Metody Obliczeniowe w Nauce i Technice
Laboratorium 4
Singular Value Decomposition

24 marca 2020

Pojęcia

- Latent Semantic Indexing
- Information Retrieval
- Inverse Document Frequency

Zadanie 1 Wyszukiwarka

1. Przygotuj duży (> 1000 elementów) zbiór dokumentów tekstowych w języku angielskim (np. wybrany korpus tekstów, podzbiór artykułów Wikipedii, zbiór dokumentów `HTML` uzyskanych za pomocą *Web crawlera*, zbiór rozdziałów wyciętych z różnych książek)

2. Określ słownik słów kluczowych (termów) potrzebny do wyznaczenia wektorów cech *bag-of-words* (indeksacja). Przykładowo zbiorem takim może być unia wszystkich słów występujących we wszystkich tekstach.

3. Dla każdego dokumentu j wyznacz wektor cech *bag-of-words* $\mathbf{d}_j$ zawierający częstości występowania poszczególnych słów (termów) w tekście.

4. Zbuduj rzadką macierz wektorów cech *term-by-document matrix* w której wektory cech ułożone są kolumnowo $A_{m\times n} = [\mathbf{d}_1|\mathbf{d}_2|\ldots|\mathbf{d}_n]$ (m jest liczbą termów w słowniku, a n liczbą dokumentów)

5. Przetwórz wstępnie otrzymany zbiór danych mnożąc elementy *bag-of-words* przez *inverse document frequency*. Operacja ta pozwoli na redukcję znaczenia często występujących słów.
$$IDF(w) = \log \frac{N}{n_w}, \tag{1}$$

gdzie n_w jest liczbą dokumentów, w których występuje słowo w, a N jest całkowitą liczbą dokumentów.

6. Napisz program pozwalający na wprowadzenie zapytania (w postaci sekwencji słów) przekształcanego następnie do reprezentacji wektorowej $\mathbf{q}$ (*bag-of-words*). Program ma zwrócić k dokumentów najbardziej zbliżonych do podanego zapytania $\mathbf{q}$. Użyj korelacji między wektorami jako miary podobieństwa

$$\cos\theta_j = \frac{\mathbf{q}^T\mathbf{d}_j}{\|\mathbf{q}\|\|\mathbf{d}_j\|} = \frac{\mathbf{q}^T\mathbf{A}\mathbf{e}_j}{\|\mathbf{q}\|\|\mathbf{A}\mathbf{e}_j\|} \tag{2}$$

7. Zastosuj normalizację wektorów cech $\mathbf{d}_j$ i wektora $\mathbf{q}$, tak aby miały one długość 1. Użyj zmodyfikowanej miary podobieństwa otrzymując

$$|\mathbf{q}^T\mathbf{A}| = [|\cos\theta_1|, |\cos\theta_2|, \ldots, |\cos\theta_n|] \tag{3}$$

8. W celu usunięcia szumu z macierzy $\mathbf{A}$ zastosuj SVD i *low rank approximation* otrzymując

$$\mathbf{A} \simeq \mathbf{A}_k = \mathbf{U}_k\mathbf{D}_k\mathbf{V}_k^T = [\mathbf{u}_1|\ldots|\mathbf{u}_k] \begin{bmatrix} \sigma_1 & & \\ & \ddots & \\ & & \sigma_k \end{bmatrix} \begin{bmatrix} \mathbf{v}_1^T \\ \vdots \\ \mathbf{v}_k^T \end{bmatrix} = \sum_{i=1}^{k} \sigma_i \mathbf{u_i}\mathbf{v_i}^T \tag{4}$$

oraz nową miarę podobieństwa

$$\cos\phi_j == \frac{\mathbf{q}^T\mathbf{A}_k\mathbf{e}_j}{\|\mathbf{q}\|\|\mathbf{A}_k\mathbf{e}_j\|} \tag{5}$$

9. Porównaj działanie programu bez usuwania szumu i z usuwaniem szumu. Dla jakiej wartości k wyniki wyszukiwania są najlepsze (subiektywnie). Zbadaj wpływ przekształcenia IDF na wyniki wyszukiwania.