|  |  |  |
| --- | --- | --- |
|  |  | |
| RAPPORT DE PROJET |  | |
|  | logo-Université | |
|  | Juin 2021    **PROJET GRAPHES & OPEN DATA** | |
|  | Par Aimed BOUGUERRA  Pour Monsieur François DELBOT | |
| RESUMECe document est vise à présenter le projet Graphes & Open Data réalisé au cours du mois de juin 2021 dans le cadre de ma Licence 3 MIAGE au sein de l’Université Paris-Nanterre. J’y développerais les éléments qui composent ledit projet.Mots-clés : MIAGE, Graphes, Open Data, developpement, projet, résumé.ABSTRACTThis document aims to present the Graphs & Open Data project achieved in june 2021 as part of my MIAGE Licence in Paris-Nanterre University. I will explain all the elements of this project.Key words: MIAGE, Graph, Open Data, development, project, abstract. | |  | |

Table des matières

[RESUME 2](#_Toc75694028)

[ABSTRACT 2](#_Toc75694029)

[GRAPHES & OPEN DATA, quesako ? 5](#_Toc75694030)

[De l’Industrie à l’Information 5](#_Toc75694031)

[La Théorie des Graphes 5](#_Toc75694032)

[La Donnée Ouverte 6](#_Toc75694033)

[DESCRIPTIF DU SUJET ET DU JEU DE DONNEES 6](#_Toc75694034)

[LA NOTION DE SIMILARITE 8](#_Toc75694035)

[CONSTRUCTION DU GRAPHE 8](#_Toc75694036)

[Outils de traitement des données 8](#_Toc75694037)

[Outils de création de graphes 9](#_Toc75694038)

[Description du code 9](#_Toc75694039)

[DESCRIPTION DU GRAPHE 11](#_Toc75694040)

[Graphe brut 11](#_Toc75694041)

[DETECTION DE COMMUNAUTE(S) 12](#_Toc75694042)

[Définition des communautés 12](#_Toc75694043)

[Les communautés de notre graphe 13](#_Toc75694044)

[Réalisation du graphe avec Gephi 14](#_Toc75694045)

[REGARD SUR LE TRAVAIL EFFECTUE 17](#_Toc75694046)

[Bilan 17](#_Toc75694047)

[Difficultés et problématiques rencontrées 17](#_Toc75694048)

[Bibliographie & Webgraphie 19](#_Toc75694049)

|  |  |  |  |
| --- | --- | --- | --- |
|  |  |  |  |
|  | Introduction | L’année 2020-2021 aura été encore fortement marquée par une crise sanitaire sans précédent, pour nous étudiants, elle a marqué un tournant dans la façon d’apprendre et nous a contraint à bien plus d’autonomie dans nos apprentissages.  Le marché du travail s’étant resserré tant sur le plan des stages que pour les emplois étudiant, il a fallu maintenir détermination et motivation pour assurer le quotidien, sans trop pénaliser le parcours étudiant choisi.  Ce faisant il s’agira dans ce document de préciser le travail effectué dans le cadre du projet réalisé autour du cours « Graphes et Open Data » du 6ème semestre de Licence MIAGE.  Ainsi, dans un premier temps, je détaillerais ce que sont les Graphes et l’Open Data. Puis je m’attacherais à préciser le sujet que j’ai choisi d’étudier, et les données sélectionnées.  Par la suite, je préciserais ma démarche (construction et description du graphe, détection de communautés, etc). Enfin, je terminerais par une analyse du travail réalisé tant du point de vue de ce qu’il m’a apporté qu’au regard des limites et obstacles que j’ai pu rencontrer. |  |

|  |  |  |  |
| --- | --- | --- | --- |
|  | GRAPHES & OPEN DATA, quesako ?De l’Industrie à l’Information Si les révolutions industrielles du XIXème siècle ont bouleversé l’industrie textile, des transports ou de l’énergie ; les différentes vagues de transformation digitale, liée à l’accélération des évolutions technologiques bouleversent chaque jour un peu plus tant notre quotidien que celui des entreprises.  En à peine 25 ans, nous sommes passés d’une ère de l’information **centrée produit** à une ère **orientée client**, donnant l’occasion de voir naître de grande évolutions technologiques (création de datawarehouse, SGBD etc. pour le premier, ou d’outils de CRM et services multi-canaux pour le second).  Depuis quelques années, nous sommes entrés dans **l’ère de la Data**.  En effet, par leur grand nombre et le volume important de leurs sources, le potentiel d’exploitation de celles-ci représente une mine d’or à ciel ouvert pour celles et ceux qui sauraient en tirer parti. De grands groupes s’y intéressent depuis plusieurs années, mais il est important de noter que même les grandes banques investissent désormais dans l’exploitation des données en leur possesion, preuve de leur potentiel non négligeable économiquement parlant.  Détenir l’information n’est pas suffisant, savoir la faire parler et l’exploiter à bon escient est essentiel. (Levy, 2019) La Théorie des Graphes Ainsi, en alliant le pouvoir des statistiques liées à ces datas et la puissance de l’informatique pour les visualiser, l’exploitation de ces données devient plus concrète et donc très intéressante. Notamment en utilisant des graphes pour visualiser les données brutes.  Ainsi, il est utile de se servir d’un **graphe** lorsqu’on souhaite considérer **un ensemble d’objets** (généralement cohérents) et que l’on s’intéresse **aux liens** (orientés ou non) qui existent ou non entre ces objets.  Les **objets** sont qualifiés de **nœuds** ou **sommets**, alors que les **liens** entre eux sont également appelés **arêtes**. On peut par ailleurs qualifier le lien avec des valeurs complémentaires pouvant représenter des distances (utile par exemple quand on cherche à calculer le chemin le plus court entre deux points), des coûts, des durées, des capacités, des notions particulières. On parlera alors de graphe valué. (Marie-Claude Portmann, 2014)  Pensé par Euler, lorsqu’il s’est penché sur le problème des 7 ponts de Könisberg, la théorie des graphes comme la recherche opérationnelle ouvrent un champ des possibles immense dans de très nombreux domaines : optimisation des ressources, transport d’énergie, gestion des réseaux de transports ou de télé-communication, etc. (Delbot, 2019) La Donnée Ouverte A la frontière de cette révolution de la Data et en soutient des connaissances toujours plus grandes en matière de recherche opérationnelle et d’analyses, grâce entre autre au machine learning, la notion « d’Open Data » enrichit encore plus les possibilités.  En effet, la « donnée ouverte » est totalement publique et libre de droit, d’exploitation et de réutilisation. Selon la définition donnée par l’Open Knowledge Foundation en 2005, 3 critères sont essentiels :   * **Disponibilité et accès** : Les données doivent être pleinement accessibles, moyennant un coût de reproduction raisonnable. De préférence, elles se téléchargent sur Internet. La forme doit être confortable et modifiable. * **Réutilisation et redistribution** : Les données doivent être fournies sous des conditions permettant la réutilisation et la redistribution, incluant le mélange avec d’autres ensembles de données. * **Participation universelle** : Tout le monde doit être en mesure d’utiliser, de réutiliser et de redistribuer les données. Il ne doit y avoir aucune discrimination concernant les fins d’utilisation, ou contre des personnes ou des groupes. Par exemple, des restrictions non commerciales qui empêchent l’utilisation commerciale, ou les restrictions d’usage à certains secteurs, ne sont pas compatibles avec l’Open Data.   Ces trois critères sont l’essence de l’Open Data, **car ils autorisent l’interopérabilité**. L’interopérabilité désigne la capacité de différentes entreprises ou systèmes à travailler ensemble. En l’occurrence, l’interopérabilité est la capacité de mélanger différents ensembles de données. (L, 2019) | |  |
|  | DESCRIPTIF DU SUJET ET DU JEU DE DONNEES Dans le cadre de ce projet, je vais donc m’attacher a **étudier les intéractions** entre les différents personnages de la triple trilogie de Star Wars, l’œuvre cinématographique de George Lucas. Il faut noter que les éléments des épisodes 8 et 9 sont absents du jeu de données proposé.  Dans un premier temps, j’avais la possibilité d’utiliser d’autres jeux de données proposés, mais celui-ci m’est apparu comme intéressant en tant que cinéphile. En effet**, la franchise Star Wars est un univers complet** à part entière. Lucas a su y méler science-fiction, questionnement philisophique et également moderniser la mythologie classique. Que ce soit par la force de son histoire ou par son univers musical, il est rare que quiconque soit resté hermétique à une telle œuvre. A lui seul, ce momument de la pop-culture est une source infinie de données. Aussi, il me semblait intéressant de m’attacher aux intéractions entre les personnages et de mettre en évidence leurs liens éventuels ou non.  Le jeu de données est disponible sur le site web kaggle.com (kaggle.com, s.d.). Il est mis à disposition Ruchi Bhatia, Data Scientiste chez HP & Nvidia. Le lot de données est basé sur les scripts de 7 des 9 films de la saga. Ainsi, lorsque deux personnages parlent ensemble lors d’une même scène, ils seront alors connectés l’un à l’autre. Le jeu de données est proposé au format JSON, ce qui me demandera un travail supplémentaire de conversion pour utiliser les datas dans Gephi.  Les lots de données sont détaillés par film, mais aussi avec une vision globale sur l’ensemble de la série. Il m’a donc semblé intéressant de m’attacher à un épisode en particulier, c’est celui-ci que j’utiliserais pour détailler ma démarche ici. Chacun a ses préférences concernant cette œuvre magistrale, pour ma part, je m’attacherais à détailler les intéractions entre les personnages du premier film historique de George Lucas, à savoir *Star Wars – Episode 4 – Un Nouvel Espoir*. Cet épisode pose selon moi les bases de toute l’histoire développée par Lucas, que ce soit pour la trilogie historique, ou pour les 2 autres qui se placeront historiquement avant puis après ce premier opus. Aussi, il est donc intéressant de comprendre comment les intéractions et le réseau social qui en découle entre les personnages se présente dans ce film.  **Contenu du fichier :** (Bhatia, 2020)   * *starwars-episode-N-interactions.json* contient les intéractions sociales entre les personnages extrait de l’Episode N, où les liens entre les personnages sont définis par le nombre de fois où les personnages parlent ensemble dans une même scène. * *starwars-episode-N-mentions.json* contient les intéractions sociales entre les personnages extrait de l’Episode N, où les liens entre les personnages sont définis par le nombre de fois où les personnages sont mentionné ensemble dans une même scène. * *starwars-episode-N-interactions-allCharacters.json* correspond au réseau d’interactions avec R2-D2 et Chewbacca ajouté à partir du réseau de mentions. * *starwars-full-…* contient l’ensemble des réseaux sociaux correspondants pour l’ensemble des épisodes.   **Nœuds** :   * *name* : Nom du personnage (chaine de caractères) * *value* : Nombre de scènes où le personnage apparaît * *colour* : Couleur dans la visualisation   **Liens** (connexions entre les personnages) :   * *source* : clé d’index du personnage qui est l’un des sommets de l’arête. L'ordre des nœuds est l'ordre dans lequel ils sont répertoriés dans l'élément « nœuds » * *target* : clé d’index du personnage qui est l’autre sommet de l’arête. * *value* : Nombre de scènes où le “personnage source” et le “personnage cible (target)” du lien apparaissent ensemble.   Le réseau n’est pas dirigé et chaque personnage représentent arbitrairement la source ou la cible, ils correspondent simplement aux deux sommets d’un lien. | |  |
|  | LA NOTION DE SIMILARITE Dans l’analyse des données proposées, il est intéressant de mettre en évidence des **similarités**, et comprendre ainsi ce qui permet de créer des liens entre les personnages de la saga Star Wars.  En sciences sociales, on met en évidence, par exemple, les classes sociales pour procéder à une analyse et tirer partie de **comportements similaires ou différenciants** afin d’en dresser une vue plus générale. (Wikipedia, s.d.)  Il conviendra donc de porter un regard sur le réseau de relations entre les personnages. On pourrait se demander si les padawans ont tous les mêmes intéractions avec leur maître Jedi. Luke Skywalker et la Princesse Leia, les jumeaux d’Anakin et Padmé, ont-ils les mêmes relations avec leur père, passé du côté obscur de la Force, Darth Vador ? Retrouvera-t-on des similarités dans le réseau ainsi visualisé qui pourront être reliées à l’histoire du film, aux groupes sociaux, « ethniques », ou « étiques » des personnages ?  Sur Kaggle, l’un des utilisateurs a également proposé un versant intéressant d’étude. Il s’est arrêté sur la notion **d’homophilie**. (Ganesh, 2020)  En effet, il est socialement admis que les individus ont tendance à préférer entretenir des relations avec leurs semblables. C’est ainsi que les structures sociales et culturelles se reproduisent et perdurent. Aussi, il serait logique autant qu’intéressant que tout cela soit mis en évidence par l’analyse des graphes, notamment grâce à la notion de communautés. Inspiré par le travail de cet étudiant de NC State University, je tenterais à mon tour d’interroger les données proposées pour répondre à cette question autour des intéractions sociales et de la notion d’homophilie. | |  |
|  | CONSTRUCTION DU GRAPHEOutils de traitement des données Pour réaliser l’analyse des données, j’ai utilisé Anaconda (Anaconda, s.d.), une des meilleures « boîte à outils » open-source en matière de data science. Facile à prendre en main, il embarque l’ensemble des librairies nécessaires à l’analyse et la création de graphes en Python. Il est déployé avec un IDE dédié, Spyder, qui permet de compiler immédiatement le code et de visualiser rapidement les graphes construits. Outils de création de graphes Pour réaliser des graphes, plus simplement, j’utiliserais dans un second temps Gephi (Gephi, s.d.). Son interface graphique et l’ensemble des outils mathématiques qui y sont embarqués facilitent grandement la réalisation de graphes. Version utilisée : 0.9.2 Description du code Pour réaliser des graphes, plus simplement, j’utiliserais dans un second temps Gephy (Gephi, s.d.). Son interface graphique et l’ensemble des outils mathématiques qui y sont embarqués facilitent grandement la réalisation de graphes.   |  | | --- | | import os  import json  import networkx as nx  import matplotlib.pyplot as plt  from collections import defaultdict  from networkx.algorithms.community import greedy\_modularity\_communities | | D’abord, j’importe les packages nécessaires : os pour créer des dossiers où seront stockés les graphes réalisés, json pour traiter les données importées au format json, les librairies networkx et matplotlib permettent de traiter les données, j’ajoute également 2 modules qui me permettront de gérer les collections et le calcul de la modularité pour la détection des communautés.  Ensuite je développe plusieurs fonctions pour récupérer les informations contenus dans le fichier json de données : | | def **get\_value**(name, maps):  for nodes in maps:  if nodes["name"] == name:  return nodes["value"] | | get\_value(name,maps) revoie la valeur des noeuds. | | def **get\_info**(data):  connections = defaultdict(int)  interactions = defaultdict(int)  for link in data["links"]:  connections[data["nodes"][link["source"]]["name"]] += 1  connections[data["nodes"][link["target"]]["name"]] += 1  interactions[data["nodes"][link["source"]]["name"]] += link["value"]  interactions[data["nodes"][link["target"]]["name"]] += link["value"]  return connections, interactions | | get\_info(data) renvoie les interations et les connexions entre les personnages. Ces deux premières fonctions sont inspirées de l’analyse proposée Ganesh A. | | def **get\_communities**(G):  partition = greedy\_modularity\_communities(G)  pos = nx.spring\_layout(G)  node\_groups = []  node\_groups = list(partition)  color\_map = []  for node in G:  if node in node\_groups[0]:  color\_map.append('blue')  else:  if node in node\_groups[1]:  color\_map.append('yellow')  **(…)**  nx.draw(G,pos, alpha=0.5,node\_size=50,node\_color=color\_map,with\_labels=True)  plt.savefig(f'results/graphs/Episode\_{episode}\_Graph\_communities.png')  plt.show()  plt.close() | | get\_communities(G) permet de créer un partitionnement via le calcul de la modularité. Selon les groupes créés, on affecte une couleur aux sommets, puis on crée le graphe. | | def **get\_graph**(data, episode):  nodes = [ node['name'] for node in data["nodes"] ]  edges = [ (nodes[link['source']], nodes[link['target']]) for link in data["links"] ]  G = nx.Graph()  for node in nodes:  G.add\_node(node)  for edge in edges:  G.add\_edge(edge[0],edge[1])  nb\_nodes = G.number\_of\_nodes()  nb\_edges = G.number\_of\_edges()  plt.figure(figsize=(30,15))  plt.subplot(121)  nx.draw(G,with\_labels=True)  plt.savefig(f'results/graphs/Episode\_{episode}\_Graph.png')  plt.show()  plt.close()  G = get\_communities(G)  return G,nb\_nodes,nb\_edges | | get\_graph(data,episode) est conçue pour renvoyer un graphe à partir d’un lot de données et du numéro de l’épisode. En effet, la fonction a été prévue pour lire les fichiers de l’ensemble des épisodes. | | def **get\_episode**(episode, feature):  e = "episode-" + str(episode)  if episode == 0:  e = "full"  with open(f'data/starwars-{e}-{feature}.json') as f:  data = json.load(f)  connections, interractions = get\_info(data)  Graph,nb\_nodes,nb\_edges = get\_graph(data, episode)    temp = set()  for i,j in zip(connections.items(), interractions.items()):  temp.add((i[0], i[1], j[1], get\_value(i[0], data["nodes"])))  temp = sorted(temp, key = lambda x: x[1])[::-1]  plt.figure(figsize=(25,10))  plt.title(f'Episode-{episode} {feature}')  plt.plot(list(zip(\*temp))[0], list(zip(\*temp))[1], list(zip(\*temp))[0], list(zip(\*temp))[2], list(zip(\*temp))[0], list(zip(\*temp))[3])  plt.xticks(list(zip(\*temp))[0][::1], rotation='vertical')  plt.savefig(f'results/images/Episode\_{episode}\_{feature}.png')  plt.show()  plt.close()    return Graph,nb\_nodes,nb\_edges, (connections, interractions) | | get\_episode(episode, feature) est prévue pour renvoyer les éléments d’un épisode, elle appelle les fonctions précédentes. | | if \_\_name\_\_ == "\_\_main\_\_":  if not os.path.exists('results'):  os.makedirs('results')  os.makedirs('results/images')  os.makedirs('results/graphs')  feature = "interactions-allCharacters"  episode = 4  task = get\_episode(episode, feature)  print(f"Nombre de sommets :{task[1]}, Nombre de noeuds : {task[2]}") | | Enfin, je crée un dossier où stocker mes résultats, puis je fais appelle à mon code pour analyser les données et créer les graphes. | | |  |
|  | DESCRIPTION DU GRAPHEGraphe brut Le premier graphe obtenu présente la forme ci-contre. Il présente **22 sommets**, représentants les 22 personnages intéragissant les uns avec les autres dans l’épisode 4 de la Saga. Il **existe 60 liens** entre ces personnages. Or, si chaque sommet était relié à tous les autres on aurait 231 liens selon la formule suivante : avec n correspondant au nombre de sommets du graphe.  On peut donc en déduire, et ce sans même visualiser le graphe, que tous les personnages du film n°4 n’intéragissent pas les uns avec les autres.  Quand on visualise le graphe, on constate qu’un personnage n’a aucun lien avec les autres. En l’occurrence, on pourrait considérer cet élément comme non pertinent. Il n’a semble-t-il pas d’échange avec les autres. Il s’agit de Gold Five, un membre d’un des escadrons de l’Alliance Rebelle, qui n’a probablement pas de dialogue.  Hormis cette exception, on constate un regroupement homogène des personnages (cf. capture ci-dessous sans l’élément non pertinent). Un sommet centralise un très grand nombre de liens, il s’agit de Luke Skywalker, ce qui n’a rien d’étonnant puisque c’est le personnage central du film.  On note que C3PO, R2D2 et la Princesse Leia ont eux aussi de nombreuses intéractions.  On constate aussi que certains personnages n’ont qu’un seul lien, Han Solo est ainsi le seul lien avec Jabba The Hutt et Greedo. Red Ten n’intéragit qu’avec Red Leader.  Graphe zoomé sans l’élément non-pertinent | |  |
|  | DETECTION DE COMMUNAUTE(S)Définition des communautés Une communauté est formée par un groupe d’individus qui interagissent ensemble plus souvent entre eux qu’avec d’autres. La détection de communautés a donc pour rôle de mettre en évidence ces groupes qui se sont formés implicitement. L’intérêt est multiple : identifier des profils types d’individus, effectuer des actions ciblées, identifier des acteurs centraux ou influents, etc. (Rakotomalala)  Il est intéressant de noter que par ces détections de communautés, on peut constater des partitions dites « crisps », des groupes clairement distincts ; ou bien, observer des groupes qui se chevauchent appelés « partitions floues ». Dans ce dernier cas, on peut constater que l’appartenance à une communauté n’est pas univoque, c’est-à-dire qu’un individu peut appartenir à deux communautés et joue ainsi le rôle d’interface. C’est là qu’apparaît la notion d’influence ou de centralité.  Pour réaliser ces détections, il existe plusieurs algorithmes de partitionnement, certains par coupe d’arêtes comme le Clustering spectral ou le Girvan Newman, d’autres permettent d’identifier les sous-groupes denses. C’est ici ce second type que j’utiliserais en faisant appelle à la modularité.  En pratique, il s’agit de comparer, dans un groupe de nœuds donnés, le nombre de liens réel par rapport au nombre de liens espéré. Lorsqu’il y a plus d’arêtes qu’espéré, on obtient un groupe. (Tout est graphe ! Détection de communautés : théorie et retour d’expérience, 2018)    Graphe des communautés réalisé sous python Les communautés de notre graphe On constate donc la détection de 4 communautés différentes. La quatrième étant négligeable et représentée par le personnage déjà à l’écart.  La première communauté est représentée par les sommets de couleur bleue. On constate qu’elle est centrale, elle fait le lien entre les deux autres communautés. Elle est composée de Beru et Owen, la tante et l’oncle de Luke, R2D2 et C3PO, les droïdes achetés par l’oncle de Luke et qui l’accompagneront ensuite dans l’aventure, Chewbacca et Han Solo, qui deviendront les compagnons d’aventure de Luke, et Jabba et Greedo qui sont à la recherche de Han Solo dont la tête est mise à prix. Ce groupe de personnages ne compte aucun des personnages centraux de l’histoire, mais ils sont pour autant incontournables et clairement essentiels pour articuler l’histoire et visiblement ici les relations entre les groupes. Notamment le quatuor, C3PO, R2D2, Han et Chewwie sert clairement d’interface entre les autres groupes.  Le deuxième groupe d’individus est rassemblé sous la couleur rouge. Il s’agit d’un groupe relativement « étonnant » au regard de l’histoire du film. Darth Vador, Tarkin et Motti sont 3 personnages du côté obscur de la force, qu’ils aient des liens forts entre eux n’est donc pas étonnant. La présence de Leia et d’Obi Wan Kenoby dans ce groupe peut paraître étonnante. Cependant, on peut comprendre qu’ils sont tous deux des interfaces avec le reste des personnages de l’histoire. Leia a été élevée dans la crainte du seigneur Vador sans savoir qu’il s’agissait autrefois de son père. Elle a donc de nombreuses interactions avec lui, et subit même la torture de sa part dans cet épisode mythique. Quant à Obi Wan, Vador était autrefois son apprenti Jedi, son padawan. Ils se connaissent étaient amis, aucune surprise que leurs relations soient nombreuses.  La troisième communauté est colorée en jaune. Elle rassemble essentiellement des membres de l’Alliance Rebelle : des officiers, des pilotes de vaisseaux et leurs leaders d’escadrons. Tous ont majoritairement lien avec Luke Skywalker, c’est bien lui le principal point d’influence de cette communauté en direction de l’extérieur. C’est d’ailleurs plus ou moins la place qu’il prend à la fin de cet épisode de la saga où de son vaisseau, il arrive à détruire héroïquement l’Etoile Noire, l’arme de l’Empire capable de détruire une planète. Réalisation du graphe avec Gephi Sous Gephi, j’utiliserais le même lot de données, cependant, Gephi ne propose pas d’import au format JSON, j’ai donc dans un premier temps converti mon fichier JSON en un fichier Excel, que j’ai retravaillé en créant 2 onglets : l’un pour les nœuds, l’autre pour les liens.  Après cette conversion j’ai importé les feuilles de calcul dans mon projet. Le graphe de départ est celui présenté ci-contre (Figure 1).  Figure  Je commencerais par lui appliquer un algorithme de spatialisation « Force Atlas », pour mieux répartir l’affichage des données (cf. Figure 2).  Figure  Je continuerais en appliquant des couleurs sur les nœuds pour mieux les identifier. Tout d’abord, en me basant sur la valeur intégrée au fichier de données (cf. Figure 3).  Puis après avoir appliqué l’algorithme du plus court chemin aux liens, j’utiliserais les données issues de cet algorithme pour appliquer les nouvelles couleurs (cf. Figure 4).    Figure 5  Figure 4  Figure 3  Ensuite, j’ai appliqué un classement par taille des sommets (cf. Figure 5). Le classement par degré correspond au nombre de relations qu’a un personnage (c’est la somme des arêtes d’un nœud), utiliser la couleur et la taille permet ainsi de mettre en évidence les membres importants du réseau. J’ai ajouté à la spatialisation la notion d’ajustement par taille ainsi que l’affichage des noms des nœuds avant de procéder à la détection des communautés.  En appliquant l’algorithme de modularité, il est détecté 4 communautés (comme lors de notre script sous Python). Pour les rendre visibles, j’ai ainsi modifié les couleurs des nœuds en fonction de leur appartenance à l’une ou l’autre des communautés (cf. Figure 6).  Pour finir, suivant l’analyse proposé lors de mon travail sous Python, j’ai choisi de filtrer les nœuds dont la plage de degrés est inférieure à 2 (cf. Figure 7).  Figure 6  Figure 7  Graphe exporté de Gephi  Au final, après export on obtient le graphe ci-dessous (cf page précédente), pour lequel l’importance des personnages, leur rôle central dans l’interaction entre les communautés, et donc leur importance dans le film est encore plus parlante.  On notera que ce dernier graphe diffère légèrement de celui créé sous Python.  Les éléments des communautés sont quelque peu différents.    Ainsi, le groupe des personnages secondaires, le groupe bleu, compte Obi-Wan en plus.  On note que Han Solo est le personnage central de ce groupe, c’est le plus important. Viennent ensuite C3PO, R2D2 et Obi-Wan.  Le deuxième groupe, le groupe vert est composé des 3 personnages du coté obscur, et de Leia. Il ressort clairement son rôle d’interface avec le reste des personnages. C’est elle qui joue un rôle central dans cet équilibre contre le côté obscur, est-ce elle qui incarne le titre du film « *Un nouvel espoir* » ?  Le troisième et dernier groupe est quant à lui inchangé par rapport au résultat découvert sous python. Il est composé des membres de l’Alliance Rebelle dont Red Leader et Biggs sont les membres les plus influents après Luke Skywalker. C’est en effet le personnage central, qui est l’interface entre toutes les communautés et avec une grande part des personnages (il compte plus d’une quinzaine de liens à son actif). Il apparaît clairement comme le personnage majeur du film.  Le trio Luke, Leia et Han est bel et bien le centre de l’intrigue de ce film et les interactions qui ressortent de ce réseau social l’illustre parfaitement.  Plutôt j’évoquais la notion d’**homophilie**, qui se caractérise par le fait que des relations sociales s’établissent entre des personnes de même classe, de même groupe social, ethnique ou culturel. « *Qui se ressemble s’assemblent* » comme le dit le proverbe.  On note par l’analyse de ce graphe que les communautés en présence illustrent bel et bien ce fait. Hormis la Princesse Léia qui déroge à la règle en étant rattachée aux membres du côté obscur de la Force, les groupes formés rassemblent bien des personnes de même horizon.  Ainsi, le groupe social bleu rassemble des personnages de même ordre, et de même importance. Pas nécessairement emprunts de rebellion, en tout cas pas au départ de l’histoire, mais pas non plus soumis à l’Empire. Hormis Obi-Wan, à la lueur du film, on pourrait les qualifier de rebels malgré eux…  Il en va de même pour le groupe rouge, il rassemble les membres de l’Alliance rebelle sans aucune exception. C’est la clairement une marque d’homophilie.  Enfin, le groupe social vert, hormis une exception rassemble des personnage du côté obscur de la Force. Léia étant ici l’unique surprise de cette notion.  Ainsi, le présent graphique confirme bien ce concept de la tendance à l’homophilie. Il pourrait être intéressant d’ailleurs d’approfondir ce constat et de le mettre en parallèle avec la notion de Capital social évoquée par Pierre Bourdieu (Eloire, 2014). Il semble bien que l’œuvre de George Lucas soit encore plus riche que cela a pu être pensé par beaucoup, et ce modeste travail d’analyse des données d’un seul de ses films me semble en tout cas illustrer la véritable capacité analytique de la société que possède le cinéma. | |  |
|  | REGARD SUR LE TRAVAIL EFFECTUEBilan Réaliser un tel projet seul n’a pas été d’une grande facilité. En effet, l’année en distanciel a rendu les apprentissages plus difficiles. Notamment pour suivre les cours en visio sans souvent pouvoir manipuler en live les apprentissages. Cela m’a demandé des efforts plus conséquents pour intégrer de nouvelles compétences.  Cependant, réaliser ce projet sur les graphes m’a permis d’intégrer une compréhension plus forte de leur utilité. Et, bien que je ne me destine pas à l’avenir au domaine de la data science, j’ai une meilleure vision du potentiel des datas et des usages qui peuvent en être faits.  J’ai pris également plaisir à valoriser ici mes connaissances en matière de sciences sociales mais surtout en matière cinématographique, un bonus non négligeable. Difficultés et problématiques rencontrées Concernant les difficultés rencontrées, j’en dénombre au moins 3.  Au début du projet, je devais travailler sur un autre lot de données. Le data set était dédié à un vin Portugais ‘Vinho Verde’ et présentait l’ensemble des données chimiques relevées sur une partie de la production. Le lot de données ne présentait pas d’arêtes, ce qui m’est apparu comme perturbant. Après avoir effectué ce projet-ci, je note les éléments que j’aurais pu établir, le graphe n’aurait alors pas pris la forme d’un réseau mais d’autres aspects auraient pu être valorisés et visualisés.  Cependant, le travail sur le réseau social me semblait plus parlant et plus proche de ce que je cherchais à établir. Ainsi, parmi les jeux de données proposés je me suis tourné vers celui concernant Star Wars.  Si sous python il a été assez simple d’utiliser les données au format JSON grâce à un import de package. Il m’a tout de même fallu retravailler le fichier de données natif pour l’utiliser sous Gephi. Ainsi, pour simplifier l’interprétation des données, j’ai séparé le JSON en deux onglets distincts, l’un dédié aux liens, l’autre pour les nœuds. J’ai alors renommé les colonnes pour qu’elles correspondent au format attendu par Gephi :  **Nœuds** :   * ajout d’une colonne *id* basée sur le numéro correspondant dans le JSON * *name* > *label* * *value* et *colour* rentant inchangées   **Liens** :   * ajout d’une colonne *id* basée sur le numéro correspondant dans le JSON * *source, target* et *value* : rentant inchangées   Dernière difficulté et non des moindres, il a fallu réaliser l’ensemble de ce travail en parallèle des sessions de rattrapages des examens de l’année. C’est pourquoi au regard du résultat, je peux en toute humilité me sentir fier du résultat proposé. | |  |
|  |  |  |  |
|  | En conclusion, je peux me féliciter d’avoir su mener à bien l’objectif que je m’étais fixé.  Ce projet m’a permis de manipuler concrètement des données, de les analyser et d’en donner une interprétation, la mienne, qui me semble non seulement confirmer que j’ai su maîtriser les outils proposés en réalisant des graphes tant sous Python qu’avec Gephi mais en valorisant également mes connaissances propres en termes de cinéma et de sciences sociales.  La théorie des graphes est un vaste domaine d’étude, et pour ma part, l’analyse des réseaux sociaux, des interactions entre les individus, me semble un volet intéressant d’analyses. Cela ouvre clairement des perspectives sur les usages possibles dans de nombreux domaines.  Allier théorie mathématique et technique informatique a déjà révolutionné l’usage de nos données. Mais ce volet invitant l’analyse sociale des relations entre les membres d’un réseau me semble un domaine plein d’opportunités et de possibilités loin d’être encore toutes exploitées. | Conclusion |  |

# Bibliographie & Webographie

(s.d.). Récupéré sur kaggle.com: http://www.kaggle.com

(s.d.). Récupéré sur Anaconda: https://www.anaconda.com/products/individual

Bhatia, R. (2020, Août 11). *Star Wars Social Network*. Récupéré sur Kaggle: https://www.kaggle.com/ruchi798/star-wars

Delbot, F. (2019). Théorie des Graphes - Concepts Fondamentaux. *CM - Graphes et Open Data*, (pp. 2-9).

Eloire, F. (2014, Mai). *Qui se ressemble s'assemble ?* Récupéré sur Cairn.info: https://www.cairn.info/revue-actes-de-la-recherche-en-sciences-sociales-2014-5-page-104.htm

Ganesh, A. (2020, Septembre 8). *Star Wars character analysis*. Récupéré sur Kaggle: https://www.kaggle.com/adithyarganesh/starwars-character-analysis/notebook

*Gephi*. (s.d.). Récupéré sur https://gephi.org/

L, B. (2019, Novembre 5). *Définition de l'Open Data*. Récupéré sur Le Big Data: https://www.lebigdata.fr/open-data-definition

Levy, M. (2019, Mars 18). *Big Data, nouvelle ère*. Récupéré sur Business Decision: https://fr.blog.businessdecision.com/big-data-nouvelle-ere/

Marie-Claude Portmann, X. D. (2014, Février 3). *Introduction à la théorie des graphes*. Récupéré sur sup-numerique.gouv.fr: http://www.unit.eu/cours/EnsROtice/module\_de\_base\_voo7/co/introintrographe.html

Rakotomalala, R. (s.d.). *Détection de communauté dans les réseaux sociaux.* Université Lyon 2. Récupéré sur Université de Lyon 2: https://eric.univ-lyon2.fr/~ricco/cours/slides/WM.C%20-%20Detection%20de%20communautes.pdf

*Tout est graphe ! Détection de communautés : théorie et retour d’expérience*. (2018, Janvier 19). Récupéré sur Quantmetry: https://www.quantmetry.com/blog/tout-est-graphe-detection-de-communautes-theorie-et-retour-dexperience/

Wikipedia. (s.d.). *Analyse de Similitude*. Récupéré sur Wikipédia: https://fr.wikipedia.org/wiki/Analyse\_de\_similitudes

### Documents joints au projet :

* Le présent rapport : RAPPORT\_PROJET\_GRAPHES\_AimedBouguerra.docx
* Le data set : dossier ‘data’ => 25 fichiers JSON
* Le script Python : scriptSW.py
* Le dossier des graphes créés sous Python : dossier ‘results’ => 2 sous-dossiers, 3 fichiers image
* Le projet Gephi => project\_SW\_last.gephi
* Le data set en CVS : dossier ‘data\_cvs’ => starwars-episode-4-interactions-allCharacters\_new.xlsx
* L’ensemble des capture d’écran et opérations effectuées sous Gephi : dossier ‘gephi\_graph’ => 9 captures d’écran, 1 graphe exporté (‘SW4\_graphe\_gephi.png’), 2 sous-dossiers l’un pour le calcul du chemin le plus court, l’autre pour le calcul de la modularité.

|  |  |  |
| --- | --- | --- |
|  |  |  |
| Obi-Wan Kenobi à Luke Skywalker, Star Wars épisode 4 : « **Que la Force soit avec toi. À jamais.** » |
|  |