Metody Obliczeniowe w Nauce i Technice

Laboratorium 6
Singular Value Decomposition

# Projekt Wyszukiwarka

Olga Słota

# 1. Web Crawler

Przygotowanie zbioru danych dla programu przeprowadziłam za pomocą Web Crawlera. Rozpoczynając przeszukiwanie i pobieranie stron internetowych od adresu https://en.wikipedia.org/wiki/Universe , przechodząc do kolejnych linków i rekurencyjnie wykonując te same czynności. Zgromazone w ten sposób dokumenty(1028 plików) znajdują się w katalogu Universe.

# 2. Program wyszukujący

Wyszukiwanie zaimplemetowałam w większości w Javie z użyciem pojedynczego skryptu napisanego w Pythonie w celu dokonania przekształceń na macierzy.

a) Słownik termów

Zbiór wszystkich słów występujących we wszystkich tekstach z przypisanymi im indeksami zgodnymi z kolejnością dodawania słów zaimplementowany z użyciem TreeMap.

b) Bag-of-words , term-by-document matrix

Wypełniane wartościami częstości występowania poszczególnych słów w poszczególnych plikach. Macierz jest tworzona z użyciem HashMap poprzez przyporządkowanie pozycji w macierzy(słowo, plik) jej wartości ( częstości).

c) IDF

Po skonstruowaniu macierzy następuje przemnożenie jej elementów przez odpowiedni czynnik, który redukuje znczenie słów często występujących.

d) Zapisanie macierzy do pliku.

Dalsze operacje wykonywane są w języku Python, a wyniki zwracane do

programu głównego.

e) korelacja

Po przyjęciu jako argument i znormalizowaniu wektora q - wejsciowego zapytania oraz znormalizowaniu wektorów bag-of-words oblicznana jest ich korelacja (podobieństwo) jako cosinus kąta między nimi.
Następnie zwracane jest k z nich najbardziej zbliżonych do wektora q.

f) SVD

W celu usunięcia szumu faktoryzacja SVD odbywa się ze zredukowanym rzędem macierzy, a następnie aproksymuje początkową macierz.

# 3. Wyniki , skuteczność, analiza

Prześledźmy działanie programu na przykładzie wyszukiwania słów “cartesian” oraz “Euler”.Najczęstsze wystąpienia tych słów w plikach przedstawiają się następująco:

| | Plik | “Cartesian” | “Euler” |
|---|---|---|---|
| | 48.txt | 8 | - |
| | 49.txt | 16 | - |
| | 50.txt | 15 | 1 |
| | 55.txt | 1 | 7 |
| | 56.txt | 2 | 2 |
| | 57.txt | - | 62 |
| | 59.txt | 2 | 5 |
| | 62.txt | 1 | - |
| | 125.txt | 3 | - |
| | 1009.txt | 1 | - |
| | 1187.txt | - | 1 |

## Bez usuwania szumu za pomocą SVD :

```
Searching for : cartesian euler  in directory /home/olga/workspace/Search/Universe
With settings:
number of best files to find 20
SVD off

words : 78914
files: 1028
[1009.txt, 62.txt, 419.txt, 124.txt, 1186.txt, 1187.txt, 57.txt, 1209.txt, 1210.txt, 51.txt, 113.txt, 1067.txt, 897.txt, 798.txt, 418.txt, 316.txt, 896.txt, 120.txt, 1011.txt, 1279.txt]
execution time in seconds: 423
```

Jak widać wyniki niewiele mają wspólnego z rzeczywistą zawartością plików.
Na początkowych pozycjach pojawiają się pliki, w których szukane słowa nawet nie występują. Z kolei pliki zawierające wskazane słowa pojwiają się dopiero na dalszych pozycjach.

## Usuwanie szumu za pomocą SVD ( rank = 200):

```
Searching for : cartesian euler  in directory /home/olga/workspace/Search/Universe
With settings:
number of best files to find 20
SVD with rank = 200

words : 78914
files: 1028
[59.txt, 55.txt, 1009.txt, 62.txt, 419.txt, 1215.txt, 1187.txt, 57.txt, 1210.txt, 113.txt, 1188.txt, 828.txt, 897.txt, 51.txt, 762.txt, 1067.txt, 237.txt, 418.txt, 316.txt, 1201.txt]
execution time in seconds: 469
```

Widzimy, że plik 57.txt, czyli artykuł “Euler diagram” zawierający najwięcej wystąpień ( aż 62 słowa Euler )  jest dopiero na 8. pozycji.
Spróbujmy SVD z większym rankiem.

## Usuwanie szumu za pomocą SVD ( rank = 250):

```
Searching for : cartesian euler  in directory /home/olga/workspace/Search/Universe
With settings:
number of best files to find 20
SVD with rank = 250

words : 78914
files: 1028
[59.txt, 419.txt, 50.txt, 1186.txt, 124.txt, 1187.txt, 62.txt, 1209.txt, 1215.txt, 57.txt, 113.txt, 1210.txt, 51.txt, 654.txt, 1145.txt, 1181.txt, 1083.txt, 152.txt, 642.txt
execution time in seconds: 510
```

Wyniki są jednak mniej zadawalające niż z rankiem 200. Spróbujmy z mniejszym.

## Usuwanie szumu za pomocą SVD ( rank = 150):

```
Searching for : cartesian euler  in directory /home/olga/workspace/Search/Universe
With settings:
number of best files to find 20
SVD with rank = 150

words : 78914
files: 1028
[1009.txt, 62.txt, 419.txt, 124.txt, 1186.txt, 1187.txt, 57.txt, 1209.txt, 1210.txt, 51.txt, 113.txt, 1067.txt, 897.txt, 798.txt, 418.txt, 316.txt, 896.txt, 120.txt, 1011.txt
execution time in seconds: 451
```

Widać, że to krok w dobrą stronę. Spróbujmy jeszcze zmniejszyć rank .

## Usuwanie szumu za pomocą SVD ( rank = 130):

```
Searching for : cartesian euler  in directory /home/olga/workspace/Search/Universe
With settings:
number of best files to find 20
SVD with rank = 130

words : 78914
files: 1028
[1009.txt, 62.txt, 124.txt, 59.txt, 1186.txt, 126.txt, 1187.txt, 57.txt, 1210.txt, 51.txt, 1188.txt, 897.txt, 798.txt, 418.txt, 1201.txt, 316.txt, 52.txt
execution time in seconds: 388
```

Wyniki jednak się pogorszyły. Spróbujmy w drugą stronę.

## Usuwanie szumu za pomocą SVD ( rank = 180):

```
Searching for : cartesian euler  in directory /home/olga/workspace/Search/Universe
With settings:
number of best files to find 20
SVD with rank = 180

words : 78914
files: 1028
[49.txt, 59.txt, 62.txt, 1186.txt, 1187.txt, 124.txt, 57.txt, 126.txt, 1210.txt, 51.txt, 745.txt, 1067.txt, 418.txt, 798.txt, 1201.txt, 715.txt, 1311.txt
execution time in seconds: 469
```

SVD z rank = 180 daje najlepsze wyniki. Dotychczasowe testy wykonywane były bez IDF, w tym przypadku psuje ono wyniki. Zapewne jest tak dlatego ze szukane słowa pojawiają się w wielu plikach ( np. jednokrotnie ).