University of Asia Pacific

Team B1-G3

Md Shamaun Nabi (18201050)
Md. Musfiqur Rahman (18201054)
Al Amin (18201063)

Project Title

The title of our project is “**Movie Recommendation Engine**”.

INTRODUCTION

This data set contains information about movies. Which is collected from The Movie Database (TMDb), including user ratings and revenue.

Dataset

In this project, we are using The **TMDb** Dataset. We have collected it from Kaggle.. Our dataset is in two parts of a csv file. They are -

- tmdb_5000_credits.csv
- tmdb_5000_movies.csv

There are almost 5000 movies in our dataset in which we will train our model.

Data Source Summary

We (Kaggle) have removed the original version of this dataset per a DMCA takedown request from IMDB. In order to minimize the impact, we're replacing it with a similar set of films and data fields from The Movie Database (TMDb) in accordance with their terms of use. The bad news is that kernels built on the old dataset will most likely no longer work.

The good news is that:

- We can port our existing kernels over with a bit of editing. This kernel offers functions and examples for doing so. We can also find a general introduction to the new format here.

- The new dataset contains full credits for both the cast and the crew, rather than just the first three actors.

- Actors and actresses are now listed in the order they appear in the credits. It's unclear what ordering the original dataset used; for the movies I spot checked it didn't line up with either the credits order or IMDB's stars order.

- The revenues appear to be more current. For example, IMDB's figures for Avatar seem to be from 2010 and understate the film's global revenues by over $2 billion.

- Some of the movies that we weren't able to port over (a couple of hundred) were just bad entries. For example, this IMDB entry has basically no accurate information at all. It lists Star Wars Episode VII as a documentary.

Data Source Details

Several of the new columns contain json. We can save a bit of time by porting the load data functions [from this kernel]().
Even in simple fields like runtime may not be consistent across versions. For example, the previous dataset shows the duration for Avatar's extended cut while TMDB shows the time for the original version.
There's now a separate file containing the full credits for both the cast and crew.
All fields are filled out by users so don't expect them to agree on keywords, genres, ratings, or the like.
Our existing kernels will continue to render normally until they are re-run.
If you are curious about how this dataset was prepared, the code to access TMDb's API is posted here.

All columns:

```
Data columns (total 29 columns):
 #   Column                Non-Null Count  Dtype
---  ------                --------------  -----
 0   budget                4845 non-null   int64
 1   genres                4845 non-null   object
 2   homepage              1719 non-null   object
 3   id                    4845 non-null   int64
 4   keywords              4845 non-null   object
 5   original_language     4845 non-null   object
 6   original_title        4845 non-null   object
 7   overview              4842 non-null   object
 8   popularity            4845 non-null   float64
 9   production_companies  4845 non-null   object
 10  production_countries  4845 non-null   object
 11  release_date          4844 non-null   object
 12  revenue               4845 non-null   int64
 13  runtime               4843 non-null   float64
 14  spoken_languages      4845 non-null   object
 15  status                4845 non-null   object
 16  tagline               4001 non-null   object
 17  title                 4845 non-null   object
 18  vote_average          4845 non-null   float64
 19  vote_count            4845 non-null   int64
 20  movie_id_x            4845 non-null   int64
 21  cast_x                4845 non-null   object
 22  crew_x                4845 non-null   object
 23  movie_id_y            4845 non-null   int64
 24  cast_y                4845 non-null   object
 25  crew_y                4845 non-null   object
 26  movie_id              4845 non-null   int64
 27  cast                  4845 non-null   object
 28  crew                  4845 non-null   object
```

Selected columns:

```
Data columns (total 7 columns):
 #   Column    Non-Null Count  Dtype
---  ------    --------------  -----
 0   movie_id  4845 non-null   int64
 1   title     4845 non-null   object
 2   overview  4842 non-null   object
 3   genres    4845 non-null   object
 4   keywords  4845 non-null   object
 5   cast      4845 non-null   object
 6   crew      4845 non-null   object
```

Lost columns:

- actor1facebook_likes
- actor2facebook_likes
- actor3facebook_likes
- aspect_ratio
- casttotalfacebook_likes
- color
- content_rating
- directorfacebooklikes
- facenumberinposter
- moviefacebooklikes
- movieimdblink
- numcriticfor_reviews
- numuserfor_reviews

Algorithm

We have used K-Nearest Neighbor(KNN) for our dataset. KNN makes inference about a movie, KNN will calculate the “distance” between the target movie and every other movie in its database, then it ranks its distances and returns the top K nearest neighbor movies as the most similar movie recommendations.

Conclusion

This dataset was generated from The Movie Database API. This product uses the TMDb API but is not endorsed or certified by TMDb.Their API also provides access to data on many additional movies, actors and actresses, crew members, and TV shows.