



De C++ à Python, le calcul de la distance d'édition de graphes par la librairie GedLib

Natacha Lambert
Réunion du projet AGAC
3 mai 2019

À propos de la distance d'édition de graphe

- Les graphes permettent de représenter beaucoup de choses et de données
 - Difficiles parfois à manipuler, notamment pour les comparer
- Distance d'édition = coût de transformation d'un graphe en un autre

À propos de la distance d'édition de graphe

- Problème NP-dur → inutilisable sauf très petit graphe (10/15 nœuds max)
 - Plusieurs heuristiques possibles (par sous-graphe, etc)
 - Fonction de coût variable selon le type de graphe
- Regrouper le tout au même endroit

Gedlib

- Librairie open source créée par David Blumenthal
- Permet de faire des calculs de distance d'édition
- Comprend beaucoup de méthodes de calcul et de fonctions de coût
- Optimisée et en C++

Gedlib

- Besoin de passer des structures Python
 - Comporte des choses qu'on ne veut pas recoder
 - A tous les avantages cités
- Binding Python

Cython

- Langage dérivé du Python
- Permet d'écrire du C dans un script Python
- Peut appeler des programmes externes écrit en C ou C++
- Comprend la gestion des classes, types et templates de C++

Démarche

- Déclaration des fonctions, classes et types C++ utilisés en Python
- Encapsulage du C++ en Python
- Compilation pour créer un wrapper.so pour faire le lien
- Importation dans un programme Python pour utiliser les fonctions

Lors du binding

- Tentative d'importer les types de Gedlib → abandon car trop de choses à déclarer (types implicites)
- Obligation de tout déclarer
- Choix d'ajouter un code C++ qui encapsule Gedlib pour ce qu'on a besoin réellement

Choix lors du binding

- Importation des types C++ pour la cohérence (size_t , long unsigned int)
- Ajout de listes pour les méthodes et fonction de coût accessibles à Python
- Surcouche des erreurs pour les afficher quelque soit la façon d'utiliser Python

Etat actuel

- Peut faire le calcul entre deux graphes en appelant successivement les fonctions
- Fonction python pour comparer paire par paire un corpus de graphes
- Seulement en chargeant des graphes GXL

Prochaines étapes

- Vérifier la non duplication des données (particulièrement des matrices)
- Ajout de méthodes et surcouches selon les paramètres
- Possibilité d'ajouter « from scratch » des graphes
- Calcul de médian

Conclusion

- Librairie complète et optimisée
- Passage Python compatible
- Encore du travail pour la robustesse
- Demande de réflexion sur la librairie pour manipuler certaines options (matrice, environnement)

Bibliographie

- Sébastien Bogleux, Luc Brun. *Linear Sum Assignment with Edition*. [Research Report] Normandie Université; GREYC CNRS UMR 6072. 2016.
- Michel Neuhaus and Horst Bunke, *Bridging the Gap between Graph Edit Distance and KernelMachine*, World Scientific
- Sébastien Bogleux, Luc Brun, Vincenzo Carletti, Pasquale Foggia, Benoit Gaüzère, et al.. *A Quadratic Assignment Formulation of the Graph Edit Distance*. [Research Report] Normandie Université; GREYC CNRS UMR 6072; LITIS. 2015.
- Gedlib : <https://github.com/dbblumenthal/gedlib>
- Documentation Cython : <https://cython.readthedocs.io/en/latest/>