Tugas Machine Learning Praktikum

Sistem Rekomendasi



Oleh:

 $Ferry\ Triwantono-082111633094$

FAKULTAS SAINS DAN TEKNOLOGI PROGRAM STUDI SISTEM INFORMASI UNIVERSITAS AIRLANGGA

2023

Sistem Rekomendasi

Syntax beserta output:

```
import pandas as pd
from scipy import sparse
ratings = pd.read_csv(r'D:\Coolyeah\Mata Kuliah\SMT 4\Machine Learning
Praktikum\Modul 8\ratings.csv')
movies = pd.read csv(r'D:\Coolyeah\Mata Kuliah\SMT 4\Machine Learning
Praktikum\Modul 8\movies.csv')
ratings = pd.merge(movies,ratings).drop(['genres','timestamp'], axis=1)
print(ratings.shape)
ratings.head()
 (100836, 4)
     movield
                        title userld rating
           1 Toy Story (1995)
                                         4.0
             Toy Story (1995)
                                         4.0
  2
           1 Toy Story (1995)
                                         4.5
              Toy Story (1995)
                                         2.5
  4
           1 Toy Story (1995)
                                  17
                                         4.5
```

Penggabungan dataframe Ratings dengan movies menggunakan fungsi merge() dari library pandas yang memungkinkan menggabungkan dua dataframe berdasarkan kolom yang sama pada kedua data. Kemudian setelah digabungkan, kolom 'genres' dan 'timestamp' dihapus dari dataframe menggunakan fungsi drop() dari library pandas

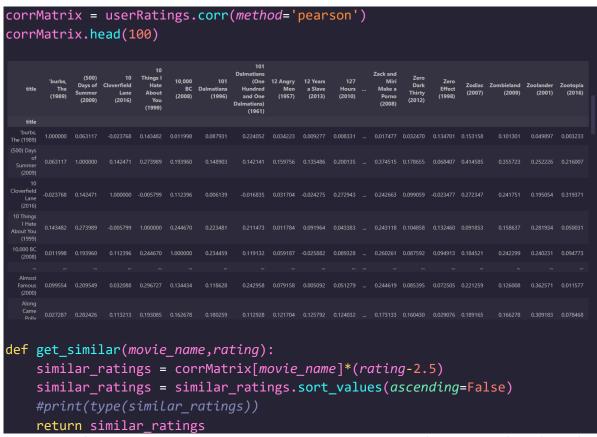
Output dari code diatas adalah berupa ukuran dataframe 'ratings' setelah dilakukan merge serta penghapusan kolom 'genres' dan 'timestamp dan memunculkan 5 baris pertama dari dataframe 'ratings' yang telah diolah

```
userRatings = ratings.pivot_table(index=['userId'], columns=['title'],
values='rating')
userRatings.head()
print("Before : ", userRatings.shape)
userRatings = userRatings.dropna(thresh=10, axis=1).fillna(0, axis=1)
print("After : ", userRatings.shape)
```

```
Before : (610, 9719)
After : (610, 2269)
```

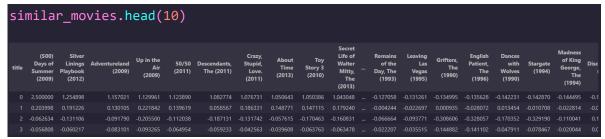
Disini terjadi pengubahan dataframe 'ratings' menjadi pivot table yang akan menunjukan nilai rating untuk setiap user pada setiap judul film yang diberikan. Dengan parameter index diisi dengan kolom 'userld', parameter columns diisi dengan kolom 'title', dan parameter values diisi dengan kolom 'rating'.

Kemudian ukuran dari pivot table sebelum dan sesudah penghapusan kolom dengan nilai null dicetak ke layer menggunakan fungsi shape(). Kemudian ada syntax dropna() untuk menghapus kolom dan fillna() untuk mengisi kolom yang memiliki nilai null dengan 0



Penggunaan fungsi get_similar yang menggunakan 2 parameter, yaitu 'movie_name' dan 'rating' akan menghasilkan daftar film film yang mirip dengan film yang dipilih berdasarkan perhitungan korelasi antara film tersebut

```
romantic_lover = [("(500) Days of Summer (2009)",5),("Alice in Wonderland
(2010)",3),("Aliens (1986)",1),("2001: A Space Odyssey (1968)",2)]
similar_movies = pd.DataFrame()
for movie,rating in romantic_lover:
    similar_movies =
similar_movies.append(get_similar(movie,rating),ignore_index = True)
```



Fungsi get_similar() digunakan lagi pada setiap film dan rating dalam list 'romantic_lover' untuk mencari daftar film film yang mirip dengan film yang dipilih pada list 'romantic_lover'. Hasil dari fungsi tersebut akan di append ke dataframe 'similar_movies'

```
similar_movies.sum().sort_values(ascending=False).head(20)
 (500) Days of Summer (2009)
                                               2.584556
Alice in Wonderland (2010)
                                               1.395229
Silver Linings Playbook (2012)
                                               1.254800
 Yes Man (2008)
                                               1.116264
Adventureland (2009)
                                               1.112235
Marley & Me (2008)
                                               1.108381
 About Time (2013)
                                               1.088757
50/50 (2011)
 Help, The (2011)
Up in the Air (2009)
                                               1.053037
Holiday, The (2006)
Friends with Benefits (2011)
Notebook, The (2004)
                                               1.025880
                                               1.015771
Secret Life of Walter Mitty, The (2013)
                                               0.997979
Perks of Being a Wallflower, The (2012)
                                               0.967425
                                               0.963276
Ugly Truth, The (2009)
                                               0.959079
Harry Potter and the Half-Blood Prince (2009)
                                             0.954180
```

Penghitungan nilai kumulatif dari seluruh nilai korelasi pada dataframe 'similar_movies' untuk setiap judul film. Kemudian, hasil perhitungan tersebut akan diurutkan dari nilai yang paling besar ke yang paling kecil.

```
action_lover = [("Amazing Spider-Man, The (2012)",5),("Mission: Impossible
III (2006)",4),("Toy Story 3 (2010)",2),("2 Fast 2 Furious (Fast and the
Furious 2, The) (2003)",4)]
similar_movies = pd.DataFrame()
for movie,rating in action_lover:
    similar_movies =
similar_movies.append(get_similar(movie,rating),ignore_index = True)
similar_movies.head(10)
similar_movies.sum().sort_values(ascending=False).head(20)
```

```
Amazing Spider-Man, The (2012)
                                                       3.233134
                                                      2.874798
Mission: Impossible III (2006)
2 Fast 2 Furious (Fast and the Furious 2, The) (2003) 2.701477
Over the Hedge (2006)
                                                      2.229721
Crank (2006)
                                                      2.176259
Mission: Impossible - Ghost Protocol (2011)
                                                      2.159666
Hancock (2008)
                                                      2.156098
The Amazing Spider-Man 2 (2014)
                                                      2.153677
Hellboy (2004)
                                                      2.137518
Snakes on a Plane (2006)
                                                      2.137396
Jumper (2008)
                                                      2.129716
Chronicles of Riddick, The (2004)
                                                      2.121689
Tron: Legacy (2010)
                                                      2.111843
Fantastic Four (2005)
                                                       2.083022
X-Men: The Last Stand (2006)
                                                       2.077530
Wreck-It Ralph (2012)
                                                       2.067907
Kung Fu Hustle (Gong fu) (2004)
                                                       2.067457
Godzilla (2014)
                                                       2.061653
Incredible Hulk, The (2008)
                                                       2.050104
Quantum of Solace (2008)
                                                       2.016189
dtype: float64
```

Sama halnya dengan 'romantic_lover' namun hanya diganti pada list film apa saja yang dipilih dan dimasukan ke dataframe 'action lover'