# Steps to execute the Hybrid Recommender System.

Step 0: Downloading required Libraries and Datasets:

Text

Description automatically generated

The required Libraries are mentioned above, to install them I would recommend creating a conda virtual environment. This can be done by:

1. Open a Terminal/shell command window (Mac or Windows)
2. Check conda version by running:

conda -v

If it needs to be updated you can run:

conda update conda

1. Once conda is updated we can run the command to create a virtual environment:
   1. conda create -n “your\_project\_name” pip python = 3.9
   2. conda activate “your\_project\_name”

Once conda is setup and activated you can begin installing the libraries by running the following commands sequentially:

1. conda install pandas
2. conda install numpy
3. conda install -c conda-forge regex
4. conda install ­-c conda-forge matplotlib
5. conda install seaborn
6. conda install sklearn
7. conda install –ignore-installed –upgrade tensorflow==2.5.0

This will provide you with all the libraries needed.

The dataset is freely available on Kaggle or the article hosting the dataset. MovieLens has a webpage that you can download the dataset from and can be traced by the link mentioned below:

<https://grouplens.org/datasets/movielens/latest/>

The alternate download location is attached as a link below, this is a direct download zip folder.

<https://files.grouplens.org/datasets/movielens/ml-latest.zip>

* Troubleshooting

In case of library installation issues which can be of type:

1. Wheel creation failure for tensorflow.
2. Dependency un installable.
3. Error during downloading.

You can recreate the conda virtual env setup and make sure it is activated before package installation. Else, you could try installing the libraries by the commands mentioned below:

1. conda config –env –add channels conda-forge
2. conda install pandas = 0.13.1
3. conda install numpy
4. conda install -c conda-forge/label/gcc7 regex
5. conda install -c conda-forge/label/testing/gcc7 matplotlib
6. conda install -c anaconda seaborn
7. conda update -f -c conda-forge tensorflow

Step 1: Loading the Dataset:

The Dataset can easily be loaded by running the cell as shown below:

Text

Description automatically generated

Here, the path where the dataset is downloaded and stored needs to be replaced as per your directory layout.

"../../../../DataSets/ml-latest-small/movies.csv”

Once the dataset is loaded we are ready to visualise the dataset.

Step 2: Visualising the Dataset:

The dataset can be converted as a Dataframe using the pandas library and can be visualised as below on running the next cell.

Text

Description automatically generated

Step 3: Visualising the spread and Variation of the dataset:

To understand the dataset spread and how it is mapped out, we can use seaborn heatmaps to visualise the data, after running each cell this is the how the dataset can be visualised.

Chart

Description automatically generated

Chart, bar chart

Description automatically generated

Step 4: Pivoting the dataset:

One of the major parts of the data preparation is to Pivot the ratings dataframe and merge it with the movies dataframe. For a large data set this can be a problem and a potential failing point. The cell of pivoting the table is shown below:

A picture containing text

Description automatically generated

If this cell fails to execute troubleshooting can be carried out.

* Troubleshooting

Troubleshooting pivoting cell can be done if:

1. Size of the dataset is too large.
2. Chunk Size overflow.
3. No response from kernel.
4. Kernel Failure.

The solution is simple and efficient, add another cell below this code and insert the following commands:

tmp = dtf\_users.copy()

chunker = pd.read\_csv(tmp,sep=',', chunksize=1)

tot = pd.DataFrame()

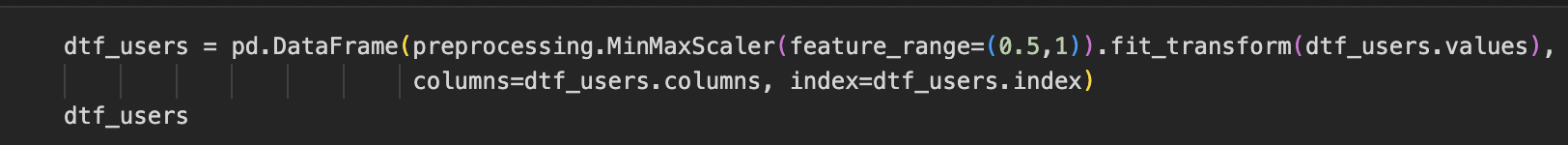
for piece in chunker:

tot = piece.pivot('r\_id', 'g\_id', 'exp')

dtf\_users = tot

This code is already commented in a block below, you just need to uncomment it (command+/) and run this cell.

Step 5: Pre-processing Execution.



The pre-processing codes can be run one after the other without delay. We use min-max scalar to scale the data to avoid overfitting or outliars.

Step 6: Output Parameters:

Finally, once all the cells are executed the results can be observed.

Graphical user interface, text, application

Description automatically generated

These parameters are:

1. y\_test: This shows the actual movie id for the user that he/she would like to watch.
2. predicted: This shows the predicted movie id for the user by our Model.
3. true positive: This shows the right movies predicted and in the right order. Here all movies predicted exist in the movies the user wishes to watch however only 3 movies are in the right order ie. 8305, 8550 and 8063
4. accuracy: percentage of right order and right movies predicted.
5. mrr: Mean Reciprocal Rank, gives the error in prediction compared to actual.

The final output can be visualised as:

Text

Description automatically generated