Recommending Movie Systems

Phong Nguyen, Ai Nguyen, Andrew Carranza

UNIVERSITY OF HOUSTON

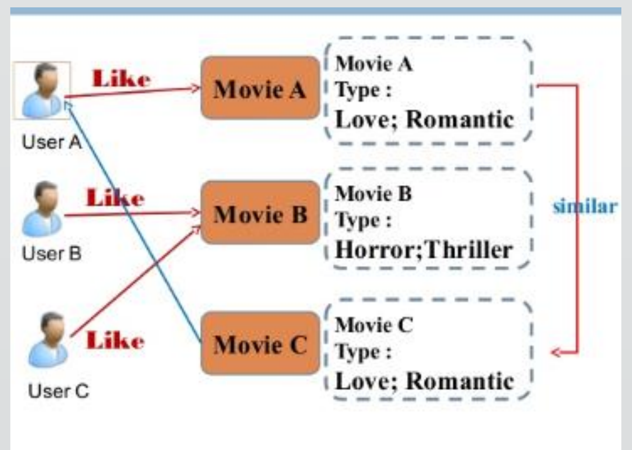
College of Natural Sciences & Mathematics, University of Houston

Abstract

- Movies can be watched at home or in the movies, of all genres, ages, languages, and to meet every audience's requirement.
- This project will provide a system that uses data and a filter to help audiences find movies that match their personal preferences.
- The project is constructed using data source from TMDB 5000 Movie Dataset, combined with models from Machine Learning to give appropriate results. We apply IMDb weighted rating (WR) to calculate the rating for each movie.
- We also included Neural Network in the project to predict the recommendations by training the model on the training movie dataset. The goal of the project is divided into two parts.
- For the first part, team needs to achieve the goal of accurately assessing the ratings of the movies in the dataset.
- For the second part, based on the audience's request, team will give recommendations about the most suitable movie.

Background

	title	vote_count	vote_average	score
1881	The Shawshank Redemption	8205	8.5	8.059258
662	Fight Club	9413	8.3	7.939256
65	The Dark Knight	12002	8.2	7.920020
3232	Pulp Fiction	8428	8.3	7.904645
96	Inception	13752	8.1	7.863239
3337	The Godfather	5893	8.4	7.851236
95	Interstellar	10867	8.1	7.809479
809	Forrest Gump	7927	8.2	7.803188
329	The Lord of the Rings: The Return of the King	8064	8.1	7.727243
1990	The Empire Strikes Back	5879	8.2	7.697884

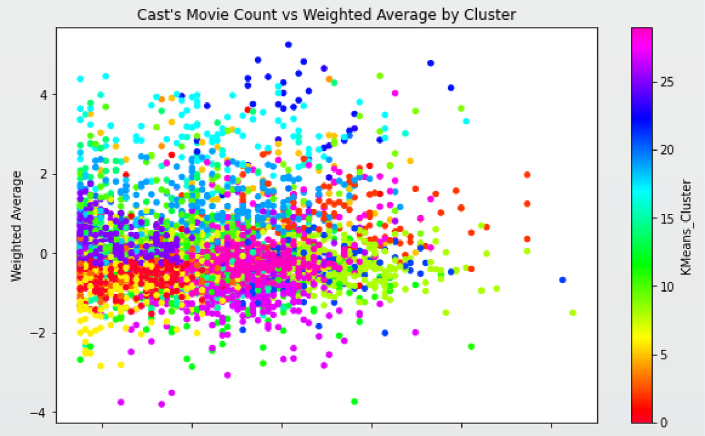


- It is necessary to have a system of filtering, allocating, and prioritizing information to help users avoid overloading when dealing with excessive amounts of data.
- The movie recommendation system will solve the above problem by reviewing a huge amount of information to provide users with personalized services and content.
- A fantastic more complex example of this would be the streaming service YouTube. Which has a great algorithm that helps users stay in a pool of related information. Where they can receive and expand on similar information on specific topics.
- The dataset that we use has information about movie title, budget, genre, release date, revenue, vote count etc.
- Moreover, research and development in this type of field can have a major impact on the way people are able to shape and receive their entertainment and information to suit their own preferences based on the rating of the content and the related genres

Methods

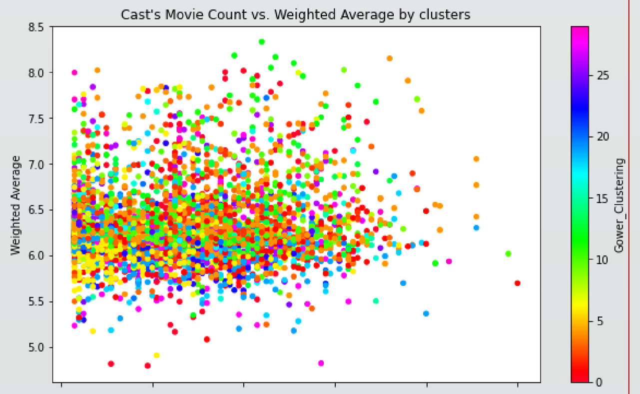
- Model 1 Kmeans:

The Kmeans, uses the Euclidean distance between instances to calculate their similarities. So, when clustering, the difference between the budgets of movies will have similar impact with the difference between their average ratings. Then we calculated the silhouette score for our Model. We used a plot to display our clustering results of our data.



- Model 2 Gower clustering:

The Gower Distance is a distance measure that can be used to calculate distance between two entity whose attribute has a mixed of categorical and numerical values. We used a plot to display our clustering results of our data.



- Model 3 Cosine similarity:

We chose to implement the Cosine similarity algorithm for our 3rd machine learning model. We aim to discover how we can find movies related with their categorical attributes. We aim to measure how similar movies. Since that unlike the distance models above. the Cosine similarity measure suggests that the points between our distances are more similar to each other.

- Model 4 CNN model:

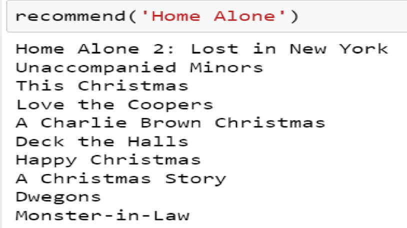
Our implementation was to compute the cosine similarity of the given movie and the whole movies' feature matrix.

We plan on doing this by Returning the top k max similarity values and then by selecting randomly to make sure each recommendation id distinct.

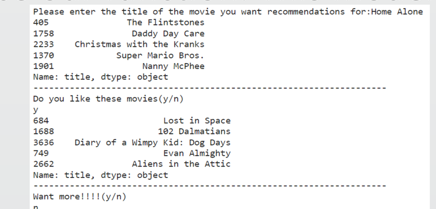
Results

- K Means clustering scatter plot: Displays the cast movie count vs the weighted average by cluster. This model performed the worst. Home Alone is a family/comedy movie, whereas our model recommended a horror movie

- Gower clustering scatter plot displays and, attempts to visualize suffer even more from high dimensionality. The clusters are not separable by two features. This model performed okay. Most of the movies are comedy / family movies.



- Cosine similarity classification: Because our similarity function calculates the similarities of the soup strings, we are not able to measure or visualize the performance our function. This model performed the best.



- The CNN model didn't perform as well as the machine model Cosine similarity. The results seemed to be random and unrelated to each other when searching for similar genres this could be the result of how we trained the data in our CNN model.

Conclusion

In conclusion, we found that the results that we were able to gather from our experiment. Indicate to us that the machine learning Model 3: cosine similarity performed better. Then our other machine learning models along in comparison to our CNN implementation.

Future Direction

However, our results could have been impacted by how well we implemented our training data for our CNN model. And we suspect that it could have performed better than the results we received.

Acknowledgments

We found these resources helpful in guiding us in understanding recommendation systems better

Kniazieva, Y. (2022, April 14). Guide to movie recommendation systems using machine learning. High quality data annotation for Machine Learning.