RAPPORT DE MODAL : STRUCTURE GÉOGRAPHIQUE DU TERRORISME MONDIAL

INF473G - Graphe Global Géant

30 mai 2019

Charlotte GALLEZOT, Gauthier GUINET, Hilaire PERCHENET

INTRODUCTION ET BUT DU PROJET

Notre projet porte sur l'étude du terrorisme à l'échelle mondiale. Son objectif est d'utiliser les graphes et les bases de données afin de visualiser et d'analyser certaines informations d'une manière originale et pertinente. Nous nous sommes particulièrement intéressés à la structure géographique et organisationnelle des groupes terroristes.

SOURCES DE DONNÉES

Nous avons utilisé des bases de données très complètes et accessibles en ligne : *Global Terrorisme Database* `kaggle.com/START-UMD/gtd` par START Consortium et *Terrorist and Insurgent Organization Social Services Dataset* `oefresearch.org/datasets/tios` par Lindsay L. Heger et Danielle F. Jung. D'autres bases annexes ont également été employées pour certains points précis. Ces bases de données sont au format CSV. Nous avons donc utilisé `Python`, `Java`, `Gephi` et `Neo4j` pour lire et traiter les données et nous appuyer sur les outils découverts en cours.

Base de données GTD : elle regroupe plus de 180 000 attaques terroristes réparties entre 1970 et 2017. Chaque attaque est décrite par 135 colonnes rassemblant de nombreuses features.
Base de données TIOS : elle recouvre plus de 400 organisations terroristes dont elle décrit les relations au sein des pays où ces dernières sont implantées. Les données sont disponibles sous forme d'indicateurs pondérés ou de mesures classées.

CONTENU TECHNIQUE DU PROJET

Conversion de format de données : Les databases étant au format CSV, nous avons pu la plupart du temps les utiliser directement, sur `Python` ou sur `Gephi`. Cependant, pour `Neo4j`, nous avons rencontré un problème de compatibilité qui a nécessité une conversion au format `text`. C'est l'unique conversion dont nous avons eu besoin.
Nettoyage et modifications de données : Les bases de données utilisées présentaient des nombreuses incohérences entre elles. Il nous a fallu dans un premier temps nettoyer les données afin de pouvoir les utiliser simultanément. Sur le plan du contenu, certains noms (groupes terroristes, pays..) ne coincidaient pas selon les databases ou n'étaient pas présents dans les deux. De nombreuses informations furent donc rajoutées à la main ou à l'aide de fonctions. Sur le plan informatique, divers points furent traités : par exemple, l'un des problème de compatibilité fut de modifier le séparateur de colonnes, qui était **;** pour la première database et **,** pour la seconde. Evoquons également les incohérences au sein des données (valeurs manquantes ou incomplètes, nombres de victimes non entier...). Enfin, *GTD* comporte plus d'une centaines de colonnes que nous n'utilisions pas la majorité. Pour accélérer les calculs et les requêtes, nous avons supprimé celles qui nous semblaient inutiles pour cette étude. Le travail de nettoyage de données a donc consisté à corriger ces imperfections afin de pouvoir travailler convenablement sur nos deux databases, il a été réalisé sur `Python` et `Java`.

Interrogation des données : Une fois les données nettoyées, nous les avons interrogé en `Cypher` et sur `Gephi`. Notre objectif était double : lier les pays et les zones géographiques qui étaient victimes d'attaques provenant du même groupe terroriste, et faire apparaître des liens entre les groupes terroristes ayant des points communs. D'autres requêtes donnent le pays et la zone les plus touchés et les attaques les plus meurtrières.

Intégrer des données dans un entrepôt : Nos bases de données étaient principalement stockées, enrichies et interrogées sur `Neo4j` ainsi que `Python`.

Enrichir les données par l'extraction : Afin d'étudier la géographie du terrorisme mondial, il a été nécessaire de créer des liens entre pays et zones géographiques. Partant des databases intiales traitant des évènements terroristes, nous avons créé une nouvelle database de groupes terroristes regroupant près d'une dizaines de features pertinentes extraites grâce à `Neo4J`, `Python` et `Gephi`. L'usage des classes de modularité de `Gephi` nous a permis ensuite de créer 4 nouvelles databases, liées aux clusters et nous avons adopté la même démarche pour chacune.

Visualiser les données : C'est là l'enjeu principal du MODAL et de notre projet. Tout d'abord, avec `pandas`, nous avons tracé différents historigrammes permettant d'observer la répartitions des attaques selon plusieurs critères et nous en concluons des tendances générales. Nous avons représenté les zones connectées par le terrorisme, avant de comparer la carte obtenue avec celle du commerce mondial. Il semble visuellement que les deux sont complémentaires, ce qui permet de valider l'intuition d'un commerce source paix. De plus, il apparaît que les connexions terroristes ne correspondent pas à une proximité géographique mais plutôt à des enjeux historiques (colonisation...), culturels et religieux, comme entre le Moyen-Orient et l'Asie du Sud-Est. Sur `Gephi`, nous avons défini une notion de distance entre deux groupes terroristes, puis le logiciel a détecté des communautés et a représenté ces clusters. Ces derniers présentent de nombreuses caractéristiques intéressantes. Ils permettent entre autre d'appréhender les groupes terroristes semblables et pourraient être une source d'anticipation d'actions terroristes. D'autre part, nous pouvons tracer des évolutions temporelles qui peuvent permettre de prédire à court terme les attaques possibles. C'est donc une manière différente de traiter une base de données massive, mais on a ici aussi regroupé les données par caractéristiques communes.

CONCLUSION

Dans ce projet, nous avons voulu comprendre les liens entre les différentes organisations terroristes et les représenter sur des graphes pour comprendre leurs relations. Nous avons également présenté une géographie du terrorisme qui permet la compréhension du terrorisme mondial. En effet, l'approche graphique et visuelle est souvent sous-exploitée, mais elle permet de comprendre des corrélations que d'autres études omettent.

Nous en retiendrons également que le travail manuel est nécessaire pour compléter le travail logiciel, et que dans du traitement de données massives, il faut aussi consacrer du temps aux détails techniques. Enfin, la lenteur de nos ordinateurs est un frein à l'exploitation de données, et nous avons du optimiser nos requêtes et faire des choix dans les données pour obtenir les résultats souhaités.