Metody obliczeniowe w nauce i technice

Laboratorium 10
Zastosowania dekompozycji
7-8.01.2019

Przydatne linki:
https://pypi.org/project/wikipedia/
https://github.com/shaypal5/awesome-twitter-dat
http://datameetsmedia.com/bag-of-words-tf-idf-explained/

Zadanie 1.

a) Przygotuj duży korpus tekstów w języku angielskim, np. korzystając z web crawlera lub wikipedii (patrz linki).
b) Przygotuj zbiór słów występujących w dokumentach (słownik), następnie wykonaj embedding *bag of words*. Policz także częstość występowania poszczególnych słów.
c) Przygotuj program akceptujący zapytanie użytkownika. Wektory zapisz w formie macierzowej (kolejne wiersze to kolejne słowa).
d) Zaproponuj reprezentację całego dokumentu (oraz zapytania) w oparciu o reprezentacje poszczególnych słów. Poprawna reprezentacja nie powinna faworyzować dokumentów ze względu na ich długość (warto sprawdzić czemu).
e) Do oceny podobieństwa zastosuj metrykę cosinusową. Zwróć *k* najbardziej podobnych wektorów (najlepiej w przyjaznej dla usera formie - tytuły dokumentów?)
f) Zbadaj jak zachowa się wyszukiwarka po aproksymacji macierzy BoW metodą SVD low rank approximation. Dla jakiego rzędu macierzy (bezwzględnego i względem rozmiaru macierzy BoW) wyniki są najlepsze/najgorsze, dlaczego?
g) Aby zmniejszyć wagę występujących słów, które występują w dużej ilości dokumentów, pomnóż każdy *one hot vector* przez odpowiednią liczbę, wyliczoną jako logarytm ze stosunku ilości dokumentów do ilości dokumentów, w których dane słowo występuje co najmniej raz - podejście to znane jest jako TF-IDF. Sprawdź wpływ TF-IDF na działanie wyszukiwarki