Big Data Analytics - Winter'20

SOEN 691-UU

Dr. Tristan Glatard

Implementation of Recommendation System Algorithms for Anime Dataset

Submitted by:

Arsalaan Javed – 40085994

Gunvansh bhatia – 40082036

Manpreet Singh - 40083737

1. Abstract

As a part of analysing big data and extracting information from it, in this project we have created a recommender system for anime dataset with user ratings. In the dataset there are around 12000 animes so, to select an anime will be very difficult for the user. This project has emphasized over analysing user ratings for different animes and tried to recommend animes as per the user profile. After proper analysis and preprocessing of the dataset we have implemented two algorithms of the recommendation system that were discussed in the class. Later, we have analysed the difference between the two approaches and different results that were obtained.

2. Introduction:

2.1 Context

Recommendation systems help users find and select items (e.g., books, movies, restaurants) from the huge number available on the web or in other electronic information sources. Given a large set of items and a description of the user's needs, they present to the user a small set of the items that are well suited to his/her profile. This helps in overcoming the long tail issue by not just recommending the famous items but the most relevant items.

2.2 Objectives:

- Implement big data analytics algorithms, Collaborative Filtering and content based filtering for recommendation systems that were discussed in class.
- Parsing data retrieved from a dataset and applying it over the developed model.
- Evaluating different approaches of recommender systems and analysing them.

2.3 Problem Statement

Using an anime dataset, to create a recommendation system based on two famous algorithms of content based and collaborative filtering recommendation systems. These algorithms are implemented from scratch with other variations to enhance the performance. To implement evaluation metrics that measure the performance

of the system. Also, to recommend most appropriate animes to the user as per his/her profile.

2.4 Related Work

Recent work in recommendation systems includes intelligent aides for filtering and choosing web sites, news, stories, TV listings, and other information. The user of such systems often has diverse, conflicting needs. Differences in personal preferences, social and educational backgrounds, and private or professional interests are pervasive. As a result, we wanted to have a personalized intelligent system that processes, filter, and display available information in a manner that suits everyone using them.

3. Materials and Methods

3.1 Dataset description

For this project, we are using the Kaggle Dataset for Anime, which is available for public use. The database contains two csv files: anime.csv with 7 attributes and rating.csv with 3 attributes

link to dataset: https://www.kaggle.com/CooperUnion/anime-recommendations-database#rating.csv

- anime.csv: This file contains information about anime with 7 attributes: Anime_id, Name, Genre, Type, Episodes, Rating, members.
    - Total 12,294 rows or we can say 12,294 unique anime and their details
    - Missing values in columns Genre, Type and Rating
- Rating.csv with 3 attributes: User_id, Anime_id, rating i.e. the rating given by each user to the anime they have watched
    - Total entries: 7,813,737
    - Total unique users: 73,516
    - Missing values for rating column (value of -1)
    - Ratings are in range from 1 - 10 with global avg rating of 7.80

3.2 Technologies:

Since the dataset is huge, we have implemented the model using:

- PySpark: Spark's RDD has been an aid for processing huge dataset and analysing it.
- Python: For writing basic algorithms using data structures of python.
- Pandas: Pandas dataframe for querying over the data tables.
- Matplotlib: For visualizing the data.

The online ide used is Google Colab for faster processing speed.

3.3 Algorithms

3.3.1 **Content Based Recommendation System**

The main idea revolves around recommending the user with similar products as was rated highly by him/her in the past. The word "content" here refers to the attributes of the product that a user has liked or disliked. To do so the algorithm requires user profile and item profile.

Item Profile: We have taken columns Genre and Type of each anime and the final vector that we will get for each item will be of 1 x 51 dimensions. To calculate the weightage of each content word like "action" we implemented TF x IDF score w.r.t. each document.

User Profile: Will also be a vector of dimension 1 x 51 with the same set of content words. It is calculated by taking the weighted sum of each anime item vector that the user has watched and rated. The weight for each anime is calculated by taking the mean rating of the user and subtracting it with the rating for each anime. This will shift the rating values about zero. So, the animes with a positive rating now are the ones that the user has liked while vice versa for the not liked animes.

A cosine similarity of user profile with all the item profile will provide the best matched anime list and this can be our recommendation list.

3.3.2 **Collaborative Filtering Recommender System**

The main idea here is to find a set N of other users whose ratings are "similar" to user x's ratings. Then estimate user x's ratings based on ratings of users in N. So, we are implementing a user - user based collaborative filtering approach.

This process is mainly divided into 3 tasks:

Task 1: To calculate most similar users and for this we have used Pearson Correlation Coefficient between user x with rest of all users. Once done it will return a list of N most similar users.

$$sim(x,y) = \frac{\sum_{s \in S_{xy}} (r_{xs} - \overline{r_x})(r_{ys} - \overline{r_y})}{\sqrt{\sum_{s \in S_{xy}} (r_{xs} - \overline{r_x})^2} \sqrt{\sum_{s \in S_{xy}} (r_{ys} - \overline{r_y})^2}}$$

Where,

r_xs: rating given by user x to anime s
r_bar: mean rating for user x
sim(x, y): similarity b/w user x and y

Task 2: After getting a list of similar users as user x we can predict the ratings of the anime that this user has not watched but similar users have watched.

$$r_{xi} = \frac{\sum_{y \in N} s_{xy} \cdot r_{yi}}{\sum_{y \in N} s_{xy}}$$

Where,

s_xy = Sim(x, y)
r_yi = rating of user y for anime i

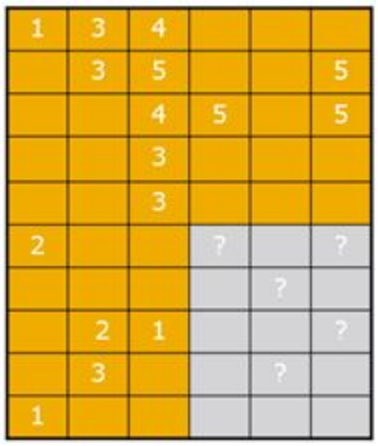
Task 3: Applying common practises i.e. regularization in predicting the rating for the user x to improve the prediction result.

$$r_{xi} = b_{xi} + \frac{\sum_{j \in N(i;x)} s_{ij} \cdot (r_{xj} - b_{xj})}{\sum_{j \in N(i;x)} s_{ij}}$$

Where, μ = overall mean anime rating
b_x = rating deviation of user x = (x_bar) – μ

b_i = rating deviation I $b_{xi} = \mu + b_x + b_i$

4. Results:



This utility matrix has users as rows and anime as columns.

For comparing the performance of both the algorithms we have used the same test data to measure performance. We have taken 995 randomly selected users and 3304 randomly selected anime.

- Content based recommender system: To evaluate this model we have created an evaluation metric with the anime which were rated by the user and comparing with predictions made by the model.

Actual Positive case: Is when the user liked the anime (+ve rating after shifting the rating about zero).

Predicted Positive case: When the cosine value for an anime is positive i.e. threshold = 0.

And vice versa for the negative cases.

True Pos 893	False Pos 415
False Neg	True Pos

509	933

Precision: 68.27217125382263
Recall: 63.69472182596291
Accuracy: 66.4
Recommending movies for user 3:

	anime_id	name	genre	type
122	11771	Kuroko no Basket	comedy, school, shounen, sports	TV
3536	20473	Teekyuu 3	comedy, school, shounen, sports	TV
4261	15125	Teekyuu	comedy, school, shounen, sports	TV
254	18689	Diamond no Ace	comedy, school, shounen, sports	TV
3710	18121	Teekyuu 2	comedy, school, shounen, sports	TV
58	24415	Kuroko no Basket 3rd Season	comedy, school, shounen, sports	TV
100	30230	Diamond no Ace: Second Season	comedy, school, shounen, sports	TV
11038	31422	Minami Kamakura Koukou Joshi Jitensha-bu	school, shounen, sports	TV
3050	32494	Days (TV)	school, shounen, sports	TV
1484	183	Whistle!	school, shounen, sports	TV

- Collaborative Filtering Recommender System: To evaluate this model we have used RMSE score for the same test dataset since we are predicting the rating again.

RMSE without regularisation: 1.7525965512669202
RMSE With regularisation: 1.4800267481453462

Recommending movies for user 3:

anime_id	name	genre	type
1488	Area 88	Action, Adventure, Drama, Military, Romance	OVA
1577	Taiho Shichau zo	Action, Comedy, Police, Seinen	OVA
1901	11-nin Iru!	Action, Adventure, Drama, Mystery, Romance, Sc..	Movie
1951	Manie-Manie: Meikyuu Monogatari	Adventure, Fantasy, Horror, Sci-Fi, Supernatural	Movie
2182	Robot Carnival	Fantasy, Sci-Fi	OVA
2699	Uchuu Kaizoku Mito no Daibouken	Action, Comedy, Sci-Fi	TV
4475	Toshi wo Totta Wani	Drama	Movie
6361	ef: A Tale of Memories. - Prologue	Drama, Music	Special
10278	The iDOLM@STER	Comedy, Drama, Music	TV
20159	Pokemon: The Origin	Action, Adventure, Comedy, Fantasy, Kids	Special

5. Discussion:

The implementation of both the algorithms provided us with a personalized solution for recommending a best matched anime according to the user profile. Both algorithms work well in predicting anime, but both have their own specific features.

Content based recommender system tries to predict the items based on the contents of the user profile so all the items that will be ranked higher for a particular user would be of almost same nature i.e. less diversity. Same thing can be seen in our results while we predicted 10 best matches for user_id 3. To detect a good feature for a user is a difficult task as the best accuracy we have got in our project is 66.4% and if we have a better method of creating the user profile this percentage will go up. This algorithm works very well in finding a similar anime. So, if the problem statement is to find the most similar item then this is the algorithm you must consider. There is not starter issue in the approach as ratings for a user are not required as long as users preferences are known we can make an appropriate recommendation.

The evaluation of this model was a challenge for us in this project, which we finally overcome by creating an evaluation metric as discussed in the result session which focusses over how many items were predicted correctly and which are not.

Collaborative filtering recommender system tries to predict the items based on the similar users. This approach gives a diverse result which is more suitable as users in general like different types to items. Although, there will be a starter problem i.e. in case where the user is new to the system and has no prior ratings or less. The accuracy achieved in this project is 1.48 RMSE as here we are predicting the ratings that the user would have given to the anime and the best result would be returned.

Future Possible Ideas:

CBRS: Creating a user profile with gradient descent type approach, as the more appropriate a user profile is the better would be the result.

Collaborative Filtering: Modeling Item- Item based CF

References:

- Class Notes : Recommender Systems: Content-based Systems & Collaborative Filtering (http://www.mmds.org)
- https://www.kaggle.com/CooperUnion/anime-recommendations-database#rating.csv
- https://www.analyticsvidhya.com/blog/2015/08/beginners-guide-learn-content-based-recommender-systems/
- https://medium.com/towards-artificial-intelligence/content-based-recommender-system-4db1b3de03e7