Challenge Open data

Where Should I Live?

Rapport

Berard Pierre, Ghoudan Youssef, Perroud Florian, Rupp Alexandre

05 févirer 2016

Cadre général:

Le projet Challenge Open Data s'inscrit dans le cadre de la troisième année du cursus ISI de l'Ensimag. Le but du projet est d'exploiter des données ouvertes et de réaliser des traitements ainsi que de la visualisation dessus.
Notre projet s'inscrit donc dans ce cadre et porte sur la réalisation d'une application Web pour la visualisation de données à partir d'un jeu de données libre.
Le thème que nous avons choisi est l'installation d'un citoyen dans une ville qu'il ne connait pas encore. L'application doit donc le guider et lui permettre de choisir un nouveau lieu de résidence. Pour ce faire, le site web doit lui permettre de comparer les quartiers de la ville en fonction de plusieurs critères. Différentes visualisations sont proposées afin de rendre les données plus claires.

I. Scénario : objectif de l'application

Notre application Where Should I Live vise à assister l'utilisateur dans le choix d'un quartier où emménager dans Grenoble même. Elle permet donc de comparer les différents quartiers selon de multiples critères (transport, réseaux, commerces…) afin de trouver celui qui convient le mieux aux priorités de l'utilisateur.

II. Les données utilisées

A. Les jeux de données

Les données sur les quartiers et les aménagements Grenoblois que nous avons utilisées sont toutes issues de portails OpenData :

Données sur les quartiers :

- Les unions de quartier (http://data.beta.metropolegrenoble.fr/dataset/les-unions-de-quartier).
- Les limites de la ville de Grenoble (OpenStreetMap)

Données sur les critères :

- Les antennes gsm (http://data.beta.metropolegrenoble.fr/dataset/antenne-gsm-sur-le-territoire-grenoblois)
- Les arrêts de bus, tramway, car et lignes sncf (http://data.beta.metropolegrenoble.fr/dataset/arrets-bus-et-tramways)
- Les stations Cité Lib (http://data.beta.metropolegrenoble.fr/dataset/stations-citelib)
- Les pistes cyclables (http://data.beta.metropolegrenoble.fr/dataset/pistes-cyclables)
- Les restaurants Grenoblois (http://overpass-api.de/api/interpreter?data=[out:json];area[name=%22Grenoble%22]-%3E.a;(node(area.a)[amenity=restaurant];);out;)

- Les supermarchés Grenoblois (http://overpass-api.de/api/interpreter?data=[out:json];area[name=%22Grenoble%22]-%3E.a;(node(area.a)[shop=supermarket];);out;)
- Le contour de Grenoble (http://overpass-api.de/api/interpreter?data=[out:json];(rel[name=Grenoble];%3E;);out;)

B. Traitements effectués

Le principal traitement effectué sur les données a été de pré-compter les occurrences de chaque objet (station, arrêt, …) au sein d'un quartier dès l'étape d'importation des données. Ces opérations utilisent des opérations GeoJSON, et les différents compteur sont ensuite ajoutés aux données sur les quartiers. En effet, la plupart des visualisations ne nécessite pas le détail des positions et caractéristiques de chaque objet. A cette étape, les jeux de données comportant plusieurs types d'objets (arrêt de tram, de bus, …) ont été séparés en plusieurs compteurs.

Afin d'obtenir des visualisations claires et représentatives, certaines visualisations utilise un ou les deux traitements suivant :

- Normalisation des critères par rapport à la surface du quartier afin de réduire les déséquilibres dus aux tailles variables des quartiers.
- Étalage des critères entre 0 et 100 afin de s'abstraire de l'unité *critère/m²*.

III. Visualisations

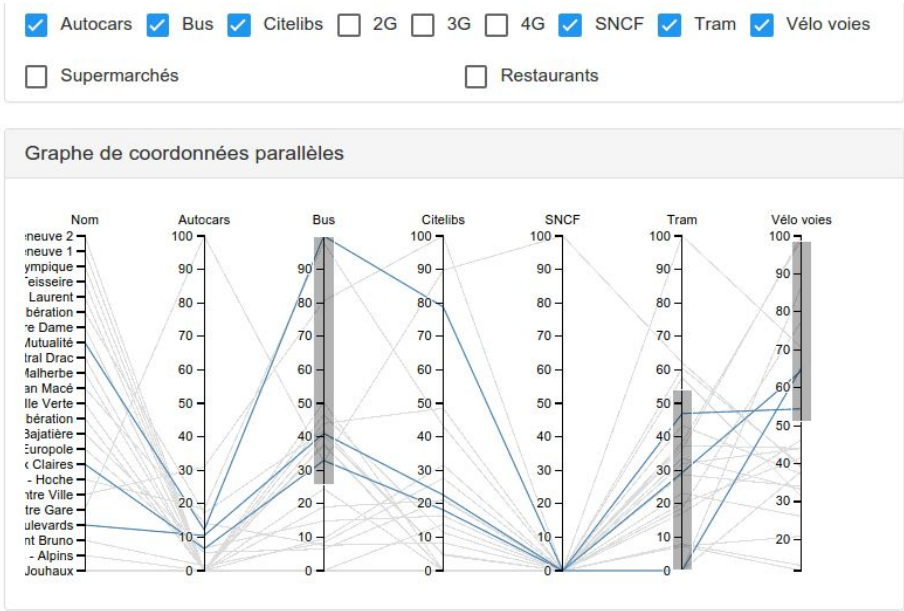
A. Carte

La principale visualisation de l'application est une carte représentant les différents quartiers de Grenoble. Les quartiers sont colorés, du plus clairs aux plus foncés, en fonction de la note du quartier pour un ou plusieurs critères choisis. Dans le cas où plusieurs critères sont sélectionnés, la couleur représente la moyenne des notes du quartier pour chaque critère.

La carte permet également d'afficher les points d'intérêts (pistes cyclables, restaurants, …), et de sélectionner un quartier pour obtenir des détails sur celui-ci.

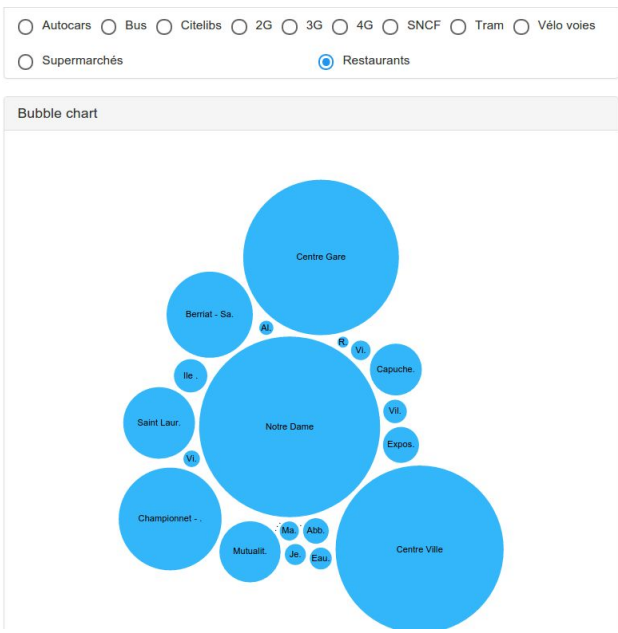
B. Coordonnées parallèles

La figure de coordonnées parallèles représente les mêmes informations que la carte, c'est à dire des comparaisons multi-quartiers et multi-critères. Des outils de séléction permettent de faire ressortir certains quartiers particuliers, et/ou les quartiers dont la note pour un critère appartient à une certaine plage de valeurs.



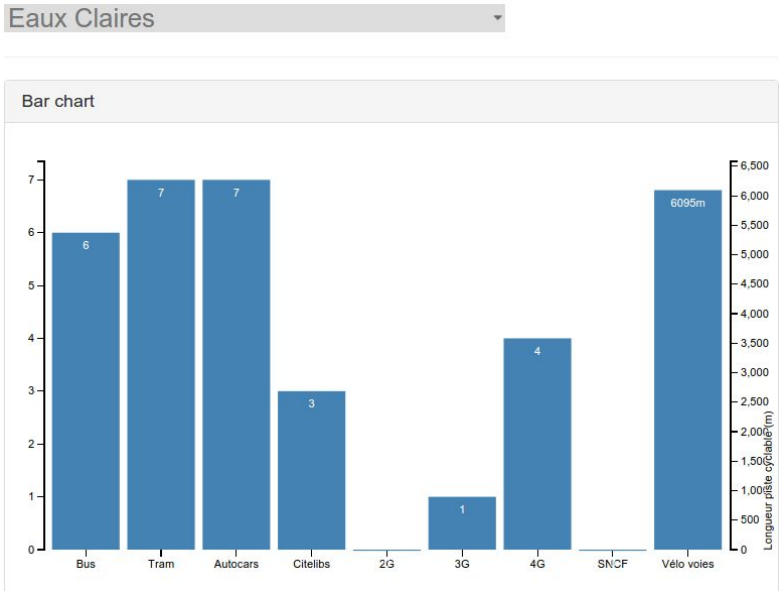
C. Bubble chart

Un graphique sous forme de bulle permet de comparer tous les quartiers selon un critère particulier, et fait immédiatement ressortir les quartiers ayant des notes élevées pour le critère choisi.



D. Histogramme

Un graphique histogramme permet enfin une visualisation de tout les critères pour un quartier particulier.



IV. Architecture de l'application

Les données récupérées par API via un script bash minimaliste sont dans un premier temps importées dans une base de données MongoDB. Celle-ci sert principalement a effectué facilement et rapidement les opérations sur GeoJSON ($geoWithin, $geoNear, …).

Un pré-traitement des données est effectué juste après l'importation via une mini application node.js utilisant un driver mongo pour accéder à la base.

Afin de simplifier l'accès aux données depuis l'application, un serveur d'API REST (python-eve) a été installé. Celui-ci permet un acces direct aux données, mais également des requêtes utilisant toutes les fonctionnalités de mongoDB.

L'application en elle même a été développé avec AngularJs (1.5), et utilise les utilitaires habituels pour la gestion des dépendances (bower, grunt). Elle utilise enfin plusieurs bibliothèque javascript, dont D3.js pour les visualisations et LeafLet pour la carte interactive.

V. Améliorations envisagées

L'application est très dépendante du jeu de données sur les quartiers, puisque tout les autres jeux de données sont corrélés à celui là. On pourrait donc significativement l'améliorer en utilisant des subdivisions des quartiers (réelles ou arbitraires) qui augmenteraient la pertinence des comparaisons sur un quartier entier.
Les quartiers étudiés pourrait également être étendus en dehors de Grenoble même.

On peut également envisager de rajouter de nombreux autres critères (loyer, écoles, santé, …), la seule limitation étant les données disponibles. Si le nombre de critères devient trop élevé, ceux-ci pourront être divisés en catégories (transport, éducation, environnement) pour simplifier l'utilisation de l'application.

Concernant les visualisations en elle-même, davantage d'interactivité pourrait leur être ajouté, notamment :

- Tooltip contenant les détails d'un objet (antenne, restaurant) lorsqu'il est sélectionné sur la carte
- Exclusion de certains quartiers sur les représentations bubbles et coordonnées parallèles, ou de certains critères sur l'histogramme
- Réorganisation manuelle des critères sur le diagramme de coordonnées parallèles
- Liste des objets situés à moins de 200m d'un point choisi sur la carte