Projet de Technologies NoSQL

Françoise Goujard

21 février 2017

Si je devais habiter à New York (ou dans n'importe quelle ville d'ailleurs), le fait d'avoir des bibliothèques à proximité serait vraiment apréciable car j'adore lire. De plus New York a de nombreux musées de qualité dont il serait dommage de ne pas profiter. Enfin, le fait d'avoir des spots Wifi à disposition est également agréable afin d'éviter de vider son forfait internet en moins d'une semaine.

1 Les bases de données

J'ai choisi trois bases de données afin de répondre à mes critères. La première est la liste des bibliothèques de New York, la deuxième celle des musées et la troisième des spots Wifi de la ville. Ces bases contiennent le nom, l'adresse (notament le quartier) et les coordonnées du lieu. Il y a également des informations spécifiques à chacune (comme le numéro de téléphone pour les musées par exemple) qui ne seront pas intéressantes pour la suite. Pout télécharger les différentes bases, il faut lancer le script 'telechargerBases.sh'.

J'ai choisi d'utiliser MongoDB comme système de base de données car elle me semblait la plus pratique pour stocker mes données par rapport à leur format (par exemple, les stocker sous la forme de graphes en utilisant neo4J n'aurait pas eu d'intérêt). De plus, le système de requêtage me semblait le plus pratique pour ce que je voulais faire.

La script 'installationMongodbEtR.sh' permet, comme son nom l'indique, de télécharger MongoDB ainsi que R dont je me suit servi pour faire le traitement des données.

2 Traitement des bases de données

Les bases ont été exportées sous le format JSON. Le script R 'importBases.R' permet de nettoyer les fichiers en enlevant toutes les métadonnées qui n'auront pas d'utilité par la suite et en créant les variable de longitude et latitude à partir de la variable des coordonnées GPS fournie dans le JSON, afin de pouvoir par la suite les exploiter. Les JSON résultants seront 'museumFinal.json', 'libraryFinal.json' et 'wifiFinal.json'.

Pour utiliser le code R, cela necesscite tout d'abord de télécharger le package 'jsonlite' afin de lire des fichier JSON. Pour cela, il faut lancer R dans la console avec la commande 'R' puis lancer dans la console R obtenue 'install.packages("jsonlite")' (choisir le miroir de Paris, bien que ça n'ait pas réellement d'importance). Après avoir quitter la console R avec la commande 'q()', il suffit de lancer le script R avec la commande 'Rscript importBases.R' en se placant préalablement dans le dossier contenant les fichiers JSON des bases de données. Normalement, cela crée trois nouveaux fichiers JSON 'libraryFinal.json', 'museumFinal.json' et 'wifiFinal.json'.

3 Import des bases dans MongoDB

Pour importer les bases de données dans MongoDB, il faut lancer les trois lignes suivantes :
mongoimport –db projet –jsonArray –collection wifi –file wifiFinal.json
mongoimport –db projet –jsonArray –collection museum –file museumFinal.json
mongoimport –db projet –jsonArray –collection library –file libraryFinal.json

4 Requêtage des bases de données

Nous allons dans un premier temps regarder dans quels quartiers les différents items sont les plus présents, à l'aide des trois requêtes suivantes :

```
db.library.aggregate([
     { $group : { _id : "$CITY" , count: { $sum: 1 }} },
     { $sort : { count : -1}}
   ])

db.museum.aggregate([
     { $group : { _id : "$CITY" , count: { $sum: 1 }} },
     { $sort : { count : -1}}
   ])

db.wifi.aggregate([
     { $group : { _id : "$CITY" , count: { $sum: 1 }} },
     { $sort : { count : -1}}
   ])
```

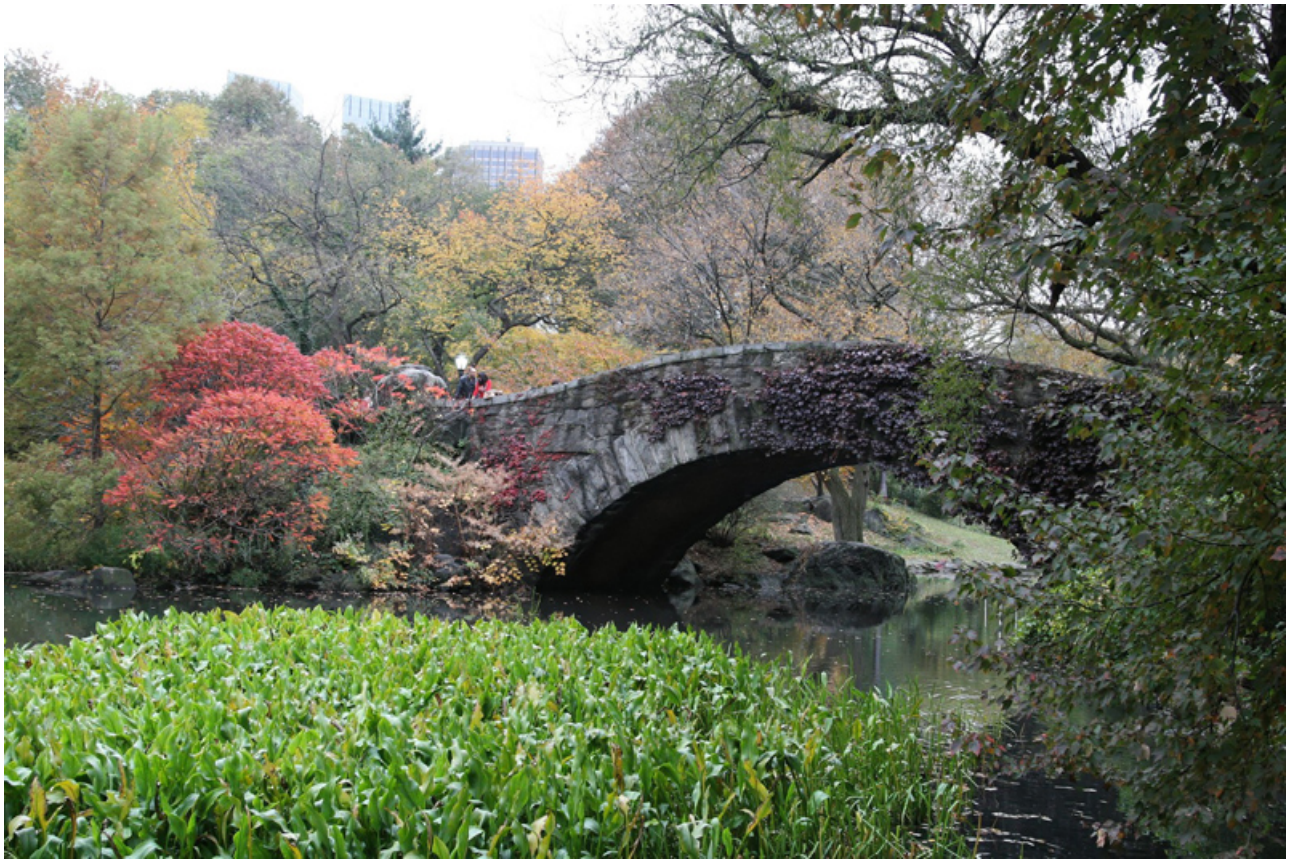
On s'apercoit que Manhattan (dans les données New York) arrive en premier pour les musées avec 89 musées et les spots Wifi avec 1014 alors que c'est Brooklyn pour les bibliothèques avec 59 bibliothèque (Manhattan arrive en second avec 44). Compte tenu cette répartition, j'ai choisi le quartier de Manhattan.

Pour essayer de préciser cela un petit peu, j'ai ensuite exploiter les coordonnées GPS.

Le code suivant permet de faire la moyenne de la latitude et la longitude pour chaque type d'items présents dans Manhattan :

```
db.library.find().forEach(function(data) {
    db.library.update({_id:data._id},{$set:{long:parseFloat(data.long)}});
    db.library.update({_id:data._id},{$set:{lat:parseFloat(data.lat)}});
})

db.wifi.find().forEach(function(data) {
    db.wifi.update({_id:data._id},{$set:{long:parseFloat(data.long)}});
    db.wifi.update({_id:data._id},{$set:{lat:parseFloat(data.lat)}});
})

db.museum.find().forEach(function(data) {
    db.museum.update({_id:data._id},{$set:{long:parseFloat(data.long)}});
    db.museum.update({_id:data._id},{$set:{lat:parseFloat(data.lat)}});
})

db.library.aggregate([
     { $group : { _id : "$CITY" , moyLong: { $avg: "$long" } , moyLat: { $avg: "$lat" } } } ,
     { $match: { _id: "New York" } }
   ])

db.museum.aggregate([
     { $group : { _id : "$CITY" , moyLong: { $avg: "$long" } , moyLat: { $avg: "$lat" } } } ,
     { $match: { _id: "New York" } }
   ])

db.wifi.aggregate([
     { $group : { _id : "$CITY" , moyLong: { $avg: "$long" } , moyLat: { $avg: "$lat" } } } ,
     { $match: { _id: "New York" } }
   ])
```

En faisant la moyenne des trois résultats obtenu, on obtient -73.97436282879269 en longitude et 40.76656844529227 en latitude, ce qui correspond, d'après le site http ://www.coordonnees-gps.fr/, à Gapstow Bridge, New York, NY 10019, États-Unis.



Ce pont se situant dans le sud de Central Park, il n'est sans doute pas très légal d'y habiter malgrès le cadre visiblement idyllique.

Les quartiers de Manhattan les plus près sont Midtown et Lenox Hill, donc ce serait là que je chercherais (la proximité avec Central Park étant un vrai plus).