Regăsirea Informațiilor pe Web

P01 – Indexare și căutare

Student: Neştian Mihai

1. Descriere

Proiectul conține două părți: partea de indexare și cea de căutare.

Modulul de indexare primește ca intrare calea și numele unui director ce conține un set de fișiere de tip *.txt*. Acest director este parcus recursiv și pentru fiecare fișier este construit *indexul direct* , stocat apoi într-o bază de date *MongoDB*. La construirea indexului direct, este parcurs fiecare fișier caracter dupa caracter și sunt determinate cuvintele. Fiecare cuvânt trece prin două teste înainte de a fi adăugat în index. În primul rând se verifică dacă face parte din lista de excepții (din fișierul *exceptions*) . Dacă nu este o excepție, se verifică dacă se găsește în lista de cuvinte ignorate (din fișierul *stopwords)*. Dacă trece aceste teste, cuvântul este adus la forma canonică prin trecerea lui prin algorimul *Porter Stemming din biblioteca nltk* și este adăugat apoi în index, împreună cu numărul de apariții în document. Pe baza indexului direct, se construiește *indexul indirect cantitativ*, care este stocat, la fel ca cel direct, în baza de date *MongoDB*.

Am ales să folosesc algoritmul lui Porter deoarece are o acuratețe mai mare fata de cel al lui Lancaster.Timpul de prelucrare este același, prelucrearea fișierelor a durat 12 secunde cu ambii algoritmi. Am mai incercat si un lemmatizer. Acuratețea acestuia este mult mai bună față de cea a algoritmilor de stemming dar timpul de prelucrare este mult mai mare, prelucrarea celor 26 de fișiere a durat 1:53 față de 12 secunde pentru algoritmii de stemming.

Cuvant	Porter	Lancaster	Lemmatizer
programs	program	program	program
programming	program	program	program
programmers	programm	program	programmer
languages	languag	langu	language
peoples	peopl	peopl	people
programmer	programm	program	programmer
involving	involv	involv	involve
swimming	swim	swim	swim
spliting	splite	spliting	split
are	are	ar	be
going	go	going	go
tissues	tissu	tissu	tissue
bigger	bigger	big	big
smallest	smallest	smallest	small
churches	church	church	church
biggest	biggest	biggest	big
Churchill	churchil	churchil	Churchill

Modulul de căutare cuprinde două metode: căutarea booleană şi căutarea vectorială. Acesta primeşte ca intrare o expresie de căutare din partea utilizatorului şi returnează numele fişierelor relevante pentru expresia introdusă şi scorul obținut de acestea. Pentru căutarea booleană această expresie trebuie să conţină operatori booleeni între cuvinte (*AND, OR, NOT) care sunt reprezentați de (&, |, !)*. Căutarea se face pe baza indecsului indirect, iar criteriile de relevanță sunt: similaritatea cosinus pentru căutarea vectorială şi criteriul deciziei binare şi logica mulțimilor pentru căutarea booleană.

Pentru stocarea datelor s-a folosit o bază de date MongoDB, în care sunt stocați indecşii direct şi indirect, dar şi matricea de ponderi folosită la căutarea booleană, *inverse document frequency* şi valorile ponderilor documentelor (*term frequency * inverse document frequency)* folosite pentru căutarea vectorială.

Toate fişierele de test se găsesc în directorul *Structura*.Cele generate de program se găsesc in Direct_Index şi Indirect_Index.

2. Utilizare

Aplicația foloseşte o bază de date MongoDB , configurată cu setările implicite (*localhost,* port 27017). Numele bazei de date folosite este *mydatabase* şi conţine un număr variabil de colecţii, in funție de numărul fişierelor de intrare:

- ID 1-N - aici este stocat indexul direct
- direct_index – aici este stocată maparea fişierelor, ce colectie de index ii revine fiecărui fişier
- indirect_index_cantitativ – aici este stocat indexul invers
- info_collection – cuprinde TF,IDF şi vectorul asociat fiecărui document

Aplicația conține şi o interfață grafică. Interfața este de tip *tkinter* şi conţine un o căsuță text în care utilizatorul poate introduce expresia de căutare , un button: *Search şi încă o căsuîa text unde vor fi afişate documentele rezultate in urma căutării vectoriale*. Prin apăsarea butonului Search se va apela funcția de căutare vectorială.

3. Auto-evaluare

baza - **1 punct**

1. parcurgere recursivă structura de directoare – 0.5 **puncte**

2. procesare cuvinte + stemming – 2 **puncte**

3. indexare indirectă - 3 **puncte**

4. căutare (în indecşi preîncărcați)

   vectorială (cu restrângerea colecţiei de lucru) **- 3 puncte**

5. stocare indecşi în mongodb - 1 **punct**

Total: 9.5 puncte + 1 punct bonus