Apprentissage par renforcement pour l'estimation de la Graph Edit Distance

Hamon Hugo
Encadrant: Heroux Pierre

12 avril, 2024

Université sciences et techniques de Rouen (Campus du Madrillet)

Table des matières

Qu'est-ce que la GED ?

Définition

La GED est une mesure de similarité entre deux graphes, où un graphe est défini comme suit : $G = (\nu, \epsilon, \mu, \xi)$

- ν représente les nœuds.
- ϵ représente les arêtes.
- μ est une fonction associant un attribut à un nœud.
- ξ est une fonction associant un attribut à une arête.

Qu'est-ce que la GED ?

Définition

La GED est une mesure de similarité entre deux graphes, où un graphe est défini comme suit : $G = (\nu, \epsilon, \mu, \xi)$

- ν représente les nœuds.
- ϵ représente les arêtes.
- μ est une fonction associant un attribut à un nœud.
- ξ est une fonction associant un attribut à une arête.

Opérations d'édition

La GED est définie comme le nombre minimum d'opérations d'édition nécessaires pour transformer un graphe en un autre. Les opérations d'édition possibles sont :

- Insertion/suppression/substitution de nœuds.
- Insertion/suppression/substitution d'arêtes.

Qu'est-ce que la GED ?

Formalisme mathématique

Formellement, la GED est définie comme suit :

$$GED(G1, G2) = \min_{\{e_1,\cdots,e_k\}\in\Gamma(G1,G2)} \sum_{i=1}^{k} c(e_i)$$

Avec :

- $\Gamma(G1, G2)$ l'ensemble des chemins d'édition entre les deux graphes.
- $c(e_i)$ le coût de l'opération d'édition e_i.

Les méthodes exactes pour calculer la GED

Méthodes de calcul exactes

Les méthodes classiques pour calculer la GED sont :

- Recherche exhaustive
- Algorithme A^*
- Parcours en profondeur (DFS - Depth First Search)

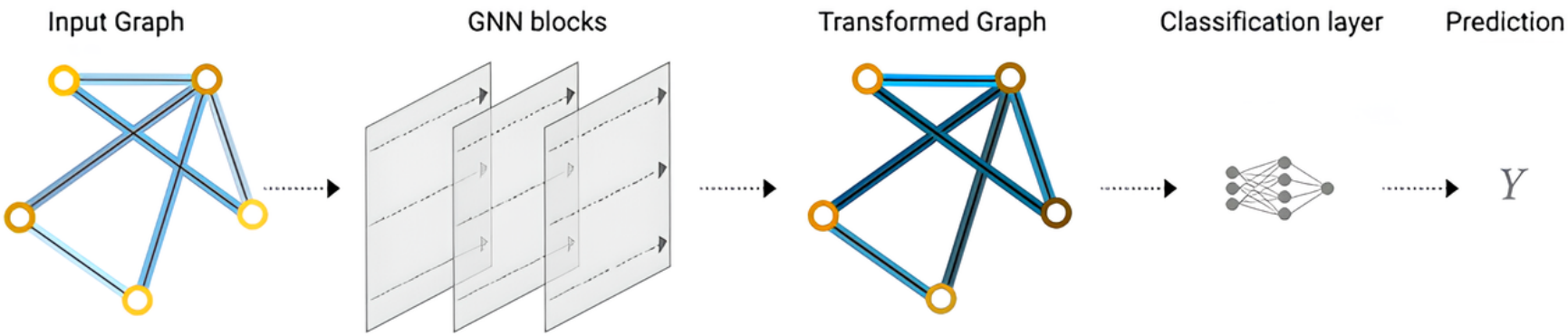
Taille des graphes ($\|V_1\| x \|V_2\|$)	Nombre de possibilités
3x3	34
5x5	1546
7x7	130922
8x8	1441729

Les méthodes pour approximer la GED

Estimation de la GED

Avec les avancées récentes de l'apprentissage automatique, la GED peut être estimée à l'aide des méthodes suivantes :

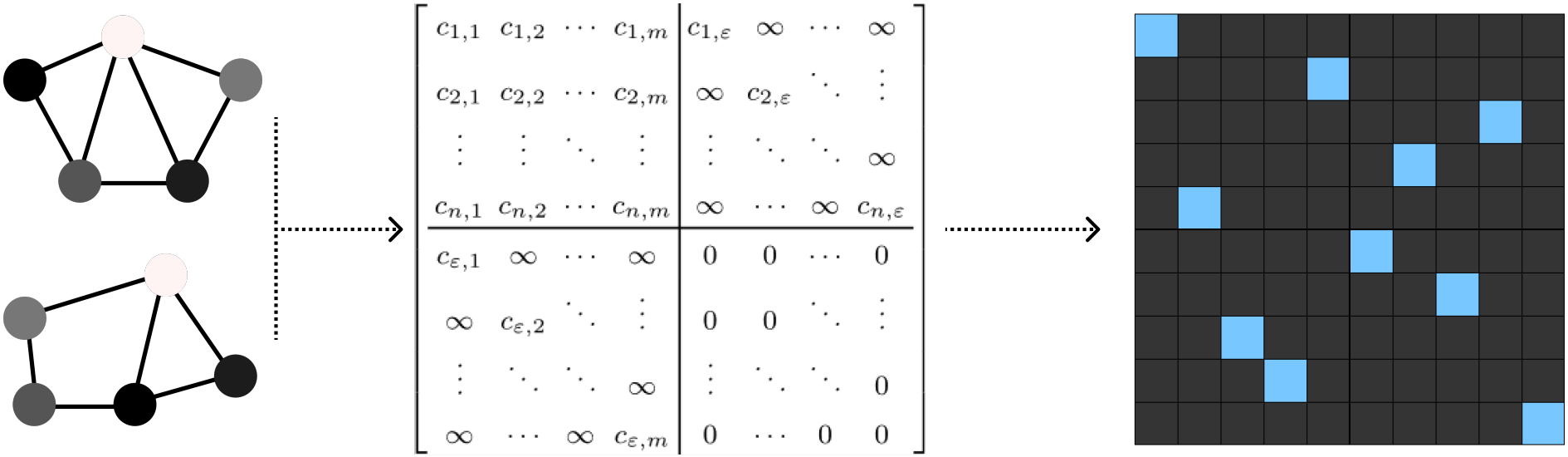
- Réseaux de neurones graphiques (GNN - Graph Neural Networks)
- Modèles Transformer
- Transport optimal
- BPGED



Fonctionnement de BPGED

Distance d'édition de graphe bipartite

L'approche bipartite a pour but de résoudre un problème d'affectation sur une matrice de coût préalablement obtenue à partir de deux graphes et composée de trois blocs (substitution, suppression et insertion) afin d'obtenir une borne inférieure pour la GED.



$$\left[\begin{array}{cccc|cccc} c_{1,1} & c_{1,2} & \cdots & c_{1,m} & c_{1,\varepsilon} & \infty & \cdots & \infty \\ c_{2,1} & c_{2,2} & \cdots & c_{2,m} & \infty & c_{2,\varepsilon} & \ddots & \vdots \\ \vdots & \vdots & \ddots & \vdots & \vdots & \ddots & \ddots & \infty \\ c_{n,1} & c_{n,2} & \cdots & c_{n,m} & \infty & \cdots & \infty & c_{n,\varepsilon} \\ \hline c_{\varepsilon,1} & \infty & \cdots & \infty & 0 & 0 & \cdots & 0 \\ \infty & c_{\varepsilon,2} & \ddots & \vdots & 0 & 0 & \ddots & \vdots \\ \vdots & \ddots & \ddots & \infty & \vdots & \ddots & \ddots & 0 \\ \infty & \cdots & \infty & c_{\varepsilon,m} & 0 & \cdots & 0 & 0 \end{array}\right]$$

Riesen, K. ; Bunke, H.
Approximate graph edit distance computation by means of bipartite graph matching.
Institute of Computer Science and Applied Mathematics, University of Bern, 2009.

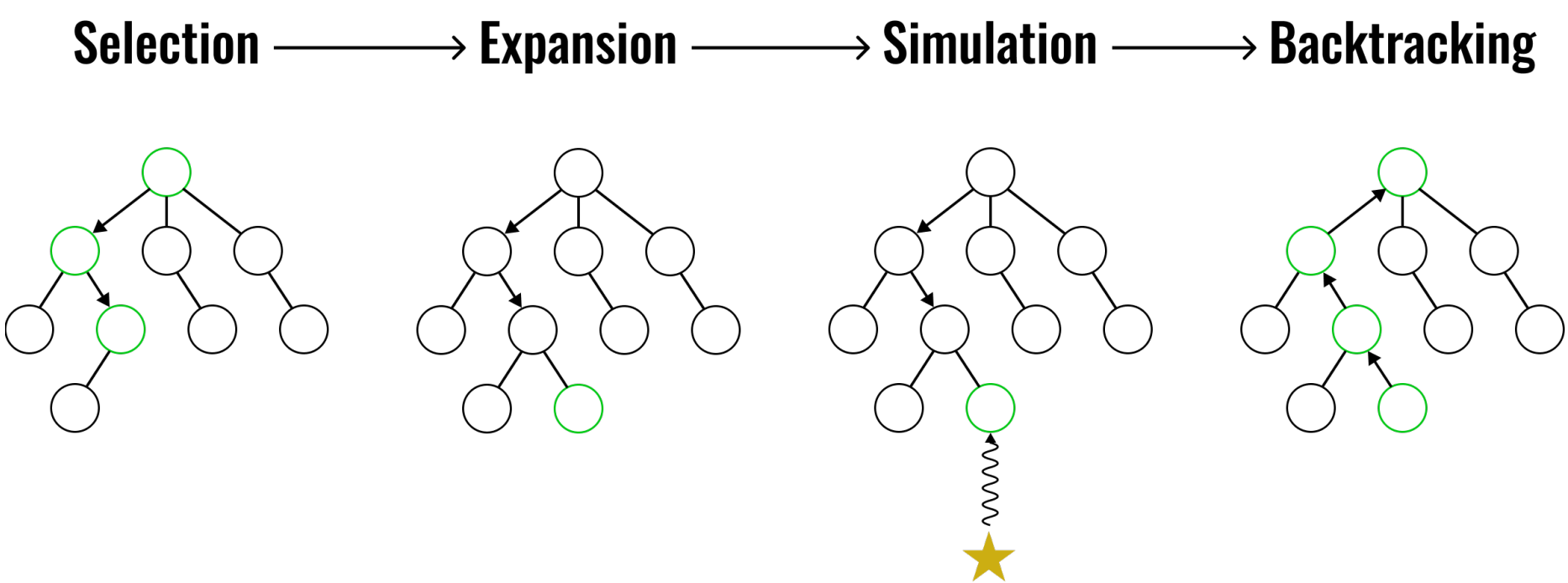
Table des matières

Fonctionnement du Monte Carlo tree search

Les quatre étapes de la méthode MCTS

La méthode MCTS se déroule généralement en quatre étapes :

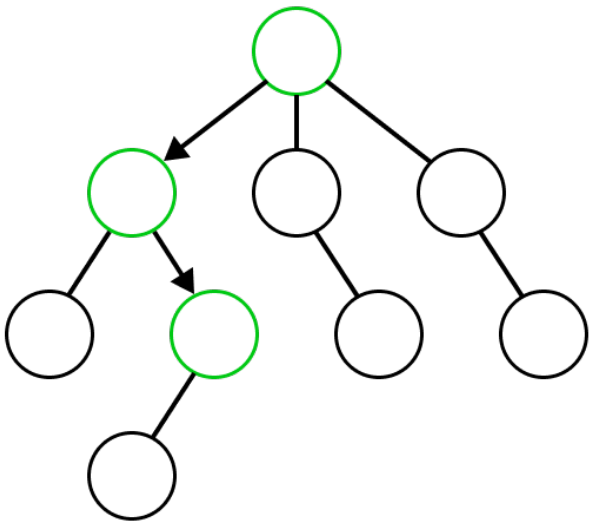
- **Sélection :** Choix d'un nœud en fonction d'une politique.
- **Expansion :** Ajout de nouveaux nœuds représentant des actions.
- **Simulation :** Simulations aléatoires jusqu'à un état final.
- **Backpropagation :** Rétropropagation des résultats vers le nœud racine.



Fonctionnement de la sélection

Processus de sélection

La sélection consiste à choisir un nœud de l'arbre de recherche en fonction d'une politique. Ce choix est important pour une exploration efficace de l'arbre et une prise de décision optimale.



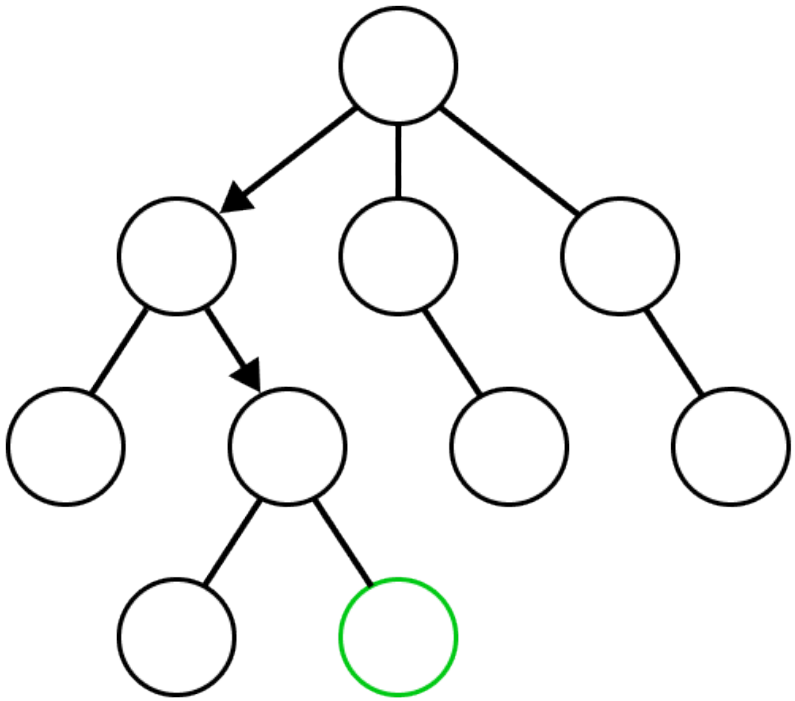
$$UCB1(S_i) = \overline{V_i} + C\sqrt{\frac{\ln N}{n_i}}$$

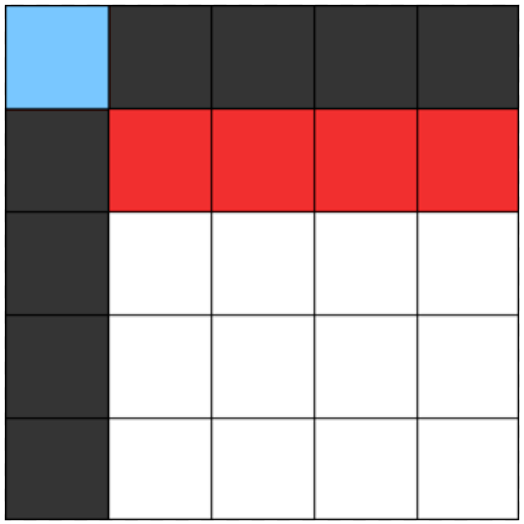
Nombre de visites parent: N
Valeur moyenne: $\bar{V}_i$
Nombre de visites: n_i

Fonctionnement de l'expansion

Processus d'expansion

L'expansion consiste à ajouter de nouveaux nœuds à l'arbre de recherche à partir du nœud sélectionné lors de l'étape précédente. Dans le contexte du LSAP, ces nouveaux nœuds représentent les cases non sélectionnées à la ligne suivante.

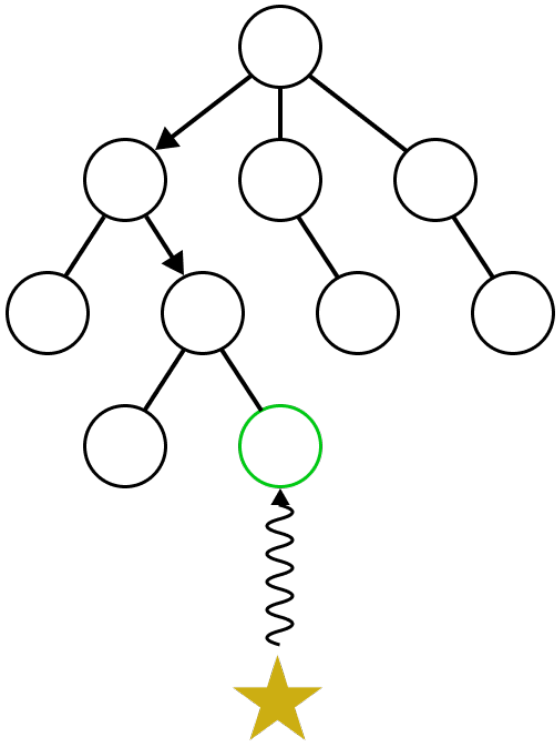


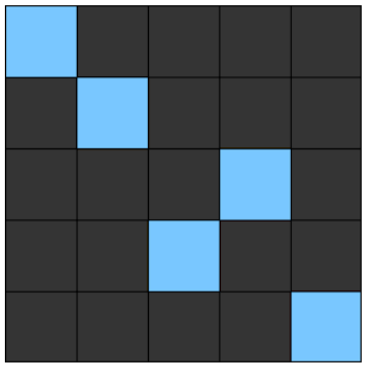


Fonctionnement de la simulation

Processus de simulation

La simulation implique de déterminer une valeur à attribuer pour une position actuelle. Il s'agit de l'heuristique permettant d'évaluer une séquence de choix. Dans notre cas, une sélection aléatoire des nœuds restants est une heuristique valide.

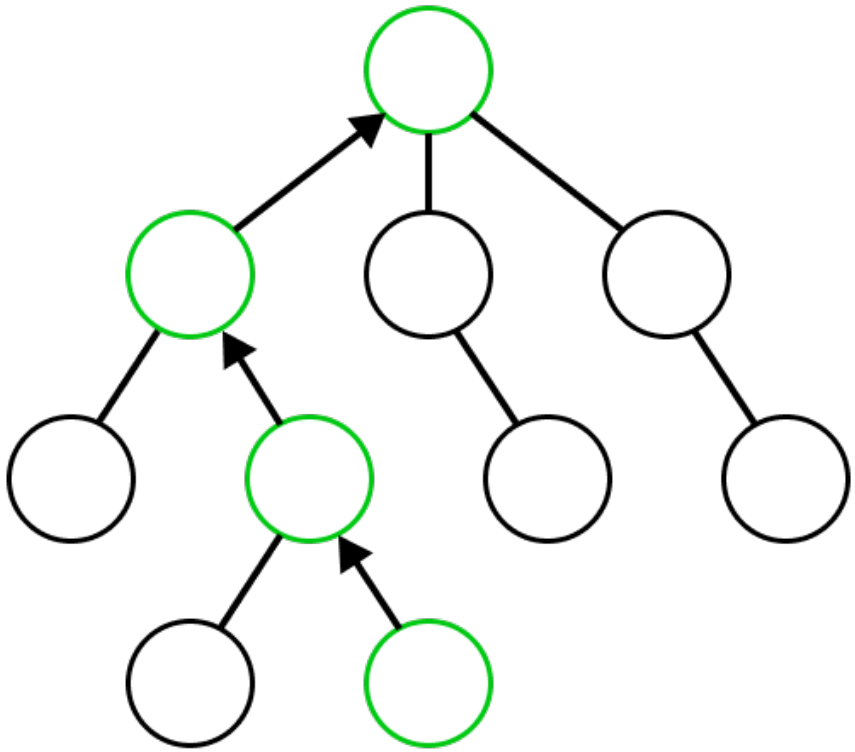




Fonctionnement du backtracking

Processus de backtracking

Le backtracking permet de propager les résultats obtenus lors de la simulation jusqu'à la racine de l'arbre, influençant ainsi le processus de sélection.



Utilisation du deep learning

Modification de la sélection

La sélection est modifiée en utilisant une probabilité calculée par le modèle. Cette probabilité donne plus de poids aux actions potentiellement plus prometteuses selon les estimations du modèle. La nouvelle formule de sélection peut être définie comme suit :

$$UCB1(S_i) = \overline{V_i} + C\sqrt{\frac{\ln N}{n_i}} * p_i$$

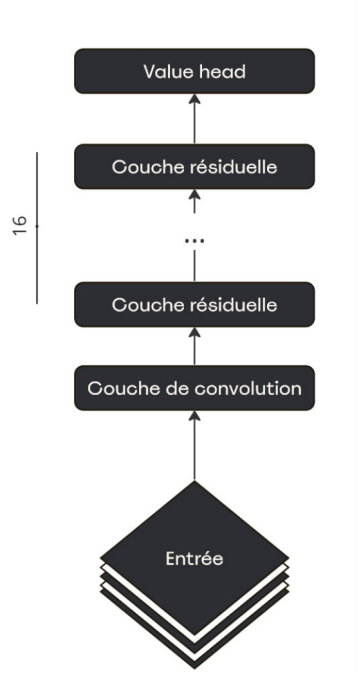
Où :

- $\overline{V_i}$ est la valeur moyenne des récompenses pour l'état S_i.
- N est le nombre de visites du parent.
- n_i est le nombre de fois où l'état S_i a été visité.
- p_i est une priorité inférée par le modèle.

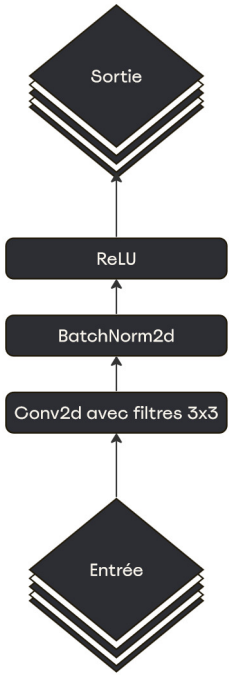
Modèle

Modèle ResNet

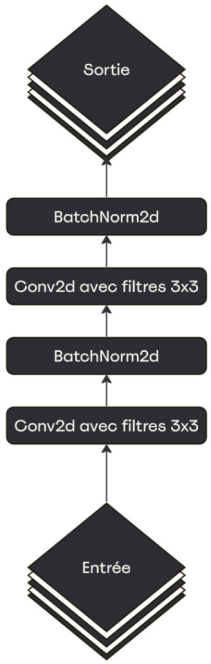
Architecture du réseau profond utilisée pour l'apprentissage dans AlphaZero pour la sélection des actions.



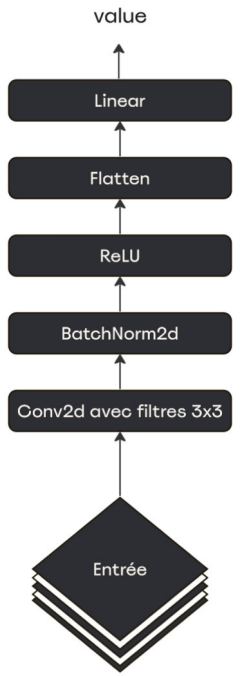
Modèle



Convolution



Couche résiduelle



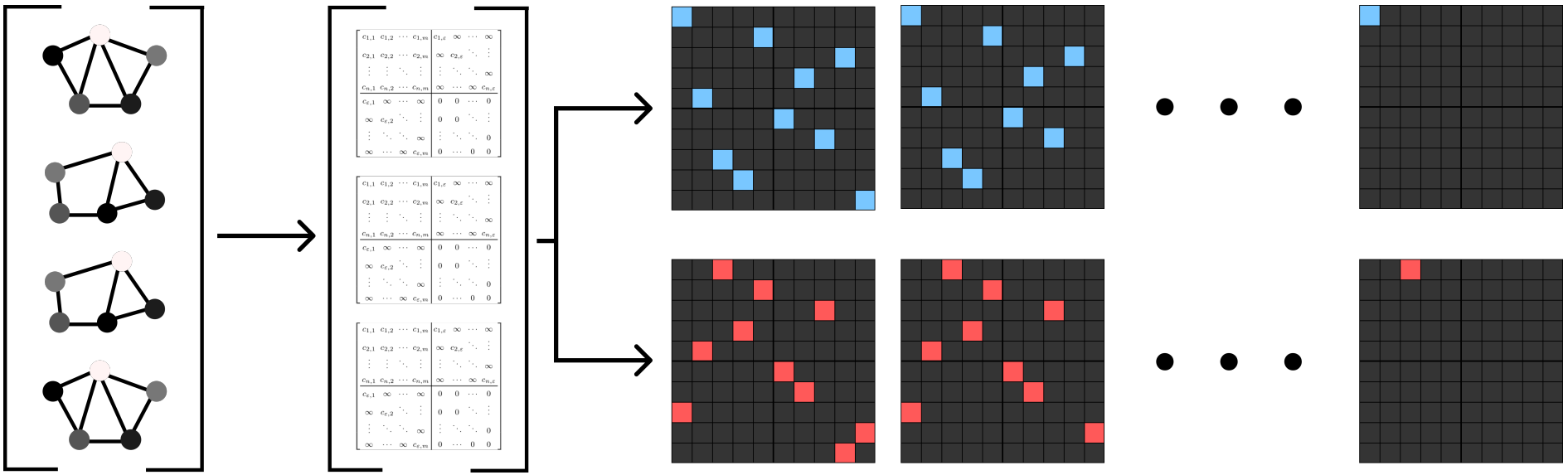
Value Head

Table des matières

Génération des données d'entraînement

Données d'entraînement

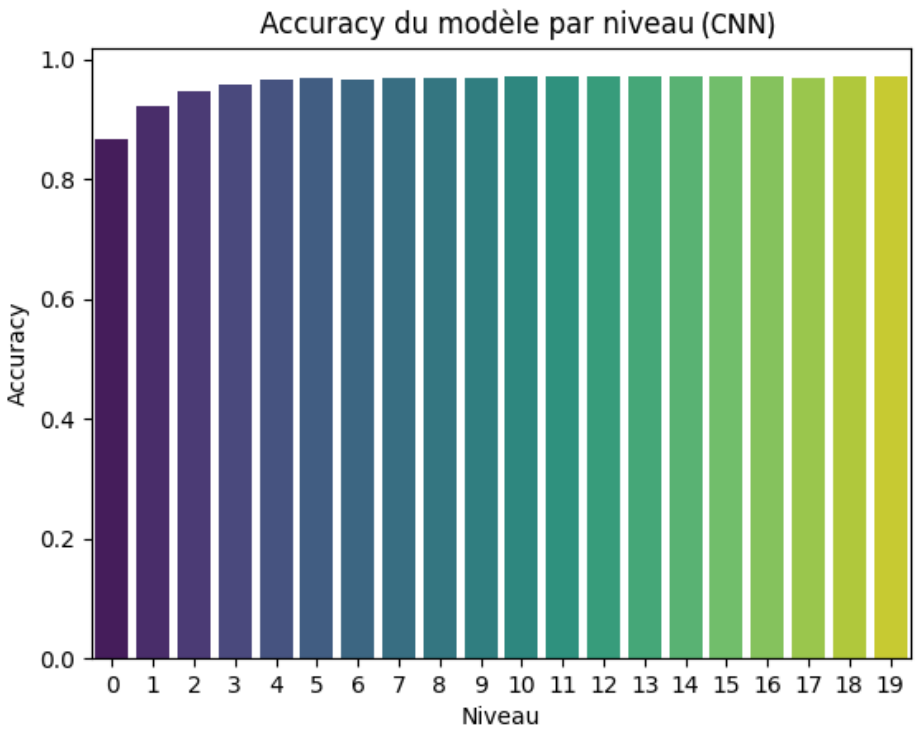
Les graphes et les matrices de coûts sont générés aléatoirement, l'algorithme Hongrois est alors utilisé pour trouver la solution optimale pour chaque matrice. Ensuite, des solutions bonnes et mauvaises sont également générées pour l'entraînement du modèle.

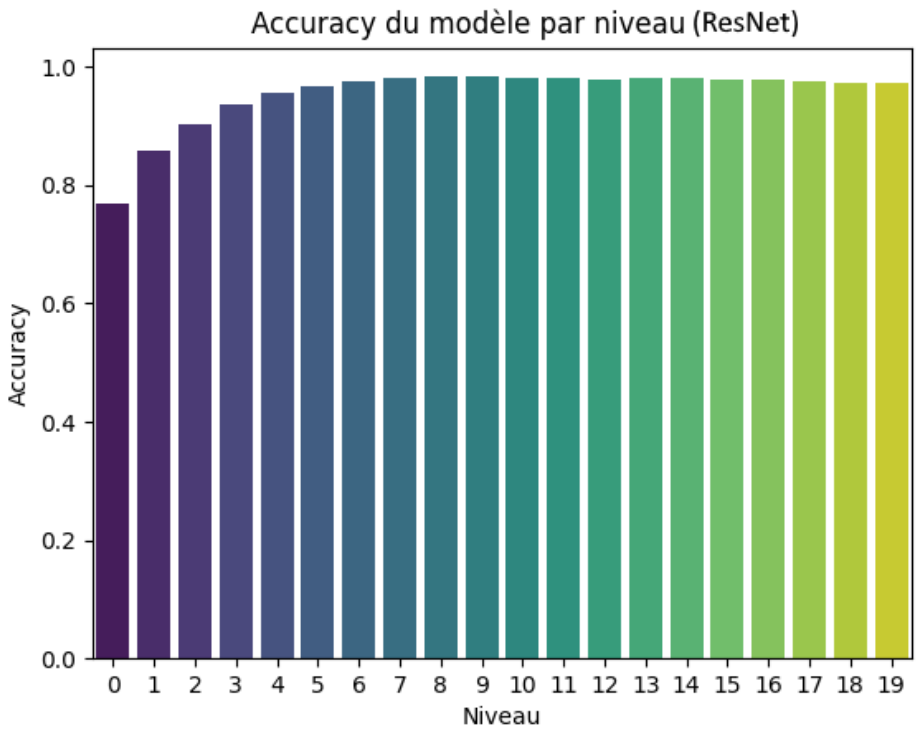


Précision

Précision du modèle

Évolution de la précision du modèle sur 20 lignes d'une matrice



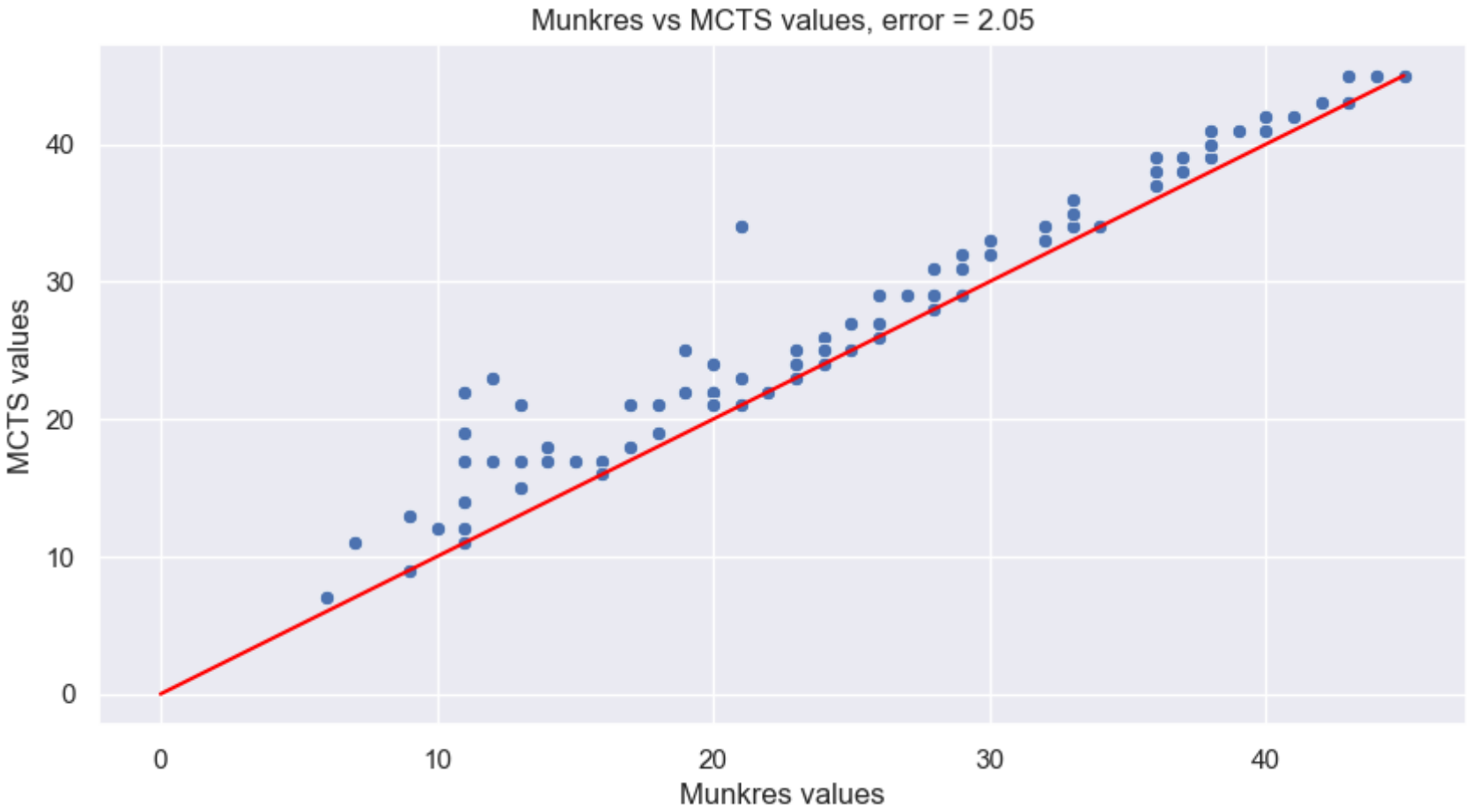


Estimation sans modèle

Données d'entraînement

Les données de test ont été générées selon la procédure expliquée précédemment.

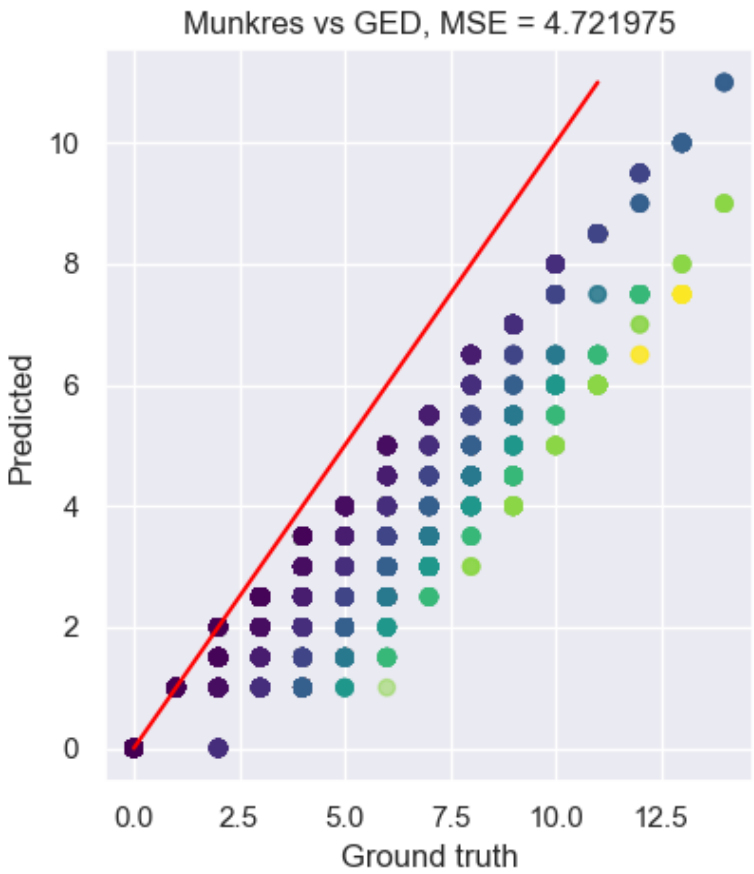
Estimation avec modèle



Estimation de la GED avec le framework

Données d'entraînement

Les données de test proviennent de la base de données Linux et contiennent 10 000 paires de graphes.



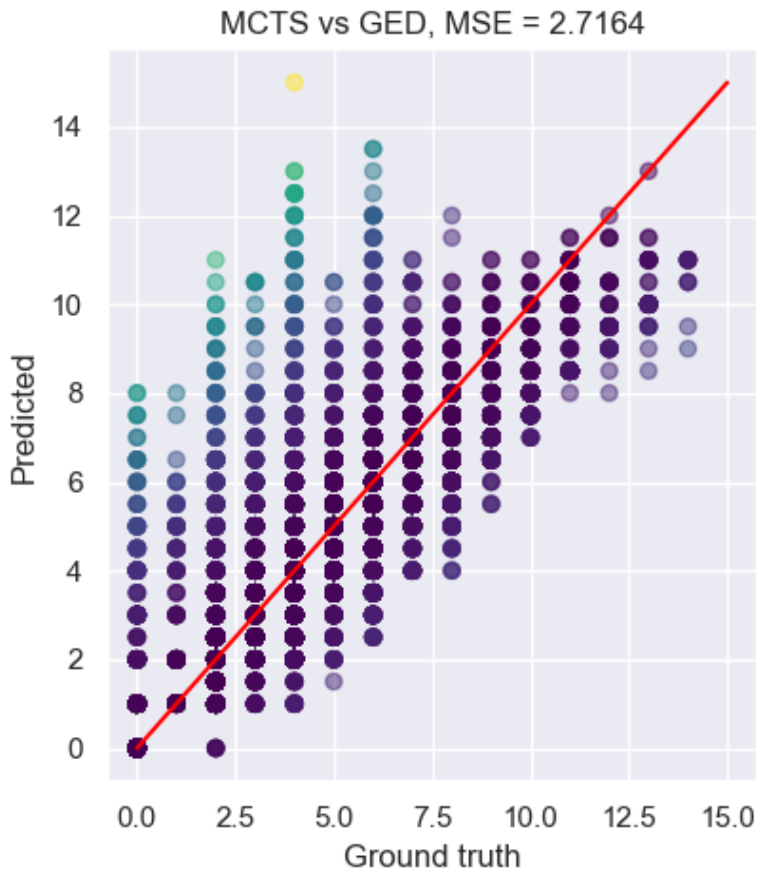


Table des matières

Points d'amélioration

Optimisation des hyperparamètres

Malgré les résultats prometteurs pour estimer le LSAP, l'optimisation des hyperparamètres du modèle pourrait être approfondie pour améliorer encore davantage les performances de l'estimation de la GED.

Autres architectures de réseaux neuronaux

D'autres architectures de réseaux neuronaux, telles que les réseaux récurrents, pourraient être exploitées et comparées au réseau actuel.

Données d'entraînement

Les données d'entraînement sont importantes pour entraîner le modèle. Un entraînement basé sur les vraies solutions des GED pourrait permettre une estimation plus précise de celles-ci.