DMG Assignment 2

Aryan Behal 2019026

Kesar Shrivastava 2019051

QUESTION 1

We have performed the following analysis:

- **Nan Value per column for each table:** There were no Nan values in the dataset except links.csv and ratings.csv. Total nan values found is 8+1 = 9.
- **Rating frequency:** Most movies have ratings of 3.0 and 4.0, whereas very few users have rated movies in lower ranges.
- **Top and least 25 frequently used Tags:** 'In Netflix queue' was the most used tag used more than 120 times whereas the least used tags are used not more than a single time.
- **Top/Bottom 25 reviewed movies:** From the plots above, we can infer that the film 'The Age of Innocence (1993)' has been examined by most users. Not more than one user has reviewed the least reviewed movies.
- **Correlation heatmaps for links.csv and ratings.csv:** There is a high correlation between the ids of movies in the two databases. This shows that the film was entered into the databases at similar times, and both used a similar indexing algorithm for new entries of movies.
- **Mean ratings and top/bottom 30 movies (average rating score-wise)**
- **Mean rating for tags and top/bottom 30 tags (avg rating wise):** People loved movies that showed classical stuff, American natives, and buildings. Whereas there was not so much love shown for the eastern flavor of the film. Films based on robbery, cheating, war was not liked a lot
- **movieID vs. time, Ratings vs. Time**
- **Rating for different tagged movies (dark comedy vs. funny):** People have liked dark humor more than comic movies.
- **Votes for each movie:** There are many movies below the movie id 25000 that any user has not reviewed.

INSIGHTS ABOUT THE DATASET

- The dataset links.csv contains the ids of the movies from IMDb and TMDb. As seen from the correlation plot, both these columns are highly correlated with each other. This can be attributed to the fact that both databases add movies to their records simultaneously.
- As seen from the number of movies reviewed by a user and the number of users that have studied a particular film, we can infer that the ratings.csv dataset is sparse.
- There are very few Nan values that save some time in preprocessing. Ratings.csv is used to create the recommendation system,'. And the only Nan value that appears is handled.
- The number of movies that were rated became exponentially higher as time passed.
- The rating system evolved. Initially, ratings were given in jumps of 1 (1, 2, 3, 4, 5), but later, ratings were given as (0.5, 1, 1.5, 2.0, …)

QUESTION 2

DATA PREPROCESSING

- The movies.csv consist of movie ids along with their titles and genres. This file is used to get the movie name corresponding to its id throughout the recommendation system.
- Since traversing the complete data frame of movies.csv to get the id of the required movie and its title, we have created two lists: one of the movie titles and the other of their ids correspondingly.
- The ratings.csv has the information about which user rated which movies.
- This dataset is used to retrieve the transactions of all the users.
- To use this dataset in the apriori algorithm, we use the 'encode_units' function. We define 2.5 to be the threshold. The ratings equal to or below 2.5 are taken to be 0. Others are assumed to be 1.
- Thus, we get the final dataset. We use association rule mining over this dataset to get the recommendations.

ASSUMPTIONS

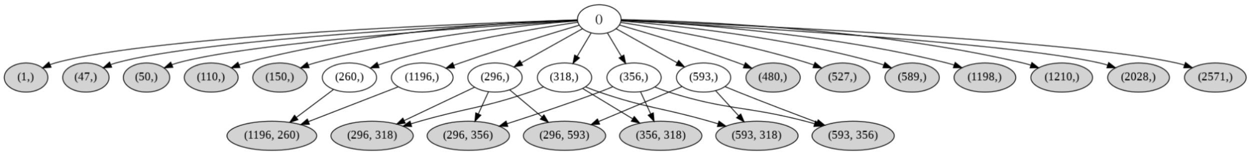
- The movies that a particular user has rated are the only movies that they have watched. Thus, this list corresponding to a user acts as its transaction.
- The movies that have a rating of 2.5 or below it are not worth recommending.
- The metric for the apriori algorithm is ‘lift.’
- The system considers itemsets with support values below 0.075 too infrequent, so they are not recommended.
- If the input consists of only one movie and is not present in the antecedents of any rule, the system recommends the random film.
- If the input consists of more than one movie, but the rules do not have the complete set of movies as an antecedent, the system picks a single film from the collection, finds its recommendations, and lists out the movie with the greatest ‘lift.’
- Even after doing all this the recommendations are less than four we go for random recommendations

QUESTION 3

ASSUMPTIONS

- Since we were required to visualize only a subset of the ‘maximal frequent itemset’, we have changed to min support = 0.3 from 0.075, so that the final set received has lesser itemsets.
- We have found and printed the maximal frequent itemset using the fp-growth algorithm just for the sake of tallying it
- For visualization, we have used the “frequent itemsets” from the apriori algorithm (as used in Q2) and have processed it to find the maximal frequent itemset.
- For the representation of the graph, we have used Digraph from Graphviz.

VISUALIZATION



- The dark nodes are the maximal frequent itemsets whereas the lighter nodes are only frequent itemsets.
- The numerical values in the nodes are the movie Ids.
- These are results

LEARNINGS

- We learned about association rule mining and apriori algorithm.
- We also learned how we should tune different parameters in apriori to get some desired results.
- We also got to know about different types of filtering like collaborative filtering and content-based filtering.
- The EDA taught us how to understand the data before starting to work with it.
- We got insights about Graphviz. It helped for the visual representation of the directed graph.

REFERENCES

1. https://www.analyticsvidhya.com/blog/2020/11/create-your-own-movie-movie-recommendation-system/
2. https://towardsdatascience.com/association-rules-2-aa9a77241654
3. https://www.kaggle.com/kerneler/starter-movie-lens-small-latest-bcd9621b-c
4. https://nb.recohut.com/movie/collaborative/2021/06/23/collaborative-filtering-mov

ielens-latest-small-01.html

5. https://github.com/WJMatthew/MovieLens-EDA
6. https://www.kaggle.com/akhilram7/affinity-analysis-of-market-basket
7. https://www.kaggle.com/ahm6644/movies-recommendations-by-association-rules