2. Projekts

Bula vaicājumu apstrāde un invertētājs indekss

Uzdevums

Dotā projektā jums tiek dots ievades teksta fails (sample.txt), kas satur dokumentu identifikatorus un teikumus (dokumentus). Balstoties uz piedāvātā teksta faila, jums ir nepieciešams izveidot invertēto indeksu, kā arī sistēmu bula vaicājumu apstrādei. Papildus jums ir nepieciešams aprēķināt dokumenta rangus izmantojot TF-IDF, kā arī sakārtot rezultātus atbilstoši TF-IDF svariem. Papildiniet sistēmu ar iespēju izpildīt aizstājēj vaicājumus. Šim mērķim uzbūvējiet bi-gram vai permutācijas indeksu.

Ievades dati

Katra teksta rinda failā sample.txt ir dokuments. Dokumenta identifikators atrodas rindas sākumā un atdalīts no paša dokumenta ar tabulācijas simbolu.

Piemēram:

```
1234   To be or not to be
1235   Let it be
```

I. Solis: Invertētā indeksa izveide

Sadaliet dokumentus terminos, izmantojot atstarpēs simbolus. Nedzēsiet punktuācijas simbolus, tādus kā punkti, komati, pēdiņas. No iegūtiem termiņiem izveidojiet vārdnīcu un norīkojumu sarakstus (postings). Norīkojumu sarakstos dokumentu identifikatoriem ir jābūt sakārtotiem augošā secībā. Piemēram:

```
Term1 = doc1  doc5  doc7
Term2 = doc2  doc5  doc8
```

Vārdnīcu un norīkojumu sarakstus ir ieteicams glabāt operatīvā atmiņā, izmantojot sarakstus. Vārdnīcas izveidei atļauts izmantot kokus vai jaucējtabulas.

II. Solis: Bula vaicājumu apstrāde

Izstrādājiet sekojošas metodes, kas izpilda bula vaicājumu kopu (AND/OR) izmantojot jūsu invertēto indeksu. Vaicājumu rezultāti ir jāsaglabā teksta failā.

1. Metode: GetPostings

Dotai metodei kā parametru ir jāsaņem terminu kopa (term1, term2, …, termN) un kā rezultāts ir jādod atpakaļ norādīto terminu norīkojumu sarakstu kopa.

Norīkojumu saraksti ir jāizvada rezultējošā failā sekojošā formātā (dokumentu identifikatoriem ir jābūt sakārtotiem augošā secībā):

```
GetPostings
term0
Postings: 1001 2002 3003 …
term1
Postings: 1010 2020 3030 …
…
termN
Postings: 1111 2222 3333 …
```

2. Metode: QueryAnd

Dotai metodei kā parametrs ir jāsaņem terminu kopa (term1, term2, …, termN) un jāsameklē dokumenti, kas satur visus norādītos terminus. Kā rezultātu ir jādod atpakaļ atbilstošs dokumentu identifikatoru saraksts.

Rezultāti ir jāizvada failā sekojošā formātā (dokumentu identifikatoriem ir jābūt sakārtotiem augošā secībā):

```
QueryAnd
term0 term1 … termN
Results: 1001 2002 3003 …
```

Ja nav neviena dokumenta, kas satur visus terminus, tad failā ir jāizvadā:

```
QueryAnd
term0 term1 … termN
Results: empty
```

3. Metode: QueryOr

Dotai metodei kā parametrs ir jāsaņem terminu kopa (term1, term2, …, termN) un jāsameklē dokumenti, kas satur vismaz vienu no norādītiem terminiem. Kā rezultātu ir jādod atpakaļ atbilstošs dokumentu identifikatoru saraksts.

Rezultāti ir jāizvada failā sekojošā formātā (dokumentu identifikatoriem ir jābūt sakārtotiem augošā secībā):

```
QueryOr
term0 term1 … termN
Results: 1001 2002 3003 …
```

Ja nav neviena dokumenta, kas satur kādu no terminiem, tad failā ir jāizvada:

```
QueryOr
term0 term1 … termN
Results: empty
```

III. Solis: TF-IDF svaru noteikšana

Izstrādājiet metodi, kas noteic svarus atlasītiem dokumentiem. Izmantojiet sekojošas formulas svara aprēķināšanai:

TF(t) = (termina t skaits dokumentā) / (kopējais terminu skaits dokumentā)

IDF(t) = (kopējais dokumentu skaits) / (dokumentu skaits, kas satur terminu t)

*TF-IDF(t) = TF(t) * IDF(t)*

Dokumenta ranga noteikšanai izmantojiet formulu:

$$Score(doc, query) = \sum_{t \in query} TF - IDF(t)$$

Izmantojiet izstrādātu metodi, lai sakārtotu dokumentu identifikatorus, atlasītus ar metodēm QueryAnd un QueryOr, atbilstoši aprēķinātiem svariem (dilstošā secībā).

Metodes rezultāti ir jāizvada failā sekojošā formātā:

```
TF-IDF
term0 term1 … termN
Results: 1001 2002 3003 …
```

Ja nav neviena dokumenta, kas atbilst bula vaicājumam, tad izvadīt:

```
TF-IDF
term0 term1 … termN
Results: empty
```

IV. Solis: Aizstājēj vaicājumu izpilde

Papildiniet izstrādātu sistēmu ar permutācijas vai bi-gram indeksu, lai nodrošinātu aizstājēj vaicājumu izpildīšanu. Izstrādājiet metodi WildCardQuery, kas kā parametru saņem šablonu, kas satur vienu vai vairākus simbolus "zvaigznīte" (*), un izvada visus vārdnīcas terminus, kas atbilst norādītam šablonam, kā arī doto terminu norīkojumu sarakstus.

Metodes rezultāti ir jāizvada failā sekojošā formātā:

```
WildCard
šablons
Results:
term0
```

```
Postings: 1001 2002 3003 …
term1
Postings: 1010 2020 3030 …
…
termN
Postings: 1111 2222 3333 …
```

Prasības programmai:

Programmai ir jābūt spējīgai ievadīt sekojošu failu nosaukumus:

- Tekstā faila vārdu, kas satur dokumentu identifikatorus un teikumus (dokumentus).

- Tekstā faila vārdu, kas satur vaicājuma terminu sarakstu. Katra rinda dotā failā atbilst vienam vaicājumam (var būt vairāki vaicājumi). Terminu skaits katrā vaicājumā var svārsties. Termini tiks atdalīti ar vienu atstarpes simbolu. Piemēram:

  ```
  term1 term2
  term3
  term4 term5 term6
  ```

  Apstrādājot katru no vaicājumiem izvadīt:
  - Katram terminam izvadīt norīkojumu sarakstu ar metodi GetPostings
  - Izvadīt vaicājuma And rezultātus
  - Izvadīt vaicājuma And rezultātus sakārtotus atbilstoši TF-IDF svariem
  - Izvadīt vaicājuma Or rezultātus
  - Izvadīt vaicājuma Or rezultātus sakārtotus atbilstoši TF-IDF svariem

- Tekstā faila vārdu, kas satur aizstājēj vaicājuma terminus (katrs termins atrodas jaunā rindā), piemēram:

  ```
  term1
  term2
  …
  termN
  ```

- Rezultējoša faila vārdu, kurā programmai ir jāizvada rezultāti.

Vērtēšana

Kopējais punktu skaits par projektu ir 45 (30% no gala atzīmes).

- Invertēta indeksa uzbūve un metode GetPostings: 10 punkti
- Bula vaicājumu izpildē (metodes QueryAnd un QueryOr): 10 punkti
- TF-IDF svaru noteikšana un rezultātu sakārtošana atbilstoši svariem: 10 punkti
- Aizstājēj vaicājumu izpilde un bi-gram vai permutācijas indeksa uzbūve – 15 punkti

Piezīme:

Piemēram, ja teksta fails, kas satur bula vaicājumu terminu sarakstu izskatās sekojošā veidā:

```
term1 term2 term3
term4 term5
```

Bet fails, kas satur aizstājēj vaicājuma terminus izskatās sekojošā veidā:

```
šablons1
šablons2
```

Tad rezultējošam failam ir jāizskatās sekojošā veidā:

```
GetPostings
term1
Postings: 1001 2002 3003 …
term2
Postings: 1010 2020 3030 …
term3
Postings: 1111 2222 3333 …

QueryAnd
term1 term2 term3
Results: 1001 2002 3003 …

TF-IDF
term1 term2 term3
Results: 3003 2002 1001 …

QueryOr
term1 term2 term3
Results: 1010 2020 3030 …

TF-IDF
term1 term2 term3
Results: 3030 2020 1010 …

GetPostings
term4
Postings: 1001 2002 3003 …
term5
Postings: 1010 2020 3030 …

QueryAnd
term4 term5
```

```
Results: 1001 2002 3003 …

TF-IDF
term4 term5
Results: 3003 2002 1001 …

QueryOr
term4 term5
Results: 1010 2020 3030 …

TF-IDF
term4 term5
Results: 3030 2020 1010 …

WildCard
šablons1
Results:
term0
Postings: 1001 2002 3003 …
term1
Postings: 1010 2020 3030 …
…
termN
Postings: 1111 2222 3333 …

WildCard
šablons2
Results:
term0
Postings: 1001 2002 3003 …
term1
Postings: 1010 2020 3030 …
…
termN
Postings: 1111 2222 3333 …
```