**PREDICTION OF MOVIE RATINGS**

**Team Members:**

**Name: Vidip Kamdar NUID:(002701593)**

**Name: Divya Dubey NUID:(002776608)**

**Introduction:**

Movies are becoming more and more important entertainment in people’s lives. There are many choices of movies, and what a person may like is an interesting topic. Movies are not the only source of recreation, but they are one of the major sources of global commerce and of marketing. Movies create a new rush among the masses, especially among young people. Not only the movie directors and box office personnel are concerned with the success of movies but also the people in general. People also talk about these things on social media. Therefore, analysis of social media data about movies is recently popular among data analysts. Other than this there remains some other scopes like analyzing a director’s previous success histories or an actor's previous popularity etc. Again, the analysis may be different in different countries. Naturally people from all regions of the world do not react in a similar way to various genres of movies. Movies are now available on the internet. There are platforms like IMDb (Internet Movie Database), TMDB, Rotten Tomatoes, Metacritic etc. where people can share their reviews about movies. Day-to-day these platforms and media are becoming famous since people are getting their honest reviews there. So, a huge amount of data is available online about the feedback and ratings of movies. In this project, we are building a movie rating predicting system.

User rating is a kind of Word of Mouth (WOM), it simplifies the procedure in which consumers express their opinions about a product. User rating is highly important to any product or service because it reflects the wisdom of the people. Undoubtedly, the user rating is a great indication to predict the future sales and performance of a product. Movie industry specialists agree that it is a key success factor of movies and helps movie production companies and investors gain financial success. To predict movie ratings, we will conduct an ablation study of various visual and textual features and evaluate their performance in the prediction accuracy. There are lots of methods that we can use to build a prediction model. Specifically, we will use linear regression, decision tree, random forest and KNN as input algorithms.

The objective of this project is to learn from the data and rate the movies which will also predict which algorithms have the proper accuracy in rating any products. The other is also sometimes such that the movies that are less popular have an exceptionally higher rating average. We aim to rate the movies and then find the difference which will help gain the results of the accuracy of the model. Also, we hope to offer insight on the determining factors of film ratings that will guide producers through film promotion.

**Description of the dataset:**

The link of our dataset is:

<https://www.kaggle.com/datasets/akshaypawar7/millions-of-movies>

The Movies Daily has an updated dataset file which consists of metadata for more than 700,000 movies listed in the IMDB dataset. The dataset is updated daily to ensure the updated movies are in the dataset. Certain data points include cast, crew, plot keywords, budget, revenue, posters, release dates, languages, production companies, countries, IMDB vote counts and vote averages, reviews, recommendations. IMDB vote counts and vote averages, reviews, recommendations, are included. Ratings are on the scale of 1 to 10 and have been obtained from the official Group Lens website. The dataset file consists of:

|  |  |  |
| --- | --- | --- |
| No. | Data Index content | Description |
| 1 | Id | IMDB id assigned |
| 2 | Title | Title of movies |
| 3 | Genres | '-' separated category that movie belongs to including various genres and categories |
| 4 | original\_language | Language that movie is made in |
| 5 | Overview | A short description of the movies |
| 6 | popularity | IMDB metric of the popularity ratio. It is dependent on various categories |
| 7 | production\_companies | '-' separated production company names that have produced in the movies |
| 8 | release\_date | Movie release date in the format yyyy-mm-dd |
| 9 | Budget | Entire gross budget of movie |
| 10 | Revenue | Total revenue generated by the movie. This is the international revenue generated by the movie. The gross earning of the movie project |
| 11 | Runtime | Total duration of the movie runtime |
| 12 | Status | Status of the movie as in Released, in production, Not released, rumored, etc. |
| 13 | Tagline | Tagline of the movie |
| 14 | vote\_average | Average votes given by the IMDB users |
| 15 | vote\_count | Count of total users who gave votes of the movie |
| 16 | Credits | '-' Separated cast of the movie. A number of the lead star cast of the movies mentioned |
| 17 | keywords | '-' separated keywords that give the description of the movie. Keywords can also be used for certain prediction |
| 18 | poster\_path | Poster image path of movies |
| 19 | backdrop\_path | Background images path of the movies |
| 20 | recommendations | '-' separated recommended movie id |

The datatype of the dataset attributes is given below:

**Table

Description automatically generated**

**Methodology:**

1. **Collecting the dataset:**

We got our dataset from kaggle from there we downloaded movies.csv dataset which is: <https://www.kaggle.com/datasets/akshaypawar7/millions-of-movies>

1. **Importing the dataset:**

The dataset is imported by read.csv() function. Head() function is used to display the content inside the dataset.

Text

Description automatically generated with medium confidence

To check the shape of the dataset use dataframe\_name.shape.

Graphical user interface, text

Description automatically generated

### **Data Cleaning:**

Data cleaning is the process of fixing or removing incorrect, corrupted, incorrectly formatted, duplicate, or incomplete data within a dataset. If data is incorrect, outcomes and algorithms are unreliable, even though they may look correct.

* Renaming column vote\_average to imdb\_ratings to have better understanding.

A picture containing text

Description automatically generated

* Checking for null values and the count of null values in the dataset.

Table

Description automatically generated Table

Description automatically generated with low confidence

* Creating a new dataframe to store the data with vote counts more than 20.

Graphical user interface, text, application, email

Description automatically generated

* Dropping the categorical columns that are not needed.

Table

Description automatically generated

* Dropping the null values and duplicate values. Dropped columns like poster\_path, Backdrop\_path, recommendations, that aren’t relevant to or updated for new movies using pandas’ .drop() function.

A picture containing timeline

Description automatically generated

Graphical user interface, text

Description automatically generated

* Replacing the 0’s in the dataset with the mean and median of the column.

Graphical user interface, text, application

Description automatically generated

1. **Data Visualization:**

Data visualization is the graphical representation of information and data. Data visualization is the practice of translating information into a visual context, such as a map or graph, to make data easier for the human brain to understand and pull insights from. The main goal of data visualization is to make it easier to identify patterns, trends, and outliers in large data sets. The term is often used interchangeably with others, including information graphics, information visualization and statistical graphics.

Data visualization is one of the steps of the data science process, which states that after data has been collected, processed, and modeled, it must be visualized for conclusions to be made.

* In the below image we used matplotlib to display the counts of imdb\_ratings.

Chart, histogram

Description automatically generated

* Here we have used seaborn to generate correlation heatmap. Correlation matrices are an essential tool of exploratory data analysis. Correlation heatmaps contain the same information in a visually appealing way.

Chart

Description automatically generated with low confidence

* In the heatmap we can see the highest correlation between revenue and profit made and the second highest correlation is between revenue and vote count.

From this we can conclude that most of the highest generated movies have the highest number of voters.

* From the perspective of building models, by visualizing the data we can find the hidden patterns, explore if there are any clusters within data and we can find if they are linearly separable/too much overlapped etc. From this initial analysis we can easily rule out the models that won’t be suitable for such a data and we will implement only the models that are suitable, without wasting our valuable time and computational resources.
* This part of data visualization is a predominant one in initial Exploratory Data Analysis (EDA) in the field of Data science/ML.
* As mentioned above, Python has several good packages to plot the data and among them Matplotlib is the most prominent one. Seaborn is also a great package which offers a much more appealing plot and even uses matplotlib as its base layer.

Chart, histogram

Description automatically generated Chart, histogram

Description automatically generated

1. **Feature Scaling:**

Feature Scaling is a technique to standardize the independent features present in the data in a fixed range. It is performed during the data pre-processing to handle highly varying magnitudes or values or units. If feature scaling is not done, then a machine learning algorithm tends to weigh greater values, higher and consider smaller values as the lower values, regardless of the unit of the values.

Python sklearn library offers us with StandardScaler() function to standardize the data values into a standard format. StandardScaler is used when the input dataset differs greatly between their ranges. It removes the mean and scales the data to the unit variance. Here, the variables that are measured at different scales do not contribute equally to the fit of the model and the learning function of the model and could end up creating a bias. We also must handle the outliers. So, in this process we clean the data and make it ready for the Training purpose.

**Graphical user interface, text, application, email

Description automatically generated**

1. **Algorithms and their accuracy:**

We used [Scikit-learn](https://scikit-learn.org/) for training our model and for loading and splitting data. Scikit-learn is a popular Machine Learning (ML) library that offers various tools for creating and training ML algorithms, feature engineering, data cleaning, and evaluating and testing models. It was designed to be accessible, and to work seamlessly with popular libraries like NumPy and Pandas.

Graphical user interface, text, application, email

Description automatically generated

* **KNN Algorithm:**

KNN can be used for classification in a supervised setting where we are given a dataset with target labels. For classification, KNN finds the k nearest data points in the training set and the target label is computed as the mode of the target label of these k nearest neighbors. The KNN algorithm involves retrieving the K data points that are nearest in distance to the original point. It can be used by aggregating the target values of the nearest neighbors to make a prediction. However, just retrieving the nearest neighbors is a very important aspect in several applications.

The KNN algorithm has the following features:

* KNN is a Supervised Learning algorithm that uses labeled input data sets to predict the output of the data points.
* It is one of the simplest Machine learning algorithms and it can be easily implemented for a varied set of problems.
* It is mainly based on feature similarity. KNN checks how similar a data point is to its neighbor and classifies the data point into the class it is most like.

Advantages of KNN:

* Simple to implement
* Can learn non-linear decision boundaries when used for classification.
* No Training Time for classification
* Constantly evolves with new data
* Single Hyperparameters
* Choice of distance metric

KNN is a very powerful algorithm. It is also called “lazy learner”. However, it has the following set of limitations:

* Doesn’t work well with a large dataset
* Doesn’t work well with a high number of dimensions
* Sensitive to outliers and missing values



The tested accuracy of KNN model is given below:

Text

Description automatically generated

* **Decision Trees:**

Decision Tree algorithm belongs to the family of supervised learning algorithms. Unlike other supervised learning algorithms, the decision tree algorithm can be used for solving regression and classification problems too.

The goal of using a Decision Tree is to create a training model that can be used to predict the class or value of the target variable by learning simple decision rules inferred from prior data (training data).

In Decision Trees, for predicting a class label for a record we start from the root of the tree. We compare the values of the root attribute with the record’s attribute. Based on comparison, we follow the branch corresponding to that value and jump to the next node.

Some advantages of decision trees are:

* Simple to understand and to interpret. Trees can be visualized.
* Requires little data preparation. Other techniques often require data normalization, dummy variables need to be created and blank values to be removed.
* The cost of using the tree (i.e., predicting data) is logarithmic in the number of data points used to train the tree.
* Able to handle both numerical and categorical data. However, the scikit-learn implementation does not support categorical variables for now.
* Able to handle multi-output problems.

The disadvantages of decision trees include:

* Decision-tree learners can create over-complex trees that do not generalize the data well.
* Decision trees can be unstable because small variations in the data might result in a completely different tree being generated. This problem is mitigated by using decision trees within an ensemble.
* Predictions of decision trees are neither smooth nor continuous, but piecewise constant approximations as seen in the above figure. Therefore, they are not good at extrapolation.
* The problem of learning an optimal decision tree is known to be NP-complete under several aspects of optimality and even for simple concepts.
* Decision tree learners create biased trees if some classes dominate. It is therefore recommended to balance the dataset prior to fitting in with the decision tree.



Text

Description automatically generated

* **Random Forest:**

A random forest produces good predictions that can be understood easily. It can handle large datasets efficiently. The random forest algorithm provides a higher level of accuracy in predicting outcomes over the decision tree algorithm. Random Forest works by understanding decision trees and applying decision trees to random forests. Classification in random forests employs an ensemble methodology to attain the outcome. The training data is fed to train various decision trees. This dataset consists of observations and features that will be selected randomly during the splitting of nodes.

A random forest system relies on various decision trees. Every decision tree consists of decision nodes, leaf nodes, and a root node. The leaf node of each tree is the final output produced by that specific decision tree. The selection of the final output follows the majority-voting system. In this case, the output chosen by most of the decision trees becomes the final output of the rain forest system. The diagram below shows a simple random forest classifier.

### 

### Features of a Random Forest Algorithm

* It’s more accurate than the decision tree algorithm.
* It provides an effective way of handling missing data.
* It can produce a reasonable prediction without hyper-parameter tuning.
* It solves the issue of overfitting in decision trees.
* In every random forest tree, a subset of features is selected randomly at the node’s splitting point.

### Advantages of random forest

* It can perform both regression and classification tasks.
* A random forest produces good predictions that can be understood easily.
* It can handle large datasets efficiently.
* The random forest algorithm provides a higher level of accuracy in predicting outcomes over the decision tree algorithm.

### Disadvantages of random forest

* When using a random forest, more resources are required for computation.
* It consumes more time compared to a decision tree algorithm.



**Text

Description automatically generated**

1. **Results and Model Comparison:**

A Classification report is used to measure the quality of predictions from a classification algorithm. How many predictions are True and how many are False. More specifically, True Positives, False Positives, True negatives, and False Negatives are used to predict the metrics of a classification report.

The report below shows the main classification metrics precision, recall and f1-score on a per-class basis. The metrics are calculated by using true and false positives, true and false negatives. Positive and negative in this case are generic names for the predicted classes. There are four ways to check if the predictions are right or wrong:

1. **TN / True Negative:**when a case was negative and predicted negative
2. **TP / True Positive:**when a case was positive and predicted positive
3. **FN / False Negative:**when a case was positive but predicted negative
4. **FP / False Positive:**when a case was negative but predicted positive

### **Precision – What percent of your predictions were correct?**

Precision is the ability of a classifier not to label an instance positive that is actually negative. For each class it is defined as the ratio of true positives to the sum of true and false positives.

**Precision – Accuracy of positive predictions.**

**Precision = TP/(TP + FP)**

### **Recall – What percent of the positive cases did you catch?**

### Recall is the ability of a classifier to find all positive instances. For each class it is defined as the ratio of true positives to the sum of true positives and false negatives.

Recall: Fraction of positives that were correctly identified.  
Recall = TP/(TP+FN)

### **F1 score – What percent of positive predictions were correct?**

### The F1 score is a weighted harmonic mean of precision and recall such that the best score is 1.0 and the worst is 0.0. F1 scores are lower than accuracy measures as they embed precision and recall into their computation. As a rule of thumb, the weighted average of F1 should be used to compare classifier models, not global accuracy.

F1 Score = 2\*(Recall \* Precision) / (Recall + Precision)

**Table

Description automatically generated**

Random Forest Model and KNN Model provided the highest accuracy for test dataset. The highest accuracy score is 64% with 7,00,000 rows corresponding to a number of movies being evaluated.

**Conclusion:**

Standard scaling of data increases the score on average. Most of the time, increasing the value of K in KNN increases the score. We had to visualize and analyze combinations of features because no single feature can accurately predict ratings of new movies. Yet, there are other numerous combinations that can affect the ratings differently than the individual values would indicate. Film industry specialists agree that it is a key success factor of movies and helps movie production companies and investors gain financial success. Feature engineering helps extract information from raw data, i.e., it has created a lot of features. This means we need to find the main features of the whole lot. Overall, we believe that our study on machine learning-based solutions opens up a promising direction and can be used as a reference guide for potential research and applications.

As a result, we choose algorithms such as KNN, Decision Trees, Random Forest and Linear Regression for model implementation as there is no particular proof for choosing one algorithm. To obtain the accuracy of prediction, we have trained the model with the same set of data with movie scripts and with classification algorithms like KNN, Random Forest classifier, Decision Trees and Logistic Regression that are identified. The accuracy of each model is calculated to compare with the remaining models. We can conclude that the model trained using Decision Trees, Random Forest algorithms has shown good performance with an accuracy of 86% in the prediction of movie names, when compared to the remaining models.

**References:**

* [*https://www.kaggle.com/datasets/akshaypawar7/millions-of-movies*](https://www.kaggle.com/datasets/akshaypawar7/millions-of-movies)
* [*https://www.kaggle.com/datasets/netflix-inc/netflix-prize-data*](https://www.kaggle.com/datasets/netflix-inc/netflix-prize-data)
* [*https://medium.com/web-mining-is688-spring-2021/predicting-imdb-ratings-of-new-movies-2b39459fee9a*](https://medium.com/web-mining-is688-spring-2021/predicting-imdb-ratings-of-new-movies-2b39459fee9a)
* [*https://ieeexplore.ieee.org/document/9076540*](https://ieeexplore.ieee.org/document/9076540)
* [*https://link.springer.com/article/10.1007/s42979-021-00592-x#Sec15*](https://link.springer.com/article/10.1007/s42979-021-00592-x#Sec15)
* [*https://www.kaggle.com/code/bhsraman/predict-movie-ratings-via-machine-learning/data*](https://www.kaggle.com/code/bhsraman/predict-movie-ratings-via-machine-learning/data)
* [*https://towardsdatascience.com/feature-engineering-in-python-part-i-the-most-powerful-way-of-dealing-with-data-8e2447e7c69e*](https://towardsdatascience.com/feature-engineering-in-python-part-i-the-most-powerful-way-of-dealing-with-data-8e2447e7c69e)
* [*https://towardsdatascience.com/all-about-feature-scaling-bcc0ad75cb35*](https://towardsdatascience.com/all-about-feature-scaling-bcc0ad75cb35)
* [*https://io9.gizmodo.com/5747305/how-much-money-does-a-movie-need-to-make-to-be-profitable*](https://io9.gizmodo.com/5747305/how-much-money-does-a-movie-need-to-make-to-be-profitable)
* [*https://developers.themoviedb.org/3/getting-started/introduction*](https://io9.gizmodo.com/5747305/how-much-money-does-a-movie-need-to-make-to-be-profitable)
* [*https://www.mpaa.org/film-ratings/*](https://io9.gizmodo.com/5747305/how-much-money-does-a-movie-need-to-make-to-be-profitable)
* *Antara Upadhyay, Nivedita Kamath, Shalin Shanghavi, Tanisha Mandvikar and Pranali Wagh, "Movie Success Prediction Using Data Mining", Department of Information Technology Shah & Anchor Kutchhi Engineering College, vol. 6, no. 4.*
* *V Anantharaman, Ebin G Job, Neha Sam and Asst.Prof. Sheryl Mariya Sebastian, "Movie Success Prediction Using data Mining", Department of Computer Science and Engineering Albertian Institute of Science and Technology Kalamassery Ernakulam Kerala. GRD JournalsCONFERENCE, May 2019*