به نام خدا

گزارشکار پروژه‌ی سیستم توصیه‌گر فیلم

جبر خطی کاربردی

استاد راهنما: دکتر پیمان ادیبی

اعضای گروه:

پوریا طلائی

محمدکاظم هرندی

زمستان ۱۴۰۲

فهرست مطالب

مسئله

سامانه‌های توصیه‌گر امروزه در همه جا حاضر شده‌اند و پیشنهادهای شخصی‌سازی شده برای فیلم‌ها، موسیقی، کتاب‌ها، محصولات و موارد دیگر را به کاربران ارائه می‌کنند. این سیستم‌ها نقش مهمی در بهبود تجربه و تعامل کاربر و همچنین پیشبرد رشد کسب و کار دارند. در این پروژه قصد داریم تا با استفاده از تجزیه مقادیر منفرد[1] یک سیستم توصیه‌گر فیلم ایجاد کنیم.

مقدمه

در این پروژه، یک سیستم توصیه بر مبنای SVD برای پیشنهاد دادن فیلم‌ها به کاربران پیاده‌سازی شده است. این پروژه از زبان برنامه‌نویسی Python و کتابخانه‌های Pandas و NumPy برای مدیریت و پردازش داده‌ها استفاده می‌کند.

در ادامه به موارد جزئی تر و اطلاعات فنی دقیق‌تر در مورد پیاده‌ســـازی و عملکرد هر بخش از کد در چند مرحله اشاره خواهیم کرد.

اطلاعات

```python
def read_data(movies_file, ratings_file):
    try:
        movies = pd.read_csv(movies_file)
        ratings = pd.read_csv(ratings_file)
    except FileNotFoundError as e:
        raise FileNotFoundError("Could not find file: " + str(e))

    return movies, ratings
```

۱خواندن اطلاعات

این تابع مسئول خواندن داده‌های مرتبط با فیلم‌ها و امتیازات کاربران از فایل‌های موردنظر است. درصورت وجود هر یک از فایل‌ها، از کتابخانه Pandas برای خواندن اطلاعات استفاده می‌شود. در صورتی که یک یا هر دوی این فایل‌ها یافت نشوند[2]، یک استثناء ایجاد می‌شود و یک پیام خطا به کاربر اطلاع داده می‌شود که فایل مورد نظر پیدا نشد. در نهایت، اطلاعات مربوط به فیلم‌ها و امتیازات کاربران به عنوان خروجی تابع برگردانده می‌شوند.

[1] Singular Value Decomposition
[2] FileNotFoundError

ادغام اطلاعات

```python
def merge_movie_data(user_movie_matrix, movies):
    movies['movieId'] = pd.to_numeric(movies['movieId'], errors='coerce')
    user_movie_matrix = user_movie_matrix.merge(movies, left_on='movieId', right_on='movieId', how='left')

    return user_movie_matrix.dropna(subset=['title'])
```

۲ادغام اطلاعات

در ابتدا، ستون 'movieId' در جدول فیلم‌ها به عدد تبدیل می‌شود. این عمل باعث مشکل در تبدیل اعداد نامعتبر به NaN می‌شـــود. ســـپس جدول فیلم‌ها را با ماتریس امتیازات کاربران ادغام می‌کند. ادغام بر اســاس ســتون 'movieId' انجام می‌شــود و روش ادغام 'left' اســت، بنابراین تمام ردیف‌های موجود در ماتریس امتیازات کاربران حفظ می‌شـــوند و اطلاعات جدول فیلم‌ها به تعداد ممکن به ماتریس امتیازات کاربران افزوده می‌شوند. در نهایت، ردیف‌هایی که اطلاعات عنوان[1] ندارند حذف می‌شوند و ماتریس نهایی با اطلاعات فیلم‌ها و امتیازات کاربران آماده می‌شود. این جدول به صورت زیر تبدیل می‌شود:

	movieId	title	genres	userId	rating	timestamp
0	1	Toy Story (1995)	Adventure|Animation|Children|Comedy|Fantasy	1.0	4.0	9.649827e+08
1	1	Toy Story (1995)	Adventure|Animation|Children|Comedy|Fantasy	5.0	4.0	8.474350e+08
2	1	Toy Story (1995)	Adventure|Animation|Children|Comedy|Fantasy	7.0	4.5	1.106636e+09
3	1	Toy Story (1995)	Adventure|Animation|Children|Comedy|Fantasy	15.0	2.5	1.510578e+09
4	1	Toy Story (1995)	Adventure|Animation|Children|Comedy|Fantasy	17.0	4.5	1.305696e+09
...	...	...	...	...	...	...
100849	193581	Black Butler: Book of the Atlantic (2017)	Action|Animation|Comedy|Fantasy	184.0	4.0	1.537109e+09
100850	193583	No Game No Life: Zero (2017)	Animation|Comedy|Fantasy	184.0	3.5	1.537110e+09
100851	193585	Flint (2017)	Drama	184.0	3.5	1.537110e+09
100852	193587	Bungo Stray Dogs: Dead Apple (2018)	Action|Animation	184.0	3.5	1.537110e+09
100853	193609	Andrew Dice Clay: Dice Rules (1991)	Comedy	331.0	4.0	1.537158e+09

100854 rows × 6 columns

۳ماتریس ادغام

[1] title

پیش پردازش داده‌ها

```python
def preprocess_data(user_movie_matrix):
    user_movie_matrix = user_movie_matrix.pivot_table(index='userId', columns='movieId', values='rating')
    user_movie_matrix = user_movie_matrix.fillna(0)
    user_movie_matrix.columns = user_movie_matrix.columns.astype(int)

    return user_movie_matrix
```

٤پیش پردازش

ابتدا، داده‌های امتیازات کاربران به یک جدول خطی تبدیل می‌شوند. مقادیر خالی[1] در جدول با مقدار صفر جایگزین می‌شوند. این به دلیل عدم ارتباط برخی از فیلم‌ها با برخی از کاربران می‌باشد. تمام ستون‌های جدول به نوع داده صحیح[2] تبدیل می‌شوند. این تغییر باعث ایجاد ماتریس زیر برای ادامه کار می‌باشد.

movieId	1	2	3	4	5	6	7	8	9	10	...	193565	193567	193571	193573	193579	193581	193583	193585	193587	193609
userId																					
1.0	4.0	0.0	4.0	0.0	0.0	4.0	0.0	0.0	0.0	0.0	...	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
2.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	...	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
3.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	...	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
4.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	...	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
5.0	4.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	...	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...
606.0	2.5	0.0	0.0	0.0	0.0	0.0	2.5	0.0	0.0	0.0	...	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
607.0	4.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	...	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
608.0	2.5	2.0	2.0	0.0	0.0	0.0	0.0	0.0	0.0	4.0	...	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
609.0	3.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	4.0	...	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
610.0	5.0	0.0	0.0	0.0	0.0	5.0	0.0	0.0	0.0	0.0	...	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0

610 rows × 9724 columns

۵ ماتریس نهایی

[1] NaN
[2] integer

تجزیه مقادیر منفرد

```python
def svd(matrix, tolerance=1e-10):
    covariance_matrix = np.dot(matrix.T, matrix)

    eigenvalues, eigenvectors = np.linalg.eigh(covariance_matrix)

    sorted_indices = np.argsort(eigenvalues)[::-1]
    eigenvalues = eigenvalues[sorted_indices]
    eigenvectors = eigenvectors[:, sorted_indices]

    singular_values = np.sqrt(np.abs(eigenvalues))
    U = matrix.dot(eigenvectors) / singular_values
    VT = eigenvectors.T

    U = np.nan_to_num(U)
    VT = np.nan_to_num(VT)

    mask = singular_values > tolerance
    singular_values = singular_values[mask]
    U = U[:, mask]
    VT = VT[mask, :]

    return U, singular_values, VT
```

٦محاسبه‌ی مقادیر منفرد

در این تابع، تجزیه به ماتریس SVD بر روی ماتریس ورودی انجام می‌شود. در ادامه توضیح دقیقتری به مراحل و عملکرد هر قسمت از کد آمده است:

محاسبه ماتریس کوواریانس: این مرحله با ضرب ترانهاده ماتریس ورودی در خودش، ماتریس کوواریانس را ایجاد می‌کند. این ماتریس مهمی در مراحل بعدی تجزیه به ماتریس SVD است.

محاسبه مقادیر و وکتورهای ویژه: این بخش با استفاده از تابع eigh مقادیر و وکتورهای ویژه ماتریس کوواریانس را محاسبه می‌کند.

مرتب‌سازی مقادیر و وکتورهای ویژه: مقادیر و وکتورهای ویژه بر اساس مقدار مرتب شده مرتب می‌شوند.

محاسبه مقادیر واحد تک‌تکرار: مقادیر واحد تک‌تکرار با محاسبه مقادیر مثبت و جذر می‌شوند.

محاســبه ماتریس‌های U و VT: ماتریس‌های U و VT به عنوان ماتریس‌های تک‌تکرار حاصــل از تجزیه SVDمحاسبه می‌شوند.

تصحیح مقادیر NaN و حذف مقادیر کوچک: در این بخش، مقادیر NaN در ماتریس‌های U و VT به صفر تبدیل می‌شوند و مقادیر واحد تک‌تکرار کمتر از یک حد تعیین شده توسط toleranc حذف می‌شوند.

خروجی: ماتریس‌های U و VT همراه با مقادیر واحد تک‌تکرار به عنوان خروجی تابع ارائه می‌شوند. این ماتریس‌ها قابل استفاده در تحلیل داده‌ها و تولید پیش‌بینی‌ها بر اساس تجزیه SVD هستند.

سیستم توصیه

```python
def recommendation_system(user_id, movies, U, Sigma, VT, num_recommendations=10):
    if user_id > U.shape[0]:
        raise ValueError("User ID is out of range")

    user_ratings = np.dot(U[user_id - 1, :] * Sigma, VT)
    recommended_movies_idx = np.argsort(user_ratings)[::-1][:num_recommendations]

    recommendations_df = movies.loc[movies['movieId'].isin(recommended_movies_idx)]
    recommendations_string = recommendations_df[['title', 'genres']].to_string(index=False)

    return recommendations_string
```

۷سیستم توصیه

با اســتفاده از ماتریس‌های U، Sigma و VT محاســبه امتیازهای تخمینی کاربر برای تمام فیلم‌ها انجام می‌شود. سپس امتیازهای کاربر بر اساس مقدار مرتب می‌شوند و تعداد مشخصی از فیلم‌های با بالاترین امتیازات برای پیشنهاد به کاربر انتخاب می‌شوند. درنهایت اطلاعات فیلم‌های پیشنهادی را از جدول اصلی فیلم‌ها با اســتفاده از شــناســه فیلم‌ها اســتخراج می‌کند و اطلاعات فیلم‌های پیشــنهادی را به یک رشــته مناسب برای چاپ تبدیل می‌کند.

نمونه اجرای برنامه

همانطور که می‌دانیم محاسبت برای بدست پیدا کردن مقادیر SVD کمی بالاست، پس باید برای انجام محاسبات کمی صبر کنیم. در ادامه با توجه به اطلاعاتی که داشتیم، به ۶۱۰ کاربر موجود می‌توانیم فیلم هایی برای مشاهده توصیه کنیم. مواردی از آنها را در خروجی زیر می توانید مشاهده کنید.

```
Enter a user ID: 13
Recommended movies for user 13:
                    title                                    genres
        Restoration (1995)                                     Drama
Singin' in the Rain (1952)                     Comedy|Musical|Romance
  Lost Weekend, The (1945)                                     Drama
         52 Pick-Up (1986)                     Action|Mystery|Thriller
Run Silent Run Deep (1958)                                       War
 Outside Providence (1999)                                   Comedy
   Yellow Submarine (1968) Adventure|Animation|Comedy|Fantasy|Musical
         Body Shots (1999)                                     Drama
        Rawhead Rex (1986)                            Horror|Thriller
   Babes in Toyland (1934)            Children|Comedy|Fantasy|Musical
Enter a user ID: 325
Recommended movies for user 325:
                     title               genres
         Crimson Tide (1995)    Drama|Thriller|War
  In the Line of Fire (1993)       Action|Thriller
              Thinner (1996)       Horror|Thriller
American in Paris, An (1951)       Musical|Romance
 Little Princess, The (1939)        Children|Drama
          Real Genius (1985)               Comedy
             In & Out (1997)               Comedy
   American History X (1998)           Crime|Drama
        Baby Geniuses (1999)               Comedy
      King and I, The (1956) Drama|Musical|Romance
Enter a user ID: 609
Recommended movies for user 609:
                         title                       genres
             Sudden Death (1995)                       Action
          Babysitter, The (1995)                Drama|Thriller
               Just Cause (1995)               Mystery|Thriller
   Miracle on 34th Street (1994)                        Drama
Secret of Roan Inish, The (1994) Children|Drama|Fantasy|Mystery
             Philadelphia (1993)                        Drama
```

۸خروجی برنامه

منابع

- machinelearningmastery.com/using-singular-value-decomposition-to-build-a-recommender-system
- analyticsindiamag.com/singular-value-decomposition-svd-application-recommender-system
- www.geeksforgeeks.org/singular-value-decomposition-svd/?ref=gcse
- www.geeksforgeeks.org/compute-the-factor-of-a-given-array-by-singular-value-decomposition-using-numpy/?ref=gcse
- black box
- bard.google.com
- chat.openai.com