Info0939 – Big Data
Master 2 DAR/IIC
Devoir 1 – Sentiment Twitter

Démarrage :

Pour ce TP il vous suffit d'une machine avec Java ou Python.

Analyse des données

Pour l'analyse des données vous allez utiliser un extrait d'un flux de Twitters et le comparer à une liste de mots. Chaque mot dans cette liste a un score positif ou négatif (ex. : beautiful → 3, denied → -2). Le sentiment d'un tweet is l'équivalent de la somme des scores de chaque mot connu dans ce tweet.

Dans le fichier zip vous trouverez donc deux fichiers, *AFINN-111.txt* (le fichier avec les scores) et *three_minutes_tweets.json* (le fichier avec la capture du flux Twitter).

Pour utiliser *AFINN-111.txt*, vous allez probablement construire un dictionnaire. Le fichier *AFINN-111.txt* est délimité par des TAB, ce que veut dire que le mot et le score sont séparés par un TAB (le caractère spécial "\t"). En Python, l'extraction des mots/scores peut se faire de cette manière:

```
afinnfile = open("AFINN-111.txt")
scores = {} # initialize an empty dictionary
for line in afinnfile:
  term, score  = line.split("\t")  # The file is tab-delimited. "\t" means
"tab character"
  scores[term] = int(score)  # Convert the score to an integer.

print scores.items() # Print every (term, score) pair in the dictionary
```

Les données dans le fichier *three_minutes_tweets.json* sont en format JSON (JavaScript Object Notation. Chaque ligne correspond à un message streaming. La plupart de ces messages correspondent à des tweets (*User update*) mais certains messages devront être ignorés (par exemple, les messages *delete*). Vous pouvez avoir une idée de ces messages et de leurs formats à l'adresse https://dev.twitter.com/streaming/overview/messages-type

L'extraction des informations JSON est simple, vous trouverez des bibliothèques en Java ou Python pour cela.

Ensuite, vous devez parcourir le fichier *three_minutes_tweets.json*, extraire le contenu des tweets et comparer chaque mot à la bibliothèque de scores *AFIN-111.txt*. La sortie de votre programme doit indiquer le score total pour chaque Tweet de type "user update" ou "0" pour les lignes des autres types de message, et stocker ce total dans un fichier *scores.txt*, avec une valeur numérique par ligne.

Évaluation

Vous devez créer un fichier .zip au format "*nom-prénom.zip*" contenant votre code source et le fichier *scores.txt*. Le zip doit être déposé sur le bureau virtuel () au plus tard le 11/10/2016 au soir. **Ce devoir est individuel.**

Attention, nous allons évaluer vos réponses, donc il faut que la valeur de la ligne *n* correspond au twitter à la ligne *n*.

Attention 2 : le fichier a été extrait d'un flux réel, donc certains tweets auront d'autres jeux de caractères. Le score doit se baser uniquement sur les mots qui correspondent à ceux dans le fichier *AFIN-111.txt*.

Pour aller plus loin

Le zip contient aussi un fichier "*twitterstream.py*". Il permet de capturer des flux directement de Twitter. Pour cela, il vous faut :

Installer la bibliothèque **oauth2 library** dans votre machine. Vous pouvez le faire avec la commande `$ pip install oauth2` (ça marche dans la plupart des systèmes). Puis, il faut retrouver les identifiants twitter :

1. Créer un compte twitter si vous ne l'avez pas encore.
2. Aller à **https://dev.twitter.com/apps** et se connecter avec vos identifiants.
3. Créer une nouvelle application avec le bouton "Create New App"
4. Remplir le formulaire d'informations. Vous pouvez utiliser une adresse web fausse si vous n'avez pas une page perso.
5. Dans la page suivante, cliquer sur l'onglet "API Keys" et descendre jusqu'à "Your Access Token"
6. Cliquer le bouton "Create My Access Token".
7. Utiliser ces données pour configurer votre code twitterstream.py. Il vous faut l'"API Key", votre mot de passe "API secret", votre "Access token" et votre "Access token secret". Toutes ces valeurs doivent se trouver dans la page API Keys. Éditez le fichier twitterstream.py et remplissez avec ces valeurs.
8. Exécutez le code de la manière suivante.

   ```
   $ python twitterstream.py > output.txt
   ```

   Si tout se passe bien, le code enregistrera les twitters sur output.txt, jusqu'à ce que vous l'interrompiez avec Ctrl-C.

9. Il est aussi possible de limiter la recherche en passant un mot clé (par exemple "microsoft") dans l'URL de la fonction *twitterreq* :

   ```
   https://api.twitter.com/1.1/search/tweets.json?q=microsoft
   ```