Anime Recommendation System

- By Project Team ID 18

1. Akshay Dhabale
2. Axel Dzeukou
3. Mrinal Rai
4. Yogesh Yadav

Research Questions

- Build an anime recommender system that provides personalized anime recommendation to its user based on explicit ratings
  - KNN item-item recommendation using cosine similarity and pearson coefficient distance
  - ALS + bias recommendation using latent factor
- Study the models and draw comparisons between them
- Evaluate the performance of each model using RMSE

Dataset

Source

- https://www.kaggle.com/datasets/hernan4444/anime-recommendation-database-2020

Details

- Input files - anime.csv and rating_complete.csv file.
- Anime.csv file contains useful information such as anime id, name of the anime and genre etc.
- Rating_complete.csv file contains useful information such as user id, anime id and rating by the users for animes.
- Rating data includes
  - 57M records
  - Distinct 31K users
  - Distinct 16K animes.
- Ratings Data statistics
  - Ratings columns range (1, 10)
  - Mean - 7.51
  - Standard Deviation - 1.69
- Sparsity of utility matrix is ~ 98.90%
- In our dataset around 10% of the animes are more famous and has more number of ratings.

Model Design

Alternating Least Squares (ALS)

- Utilized spark.ml library implementation of ALS.
  - Used Latent factor based collaborative filtering.
- Used explicit ratings from users on anime, on a rating scale from 1 to 10 and made extrapolations from data to predict user ratings.
- Hyperparameter-tuning and cross-validation - to find best model parameters
  - ParamGridBuilder - to define tuning parameters
  - RegressionEvaluator - to RMSE calculation
  - CrossValidator - to compute average evaluation metric
- Added bias to ALS to make predictions based on user-item interaction

K-Nearest Neighbour (KNN)

- Spark.ml does not include library for K-Nearest Neighbour implementation like ALS.
- Goal is to develop KNN completely in Pyspark
- In KNN, focus is to implement item-item based collaborative filtering using cosine and pearson coefficient distance.

Data Preparation

Ratings Data

- Checked for count of Null, None, NaN for all columns
- Checked for duplicate user_id-anime_id pairs
- Random Train/Test split (0.8,0.2)

ALS

- Read user id, anime id and ratings column into dataframe from ratings_complete.csv for model training
- Calculated user-item interaction using the formula rating-(user mean + item mean - global average)
- Computed the predicted rating with the formula user_item_interaction + user_mean + item_mean - global_average

KNN

- Input data : Item-User Ratings Sparse rdd
- Input to model : Intermediate rdd which store all Item-Item combination to calculate similarity distance
- For Pearson coefficient distance measure, user rating deviations is calculated using user mean ratings and actual ratings for generating the intermediate rdd of item-item pairs.

Model Implementation

ALS

- Obtained the best model (Model with the lowest RMSE) using Hyper-parameter tuning
- Fit the model using random train/test split and evaluated predictions on test dataset
- Added bias to ALS by calculating user-item interaction
- Computed predicted ratings using test results
- Used ALS build-in function to generate the top 5 animes for each user.
- Converted the recommendations into interpretable format by joining rating_complete.csv with anime.csv

KNN

- Model calculates cosine similarity, pearson coefficient similarity for each item-item combination
- Model then calculates cosine and pearson coefficient distance using ratings and similarity measure for each item-item combination
- Model recommends Top 5 nearest anime details for an input anime using similarity distance
- Model recommends Top 5 anime details for an input user using similarity distance

Model Evaluation

Root Mean Square Error

- In ALS implementation, RMSE of 1.17 without bias and RMSE of 1.16 with bias was achieved.
- In KNN implementation, RMSE of 2.09 with cosine distance and RMSE of 2.27 with pearson coefficient distance was achieved

Comparison of Models

- Performance : ALS pyspark model converge faster than KNN model.
- Popularity : ALS helped in removal of popularity bias
- Scalability : ALS did not face any issue with scalability unlike KNN
- First Rater : KNN cannot recommend an item that has not been previously rated unlike ALS

Model Recommendations

User - 68042 rated a total of 4587 animes
Overall Avg Rating by user - 68042 is 5.892343984559297
User - 68042 rated sample animes details below

anime_id	Name	Genres
11593	Ganbare! Bokura no Hit and Run	School, Sports
33957	Danball Senki Wars: All Star Battle	Action, Kids, Mecha, School
19921	Ogami Matsugorou	Action, Martial Arts, Romance, School, Shounen
38199	Bermuda Triangle: Colorful Pastrale	Music, Fantasy
17219	Sore Ike! Anpanman: Roll to Laura Ukigumojou no Himitsu	Kids, Fantasy, Comedy

Top N Recommended anime similar to input user - 68042 is shown below

anime_id	Name	Genres	Name	Genres
11245	Manga Nihonshi	Historical	Nogsaegjeoncha Hamos	Action, Adventure, Fantasy, Shounen
22975	Kaibutsu-kun: Demon no Ken	Comedy, Horror, Kids, Shounen	Guitar Shoujo!	Slice of Life
5477	Gozonji! Gekkou Kamen-kun	Parody, Comedy, Sci-Fi	Kamen Rider Den-O: Imagin Anime 3	Action, Adventure, Comedy, Kids, Super Power
35401	Hyaku-nengo no Aru Hi	Military, Sci-Fi, Supernatural	Shitcom	Comedy, Romance
15141	Yukiwatari	Fantasy	Phantasm	Dementia, Music

KNN - Cosine Similarity

ALS

Discussion / Q&A

Thank You