Méthodologie

I - Collecte

II - Mise en forme des données et construction des graphes

III - Présentation des différents graphes de leurs utilités dans l'analyse de la problématique

I – Collecte

Pour scraper les données de YouTube nous avons choisi d’utiliser la bibliothèque Selenium de Python. La récolte de données s’effectue selon l’idée suivante : Pour commencer on part d’une vidéo initiale qu’on a choisi et pour laquelle on veut analyser l’effet bulle de filtre. On récupère ensuite les 3 premières vidéos recommandées de la vidéo initiale. Puis on répète le processus sur les vidéos que l’on vient de récupérer jusqu’à une profondeur de 6. On créer ainsi un arbre de hauteur 6 et de facteur de branchement 3. Lors de la récupération des vidéos on récupère le titre, le nombre de vues, le nom de la chaine qui l’a posté, la date, le nombre d’abonnés, le genre attribué par le créateur à la vidéo ainsi que le lien de la vidéo. On génère ensuite un fichier CSV avec toute ces données.

II - Mise en forme des données et construction des graphes

Il est maintenant temps de mettre en forme les données. Le script crawl est chargé de transformer le fichier CSV de données brutes en deux fichiers compréhensibles par Gephi : le fichier des nœuds et le fichier des liens.

On ouvre ensuite ces deux fichiers dans Gephi en choisissant la stratégie de fusion des liens « Somme » pour représenter le nombre de fois qu’une vidéo a été recommandé par les autres. On utilise ensuite les algorithmes de spatialisation intégré à Gephi. On commence par « Force Atlas 2 » pour obtenir une spatialisation en fonctions des liens entre les nœuds, on applique ensuite « Force Atlas » pour aérer le graphe puis « Expansion » et « Déchevauchement » pour faciliter la lecture du graphe. On associe ensuite la taille des nœuds à leur degré entrant.

On finit par colorier les nœuds et les liens du graphes de différentes manières pour mettre en lumière différents phénomènes d’effet bulles de filtre.

III - Présentation des différents types de graphes

1) Coloration par classes de modularité

On utilise la détection de classe de modularités avec Gephi puis on colore les nœuds en fonction de leur classe. On cherche alors, au sein de ces classes de modularité, à dégager des thèmes communs grâce aux titres des vidéos, aux mots-clés et aux chaînes constituant la classe. L’analyse de ces thèmes nous permet de dire si on s’éloigne du thème de la vidéo originale ou si au contraire on reste dans une bulle. On différencie les « macro thèmes » (Musique, Gaming, Divertissement…) des « micro thèmes » ou « thèmes de niche » (Slipknot, Tutos Minecraft, Pranks…).

2) Coloration par chaînes

On colore les nœuds appartenant aux 8 chaînes les plus représentées. On utilise cette coloration pour mettre en évidence un effet de bulle de filtre au sein des chaînes elle-même. Cela permet d’observer si l’on reste toujours dans la même chaîne au fil des itérations ou si au contraire on en découvre de nouvelles.

3) Coloration par genres

On colore les nœuds en fonction du genre que leur auteur leur a attribué. On cherche ici à voir si YouTube nous enferme dans un type de vidéo en particulier associé au genre (Gaming, Science et Technologie…).

4) Carte de chaleur

On utilise la fonction « carte chaleur » de Gephi en partant du nœud correspondant à la vidéo initiale. Avec cet outil, les nœuds sont colorés en fonction de leur distance à la vidéo initiale, mais aussi par rapport au poids des liens permettant de les atteindre depuis la vidéo initiale. Ainsi plus une vidéo est recommandée par les autres, plus on a de chance d’emprunter ce « chemin » lorsqu’on parcourt YouTube. Ainsi plus une vidéo est proche de la vidéo originale ou plus elle possède des poids entrants élevé plus ce nœud est chaud. Grâce a cette représentation on a une bonne représentation des vidéos les plus susceptibles d’être visionnées en partant de la vidéo initiale. En effet plus une vidéo est chaude i.e. rouge plus elle est susceptible d’être visionnées. On peut ainsi définir les thèmes les plus recommandés en liant cette information au graphe des classes de modularité et ainsi déterminer si YouTube recommande en priorité des vidéos similaires à la vidéo initiale.

5) Courbes d’unicité

Il arrive souvent qu’on tombe sur une vidéo déjà récupérée lors du crawl, en effet il arrive régulièrement qu’une vidéo nous soit recommandée plusieurs fois via des branches différentes par exemple. On crée donc une courbe représentant le nombre de vidéos uniques rencontrées à chaque profondeur du crawl. On crée aussi une courbe des chaînes uniques et des genres de vidéos uniques rencontrés à chaque profondeur du crawl.

Grâce a ces courbes et en observant leurs tendances par rapport à la courbe majorante on peut dégager une tendance de bulle de filtre plus au moins prononcé en fonction des sujets. En général, moins on a de vidéos, chaînes ou genres uniques pour une profondeur donnée, plus on a un effet d’enfermement, en effet on retombe sans cesse sur le même type de contenu. Grâce à ces courbes on peut aussi comparer les différents effets de bulle de filtre, par exemple on peut rencontrer beaucoup de vidéos uniques mais pas beaucoup de chaînes. Ainsi on peut différencier plusieurs effets de bulles de filtre qui peuvent porter sur les vidéos, les chaînes ou les genres par exemple.