Sveučilište u Zagrebu
Fakultet elektrotehnike i računarstva
Zavod za elektroniku, mikroelektroniku, računalne i inteligentne sustave

Analiza velikih skupova podataka

Autori: Marin Šilić, Klemo Vladimir

2. laboratorijska vježba

U ovoj laboratorijskoj vježbi zadatak je ostvariti algoritme preporučivanja zasnovane na tehnici suradničkog filtriranja (eng. *Collaborative Filtering*). Postoje dva osnovna pristupa suradničkog filtriranja, *item-item* pristup suradničkog filtriranja i *user-user* pristup suradničkog filtriranja.

Algoritmi suradničkog filtriranja detaljno su opisani u sklopu predavanja Recommender Systems koje je dostupno na stranici predmeta u mapi predavanja:
https://www.fer.unizg.hr/_download/repository/AVSP_11_RecSys.pdf
(slide 26 - slide 38)

5.1 Format zapisa ulazne datoteke

Prva linija u ulaznoj datoteci sadrži broj stavki *N* (eng. *Items*) i broj korisnika *M* (eng. *Users*) koji su odijeljeni jednim praznim znakom. Pritom vrijedi (*N* <= 100, *M* <= 100).
Nakon toga slijedi zapis *user-item* matrice u kojoj su vrijednosti koje nedostaju prikazane znakom 'X'.
Zapis user-item matrice čini *N* linija od kojih svaka linija sadrži *M* vrijednosti odjeljenih jednim praznim znakom. Vrijednosti u matrici mogu biti cijeli brojevi u rasponu od 1 do 5. U slučaju da vrijednosti pojedinih elemenata matrice nisu dostupne, tada su ti elementi označeni znakom 'X'.
Nakon zapisa matrice, iduća linija u ulaznoj datoteci jest konstanta *Q* koja predstavlja broj upita (1 <= *Q* <= 100). Nakon toga, slijedi *Q* linija od kojih svaka linija predstavlja jedan upit i ima sljedeći format. Upit čine 4 broja *I*, *J*, *T* i *K* koji su odijeljeni praznim znakovima. Broj *I* (1 <= *I* <= *N*) predstavlja jednu stavku u matrici, dok *J* (1 <= *J* <= *M*) predstavlja jednog korisnika u matrici (*I*, *J* zapravo predstavljaju koordinate elementa matrice označenog znakom 'X' - element za koji je potrebno izračunati vrijednost preporuke). Vrijednost *T* određuje tip algoritma koji je potrebno koristiti. Ako *T* ima vrijednost 0, potrebno je koristiti *item-item* pristup suradničkog filtriranja. U sluačaju da *T* ima vrijednost 1, tada je potrebno koristiti *user-user* pristup suradničkog filtriranja. Vrijednost *K* (1 <= *K* <= *N*, *M*) predstavlja maksimalni kardinalni broj skupa sličnih stavki/korisnika koje sustav preporuke razmatra prilikom računanja vrijednosti preporuka.

Za svaki upit program treba ispisati vrijednost preporuke u zasebnoj liniji u skladu sa parametrima upita.

Potrebno je ispisati prve 3 decimale, koristiti rounding mode HALF_UP!

Npr. u programskom jeziku **Java**, to se postiže sljedećim programskim odsječkom:

```
DecimalFormat df = new DecimalFormat("#.000");
BigDecimal bd = new BigDecimal(result);
BigDecimal res = bd.setScale(3, RoundingMode.HALF_UP);
System.out.println(df.format(res));
```

Slično za programski jezik **Python**:

```
from decimal import Decimal, ROUND_HALF_UP
Decimal(Decimal(x).quantize(Decimal('.001'), rounding=ROUND_HALF_UP))
```

Slijedi primjer ulazne datoteke:

```
5 5
1 2 X 2 4
2 X 3 X 5
3 1 X 4 X
X 2 4 X 4
1 X 3 4 X
3
1 3 0 1
4 1 0 2
5 5 1 3
```

Za ovaj primjer, program treba ispisati sljedeći izlaz:

```
3.000
2.198
2.560
```

Napomene:

Kako algoritam prilikom računanja preporuke ne razmatra entitete koji nisu slični s trenutnim entitetom (tj. *similarity*(*A*, *B*) <= 0), **moguće je da se vrijednost preporuke računa na temelju manje od K vrijednosti!** Svi ulazni primjeri jamče da će se preporuke uvijek moći izračunati makar i na temelju samo jedne vrijednosti u matrici.

Ulazna točka za Java rješenja treba biti u razredu **CF**, a ulazna točka Python rješenja treba biti u datoteci **CF.py**.

Vremensko ograničenje na izvođenje programa za bilo koju ulaznu definiciju automata jest 10 sekundi.

5.2 Primjer za provjeru valjanosti

Na stranicama predmeta postavljen je primjer ulazne datoteke s pripadajućim očekivanim izlazom (*labCF_primjer.zip*). Preporučamo provjeru ispravnosti na temelju zadanog primjera prije predaje vježbe na susta.