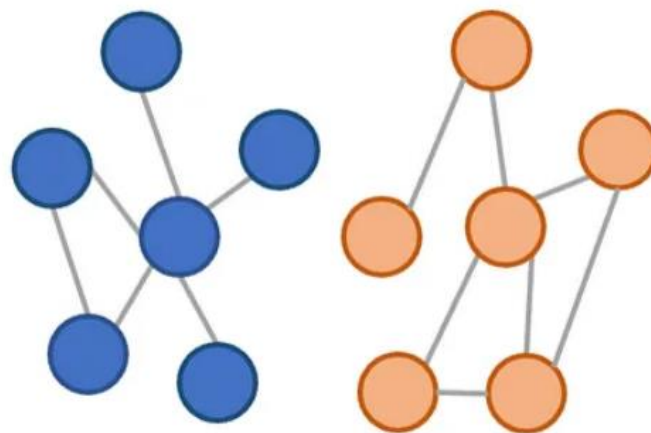


- Les tâches au niveau des liens qui se concentrent sur la classification et la prédiction des liens (Zhou et al., 2020). Cette tâche est utile dans divers domaines, tels que l'analyse des réseaux. Dans la prédiction de liens, le GNN opère sur la structure du graphe sur la base des informations des nœuds et des arêtes existantes pour apprendre une représentation des nœuds, qui peut ensuite être utilisée pour faire des prédictions sur la présence ou l'absence d'arêtes entre les nœuds (Hamilton, 2020).



**Figure 10:** Prédiction des liens (Masui, 2022)

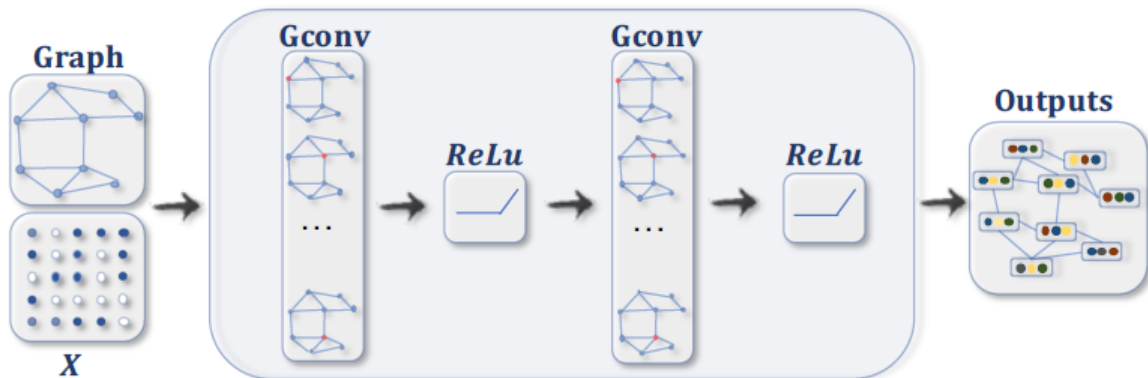
- Les tâches au niveau du graphe incluent la classification des graphes, la régression des graphes et la correspondance de graphes, qui ont tous besoin du modèle pour apprendre les représentations de graphes (Zhou et al., 2020). Dans la classification des graphes, le GNN opère sur la structure du graphe et les caractéristiques des nœuds pour apprendre une représentation du graphe dans son ensemble, qui peut ensuite être utilisée pour faire des prédictions sur l'étiquette de classe du graphe (Hamilton, 2020).



**Figure 11:** Classification des graphes (Masui, 2022)

### 2.3.2 Conception et fonctionnement des GNN

Les GNN fonctionnent en codant les nœuds d'un graphe sous forme de vecteurs, puis en utilisant ces vecteurs pour faire des prédictions sur la structure et les propriétés du graphe. L'idée de base derrière un GNN est de propager des informations à travers le graphe d'un nœud à l'autre, en utilisant des couches de convolution de graphe (Wu et al., 2020).



**Figure 12:** Exemple d'un GNN avec plusieurs couches de convolution de graphes (Wu et al., 2020)

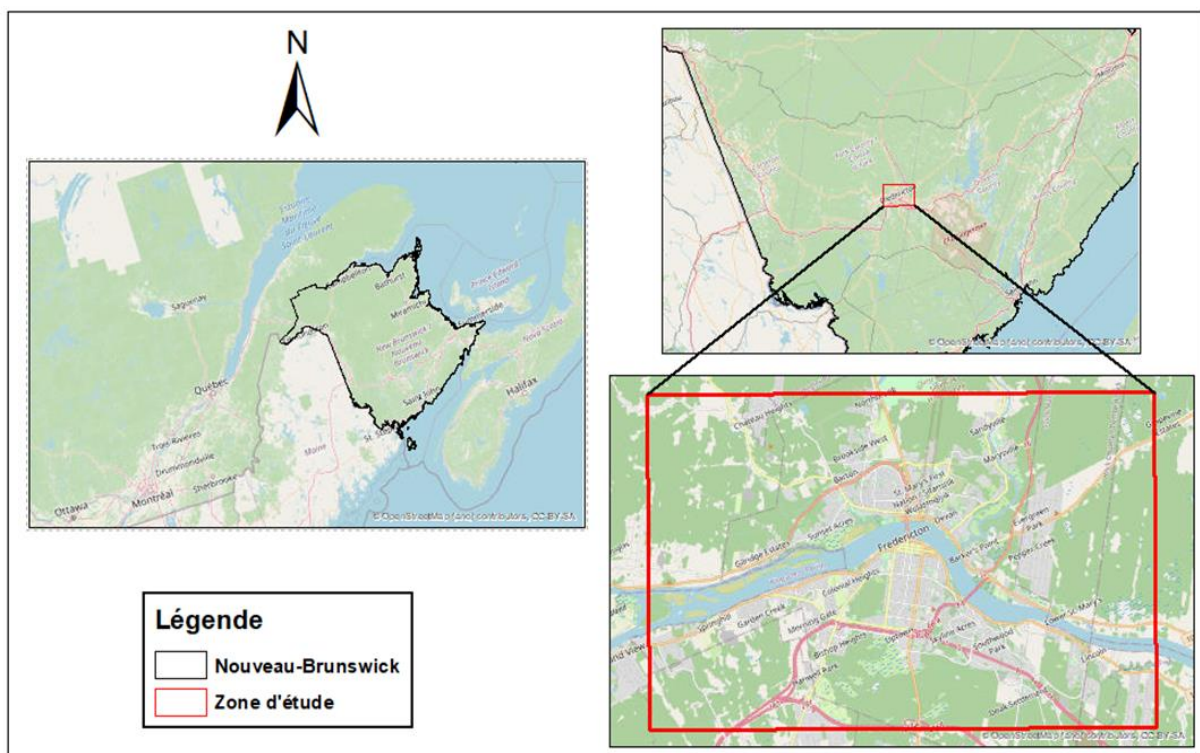
Chaque nœud du graphe se voit attribuer un vecteur de caractéristiques qui encode ses propriétés, telles que son type ou ses attributs. Ces vecteurs de caractéristiques sont l'entrée du GNN. La couche convolutionnelle du graphe effectue des opérations sur les vecteurs de caractéristiques d'un nœud et de ses voisins pour produire une représentation mise à jour du nœud. Cette couche agrège les informations des voisins du nœud, telles que leurs vecteurs de caractéristiques ou la relation entre le nœud et ses voisins, et combine ces informations avec le propre vecteur de caractéristiques du nœud pour mettre à jour sa représentation. La représentation de nœud mise à jour est ensuite transmise aux voisins du nœud sous forme de message, permettant aux informations d'être propagées dans tout le graphe. Ce processus est répété plusieurs fois pour propager les informations plus profondément dans le graphe et apprendre des représentations plus sophistiquées des nœuds. La dernière étape du GNN consiste à produire une prédiction ou une représentation du graphe dans son ensemble. Cela se fait généralement à l'aide d'une couche de lecture, qui agrège les informations de tous les nœuds du graphe pour produire un vecteur de longueur fixe qui représente le graphe (Karagiannakos, 2020).

### 3 Matériel et Méthodes

#### 3.1 Site d'étude

La zone d'étude choisie est la ville de Fredericton qui est la capitale de la province du Nouveau-Brunswick, située dans l'est du Canada. La ville est située sur les rives de la rivière Saint-Jean aux coordonnées 45 ° 57' 47.663" N, 66 ° 38' 36.146" O.

La ville de Fredericton possède un réseau routier bien développé et entretenu. Elle est desservie par l'autoroute transcanadienne, qui relie l'est et l'ouest du Canada, ainsi que par plusieurs routes provinciales importantes.



**Figure 13:** Localisation de la zone d'étude

#### 3.2 Données

Le jeu de données principale pour ce projet est le résultat de la détection du réseau routier, produit par le CCCOT, à partir d'images très haute résolution en utilisant des réseaux de neurones convolutionnels (CNN).

Pour récupérer plus d'information sur le réseau routier de Fredericton, on va essayer d'exploiter d'autres sources de données. Les données vectorielles issues de la base de données nationale canadienne Canvec sont un ensemble d'objets géométriques auxquels sont associés des attributs intéressants.

Canvec est une base de données géographiques du gouvernement du Canada qui contient des informations sur les caractéristiques géographiques du pays, notamment les routes, les rivières, les lacs et les forêts. Les données de Canvec sont disponibles gratuitement et peuvent être téléchargées sous forme des fichiers GPKG.

Les détails des données sont présentés sur le tableau ci-dessous.

**Tableau 1:** Tableau des données

<b>Nom</b>	<b>Type</b>	<b>Projection</b>	<b>Source</b>
Résultat de détection des routes	Vectorel	EPSG:2953 NAD83(CSRS)	Jeu de données produit par le CCCOT
Réseau de transport	Vectorel	EPSG :4617 NAD83(CSRS)	Données fédérales Canada CanVec (dérivé du Réseau Routier National et du Réseau Ferroviaire National)
Empreintes de bâtiments	Vectorel	EPSG :4617 NAD83(CSRS)	Données fédérales Canada CanVec (Bâtiments extraits automatiquement )
Hydrographie	Vectorel	EPSG :4617 NAD83(CSRS)	Données fédérales Canada CanVec ( les cours d'eau, les segments d'écoulement linéaire de l'eau, les obstacles hydrographiques, etc)
Images satellitaires d'une résolution de 4 m acquises le 09-11-2022	Matriciel	EPSG:32619 - WGS 84 / UTM zone 19N	Planetscope

### 3.3 Méthodologie

Le projet suit une progression conventionnelle, à partir de l'acquisition des données, jusqu'à la discussion des résultats et la conclusion (Figure 14). La première étape consiste de récupérer tous les données essentielles et les nettoyer sur notre zone d'étude. Ensuite on va se concentrer sur la formation des graphes et la recherche d'attributs pertinents pour les enrichir. Une fois les graphes formés, on va passer à la mise en place d'algorithmes de machine Learning et l'application des réseaux de neurones sur les graphes.

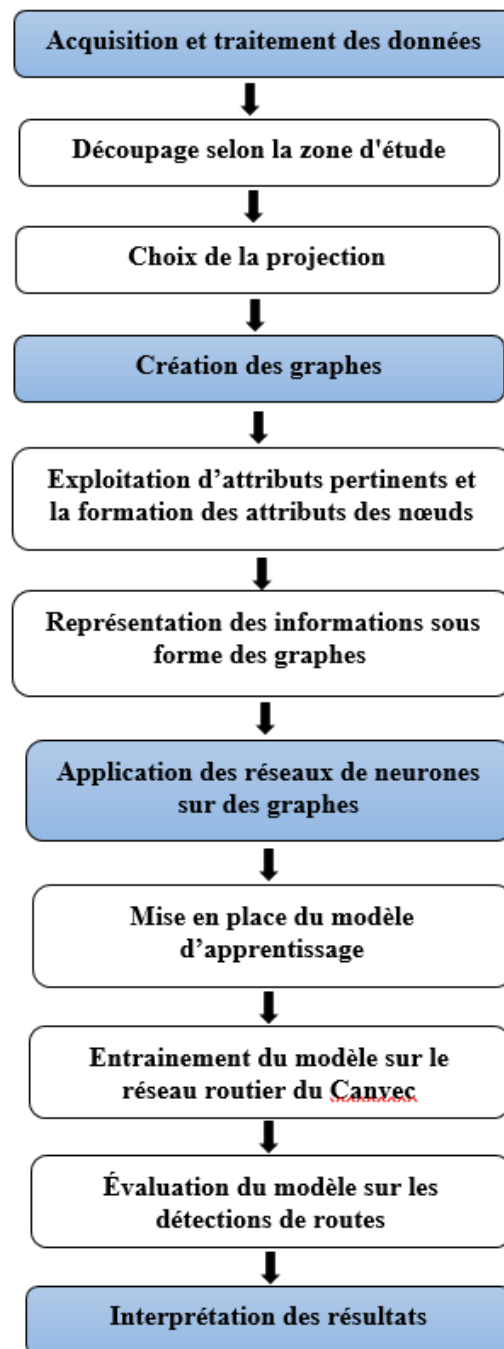


Figure 14: Organigramme méthodologique

Le logiciel QGIS sera utilisé dans un premier temps pour exploiter les couches de données et les différentes informations des tables attributaires. Il est important de mettre en forme et de nettoyer les données à notre disposition afin d'obtenir un ensemble de données homogènes et cohérentes sur la ville de Fredericton.

Une fois nos données préparées, on va utiliser Google Colab comme environnement de développement. C'est une plate-forme info-nuagique qui permet aux utilisateurs d'écrire et d'exécuter du code Python dans un navigateur Web. Il fournit un environnement gratuit avec accès à plusieurs ressources, ce qui en fait un choix populaire pour des diverses tâches comme l'analyse de données et l'apprentissage profond.

Au cours de ce projet, on va utiliser une gamme de bibliothèques différentes pour effectuer diverses tâches autour des graphes. Chacune de ces bibliothèques possède son propre ensemble unique d'outils et de fonctions qui permettra d'analyser et de visualiser les données, ainsi qu'à créer les graphes et d'entraîner les modèles d'apprentissage.

**Tableau 2:** Quelques bibliothèques utilisées dans le projet

Nom	Utilités
Geopandas	Fournit des outils pour travailler avec des données géospatiales et permet aux utilisateurs de lire, d'écrire et de manipuler les données.
Shapely	Fournit des outils pour travailler avec des points, des lignes, des polygones et d'autres objets géométriques.
Matplotlib	Fournit des outils de visualisation pour créer une variété de tracés, de diagrammes et de graphiques.
NetworkX	Fournit des outils pour l'étude des réseaux et des graphes. Il permet de créer, manipuler et analyser des graphes.
Rasterio	Fournit les outils nécessaires pour manipuler et analyser des données raster et l'analyse d'images satellite.
PyTorch	Bibliothèque Python populaire pour l'apprentissage profond qui fournit des outils pour créer des réseaux de neurones et des modèles personnalisés.



### **3.3.1 Traitement et exploration des données**

Cette étape consiste à obtenir les données requises pour ce projet. Une fois les données prêtes, on doit les découper selon la zone d'étude. Cela permet de réduire la taille des données et de se concentrer uniquement sur la zone d'intérêt.

Une fois que les couches ont été découpées, il est important de les projeter dans le même système de projection. Le système de projection (EPSG :2953) assure la cohérence spatiale entre les différentes couches de données. Cela signifie que les données géographiques sont alignées correctement et qu'elles peuvent être superposées sans problèmes lors de l'analyse.

Cette étape garantit que toutes les données sont dans un même système de projection, ce qui facilite l'analyse, la visualisation et la comparaison des différentes couches de données.

### **3.3.2 Création des graphes**

Une fois les données sont prêtes, on va passer à la création des graphes. Lors de la création du graphe, il est essentiel de transformer tous types de géométries en LineString dans le DataFrame des données pour uniformiser les données. Cette uniformisation des géométries est particulièrement importante dans le contexte de la création de graphes, car les graphes sont généralement construits à partir de segments de lignes. Ce prétraitement permettra de garantir la qualité et la cohérence des données utilisées.

#### **3.3.2.1 Exploitation des attributs géométriques et radiométriques**

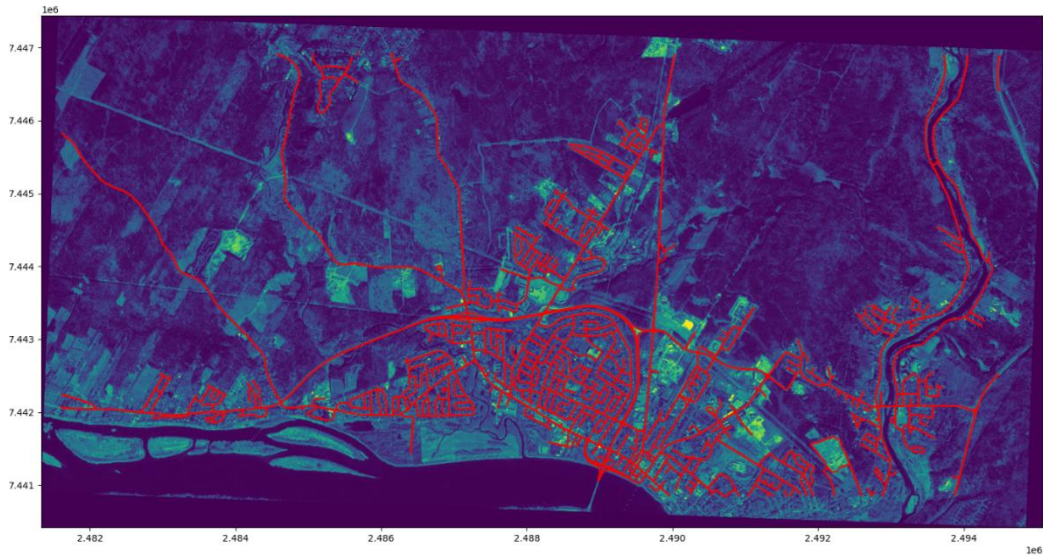
Dans cette partie, on va a essayé d'ajouter des attributs géométriques pour chaque segment de route. L'incorporation de ces attributs fournit des informations précieuses sur la structure et les caractéristiques des données du réseau et peut améliorer la précision prédictive des GNN. En particulier, les attributs géométriques tels que la courbure, l'angle et la direction d'un lien peuvent informer le GNN sur la forme et l'orientation de la structure du réseau, ce qui peut être utile dans des tâches telle que la prédiction de lien.

**Tableau 3:** Attributs géométriques

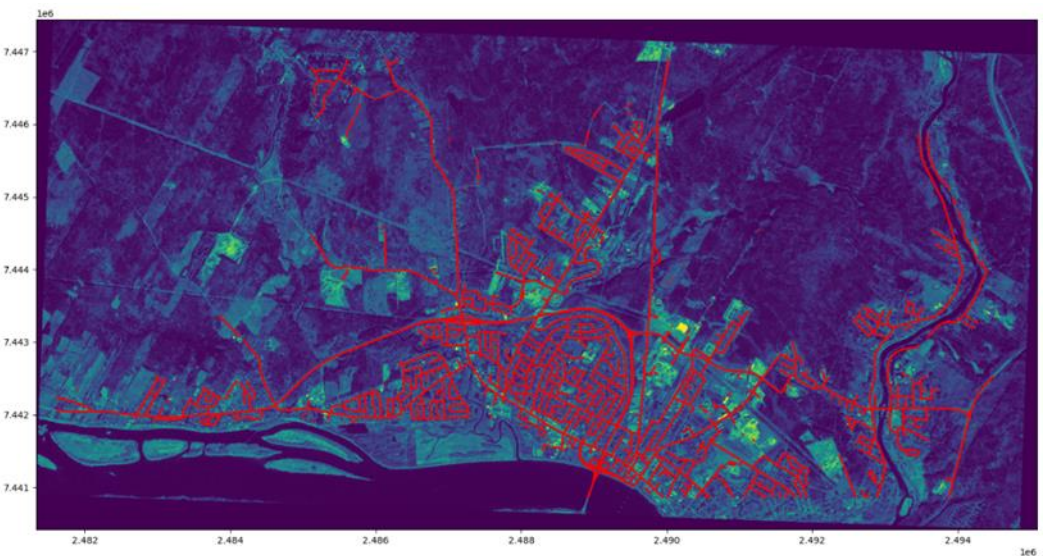
<b>Attribut</b>	<b>Définition de l'attribut</b>
K	La courbure à chaque sommet, qui est une mesure de l'écart de la courbe par rapport à une ligne droite à ce point.
t_angle	L'angle de tangente à chaque sommet, qui est l'angle entre la ligne tangente et l'axe des x.
CP_LBP	Un descripteur de modèle binaire local pour chaque sommet, qui capture la courbure locale et la forme de la courbe autour du sommet.
TAR	La représentation de l'aire du triangle pour chaque sommet, qui est une mesure de l'aire du triangle formé par le sommet et ses deux sommets voisins.
TCD	La distance centroïde triangulaire pour chaque sommet, qui est une mesure de la distance entre le centre de gravité du triangle formé par le sommet et ses deux sommets voisins, et le centre de gravité de la courbe entière.

Étant donné que la première détection est basée sur des images satellites, il sera aussi pertinent d'intégrer des informations radiométriques et de faire le lien entre les images satellites utilisées et les graphes de routes formés. Pour cela, on a utilisé les images de haute résolution (3 mètres) provenant de la plateforme PlanetScope et à l'aide de la bibliothèque Rasterio, on peut extraire les valeurs de pixels de chaque point de la géométrie pour les canaux rouge, vert et bleu de l'image.

En utilisant les valeurs de pixels extraites, on peut former des graphes de routes précis qui reflètent les caractéristiques du terrain. Ces données radiométriques mettent en évidence l'importance de la complémentarité entre le graphe et l'imagerie aérienne pour l'apprentissage et aidera aussi à améliorer la prédiction.



**Figure 15:** Aperçu des résultats des détections sur l'image satellitaire de Planetscope



**Figure 16:** Aperçu du réseau routiers Canvec sur l'image satellitaire de Planetscope

### 3.3.2.2 Représentation des couches géographiques sous forme de graphes

La création d'un graphe à partir des routes peut être réalisée en itérant à travers chaque route où on récupère les informations de géométrie pour chaque segment dans le réseau. À partir de ces informations de géométrie, le code calcule les attributs géométriques et les valeurs des pixels. Ensuite, le graphe du réseau routier est créé en utilisant les nœuds de début et de fin de chaque segment de ligne, qui sont ajoutés en tant que nœuds. Des arêtes sont ensuite ajoutées entre les nœuds de début et de fin de chaque segment et annotées avec les différents attributs.

D'autre part, on peut aussi itérer à travers chaque bâtiment et chaque entité hydrologique et créer des nœuds pour chaque sommet de leur géométrie respective. Des attributs tels que la position et le type d'entité peuvent être associés à chaque nœud en utilisant une couche

spécifique. Des arêtes peuvent ensuite être créées entre chaque paire de sommets adjacents pour représenter la connexion entre les nœuds. En utilisant cette méthode, un graphe peut être créé pour représenter les bâtiments et les entités hydrologiques.

Dans le cadre de la modélisation géospatiale, les graphes hétérogènes sont particulièrement utiles pour représenter les relations complexes entre les différentes entités géographiques. Un graphe hétérogène est un graphe dans lequel les nœuds et les arêtes peuvent avoir des types différents et permettent de combiner des entités de différents types (des routes, des bâtiments, des entités hydrologiques) dans une seule représentation, ce qui permet d'obtenir une vue d'ensemble de l'espace et des relations entre les différentes entités.

En combinant les différents graphes, on peut créer un graphe hétérogène global qui contient toutes les informations nécessaires pour l'analyse et la modélisation géospatiale. Cette représentation permet de mieux comprendre les relations et les interactions entre les différentes entités géographiques, et d'analyser les impacts potentiels de ces relations sur les processus spatiaux. Il est possible de prendre en compte des contraintes spatiales supplémentaires lors de la prédiction de liens dans les réseaux de routes, par exemple pour restreindre la création des nouveaux liens entre les nœuds du graphe qui traversent des bâtiments ou des entités hydrologiques et favoriser la création de liens qui respectent les contraintes spatiales.

Par contre l'utilisation de graphes hétérogènes peut rendre la tâche plus complexe par rapport à l'utilisation des graphes simples. En effet, la complexité de la structure des graphes hétérogènes peut rendre la tâche de prédiction de liens plus difficile et la présence de plusieurs types de nœuds et d'arêtes différents peut nécessiter l'utilisation de techniques de modélisation plus avancées pour prendre en compte ces différentes sources d'information. Mais avec les bonnes techniques de traitement, il est possible d'utiliser efficacement les graphes hétérogènes pour résoudre une grande variété de tâches et d'exploiter pleinement ces avantages pour créer des modèles de prédiction de liens précis et efficaces.

### **3.3.2.3 Création du graphe dual**

Le graphe dual est une représentation alternative d'un graphe, dans lequel les nœuds du graphe original sont représentés par des arêtes dans le graphe dual, et les arêtes du graphe original sont représentées par des nœuds dans le graphe dual.

Cette représentation permet de mettre en évidence la topologie du graphe original de manière plus intuitive et de faciliter certaines analyses. Le fait de transformer un graphe simple en un

graphe dual permet de visualiser les connexions entre les différents liens de ce réseau et en conservant les attributs des liens sur les nœuds du graphe dual, on peut conserver des informations importantes sur les caractéristiques de ces liens.

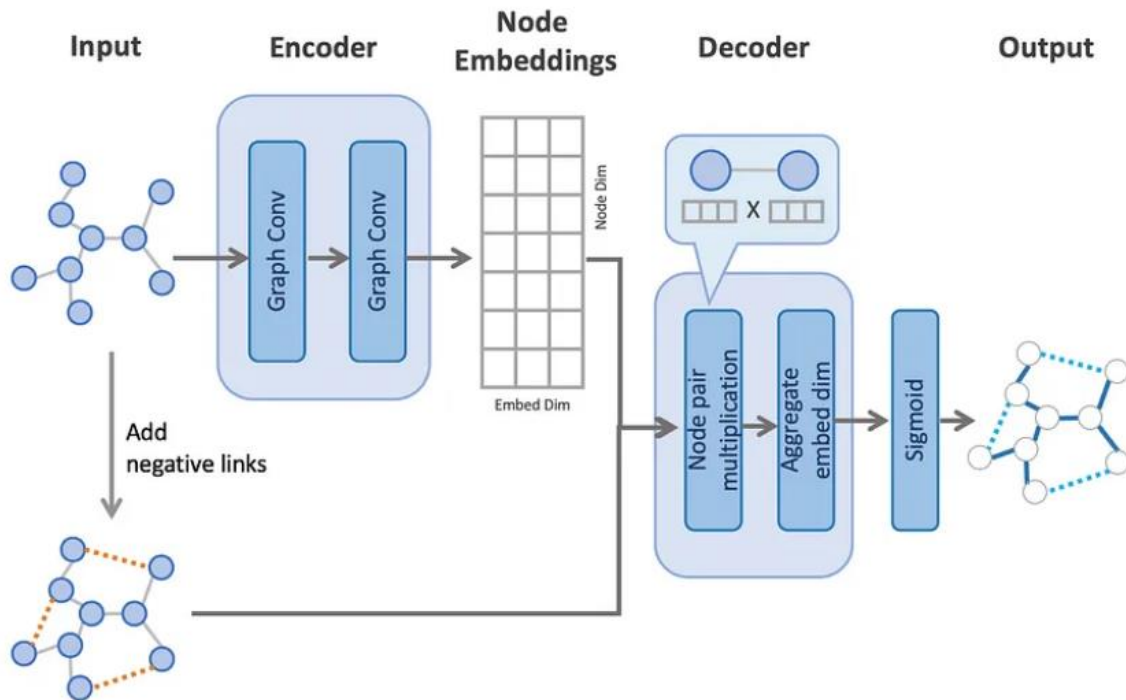
Cependant, la qualité des données d'entrée peut affecter la qualité du graphe dual résultant, et il est donc important de s'assurer que les données sont précises et complètes avant de créer le graphe dual.

### **3.3.3 Application des réseaux de neurones sur des graphes**

Lorsque on travaille avec des graphes dans le cadre de l'apprentissage, il est courant d'utiliser des bibliothèques telles que PyTorch pour entraîner des modèles d'apprentissage. Cependant, pour utiliser ces bibliothèques, il est souvent nécessaire de convertir les graphes dans un format compatible avec PyTorch. Cette conversion peut être effectuée en utilisant une fonction qui prend en entrée un graphe et renvoie un objet PyTorch. Cette conversion permet de représenter le graphe sous forme d'un tenseur d'adjacence, ainsi que des tenseurs de caractéristiques pour les nœuds et les arêtes. Cette étape est généralement effectuée avant le début de l'apprentissage, afin de préparer les données pour l'entraînement du modèle.

Dans cette partie, on va adapter une approche simplifiée en utilisant principalement la position des nœuds comme attribut dans l'application des réseaux de neurones sur les graphes de réseau routier. En se concentrant sur les coordonnées géographiques des nœuds, on vise à exploiter les informations spatiales pour analyser le réseau d'une manière simple et directe.

### 3.3.3.1 Mise en place du modèle d'apprentissage



**Figure 17:** Architecture du modèle d'apprentissage (Masui, 2022)

L'architecture du modèle de prédiction de liens (figure 17) comprend un encodeur qui traite les données du graphe d'entrée à l'aide de deux couches de convolution. Le graphe d'entrée est représenté comme un ensemble de nœuds et de liens, où chaque nœud a un vecteur de caractéristiques et chaque lien représente une relation entre deux nœuds. L'encodeur applique les deux couches de convolution aux vecteurs de caractéristiques des nœuds et des liens pour apprendre les caractéristiques pertinentes et les transformer en un ensemble de plongements de nœuds, qui représentent les caractéristiques apprises de chaque nœud. Les plongements de nœuds capturent les informations structurelles et sémantiques du graphe, qui peuvent être utilisées pour prédire la présence ou l'absence de liens entre les nœuds.

D'une autre part et après la création des nœuds d'incorporation, des liens négatifs peuvent être générés à partir de paires de nœuds qui ne sont pas connectées dans le graphe d'entrée. Le modèle peut ensuite être entraîné sur ce jeu de données augmenté, où les liens positifs et négatifs sont pondérés différemment pour éviter les biais. Le processus de formation de liens négatifs est crucial pour améliorer la capacité du modèle à généraliser et à prédire avec précision les relations entre les nœuds dans le graphe.

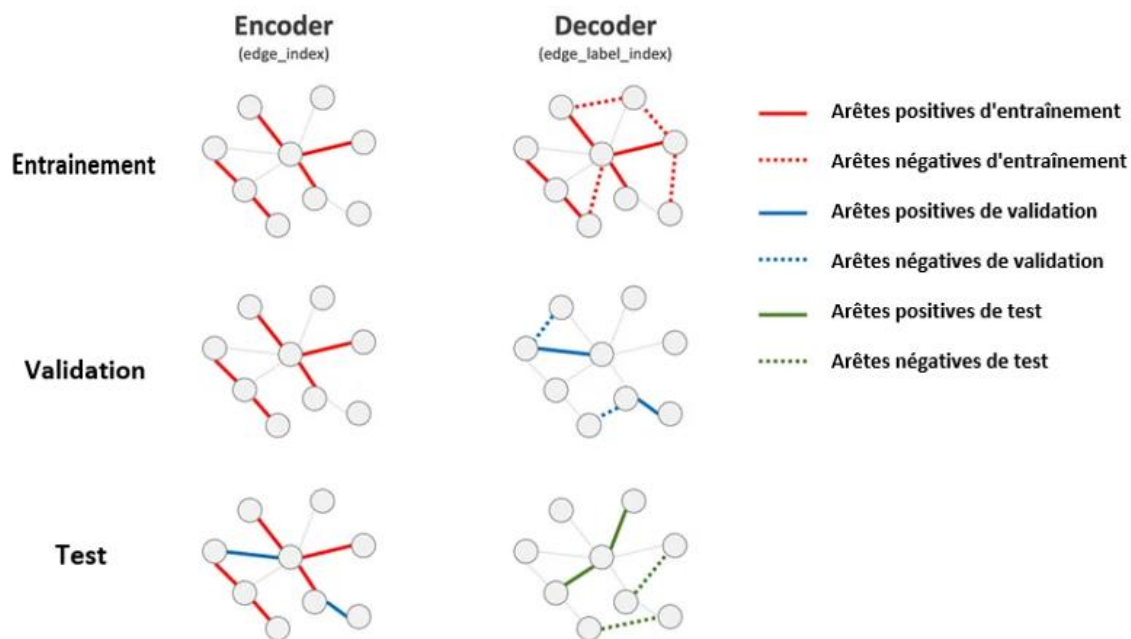
Enfin, un décodeur prend une paire de nœuds en entrée et prédit s'il existe un lien entre les nœuds correspondants. Le modèle peut être entraîné de bout en bout à l'aide d'une fonction de perte appropriée et évalué à l'aide de mesures appropriées, telle que la précision.

### **3.3.3.2 Entraînement du modèle**

Le modèle va être entraîné sur les données Canvec qui seront divisées en trois ensembles distincts : un ensemble d'entraînement, un ensemble de validation et un ensemble de test. Cette division permettra d'évaluer la capacité du modèle à généraliser et à prédire les relations entre les différentes entités dans les données.

Pour cette division, on va utiliser une méthode de séparation de liens aléatoire, qui consiste à séparer les liens dans les données en ensembles d'entraînement, de validation et de test. Cette méthode garantit que les liens dans chaque ensemble sont sélectionnés de manière aléatoire et que les ensembles sont équilibrés en termes de nombre de liens positifs et négatifs.

L'ensemble de validation sera utilisé pour ajuster les hyper-paramètres du modèle, tandis que l'ensemble de test sera utilisé pour évaluer la performance finale du modèle. L'ensemble d'entraînement sera utilisé pour entraîner le modèle en utilisant les exemples positifs et négatifs où à chaque époque d'entraînement, de nouveaux exemples négatifs sont générés en utilisant l'échantillonnage négatif des arêtes. Cela implique de sélectionner au hasard un certain nombre de paires de nœuds qui ne sont pas reliées par une arête et de les utiliser comme exemples négatifs pour l'entraînement. Ces exemples négatifs sont ensuite combinés avec les exemples positifs (paires de nœuds reliées par une arête) pour former un lot d'exemples qui est utilisé pour entraîner le modèle. La fonction de perte est ensuite calculée pour ce lot d'exemples et les poids du modèle sont mis à jour en fonction de cette perte. Ce processus sera répété pour chaque époque d'entraînement afin de permettre au modèle de s'ajuster progressivement aux données et d'apprendre à prédire les liens dans le graphe. Ces exemples négatifs permettront le modèle d'apprendre à discriminer les liens réels des liens non existants.



**Figure 18:** Résumé de la division des arêtes (Masui, 2022)

### 3.3.3.3 Principe de génération des liens négatives

Dans le cadre de la génération de liens négatifs pour le modèle, on a initialement envisagé d'utiliser la méthode de "negative sampling" de PyTorch. Cependant, on a constaté que cette approche génère des liens de manière aléatoire, sans prendre en compte la validité ou la réalité dans le contexte d'un réseau routier. Par exemple, l'utilisation de cette fonction pourrait potentiellement générer des liens entre des nœuds situés aux extrémités du graphe, qui traverseraient l'ensemble du réseau. Ce type de lien est à la fois illogique dans la réalité et peu représentatif des exemples pertinents qu'on souhaite fournir au modèle.

Par conséquent, on a entrepris de rechercher une autre méthode plus adaptée à la problématique. L'objectif était de trouver une approche qui puisse générer des liens négatifs réalistes et cohérents avec le contexte du réseau routier. On a examiné différentes alternatives et finalement on a créé une fonction plus appropriée pour cette tâche spécifique.

La fonction "proximity\_negative\_sampling" est conçue pour générer des liens négatifs basés sur la proximité dans le réseau routier. Elle utilise les coordonnées des nœuds et une méthode de recherche des nœuds les plus proches pour créer ces liens.

Tout d'abord, la fonction prend en entrée les indices des arêtes et les coordonnées des nœuds. Ensuite, la fonction essaye de trouver les nœuds voisins les plus proches d'un nœud donné. Cela permet de créer des liens négatifs réalistes qui sont cohérents avec la proximité spatiale des



nœuds dans le réseau routier. La génération des liens négatifs se fait en itérant sur un nombre défini d'échantillons négatifs à créer. Pour chaque échantillon, la fonction sélectionne un nœud source aléatoire à partir des nœuds disponibles. Ensuite, elle essaye d'obtenir les indices des nœuds les plus proches de ce nœud source, en excluant le nœud lui-même. Pour chaque nœud voisin le plus proche, la fonction vérifie s'il existe déjà un lien entre le nœud source et le nœud voisin dans les arêtes existantes ou les liens négatifs déjà générés. Elle s'assure également qu'il n'y a pas d'intersection entre le nouveau lien négatif et les arêtes existantes. Une fois qu'un lien négatif valide est trouvé, il est ajouté à la liste des liens négatifs générés. Enfin, les liens négatifs sont renvoyés sous la forme d'un tenseur contenant les indices des nœuds source et cible de chaque lien négatif.

Cette nouvelle approche tient compte de la topologie du réseau routier et des contraintes de connectivité entre les nœuds. Ainsi, elle est capable de générer des liens négatifs qui respectent les caractéristiques du réseau et qui correspondent à des exemples plausibles. Cette nouvelle fonction permettra d'améliorer la pertinence des liens négatifs qui devrait être plus réalistes et en tenant compte de la proximité spatiale des nœuds dans le réseau routier, ce qui devrait améliorer la qualité des exemples fournis au modèle pour l'apprentissage et de renforcer sa capacité à apprendre de manière efficace dans le contexte du réseau routier.

#### **3.3.3.4 Evaluation du modèle**

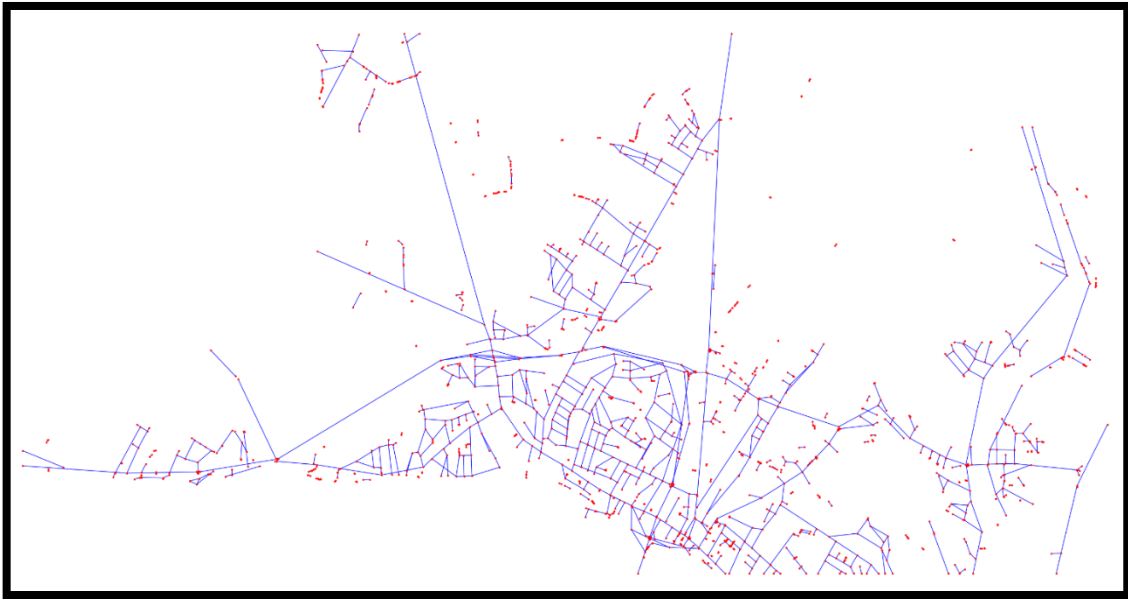
Dans cette étape, on va utiliser le modèle entraîné sur les données Canvec pour effectuer une tâche de détection de liens sur le nouveau jeu de données de détection des routes.

Ce jeu de données sera également divisé en un ensemble d'entraînement, de validation et de test. Cependant, contrairement aux données Canvec, on ne va pas utiliser une méthode de séparation de liens aléatoires. Au lieu de cela, on va utiliser les liens existants dans les données en tant que liens positifs, et générer des exemples de liens négatifs en utilisant la méthode de sous-échantillonnage négatif de l'arête. Pour ce faire, on va créer une matrice de masques pour sélectionner toutes les arêtes qui existent dans les données d'entrée. Cette matrice sera utilisée pour extraire les index des arêtes existantes et générer un nombre équivalent d'exemples négatifs en échantillonnant des paires de nœuds qui ne sont pas connectées par une arête. Ces exemples négatifs seront combinés avec les arêtes existantes pour former un nouveau jeu de données.

Enfin, on va évaluer les performances du modèle sur les données de détection et sa capacité de détecter les liens manquants entre les nœuds en utilisant plusieurs métriques comme la précision et la matrice de confusion.

## 4 Résultats

### 4.1 Résultats des graphes

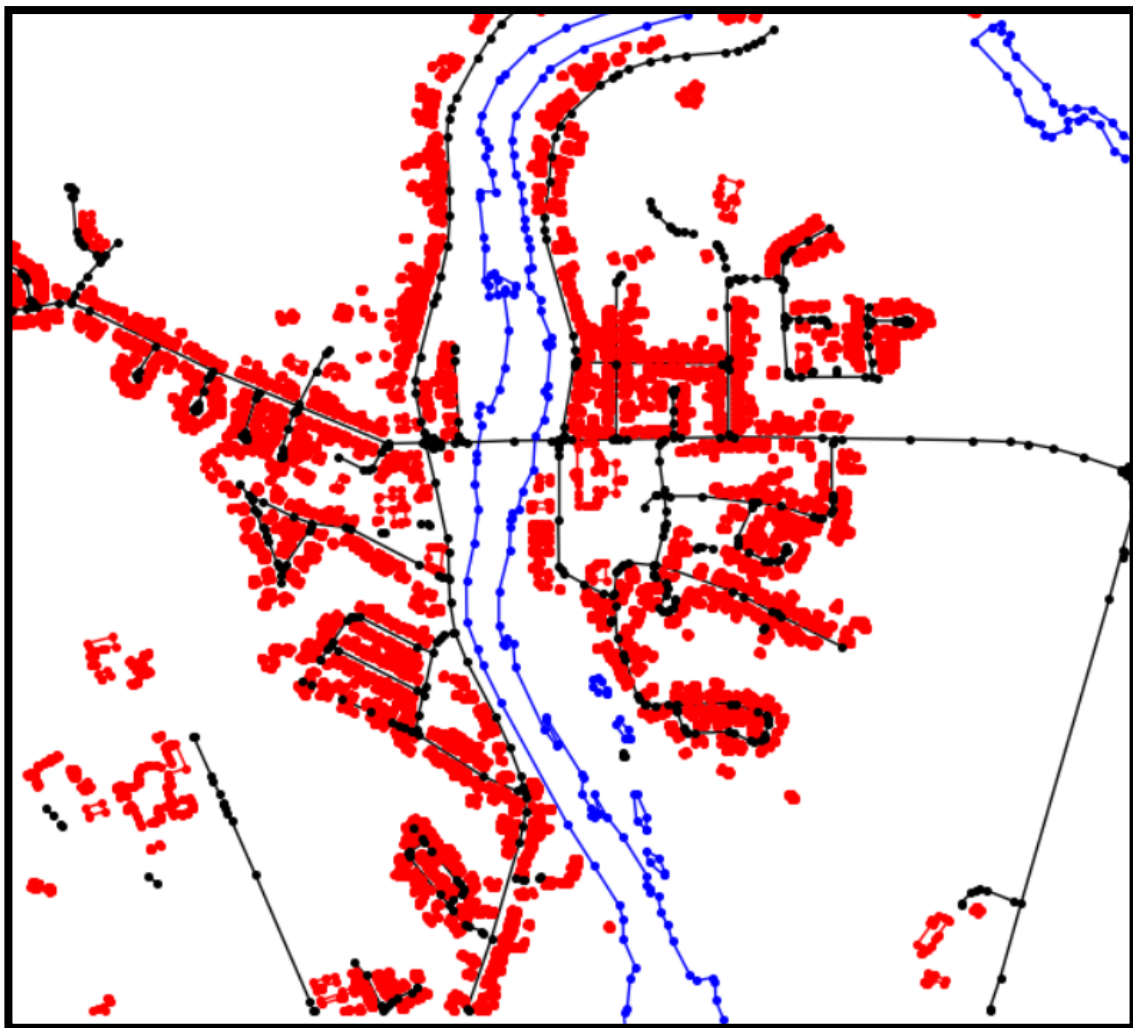


**Figure 19:** Graphe simple des détections de routes



**Figure 20:** Graphe simple de réseaux routier de Canvec

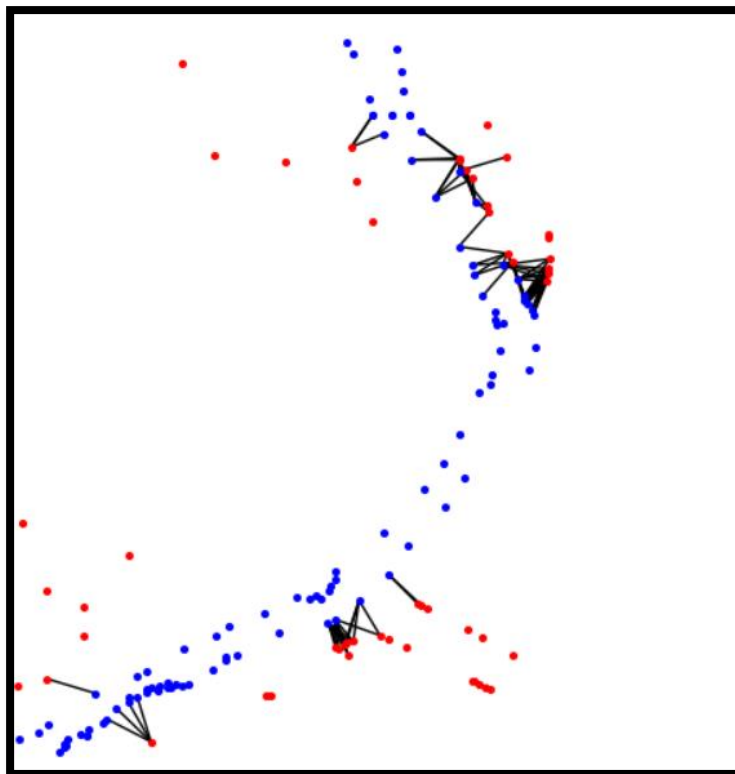
Ces résultats obtenus (figure 19 et 20) permettent de visualiser le réseau routier à partir des données d'entrée sous forme d'un graphe. Ces graphes montrent les différents segments routiers reliés entre eux par des nœuds et permet d'obtenir une vue d'ensemble de la topologie du réseau routier. Les nœuds, représentés en rouge, correspondent aux points de début et de fin de chaque segment et les lignes, représentées en bleu, correspondent aux segments routiers eux-mêmes. De plus, ce graphe permet également d'obtenir des informations détaillées sur les attributs associés à chaque segment routier, tels que les attributs géométriques et radiométriques pour chaque segment. Ces informations seront utilisées pour effectuer les analyses plus détaillées du réseau routier.



**Figure 21:** Aperçu sur le graphe composé des trois graphes simples

Ce résultat (figure 21) est un aperçu sur le graphe composé de trois types de nœuds et d'arêtes correspondants aux différents types de données. Les nœuds rouges représentent des bâtiments, les nœuds bleus représentent des caractéristiques hydrographiques telles que des rivières ou des lacs, et les nœuds noirs représentent les sommets extraits à partir de la géométrie des segments

routiers de détection. Les arêtes rouges relient les bâtiments, les arêtes bleues relient les caractéristiques hydrographiques, et les arêtes noires relient les segments routiers de détection. La combinaison de différents graphes peut permettre de créer des graphes plus complexes et plus riches en informations. Cela peut permettre d'analyser les interactions entre différents types de données, par exemple en examinant les interactions entre les nœuds hydrographiques et les nœuds routiers, on pourrait identifier des zones où les rivières et les lacs sont à proximité des routes. En combinant plusieurs types de données spatiales, on peut ainsi obtenir une vision plus complète et plus précise de la zone d'étude et ses caractéristiques.

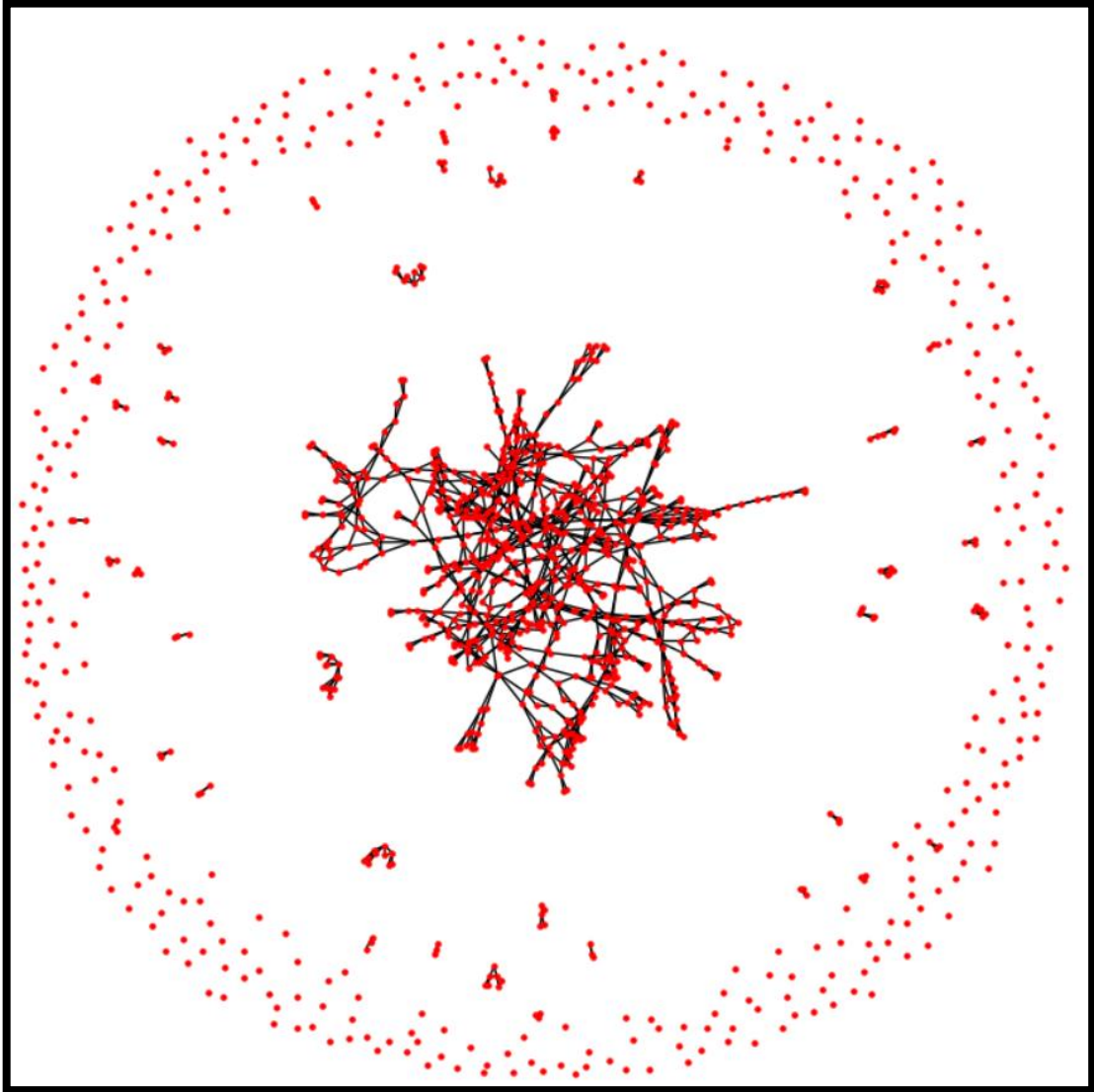


**Figure 22:** Exemple d'interaction entre deux types de nœuds

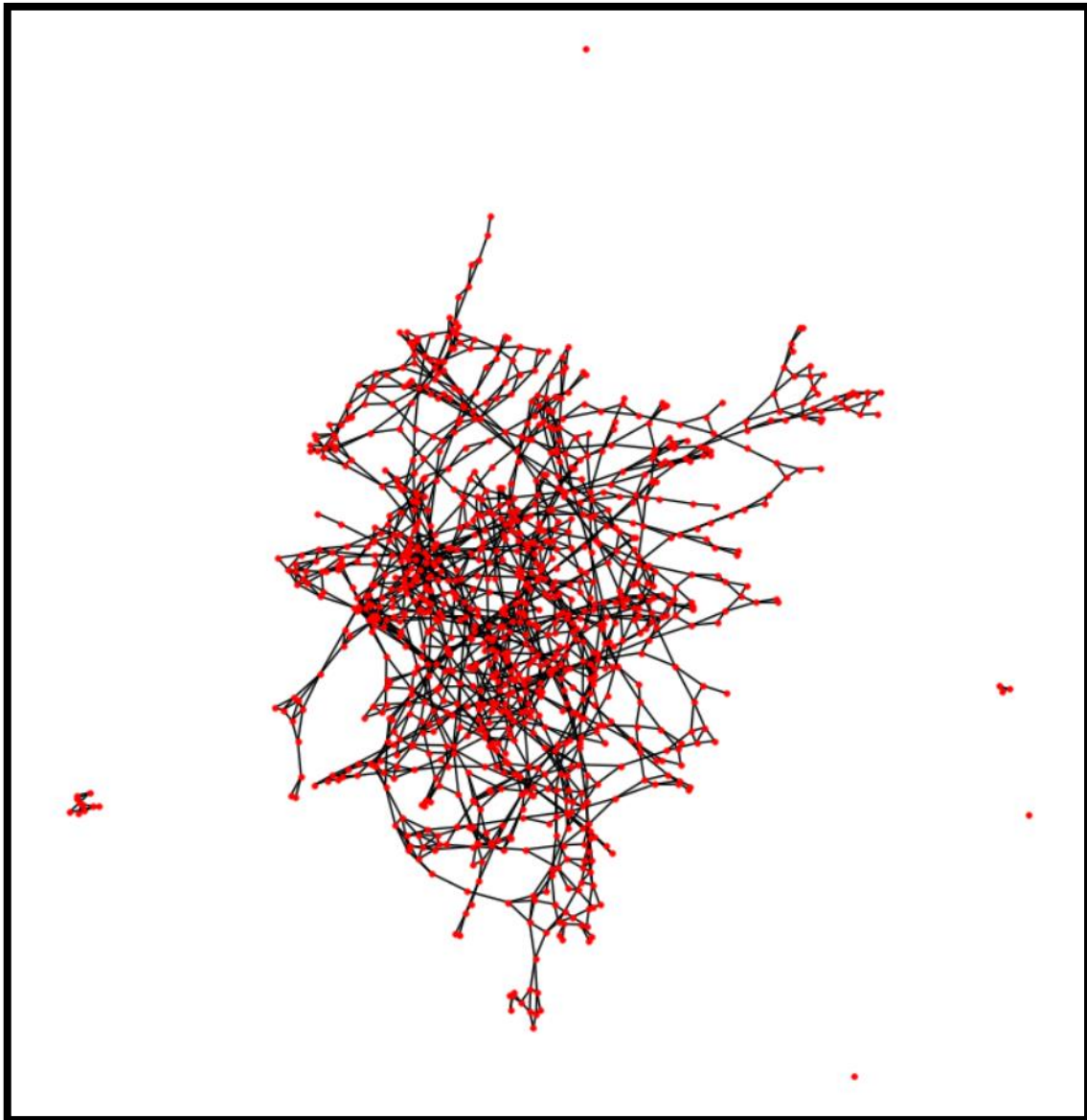
Cette exemple obtenu (figure 22) montre un graphe hétérogène contenant deux types de nœuds : des nœuds représentant des routes et des nœuds représentant des hydrographies. Des arêtes ont été ajoutées entre les nœuds de ces deux types si la distance entre les nœuds est inférieure à 100 mètres.

Ce graphe montre que les nœuds des routes et des hydrographies sont connectés via des arêtes noires représentant les liens ajoutés entre les deux types de nœuds. On observe que ces arêtes ne sont présentes que pour les nœuds qui sont proches les uns des autres, c'est-à-dire à une distance inférieure à 100 mètres.

Ces arêtes représentent donc les endroits où les routes et les cours d'eau se croisent ou sont proches les uns des autres, ce qui peut être important pour l'analyse de la connectivité hydrologique et peut être utilisé pour des analyses plus approfondies de l'interaction entre les deux types de caractéristiques géographiques.



**Figure 23:** Graphe dual des détections des routes



**Figure 24:** Graphe dual de réseaux routier de Canvec

Ces résultats sont obtenus après avoir transformé le graphe simple en graphe dual qui permet de visualiser les relations entre les segments routiers. Dans le graphe dual, chaque nœud représente un segment de route dans le graphe initial et chaque arête relie deux nœuds si les segments routiers correspondants partagent un nœud commun. Contrairement au graphe initial où les nœuds étaient des points de début ou de fin de chaque segment routier, dans le graphe dual, les nœuds représentent des segments routiers eux-mêmes. Cette représentation permet de mieux comprendre les relations entre les différents segments routiers et facilite l'identification de segments routiers similaires ou interconnectés. Par exemple, dans le graphe dual des détections des routes (figure 23), cette représentation permet de mettre en évidence plus facilement les segments de routes qui ne sont pas connectés et facilite la distinction visuelle des segments non connectés.

En outre, la fonction de transformation conserve les attributs des liens en tant qu'attributs de nœud du graphe dual, ce qui permet d'avoir une vue d'ensemble des attributs pour chaque segment routier. La visualisation du graphe dual permet ainsi d'avoir une vue d'ensemble de la topologie du réseau routier et de mieux comprendre les relations entre les différents segments routiers.

## **4.2 Résultat de l'application des réseaux de neurones**

### **4.2.1 Résultat de l'entraînement sur le réseau routier canvec**

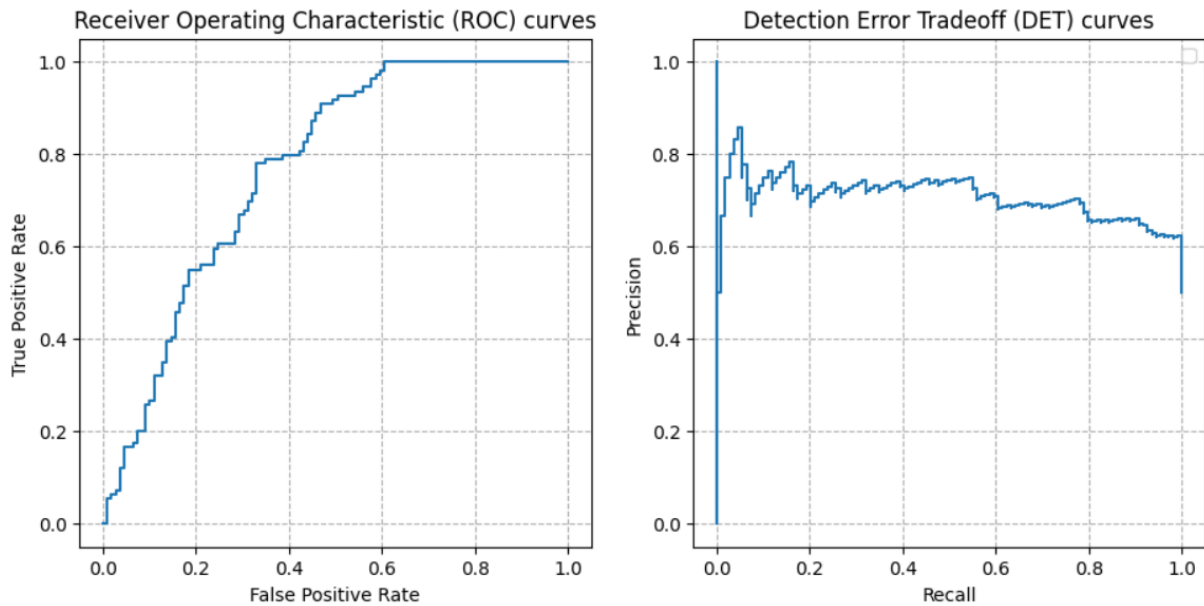
Pendant le processus d'entraînement, on cherche à obtenir le meilleur modèle en termes de performance sur les données de validation. Pour évaluer cette performance, on va utiliser le score AUC qui mesure la capacité du modèle à discriminer les exemples positifs et négatifs, en prenant en compte de tous les seuils de classification possibles. Plus le score AUC est proche de 1, meilleur est le modèle dans sa capacité à distinguer les exemples positifs des exemples négatifs. En évaluant le modèle à chaque époque d'entraînement, on va comparer le score AUC obtenu avec le meilleur score AUC enregistré jusqu'à présent. Si le nouveau score AUC est supérieur au meilleur score précédent, on va mettre à jour le meilleur score et conserver une copie du modèle correspondant. Cela permettra de suivre l'amélioration du modèle au fil des époques et de sélectionner le modèle qui présente la meilleure performance en termes de discrimination des exemples positifs et négatifs sur les données de validation.

Une fois que on a sélectionné le meilleur modèle à l'aide de la validation AUC, il est important de le tester sur les données de test pour évaluer sa performance réelle sur de nouvelles données. Cela permettra de vérifier si le modèle généralise bien au-delà des données d'entraînement et de validation.

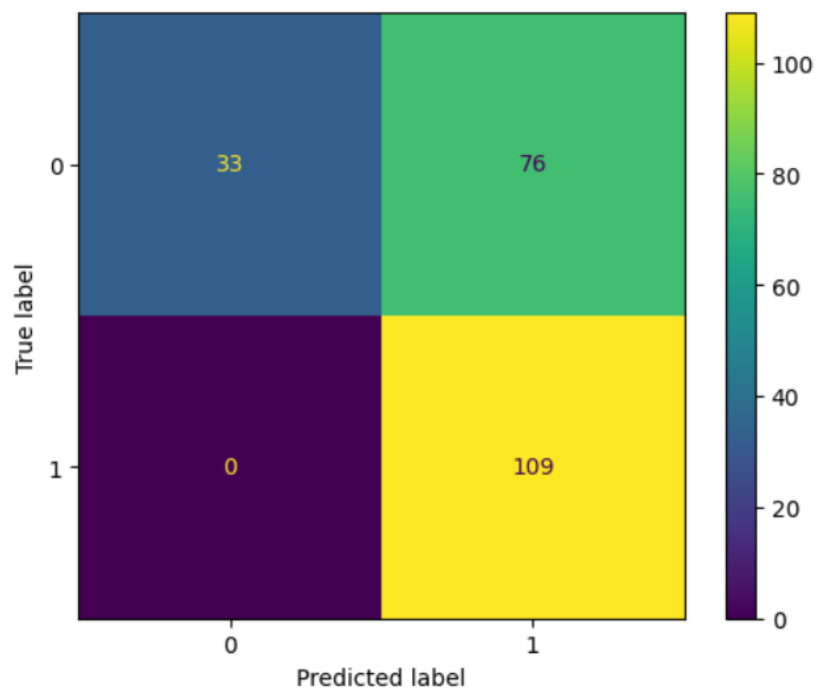
En plus, il est utile de visualiser les courbes ROC (Receiver Operating Characteristic) et DET (Detection Error Tradeoff). Les courbes ROC représentent le taux de vrais positifs (TPR) en fonction du taux de faux positifs (FPR) pour différents seuils de classification. Les courbes DET, quant à elles, permettent à évaluer visuellement comment le modèle GNN parvient à ajuster le seuil de classification pour atteindre un meilleur équilibre entre la réduction des erreurs de détection et la maximisation de la détection des exemples positifs

D'autre part, la matrice de confusion est une autre mesure importante pour évaluer la performance du modèle. Elle permet de visualiser les prédictions du modèle par rapport aux étiquettes réelles en montrant le nombre de vrais positifs, de faux positifs, de vrais négatifs et

de faux négatifs. Cela donne une vue d'ensemble de la capacité du modèle à classifier correctement les exemples positifs et négatifs. Les résultats obtenus sur les données de test sont présentés ci-dessous.



**Figure 25:** Les courbes ROC et DET obtenues en testant le modèle



**Figure 26:** La matrice de confusion obtenu en testant le modèle

Lors du test du modèle GNN après l'entraînement, on a obtenu une précision de 77 % et les valeurs de la matrice de confusion suivantes : vrai négatif (TN) : 33, faux positif (FP) : 76, faux négatif (FN) : 0 et vrai positif (TP) : 109. Les résultats montrent que le modèle a réussi à



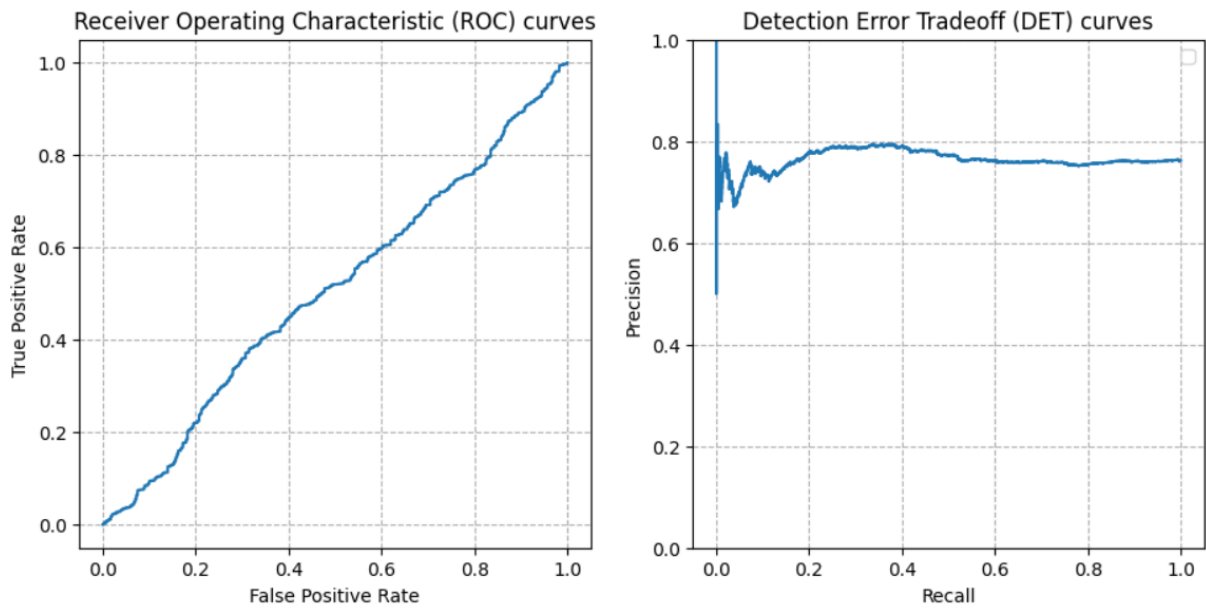
classifier correctement un nombre significatif des liens négatives, comme en témoigne le nombre élevé de vrais négatifs (TN) à 33. Cela indique que le modèle a su identifier les cas qui ne correspondaient pas à la classe positive. Cependant, le modèle présente également un certain nombre de faux positifs (FP) à 76. Cela signifie qu'il a mal classé un certain nombre de liens comme positives alors qu'elles appartenaient en réalité à la classe négative.

En revanche, les résultats montrent que le modèle n'a produit aucun faux négatif (FN), ce qui est un bon indicateur de sa capacité à identifier et classer correctement les liens de la classe positive. De plus, le nombre élevé de vrais positifs (TP) à 109 confirme que le modèle a réussi à détecter et classer de manière précise un grand nombre d'instances appartenant à la classe positive.

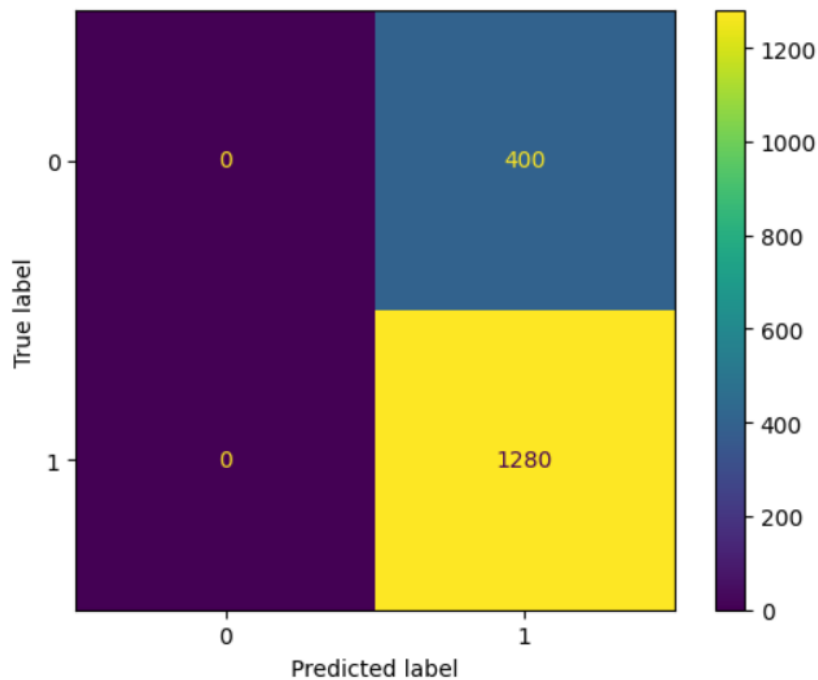
En résumé, le modèle GNN a obtenu de bons résultats en termes de vrais négatifs et de vrais positifs, mais il présente un certain nombre de faux positifs. Ces valeurs de la matrice de confusion permettent de construire les courbes ROC et DET. La courbe ROC est croissante, ce qui indique que le modèle a de bonnes performances pour distinguer les liens positifs des liens négatifs. En ajustant le seuil de classification, le modèle présente une meilleure sensibilité (taux de vrais positifs) par rapport à la spécificité (taux de faux positifs). Concernant les courbes DET, on observe une variation tout au long du test, mais globalement elles montrent une diminution légère. Cela suggère que le modèle GNN est capable de réduire les erreurs de détection à mesure que le seuil de classification est ajusté.

#### **4.2.2 Résultat de l'évaluation sur la détection des routes**

De la même manière qu'auparavant, lors de l'entraînement du modèle sur le réseau routier de canvec, on a utilisé les courbes ROC et DET pour évaluer la performance du modèle sur la détection des routes.



**Figure 27:** Les courbes ROC et DET obtenus en évaluant le modèle

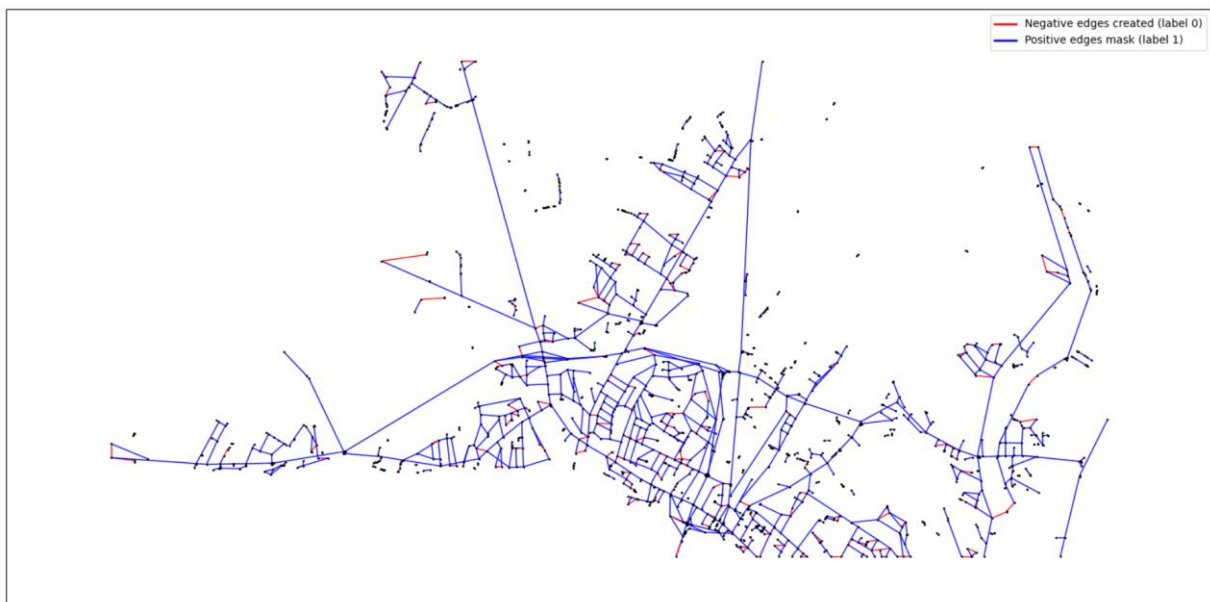


**Figure 28:** La matrice de confusion obtenue en évaluant le modèle

Lors de l'évaluation du modèle GNN, on a obtenu une précision de 50 % et les valeurs de la matrice de confusion suivantes : vrai négatif (TN) : 0, faux positif (FP) : 400, faux négatif (FN) : 0 et vrai positif (TP) : 1280. En examinant la matrice de confusion, on constate qu'il n'y a aucun faux négatifs (FN). Cela signifie que le modèle a réussi à détecter tous les exemples positifs de manière correcte, ce qui est une bonne indication de sa capacité à identifier les véritables positifs. Cependant, il y a 400 faux positifs (FP), ce qui indique que le modèle a fait des erreurs en prédisant des exemples négatifs comme positifs. Et le nombre de vrais négatifs (TN) est de 0, ce qui signifie que le modèle n'a correctement prédit aucun exemple négatif.

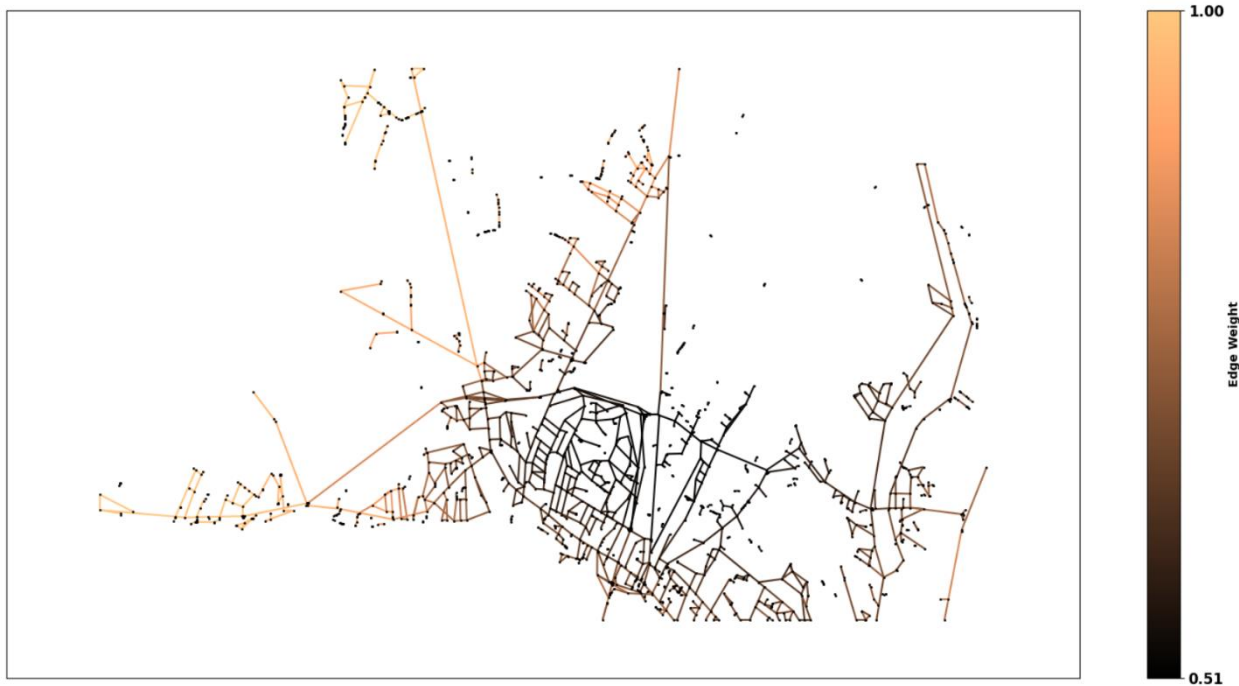
Cette performance est préoccupante, car cela indique que le modèle a une tendance à prédire des exemples négatifs comme positifs (faux positifs). Cela peut être dû à un manque de capacité du modèle à capturer les caractéristiques distinctives des exemples négatifs.

Cela est également reflété dans la courbe ROC qui, bien qu'elle soit croissante, présente une progression plus lente par rapport au test après l'entraînement. Cette performance limitée suggère que le modèle GNN a du mal à distinguer efficacement les liens positifs des liens négatifs dans cette évaluation spécifique. Malgré des fluctuations possibles dans les courbes DET, elles restent relativement stables, indiquant que le modèle n'a pas pu améliorer considérablement sa capacité de détection en ajustant le seuil de classification. Ces résultats soulignent la nécessité d'améliorer le modèle GNN pour une meilleure distinction des mauvais liens et les vrais liens dans ce scénario d'évaluation.



**Figure 29:** Aperçu sur le graphe des détection et les liens négatives créés (en rouge)

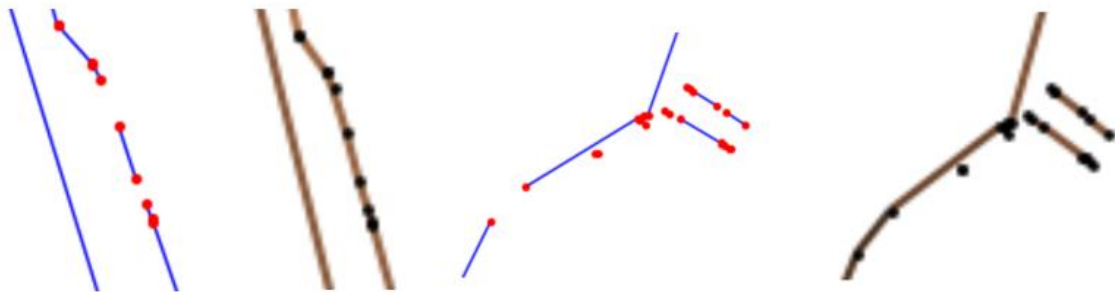
Dans la visualisation résultante (figure 29), les arêtes rouges représentent les liens négatifs créés lors de la prédiction des liens, tandis que les arêtes bleues représentent les vrais liens existants. Les arêtes rouges (étiquetées 0) indiquent les liens qui ont été créés comme négatifs mais qui en réalité ne sont pas présents dans le graphe initial. D'autre part, les arêtes bleues (étiquetées 1) représentent les vrais liens qui existent dans le graphe initial. La différenciation des couleurs permet de distinguer facilement entre les liens réels (bleus) et les liens prédits mais inexistant (rouges). Cela permet d'évaluer visuellement la précision des prédictions par rapport aux véritables connexions du graphe.



**Figure 30:** Aperçu sur le graphe des détection résultant de la prédiction des liens

Cette visualisation résultante (Figure 30) présente le graphe de détections avec les liens colorés en fonction des scores attribués par le réseau. Les nœuds du graphe sont représentés par de petits points noirs. Les arêtes du graphe sont tracées en utilisant une échelle de couleurs où les arêtes avec des poids plus faibles sont représentées par des couleurs plus sombres, tandis que les arêtes avec des poids plus élevés sont représentées par des couleurs plus claires. Cette variation de couleur permet de mettre en évidence les différences de poids entre les arêtes et facilite l'interprétation visuelle des relations dans le graphe.

## 5 Interprétation et discussion des résultats



**Figure 31:** Quelques exemples sur les liens générés après la prédiction des liens ( les liens bleus et les nœuds rouges représentent les résultats de détection avant l'apprentissage, alors que les nœuds noirs et les liens en vert foncé représentent les résultats après l'apprentissage)

Dans ces exemples (figure 31), le modèle GNN a démontré sa remarquable capacité à créer des liens pertinents entre les nœuds du graphe, en présence de ruptures et de discontinuités initiales. Ce qui est particulièrement impressionnant, c'est que le modèle a pu atteindre une précision raisonnable en se basant uniquement sur la position de chaque nœud. La capacité du modèle GNN à inférer les liens manquants en utilisant uniquement l'information de position met en évidence son efficacité à exploiter les motifs et les structures inhérents au graphe. Cela montre que, dans certains cas, la disposition spatiale des nœuds peut être une source d'information suffisamment riche pour générer des liens cohérents.

Par contre, la performance du modèle GNN peut varier en fonction de la complexité du graphe et de la disponibilité d'autres informations sur les nœuds. Durant cette étude, on a constaté que l'utilisation exclusive de la position des nœuds pour prédire les liens dans les résultats de détection de routes avec les GNN présente des limitations en termes de précision. Pour améliorer cette précision, l'une des approches que nous avons explorée consiste à incorporer les attributs géométriques calculés et les attributs radiométriques extraits. Cependant, ces attributs n'ont pas suffisamment amélioré les performances du modèle. Cette constatation suggère que les attributs utilisés jusqu'à présent ne capturent pas de manière adéquate les caractéristiques distinctives des vrais liens et des liens incorrects et qui pourraient aider à distinguer les vrais liens des liens incorrects dans la structure globale du réseau.

En ce qui concerne la génération de liens négatifs, on a développé une fonction qui pourrait améliorer la qualité des liens prédits. Cette fonction peut toutefois être gourmande en termes de calculs, mais elle a le potentiel d'améliorer les performances du modèle. Il serait donc intéressant d'évaluer l'impact de l'utilisation de cette fonction sur les performances du modèle et l'exploiter pleinement. Par exemple il serait intéressant de prendre en compte l'orientation des vrais liens lors de la génération des liens négatifs à chaque nœud. En tenant compte de l'orientation, on pourrait garantir que les liens négatifs générés sont plus adaptés et réalistes.

Une autre idée intéressante est l'utilisation du graphe complet des routes avec la géométrie complète. Cela représenterait un défi plus complexe en raison du nombre plus élevé de nœuds et de la complexité de la géométrie car cela nécessiterait une analyse approfondie et une gestion plus sophistiquée des liens négatifs. L'un des principaux défis serait d'éviter la génération de liens négatifs entre les nœuds de la même géométrie. Il faut explorer des méthodes de génération de liens négatifs qui tiennent compte de cette contrainte et qui garantissent que les liens négatifs ne se forment pas entre les nœuds qui partagent la même géométrie. Cependant, en utilisant le

graphe complet, on peut capturer des informations plus riches sur les relations spatiales entre les nœuds.

Enfin, avec la possibilité d'utiliser le graphe complet des routes, on pourrait exploiter le potentiel du graphe hétérogène créé. Cette approche pourrait permettre de prendre en compte des informations supplémentaires, telles que la localisation des bâtiments et de l'hydrographie, pour définir des conditions lors de la création des liens. Par exemple, il serait intéressant d'imposer une contrainte selon laquelle les liens générés ne doivent pas intersecter les bâtiments et les cours d'eau. Cela permettrait de rendre les liens prédits plus conformes à la réalité et d'améliorer la qualité des prédictions.

En conclusion, ce travail jusqu'à présent a montré des résultats intéressants en utilisant uniquement la position pour prédire les liens. Cependant, des améliorations sont nécessaires pour distinguer avec précision les vrais liens des liens incorrects. L'exploration de nouveaux attributs, l'utilisation de fonctions de génération de liens négatifs plus sophistiquées et l'analyse du graphe complet des routes avec des contraintes supplémentaires sont des pistes prometteuses pour améliorer la précision et l'efficacité du modèle, ce qui pourrait conduire à de meilleures performances de prédiction.

## **6 Conclusion**

Au cours de ce projet, on a réalisé une étude sur la prédiction des liens dans les graphes de réseaux routiers en utilisant seulement un attribut. On a pu observer que malgré cette approche simplifiée, le modèle a démontré des résultats encourageants en termes de performance et de précision. Cependant, on a également identifié plusieurs opportunités pour améliorer davantage ces résultats.

En utilisant uniquement la position du nœud, on a réussi à développer un modèle capable de prédire les liens dans les réseaux routiers avec une précision raisonnable. Cela montre le potentiel des méthodes d'apprentissage pour traiter des problèmes complexes même avec un nombre limité d'informations. Cependant, pour atteindre des performances optimales, il est essentiel d'explorer et d'exploiter d'autres attributs.

En intégrant des nouvelles informations dans le modèle, on pourrait améliorer la performance et la précision des prédictions. En effet, en ayant accès à des attributs plus riches et diversifiés, le modèle serait en mesure d'apprendre des patrons plus complexes et de mieux saisir les

interactions entre les différents éléments du réseau routier. Cela permettra d'obtenir des prédictions plus précises et de mieux anticiper les futurs liens potentiels.

Ce travail fut une étape cruciale dans la compréhension de la prédiction des liens dans les graphes et permet une meilleure compréhension sur l'utilisation des méthodes d'apprentissage pour résoudre les défis complexes des réseaux routiers, mais il ouvre également la voie à des nouvelles recherches et à des améliorations futures. En fin de compte, une meilleure prédiction des liens contribuera à une meilleure optimisation globale des réseaux routiers.

## 7 Documentation

Création des graphes :

<https://colab.research.google.com/drive/1eLqI10Zq6CNfie0ZRfMz8aDa8KNWHCN9?usp=sharing>

Application de GNN :

<https://colab.research.google.com/drive/1I7AkKBYgYShTwE5G5Rk6tsMYqR2orD1X?usp=sharing>

## 8 Listes des références

Béthune, L., Kaloga, Y., Borgnat, P., Garivier, A., & Habrard, A. (2020). Hierarchical and unsupervised graph representation learning with Loukas's coarsening. *Algorithms*, 13(9), 206. Disponible : <https://doi.org/10.3390/a13090206>

Boeglin, P., & Nedjaï, R. (2000). L'analyse spatiale du réseau de sentiers: apports de la théorie des graphes, des données écologiques et sociales et de l'usage des SIG. *Revue de géographie alpine*, 88(4), 47-62. Disponible : [https://www.persee.fr/doc/rga\\_0035-1121\\_2000\\_num\\_88\\_4\\_3010](https://www.persee.fr/doc/rga_0035-1121_2000_num_88_4_3010)

Conte, D., Foggia, P., Sansone, C., & Vento, M. (2003). Graph matching applications in pattern recognition and image processing. In *Proceedings 2003 International Conference on Image Processing* (Cat. No. 03CH37429) (Vol. 2, pp. II-21). IEEE. Disponible : <https://doi.org/10.1109/ICIP.2003.1246606>

Dorogovtsev, S. N., & Mendes, J. F. (2002). Evolution of networks. *Advances in physics*, 51(4), 1079-1187. Disponible : <https://doi.org/10.1080/00018730110112519>

Elinas, Pantelis. (2019). Knowing Your Neighbours: Machine Learning on Graphs. Electronic Publishing, [En ligne] : <https://medium.com/stellargraph/knowning-your-neighbours-machine-learning-on-graphs-9b7c3d0d5896>

Heinold, Brian. (2018). A Simple Introduction to Graph Theory. Disponible : [https://www.brianheinold.net/notes/A\\_Simple\\_Introduction\\_to\\_Graph\\_Theory\\_Heinold.pdf](https://www.brianheinold.net/notes/A_Simple_Introduction_to_Graph_Theory_Heinold.pdf)

Hamilton, W. L. (2020). Graph representation learning. Synthesis Lectures on Artificial Intelligence and Machine Learning, 14(3), 1-159. Disponible : [https://www.cs.mcgill.ca/~wlh/grl\\_book/files/GRL\\_Book.pdf](https://www.cs.mcgill.ca/~wlh/grl_book/files/GRL_Book.pdf)

Kaloga, Y. (2021). Learning attributed graphs representations (Doctoral dissertation, Université de Lyon). Disponible : <https://theses.hal.science/tel-03610111/>

Karagiannakos, Sergios. (2020). Graph Neural Networks - An overview. AI SUMMER. [En ligne] : <https://towardsdatascience.com/graph-neural-networks-an-overview-dfd363b6ef87>

Karger, D. R. (1995). Random sampling in graph optimization problems. stanford university. Disponible : <https://people.csail.mit.edu/karger/Papers/thesis.pdf>

Malinverni, E. S., Naticchia, B., Lerma Garcia, J. L., Gorreja, A., Lopez Uriarte, J., & Di Stefano, F. (2020). A semantic graph database for the interoperability of 3D GIS data. Applied Geomatics, 1-14. Disponible : <https://doi.org/10.1007/s12518-020-00334-3>

Masui, Tomonori. (2022). Graph Neural Networks with PyG on Node Classification, Link Prediction, and Anomaly Detection. Towards Data Science. [En ligne] : <https://towardsdatascience.com/graph-neural-networks-with-pyg-on-node-classification-link-prediction-and-anomaly-detection-14aa38fe1275>

Mnih, V., & Hinton, G. E. (2010). Learning to detect roads in high-resolution aerial images. In Computer Vision–ECCV 2010: 11th European Conference on Computer Vision, Heraklion, Crete, Greece, September 5-11, 2010, Proceedings, Part VI 11 (pp. 210-223). Springer Berlin Heidelberg. Disponible : [https://doi.org/10.1007/978-3-642-15567-3\\_16](https://doi.org/10.1007/978-3-642-15567-3_16)

Pancino, N., Bongini, P., Scarselli, F., & Bianchini, M. (2022). GNNkeras: A Keras-based library for Graph Neural Networks and homogeneous and heterogeneous graph processing. SoftwareX, 18, 101061. Disponible : <https://doi.org/10.1016/j.softx.2022.101061>



Pelle, Stéphane. (2005). Théorie des graphes. École Nationale des Sciences Géographiques, Institut Géographique National. [En ligne] : <http://cours-fad-public.ensg.eu/course/view.php?id=73>

Renton, G. (2021). Analyzing and improving graph neural networks (Doctoral dissertation, Normandie Université). [En ligne] : <https://theses.hal.science/tel-03346018/>

Rodrigue, J. P., & Ducruet, C. (2005). Graph theory: definition and properties. Electronic Publishing. [En ligne] : <https://transportgeography.org/contents/methods/graph-theory-definition-properties/>

Rodrigue, J. P. (2005). Network Data Models. Electronic Publishing. [En ligne] : <https://transportgeography.org/contents/methods/network-data-models/>

Rodriguez, A. (2020). Deep Learning Systems: Algorithms, Compilers, and Processors for Large-Scale Production. Synthesis Lectures on Computer Architecture, 15(4), 1-265. Disponible : [https://www.morganclaypoolpublishers.com/catalog\\_Orig/samples/9781681739670\\_sample.pdf](https://www.morganclaypoolpublishers.com/catalog_Orig/samples/9781681739670_sample.pdf)

Santos, L. B., Jorge, A. A., Rossato, M., Santos, J. D., Candido, O. A., Seron, W., & de Santana, C. N. (2017). (Geo) graphs-complex networks as a shapefile of nodes and a shapefile of edges for different applications. arXiv preprint arXiv:1711.05879. Disponible : <https://doi.org/10.48550/arXiv.1711.05879>

Schintler, L. A., Kulkarni, R., Gorman, S., & Stough, R. (2007). Using raster-based GIS and graph theory to analyze complex networks. Networks and Spatial Economics, 7, 301-313. Disponible : <https://doi.org/10.1007/s11067-007-9029-4>

Thomson, R. C., & Richardson, D. E. (1995). A graph theory approach to road network generalisation. In Proceeding of the 17th international cartographic conference (pp. 1871-1880). Disponible: [https://icaci.org/files/documents/ICC\\_proceedings/ICC1995/PDF/Cap354.pdf](https://icaci.org/files/documents/ICC_proceedings/ICC1995/PDF/Cap354.pdf)

Wei, Y., Zhang, K., & Ji, S. (2019). Road network extraction from satellite images using CNN based segmentation and tracing. In IGARSS 2019-2019 IEEE International Geoscience and Remote Sensing Symposium (pp. 3923-3926). IEEE. Disponible : <https://doi.org/10.1109/IGARSS.2019.8898565>

Wilson, R. J. (1979). Introduction to graph theory. Pearson Education India. Disponible : <https://www.maths.ed.ac.uk/~v1ranick/papers/wilsongraph.pdf>

Wu, C. (2021). Graph Representation Learning: from Kernel to Neural Networks (Doctoral dissertation, Institut Polytechnique de Paris). [ En ligne ] : <https://theses.hal.science/tel-03662478/>

Wu, Z., Pan, S., Chen, F., Long, G., Zhang, C., & Philip, S. Y. (2020). A comprehensive survey on graph neural networks. IEEE transactions on neural networks and learning systems, 32(1), 4-24. Disponible : <https://doi.org/10.1109/TNNLS.2020.2978386>

J. Yan, S. Ji and Y. Wei. (2022). A combination of convolutional and graph neural networks for regularized road surface extraction. IEEE transactions on geoscience and remote sensing, 60, 1-13. Disponible : <https://doi.org/10.1109/TGRS.2022.3151688>

Zhao, G., Jia, P., Zhou, A., & Zhang, B. (2020). InfGCN: Identifying influential nodes in complex networks with graph convolutional networks. Neurocomputing, 414, 18-26. Disponible : <https://doi.org/10.1016/j.neucom.2020.07.028>

Zhang, M. and Chen, Y. (2018). Link prediction based on graph neural networks. Advances in neural information processing systems. Disponible : <https://doi.org/10.48550/arXiv.1802.09691>

Zhang, T., Lu, H., & Li, S. Z. (2009). Learning semantic scene models by object classification and trajectory clustering. In 2009 IEEE conference on computer vision and pattern recognition (pp. 1940-1947). IEEE. Disponible : <https://doi.org/10.1109/CVPR.2009.5206809>

Zhu, J., Chong, H. Y., Zhao, H., Wu, J., Tan, Y., & Xu, H. (2022). The Application of Graph in BIM/GIS Integration. Buildings, 12(12), 2162. <https://doi.org/10.3390/buildings12122162>

Zhou, J., Cui, G., Hu, S., Zhang, Z., Yang, C., Liu, Z., ... & Sun, M. (2020). Graph neural networks : A review of methods and applications. AI open, 1, 57-81. Disponible : <https://doi.org/10.1016/j.aiopen.2021.01.001>