Product Recommendation System

Group No. 28

- Tarun Kumar Gupta
- Shubham Sethi
- Kartik Gupta
- Rachit Gupta
- Akshat Tilak
- Ritika Nagar

1. MOTIVATION

With the wide variety of products and services available on the web, it is difficult for users to choose the product or service that most meets their needs. To reduce or even eliminate this difficulty, recommender systems have emerged. Recommendation systems are a particular type of information filtering system. They are software applications that aim to support users in their decision-making while interacting with a vast amount of information.

The current product recommendation system suffers from two significant drawbacks:

- Recommendation redundancy.
- Unpredictability concerning new items (cold start).

These limitations occur because the legacy recommendation systems rely only on the user's previous buying behavior to recommend new items.

2. UPDATED PROBLEM STATEMENT

Our product recommendation system aims to provide users with more choices in the ways that products are recommended to them by providing multiple models for users to choose from which can help predict their preferred products depending on their needs. We are providing users with models based on content, collaborative filtering and popularity. Moving forward we would like to introduce a merged model of the above options to add more choices for the user. In Future, our models will use reviews from users on certain products to analyze and find a variety of products that the user would find interesting. Using multiple models which can use cosine similarity, Euclidean distance, etc we can uphold a higher measure of effectiveness for our ranking systems by comparing these results to user needs.

The proposed recommendation systems have already solved the problem of cold start by giving generalized recommendations to that user based on the popularity of a particular item by analyzing whatever is more popular among the general public that is more likely to be recommended to new customers.

3. LITERATURE SURVEY

Some reference papers

Reference No: 1.

Title: Study of an E-commerce recommender system based on Big data

Publication: Oxbridge college, kunning university

Author: Xuesong Zhao

This paper discussed the era of the web, in which a lot of information is overloaded over the Internet. Recommender system algorithms are widely used in e-commerce to provide personalized and more accurate recommendations to online users and enhance the sales and user stickiness of e-commerce. This study aims to build a product recommendation system on ecommerce platform according to user needs.

Reference No: 2

Title: Collaborative Filtering for Recommender Systems Publication: 2014 Second International Conference on Advanced Cloud and Big Data

Author: Michael D. Ekstrand, John T. Riedl, and Joseph A. Konstan

The report highlights the discussion of the types of the recommender systems as general and types of CF such as; memory based, model based, and hybrid model. In addition, this report discusses how to choose an appropriate type of CF. The evaluation methods of the CF systems are also provided throughout the paper.

Reference No: 3

Title: Content-Based Filtering: Techniques and Applications Publication: 2017 International Conference on Communication, Control, Computing and Electronics Engineering (ICCCCEE)

Author: Khartoum, Sudan

Content-based recommender systems make recommendations by analyzing the content of textual information and finding regularities in the content. The major difference between CF and content-based recommender systems is that CF only uses the user-item ratings data to make predictions and recommendations, while content-based recommender systems rely on the features of users and items for predictions. Both content-based recommender systems and CF systems have limitations. While CF systems do not explicitly incorporate feature information, content-based systems do not necessarily incorporate the information in preference similarity across individuals.

Reference No: 4
Title: Popularity-Based Recommendation System: International Journal of Engineering and Advanced Technology (IJEAT)
Author: Keshetti Sreekala
The paper highlights how the Popularity based recommendation system works with the current vogue. It explains how it basically uses the items which are in swing at present. Whatever is more popular among the general public that is more likely to be recommended to new customers. The generalized recommendation, not personalized, is based on the count. In this paper, they have used a class which includes the methods to create recommendations and to recommend the item to the user.

Reference No: 5
Title: Using Singular Value Decomposition Approximation for Collaborative Filtering
Author: Sheng Zhang, Weihong Wang, James Ford, Fillia Makedon
Collaborative Filtering analyzes a user preferences database to predict additional products or services in which a user might be interested. The paper highlights the Singular Value Decomposition (SVD), can be used to find a low-dimension model that maximizes the loglikelihood of observed ratings in recommendation systems.

PROPOSED METHODS:

We are currently using a sample dataset for creating our models which we later plan to convert into a more real world focused and practical dataset. Currently we are taking inputs from the user based on what all he has bought or reviewed positively in the past and on that basis we provide the user 3 options of different models to recommend a product.

Sub-Models

- **Popularity-based recommendation system**

  In this product model, we recommend products to the user according to the highest mean average rating of products. In this, we only consider the rating of those users who have rated at least 5(a threshold that can change) different products. It does not suffer from cold start problems.

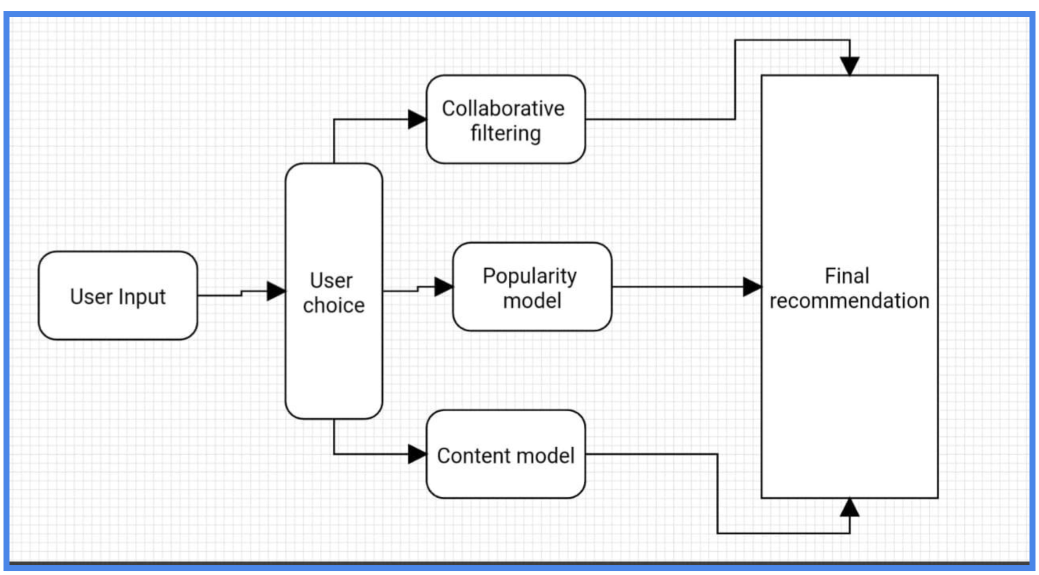
- **Content-based recommendation system**

  In this product model , we recommend products to the user based on Cosine similarity or a euclidean distance of the vector of two products(item).

  We find cosine similarity and euclidean distance with the help of pivot table using sklearn library.

- **Collabrative-based recommendation system**

  In this product model, we have used Singluar value decompostion (SVD) for the user's product prediction and have used MSE (Mean Squared Error) in the evaluation matrix.



4. BASELINE RESULTS

Data Preprocessing

The first step in the project is to preprocess the suitable dataset for the product recommendation system, by dropping the column "Timestamp" and "Unnamed 0 ".

We use the library panda for preprocessing. Further, we preprocess the data according to the different submodels used in our project.

- **OUTPUT**

Cosine similarity table :

```
similarity_score = cosine_similarity(pt)
df2 = pd.DataFrame(similarity_score)
df2
```

	0	1	2	3	4	5	6	7	8	9	...	3081	3082	3083	3084	3085	3086	3087	3088	3089	3090
0	1.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.000000	...	0.000000	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
1	0.0	1.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.000000	...	0.000000	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
2	0.0	0.0	1.0	0.0	1.0	0.0	0.0	0.0	0.0	0.707107	...	0.000000	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	1.0	0.0	0.0	1.0	0.0	1.0	0.707107	...	0.000000	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
4	0.0	0.0	1.0	0.0	1.0	0.0	0.0	0.0	0.0	0.707107	...	0.000000	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...
3086	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.000000	...	0.000000	0.0	0.0	0.0	0.0	1.0	0.0	0.0	0.0	0.0
3087	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.000000	...	0.131306	0.0	0.0	0.0	0.0	0.0	1.0	0.0	0.0	0.0
3088	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.000000	...	0.000000	0.0	0.0	0.0	0.0	0.0	0.0	1.0	0.0	0.0
3089	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.000000	...	0.000000	0.0	0.0	0.0	0.0	0.0	0.0	0.0	1.0	0.0
3090	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.000000	...	0.000000	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	1.0

3091 rows × 3091 columns

Euclidean distance table :

```
similarity_score_2 = euclidean_distances(pt)
df3 = pd.DataFrame(similarity_score_2)
df3
```

Python

	0	1	2	3	4	5	6	7	8	9	...	3081	3082	3083	30
0	0.000000	5.099020	5.099020	5.099020	5.099020	5.099020	5.099020	7.141428	5.099020	7.141428	...	7.681146	3.162278	12.845233	4.1231
1	5.099020	0.000000	7.071068	7.071068	7.071068	7.071068	7.071068	8.660254	7.071068	8.660254	...	9.110434	5.830952	13.747727	6.4031
2	5.099020	7.071068	0.000000	7.071068	0.000000	7.071068	7.071068	8.660254	7.071068	5.000000	...	9.110434	5.830952	13.747727	6.4031
3	5.099020	7.071068	7.071068	0.000000	7.071068	7.071068	0.000000	8.660254	0.000000	5.000000	...	9.110434	5.830952	13.747727	6.4031
4	5.099020	7.071068	0.000000	7.071068	0.000000	7.071068	7.071068	8.660254	7.071068	5.000000	...	9.110434	5.830952	13.747727	6.4031
...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	
3086	5.000000	7.000000	7.000000	7.000000	7.000000	7.000000	7.000000	8.602325	7.000000	8.602325	...	9.055385	5.744563	13.711309	6.3245
3087	1.414214	5.099020	5.099020	5.099020	5.099020	5.099020	5.099020	7.141428	5.099020	7.141428	...	7.549834	3.162278	12.845233	4.1231
3088	11.313708	12.328828	12.328828	12.328828	12.328828	12.328828	12.328828	13.304135	12.328828	13.304135	...	13.601471	11.661904	17.058722	11.9582
3089	5.099020	7.071068	7.071068	7.071068	7.071068	7.071068	7.071068	8.660254	7.071068	8.660254	...	9.110434	5.830952	13.747727	6.4031
3090	8.185353	9.539392	9.539392	9.539392	9.539392	9.539392	9.539392	10.770330	9.539392	10.770330	...	11.135529	8.660254	15.165751	9.0553

3091 rows × 3091 columns

Pivot table results:

Product Name user_index	Acetaminophen	Aspirin	Biotin	Calcium	Coenzyme Q10	Cranberry	Echinacea	Fish Oil	Ginkgo Biloba	Ibuprofen	...	Magnesium	Melator
0	-1.869857e-16	4.500000e+00	-2.588489e-15	9.969959e-17	-1.874741e-16	2.347696e-16	-9.461999e-16	-5.469887e-16	-1.540118e-16	2.038584e-16	...	1.229272e-15	1.099914e-
1	-3.484190e-17	1.943904e-17	3.020849e-16	-6.094348e-16	-5.206554e-16	6.821755e-16	7.628661e-16	-6.498935e-16	2.742625e-30	-9.298505e-16	...	-4.107825e-16	4.700000e+
2	-7.764292e-32	9.233928e-16	3.444360e-16	-1.422525e-14	-2.646000e-31	5.565453e-16	1.820167e-30	2.912695e-30	-2.867609e-29	-1.656212e-30	...	1.582631e-30	5.953792e-
3	-1.592134e-16	5.645436e-16	9.004009e-16	-6.186871e-17	-7.078284e-16	1.833147e-15	-1.012768e-15	1.183982e-15	-3.693519e-16	1.065814e-15	...	8.215650e-16	2.521924e-
4	6.658841e-32	-3.374533e-16	2.265628e-16	7.657663e-17	4.177411e-31	1.768651e-16	1.771292e-31	1.406635e-31	3.036315e-31	-1.077464e-31	...	-3.155022e-31	-9.78832
5	6.227681e-17	-5.964444e-16	-9.436054e-16	8.386655e-18	1.028285e-16	4.300000e+00	2.847500e-16	5.337533e-16	3.118844e-17	-2.073363e-16	...	1.386391e-15	3.269393e-
6	5.508110e-32	-6.837358e-16	5.932745e-16	-1.196131e-15	9.479124e-32	6.375387e-16	-2.471543e-31	2.002171e-30	-2.185864e-30	3.979531e-31	...	7.399638e-31	-8.48472
7	-2.542636e-16	-5.879033e-17	-6.165718e-16	-3.125805e-16	-4.093376e-16	4.341298e-16	-1.768745e-17	-1.187649e-15	-2.617755e-16	-1.371944e-16	...	1.148907e-15	-1.08115
8	-7.682139e-17	-3.275141e-15	4.600000e+00	5.227481e-17	1.231320e-16	4.928511e-16	-4.508932e-16	5.036813e-16	-2.497146e-16	4.852475e-16	...	1.026956e-15	5.153354e-
9	1.253627e-31	-7.369279e-16	-6.481590e-16	-5.291378e-16	1.413095e-31	3.826328e-16	1.332933e-30	-3.533422e-31	-1.034988e-30	-1.076688e-30	...	-5.314569e-31	1.120460e-
10	5.762991e-32	-1.777059e-16	-2.945351e-16	8.162725e-15	2.185750e-31	3.438725e-17	-8.738635e-31	-1.992001e-30	1.647118e-29	7.282641e-31	...	-8.585120e-31	-2.04235
11	-5.652582e-32	2.866759e-16	6.975433e-16	4.509435e-16	-7.340192e-32	-2.786081e-16	-1.094267e-30	4.157543e-31	8.501548e-31	1.016405e-30	...	3.609872e-31	-1.36571
12	4.146054e-16	-1.872698e-15	-1.541576e-16	1.636488e-16	-9.339045e-31	1.400668e-15	4.410232e-16	-2.726011e-17	2.513346e-17	-1.865268e-16	...	6.161738e-16	6.511407e-
13	1.356122e-17	-9.768983e-16	-4.478717e-17	4.334156e-17	-1.235645e-16	4.637585e-16	4.921059e-17	-1.405292e-15	-6.094116e-16	1.555468e-16	...	-1.283695e-15	-1.27964
14	1.439195e-31	-1.766009e-	-1.682754e-	2.880838e-16	1.159864e-	2.786335e-16	1.699236e-	4.958250e-	6.702111e-	-6.944377e-	...	-1.058968e-	-7.79795

Final suggested products to the users

```
In [64]: pred_ratings_user = pred_ratings_user.reset_index()
         pred_ratings_user.rename(columns={'ProductID':'Product ID','U1092':'Predicted Rating'}, inplace=True)
         res=pred_ratings_user.head()

In [67]: res['Product Name'].values

Out[67]: array(['Dell XPS 13', 'Samsung Galaxy S21 Ultra', 'Sony Alpha 7 III',
                'Sony PlayStation 5', 'Sony Alpha a7 III'], dtype=object)
```