

SAE 2.02: Démonstration de la Compétence 2

Contexte de la SAE

Lors de cette SAE, nous avons eu pour mission d'explorer et de résoudre un problème algorithmique en représentant les collaborations entre acteurs d'Hollywood à l'aide de la théorie des graphes. Nous devons implémenter différentes requêtes pour analyser ces collaborations et évaluer la centralité des acteurs en utilisant Python et la bibliothèque Networkx.

Description du Travail Réalisé

Analyse du Problème et Conversion des Données (AC12.01)

Pour convertir le jeu de données JSON en un graphe Networkx, j'ai d'abord écrit une fonction pour transformer les données au format JSON, puis une fonction qui parcourt ces données pour créer des arêtes entre les acteurs ayant collaboré dans un même film. Cette étape a impliqué un découpage méthodique du problème en sous-tâches simples, notamment la lecture des données et la création du graphe.

Comparaison d'Algorithmes et Complexité (AC12.02)

Lors de l'implémentation des requêtes, j'ai comparé différents algorithmes pour trouver les plus efficaces. Par exemple, pour déterminer la distance entre deux acteurs, j'ai utilisé plusieurs méthodes comme le parcours en largeur (BFS) et les fonctions optimisées de Networkx, tout en évaluant leur complexité asymptotique et leurs performances. Cela m'a permis de sélectionner les algorithmes les plus appropriés pour chaque tâche.

Mise en Œuvre des Outils Mathématiques (AC12.03)

La théorie des graphes a été au cœur de cette SAE. J'ai utilisé des concepts tels que la centralité pour modéliser et analyser les collaborations entre acteurs. Pour calculer la centralité d'un acteur, j'ai mis en œuvre des algorithmes permettant de parcourir les graphes efficacement et de déterminer la distance maximale entre un acteur donné et tous les autres.

Compétences et Apprentissages Critiques

Développement Orienté Objets

Mouad
Zouadi
14B

J'ai appliqué des principes de développement orienté objets en structurant le code en modules et en utilisant des classes pour représenter les acteurs et les collaborations. Cette approche a facilité la gestion du projet et la réutilisation du code.

Développement d'Applications avec IHM

Bien que l'interface utilisateur ne soit pas le cœur du projet, j'ai développé une interface graphique simple avec Tkinter pour rendre l'application plus interactive et accessible. Cette interface permet de charger des jeux de données, d'afficher des graphes de collaboration et d'exécuter des requêtes de manière intuitive.

Qualité de Développement

J'ai accordé une attention particulière à la lisibilité du code, aux commentaires, aux sauvegardes github et aux tests unitaires pour assurer la qualité et la robustesse de l'application. Chaque fonction a été testée rigoureusement pour garantir son bon fonctionnement et sa performance.

Théorie des Graphes

La théorie des graphes a été centrale dans ce projet. J'ai utilisé Networkx pour modéliser les collaborations entre acteurs et pour implémenter des algorithmes de parcours de graphe, de calcul de centralité et de recherche de plus courts chemins.

Conclusion :

Cette SAE m'a permis de mobiliser et de renforcer mes connaissances en algorithmique, en théorie des graphes et en développement d'applications Python. J'ai appris à analyser un problème complexe, à comparer et à implémenter différents algorithmes et à optimiser leur performance. Travailler sur ce projet a également renforcé ma capacité à travailler de manière autonome et à gérer efficacement mon temps.

De plus, cette expérience m'a fait apprécier davantage le domaine de l'algorithmique et de la théorie des graphes. J'ai particulièrement aimé chercher la meilleure complexité pour chaque algorithme et ajouter toujours plus de fonctionnalités à l'application.