Movie Recommendation System Using HIVE

Soumitra Siddharth Johri

Abstract

A fundamental data-mining problem is to examine data for similar items. We can phrase the problem of similarity as one of finding sets with a relatively large intersection.We do collaborative filtering, a process whereby we recommend to users items that were liked by other users who have exhibited similar tastes. For a given customer and his previous rating history we suggest movies and his possible ratings by finding users whoes rating profile is "similar" to the given user. We focus is on a particular notion of similarity: the similarity of sets by looking at the relative size of their intersection. This notion of similarity is called Jaccard similarity. We have used HIVE for data operations and present visualizations created using Tableau Software.

Netflix Dataset: Netfix provided a training data set of 100,480,507 ratings that 480,189 users gave to 17,770 movies. The user and movie fields are integer IDs, while grades are from 1 to 5 (integral) stars. The data set consists of two tables movie titles and movie ratings. following is the schema of the data set :

- movie titles(mid:integer, yearOfRelease:integer, title:varchar)
- movie ratings(mid:integer, custormer id:integer, date:varchar, rating: integer)

The Jaccard index, also known as the Jaccard similarity coefficient, is a statistic used for comparing the similarity and diversity of sample sets. The Jaccard coefficient measures similarity between finite sample sets, and is defined as the size of the intersection divided by the size of the union of the sample sets:

$J(A,B) = \frac{|A \cap B|}{|A \cup B|}$.

Following is the stepwise process (data pipeline) that we have employed to find movie recommendations for a given user. In our implementation we have taken user with userid '33' as our customer for whom movie recommendations are done. The corresponding hive queries are also given in each step :

1. Reduce dataset to consider only those ratings where the movie has been rated greater than 3.
2. Fetch the list of movies which the input customer (customer with customer id 33 in out case)has watched previously
3. Get aggregated list of all customers who have rated the movie which are also previously rated by the user with user id '33'
4. get the total count of movies rated by each customer
5. calculate the Jaccard variables , M_{11} , M_{01} and M_{10}
6. calculate the Jaccard coefficient J and Jaccard Distance JD by the following formulae :
   $J = \frac{M_{11}}{M_{01}+M_{10}+M_{11}}$.
   $JD = \frac{M_{01}+M_{10}}{M_{01}+M_{10}+M_{11}}$.
7. sort the result set in increasing order of Jaccard coefficient and take the top 5 customers which are most similar to the given customer with customer id 33.
8. output the movies that that are rated by these top 5 critics but not yet rated by customer with id 33 as recommended movies.

The HIVE queries for the recommendation system is given in the presentation and the attached source file.

Visualization

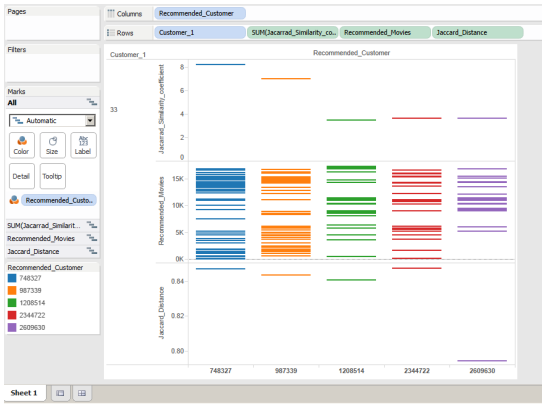

Figure 1: Recommended movies,similar users for customer 33

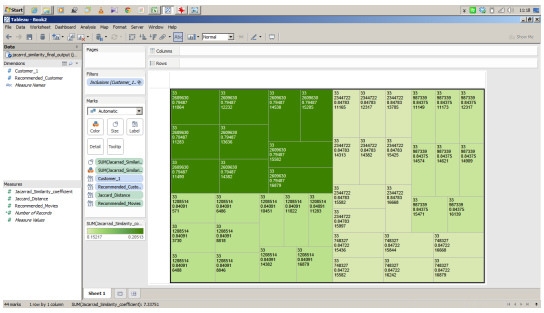

Figure 2: Recommended movies,similar users for customer 33

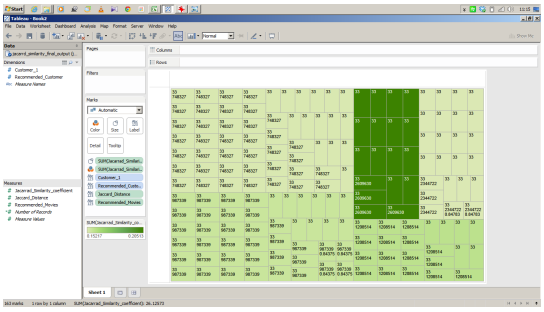

Figure 3: Recommended movies,similar users for customer 33

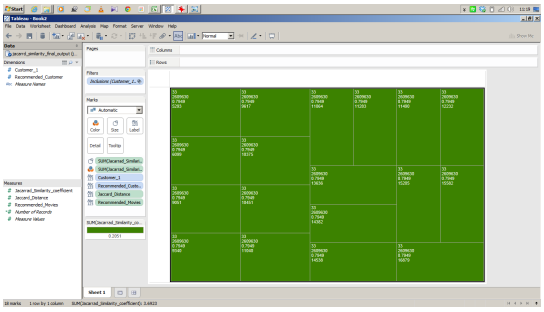

Figure 4: Recommended movies for customer 33

Other Visualizations on the data set.

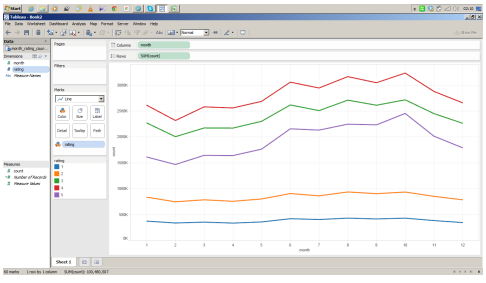

Figure 5: Trends : Movie rated per month

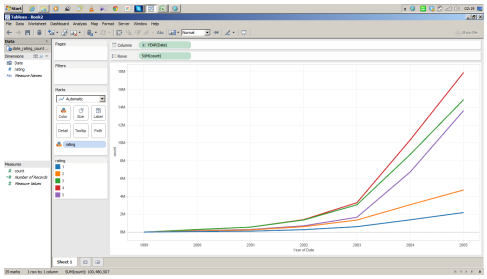

Figure 6: Trends : Movie ratings per year

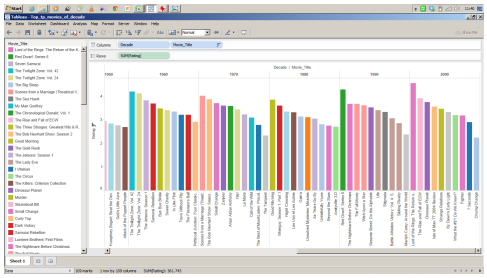

Figure 7: Trends : Top 10 Movies per decade

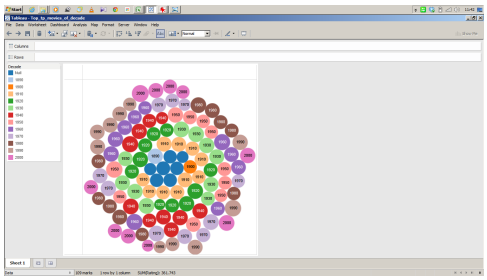

Figure 8: Detail : Top Movie per decade

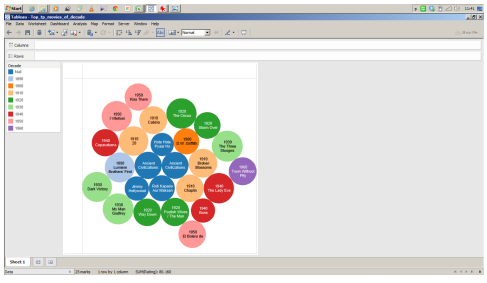

Figure 9: Detail : Top movies per decade