

SAE Graphe

A travers cette SAE se déroulant dans la continuité des cours de maths graphe, nous devons construire un programme python qui modélisera le jeu des six degré de Kevin Bacon qui est un principe que l'on retrouve sur le site de The Oracle Of Bacon et qui permet de faire le lien entre deux acteurs selon les films dans lesquels ils ont joués, ou d'établir la centralité d'Hollywood. Nous devons donc implémenter à l'aide de graphe les relations entre les différents acteurs ainsi que des fonctions permettant de parcourir ces graphes afin d'obtenir les informations recherchées.

Pour ma part j'ai d'abord proposé le code de la fonction permettant de transformer les informations d'un fichier json en graphes networkx utilisables par la suite. Pour cette première fonction il m'a fallu recherché dans les anciens cours de python ainsi que dans la première SAE python afin de me remémorer les méthodes de lecture de fichier.

Ensuite j'ai implémenté la fonction collaborateurs communs qui aura posé quelques problèmes de compréhension au sein du groupe mais qui ont vite été résolus après un court échange. En effet la fonction permet de renvoyer les personnes étant en relation directe avec les deux acteurs mis en paramètres.

Pour la fonction suivante est_proche elle a été relativement simple étant donné qu'il me suffisait de faire appel à la fonction collaborateurs_proches codée précédemment par mon binôme, il suffit de vérifier que la seconde personne est dans les collaborateurs proches de la première personne avec laquelle on fait appel à collaborateurs_proches.

En ce qui concerne la méthode `distance_naïve`, j'ai bouclé dans le nombre de sommets du graphe ce qui me permet d'avoir un compteur au fur et à mesure que je parcours mon graphe, puis j'ai fait appel à la fonction `est_proche` avec le compteur établi précédemment afin de pouvoir à chaque tour de boucle élargir la recherche au sein du graphe jusqu'à ce que la personne en question se trouve parmi les collaborateurs proches. Il suffisait alors de renvoyer le compteur afin de connaître la distance entre les deux acteurs. Cette fonction remplit son devoir mais est très longue à exécuter et très complexe.

J'ai ensuite complété la fonction `centre_hollywood` qui permet comme son nom l'indique de renvoyer la personne se trouvant au centre d'hollywood, soit la personne qui a est la plus proche de toutes les autres personnes au sein du graphe. Pour y parvenir j'ai simplement fait appel à la fonction `centralité` que mon camarade a réalisée sur l'ensemble des nœuds du graphe et je stockais la personne ayant la plus faible valeur de centralité.

De la même manière que pour `centre_hollywood`, j'ai utilisé la fonction `centralité` sur tous les nœuds du graphes et j'ai gardé la plus grande valeur pour réaliser la fonction `eloignement_max` qui permet de connaître la distance la plus élevée dans le graphe.

A travers cette SAE j'ai donc pu me remémorer mes différents acquis des cours de python et d'algorithmique du premier semestre afin de proposer des fonctions ayant un temps d'exécution optimal afin de répondre à la demande du mieux possible. J'ai bien évidemment mobilisé les nouvelles connaissances obtenues à travers les travaux pratiques de maths graphe concernant le parcours de graphes, ainsi que les méthodes de bases de networkx indispensables pour la réalisation de cette SAE. J'ai cependant montré quelques faiblesses quant à l'optimisation de certaines fonctions comme distance ou centralite qui ont fait que ma version a été abandonnée au profit d'une meilleure version proposée par Ugo .

J'ai donc, à travers cette dernière, mobiliser différents apprentissages critiques comme savoir analyser un problème avec méthode, apprentissage mobilisé lors de la complétion des différentes fonctions ou à travers les différents échanges avec mon camarade quant à la conception des algorithmes que nous allons utiliser. De plus, je pense avoir su comparer des algorithmes pour des problèmes classiques (A.C. 12.02) lors des remaniement de certaines fonctions comme collaborateurs_communs qui a présenté différentes versions jusqu'à une se démarque en termes d'optimisation. L' apprentissage critique formaliser et mettre en œuvre des outils mathématiques pour l'informatique est également présent dans cette SAE à travers les différentes utilisations de networkX.