**CSE523 - Machine Learning**

**Movie Recommendation System using Machine Learning**

**Faculty - Prof. Mehul Raval**

**Weekly Report 7**

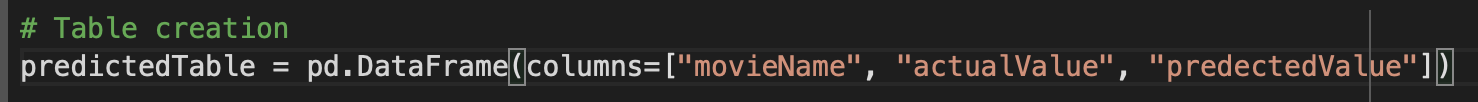
**Submission Date: 01-04-23**

**Group: Tech Titans**

| **Name** | **Roll Number** | **Email** |
| --- | --- | --- |
| Jainam Shah | AU2040186 | jainam.s4@ahduni.edu.in |
| Yash Chotaliya | AU2040193 | yashkumar.c@ahduni.edu.in |
| Akshay Parmar | AU2040199 | akshay.p@ahduni.edu.in |
| Shubham Bhatt | AU2040206 | shubham.b1@ahduni.edu.in |

In the last week, the data was pre-processed by only keeping movies that received more than 100 ratings and then splitting the data into 80% for training and 20% for testing. Then the larger dataset was used to train the models to learn the underlying patterns and relationships in the data, which can result in a more accurate and generalizable model. Then, after performing matrix normalization, Pearson correlation, and cosine similarity, a specific user's data were normalized and used to predict values using a model previously tested on a smaller dataset. The results showed that the model predicted a value of 0.031 for the movie "A Nightmare on Elm Street," normalized to -0.5, indicating that the model was 50-60% correct. Further testing of the model on all users and their respective movies watched is necessary to assess its performance fully.

After that, we created a table to store actual and predicted values for particular user movie ratings. First, we checked for the movie "A Nightmare on Elm Street" in the ranked\_item\_score data frame and then printed its corresponding movie score.



Next, we created a table named predictedTable with columns "movieName,” "actualValue", and "predectedValue".

Here in the table, after it is written code, we get that it loops through each movie in the ranked\_item\_score data frame, and for each movie, it checks if the movie exists in the test\_picked\_userid\_watched data frame. If the movie exists, it appends a new row to the predictedTable with the movie name, actual rating value from the test dataset, and predicted rating value from the ranked\_item\_score data frame.

So after doing the above, below is the output that we get for the predictedtable

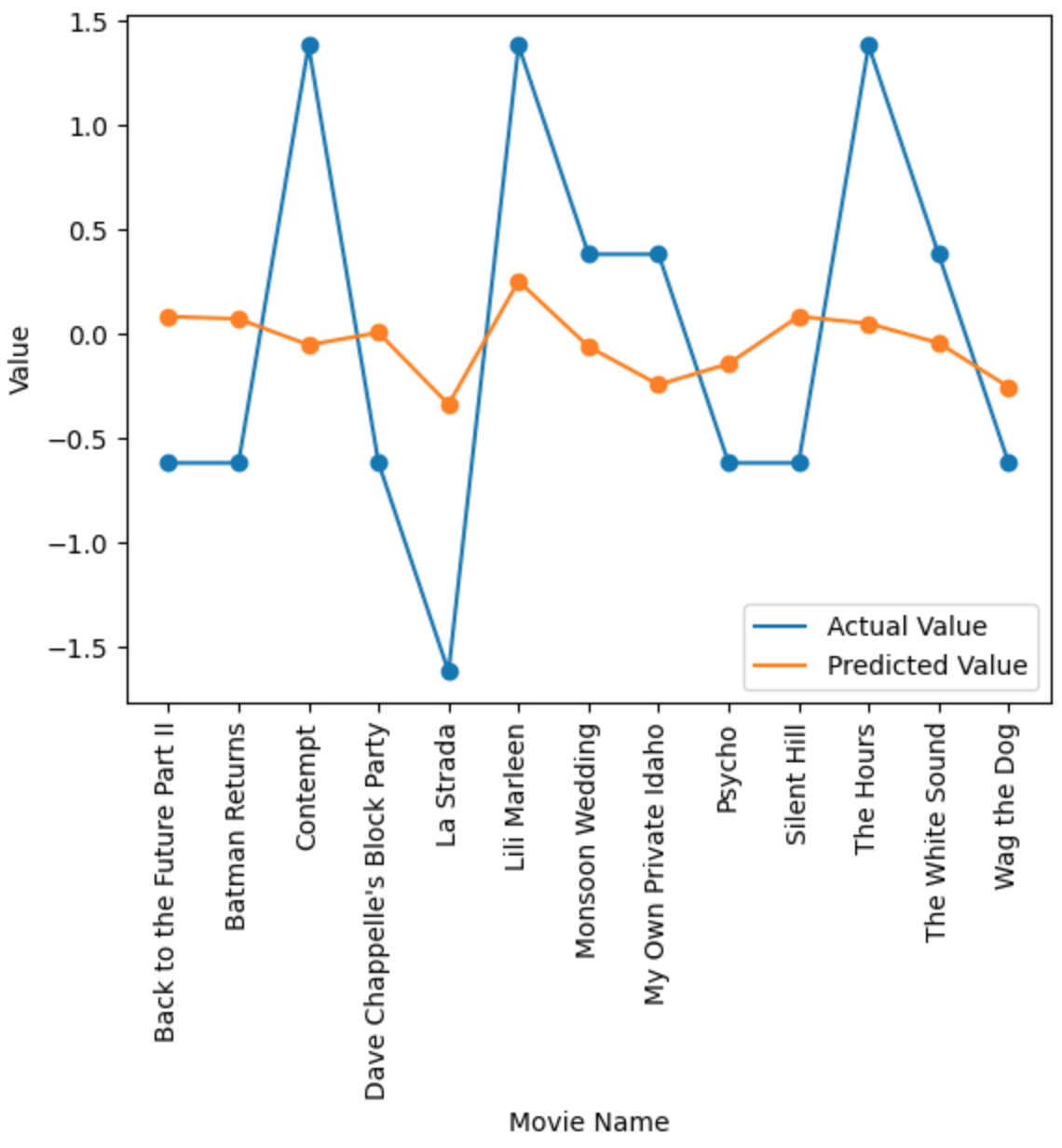


So here, we are creating a table that shows the actual and predicted ratings for a specific user's movie ratings, with the predicted values generated from the model, created earlier.

The above thing is done because this table can be helpful for evaluating the performance of the collaborative filtering model, as it compares the actual ratings given by the user to the predicted ratings generated by the model. By analyzing the differences between the actual and predicted values, we can visually understand how accurate the model is and identify areas where it may need improvement.

Also, after creating the table, we had a scatter plot and a line chart. The scatter plot displays two sets of points, one for the actual values and one for the predicted values, plotted against the movie names on the x-axis. The line chart displays the same information, with the actual and predicted values plotted as lines against the movie names on the x-axis.

Below is the scatter plot and a line chart that we got.



The above scatter plot and line chart we had made because scatter plot and line chart are helpful in visualizing and comparing the actual and predicted values for each movie. The scatter plot shows the actual and predicted values as separate points on the graph, while the line chart connects the points for each movie, making it easier to see the difference between the actual and predicted values.

By comparing the actual and predicted values visually, we can quickly identify any trends or patterns in the data, such as movies with higher or lower prediction accuracy. This can be useful in evaluating the performance of the recommendation algorithm and identifying areas for improvement.

Additionally, the scatter plot and line chart can be used to communicate the results of the recommendation system to stakeholders in a clear and concise manner.