Joseph-Marie Valleix - Février 2017

Projet de recherche d'information

Installation

Requirements:

- Python (3.5)
- nltk (3.2)

Données:

Dans un dossier `data`, copier les données des collection cacm et cs271 tel que le contenu du dossier soit :

```
data/
  - cacm/
      - cacm.all
      - common_words
      - qrels.text
      - query.text
  - cs271/
      - pa1-data/
          - 0
          - ...
```

Lancer le programme

Dans un shell :

```
python3 ri.py <source> <type>
# source : cacm or cs271
# type   : bin  or vec
python3 ri.py cacm  bin
python3 ri.py cs271 vec
```

Structure du programme

Le code est organisé de la façon suivante :

- `README.md`
- `ri.py`: point d'entrée du programme

- `data/`: collections de données
- `tools/`: contient le code spécifique aux différentes collections, chaque collection expose un ensemble de fonctions qui sont utilisés pour créer les indexes et effectuer les recherches
  - `cacm.py`: code de la collection cacm + common_words
  - `cs.py`: code de la collection cs271
  - `search.py`: code pour la recherche
  - `token.py`: code pour l'extraction des tokens

Choix d'implémentation

Au vu de la taille des jeux de données, toutes les formes d'indexes et de traitements sont effectués en mémoire.

Résultats sur cacm

Courbe Rappel/Précision

MAP

Améliorations possibles et conclusion