INTERNATIONAL JOURNAL OF ADVANCE RESEARCH, IDEAS AND INNOVATIONS IN TECHNOLOGY

ISSN: 2454-132X
Impact Factor: 6.078
(Volume 9, Issue 6 - V9I6-1241)
Available online at: https://www.ijariit.com

A comparison of Machine Learning techniques for predicting IMDb score of movies

Anubhav Mathur
anu1998.mat@gmail.com
Independent Researcher

Snigdha Patil
snigdhaspatil8@gmail.com
Independent Researcher

ABSTRACT

In the world of film industry analytics, predicting the success of movies based on various input features has garnered considerable attention in recent times. This research is important because it can help people in the movie industry make better decisions. It allows them to allocate resources effectively, minimize risks, and enhance the overall success of movie projects. This research paper presents and compares different machine learning techniques to predict the IMDb score of movies by leveraging multiple input features such as release date, genre, budget, gross revenue, profit, number of votes, country of origin, director, actor, writer, production studio, and runtime. We also account for the inflation rate over the years while considering the monetary attributes. We explore two methods: one where we transform and reduce the data using techniques like one-hot encoding and PCA, and another where we use label encoding for categorical data. In the first method, we try three models: Support Vector Regressor (SVR), RandomForest Regressor, and Recurrent Neural Network (RNN). In the second method, we use RandomForest Regressor, Gradient Boosting Regressor, and LightGBM Regressor. We measure how well these models predict by looking at Mean Squared Error (MSE) and R-squared. This research helps provide people in the film industry with insights into what factors contribute to a movie's success.

Keywords:- *Machine Learning, Principal Component Analysis, Light Gradient Boosting Machine (LGBM), Regression Analysis*

1. INTRODUCTION

In today's film industry, understanding what makes a movie successful has become crucial. The success of a movie is often gauged by how audiences respond to it. With the rise of social media platforms, people are more interconnected than ever, using these platforms to both review movies and be influenced by others' reviews. Each individual forms their own perspective and shares it through various platforms; IMDb (Internet Movie Database) is one such platform that enables movie watchers to score a movie, and this score is aggregated across all such reviews. This paints a picture for other moviegoers, indicating whether the movie is worth seeing or not. Hence, it serves as a critical indicator of how audiences will respond to a movie. Nowadays, an increasing number of people are actively engaged on such platforms. While production studios always aim to maximize profits, they also focus on strategies to ensure users rate their movies positively. This rating is influenced by several factors. For instance, certain actors and directors are widely celebrated and consistently receive high ratings for their movies. Conversely, some actors unfortunately tend to be associated with low-rated movies.

Additionally, certain audiences may be interested in specific genres of films, leading them to provide higher ratings for those genres. Moreover, a movie's rating such as R or PG-13 might also impact the type of audience entering theaters, thereby affecting the IMDb score. Thus, the IMDb score is not a simple measure but rather a combination of multiple factors. This research dives into the world of film industry analytics, aiming to predict how well a movie will do on IMDb. In our paper, we leverage this complexity to explore correlations between these factors and the eventual IMDb score using machine learning techniques. These powerful models enable us to construct a system fitted to these parameters, allowing us to predict the IMDb score when provided with these input features. Being able to know how a movie will perform using the power of predictive analytics serves as a guide for decision-makers in the film world.

For our research, we initially acquired data for movies released between 1980 and 2020. This dataset encompasses all the attributes required for constructing our machine learning (ML) models aimed at predicting IMDb ratings. We adopt two distinct approaches utilizing ML techniques to develop our IMDb rating predictor model. In the first approach, we employ one-hot encoding to convert categorical features into a format compatible with our regression ML algorithms. This encoding significantly increases the number of features. To mitigate redundancy and enhance computational efficiency, we implement dimensionality reduction through Principal Component Analysis (PCA). After reducing the number of features, we employ three different ML models to fit our data and evaluate their performance. These models comprise Support Vector Regressor (SVR), RandomForest Regressor, and Recurrent Neural Network (RNN). We assess each model on our test data, capturing measures of model fit by comparing predicted scores to actual scores.

In the second approach, instead of using one-hot encoding, we opt for Label encoding, which does not introduce additional features but labels categorical features in place. As there are no additional features, there is no need for dimensionality reduction, and we directly proceed to use ML models capable of handling label-encoded data. This set of models includes RandomForest Regressor again, alongside Gradient Boosting Regressor and LightGBM Regressor. Subsequently, we assess how well these models perform using the same measures, determining the most effective model for predicting IMDb scores.

2. RELATED WORK

You, X. et al[1] explored a multiple nonlinear regression model for predicting the movie score in exponential form. Their model used the metadata variables of the film and the characteristic variables of film actors to perform analysis on the factors that affect film scoring. They developed their model by using the concept of index. To avoid the redundancy of explanatory variables in their model, the Akaike information criterion (AIC) values of their model and its five sub-models were calculated to choose the explanatory variables. However, they did not consider inflation in their calculation.

Dhir, R. et al[2] proposed an approach to predict how successful a movie will be before it appears at the box office. Among all the different algorithms they used, Random Forest gave the best prediction accuracy. Their database had categorical and numerical information such as IMDb score, director, gross, budget, etc. However, they have not used any encoding technique for categorical variables. They have also not accounted for the inflation over a number of years. While past work has used a multiple nonlinear regression model and Random Forest without encoding, our work uses a machine learning approach that utilizes one-hot encoding and label encoding by also accounting for inflation when considering the margin and compares the results of various approaches.

3. DATASET

3.1 Dataset description

The dataset was sourced from Kaggle and spans 40 years of movie data, covering the period from 1980 to 2020. It was extracted from IMDb and encompasses various features, including release date, genre, budget, gross revenue, profit, number of votes, country of origin, director, actor, writer, production studio, and runtime of the movies. The initial dataset comprises a total of 7512 movies spanning these four decades. Data cleaning and preprocessing were used to eliminate records with missing values and introduce informative new variables such as release year, month, and profit values normalized to a logarithmic scale. Through exploratory data analysis of the dataset, we were able to determine correlations between certain features and gain an overall understanding of the data.

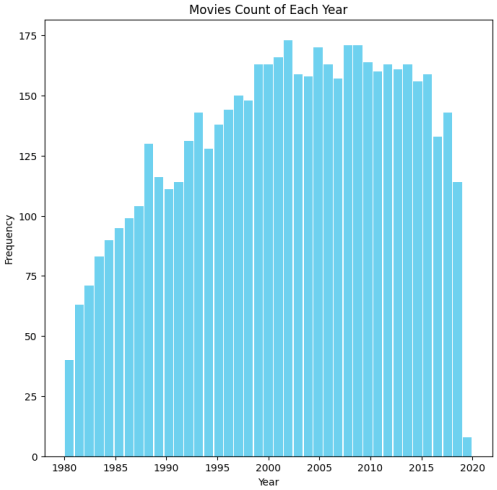

Fig-1: Movie count of each year

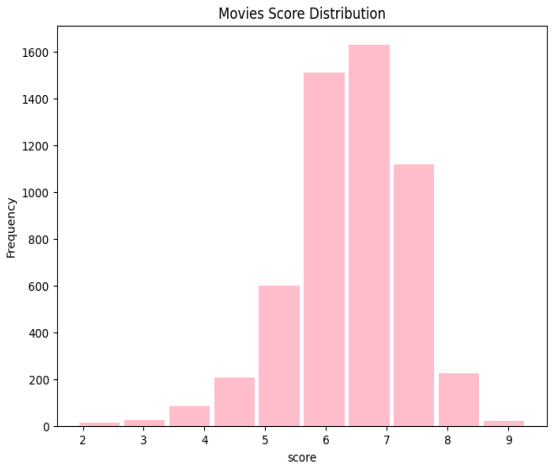

Fig-2: Movie score distribution

3.2 Data pre-processing

We began by analyzing the dataset for null values and discovered that the budget variable had null values for 2171 records, while gross revenue was null for 189 records. We eliminated all records where either the budget or gross revenue was null. We extracted the day, month, and year from the release date variable to have them as separate features for our model. Moreover, we needed to account for

inflation for movies released between 1980 and 2020. Using the average inflation rate of 2.90% per year between 1980 and 2020, compounded annually, and the delta of years between 1980 and 2020, we calculated an adjustment factor for each year. This adjustment factor was then multiplied to movies based on their respective year of release to adjust their budget and gross amounts to account for inflation.

$$avg\_inflation = 0.0290$$
$$adj\_factor = (1 + avg\_inflation) \wedge delta\_yrs$$
$$gross\_npv = gross\_revenue * adj\_factor$$

Furthermore, we calculated the profit of each movie as the difference between the gross value and the budget value, and the profit margin as the ratio of profit to the budget value. These three values—budget, revenue, and profit—were then converted to logarithmic values to scale their magnitudes relative to the other features.

$$profit\_npv = gross\_npv - budget\_npv$$
$$margin\_npv = (gross\_npv - budget\_npv) / (budget\_npv)$$

4. PROPOSED METHODOLOGY

This research focuses on predicting the IMDb score of the movie given the parameters like release date, genre, budget, gross revenue, profit, number of votes, country of origin, director, actor, writer, production studio, and runtime. We propose two novel approaches to predict the IMDb scores.

4.1 With one-hot encoding and dimensionality reduction

To represent categorical variables as binary vectors, we use the one-hot encoding technique in machine learning. It converts categorical data into a numerical format that can be provided to machine learning algorithms to improve predictions. In one-hot encoding, each category or label in the categorical variable is represented by a unique binary digit (bit).

For each category, there is a binary column, and only one of these columns is "hot" (i.e. set to 1) for each data point. We perform one-hot encoding on the columns - 'genre', 'rating', 'director', 'writer', 'star', 'country', 'company'. Thus, we can capture the impact of these features on our target i.e. IMDb score. Our entire data is now numerical, however, there is one problem.

One-hot encoding significantly increases the dimensionality of the dataset. This increase in dimensionality can result in the "curse of dimensionality," where the data becomes more sparse, which may pose challenges for certain algorithms. In our case, we get 8714 features after one-hot encoding. We use Principal Component Analysis (PCA), a dimensionality reduction technique to transform high-dimensional data into a lower-dimensional representation, capturing the most significant variability in the data. Finally we are left with 100 features. We use these features as input to our prediction models. We tried the below machine learning techniques.

a. Support Vector Regressor (SVR) - SVR is a regression algorithm that leverages the principles of support vector machines to model the relationship between features and a continuous target variable. It is suitable for a variety of regression tasks, especially when dealing with non-linear relationships in the data.

b. RandomForest Regressor - It is an ensemble learning method that builds multiple decision trees during training and combines their predictions to obtain a more robust and accurate model. Each decision tree is constructed independently and is trained on a random subset of the training data and features that helps in reducing overfitting. The randomness introduced by this technique contributes to the diversity of the individual trees and helps in creating a more robust and generalized model.

c. Recurrent Neural Network (RNN) - This is a type of artificial neural network designed for sequence modeling and processing. Unlike traditional feedforward neural networks, RNNs have connections that form directed cycles, allowing them to maintain and utilize information about previous inputs in the sequence. They maintain a hidden state that captures information about the sequence processed so far.

One-hot encoding is used on the categorical features in the dataset and the PCA is used to reduce dimensionality. This data is fed to the three models - Support Vector Regressor (SVR), RandomForest Regressor, and Recurrent Neural Network (RNN) that give us a predicted score for the movie. Then accuracy is calculated using the metrics such as Mean Squared Error (MSE) and R2 score. Fig-3. shows the architecture diagram of this approach.

When we applied one-hot encoding and PCA to the dataset, using them for these three models, we observed higher levels of Mean Squared Error (MSE) and lower R2 scores. Consequently, we decided to experiment with a different approach by utilizing label encoding.

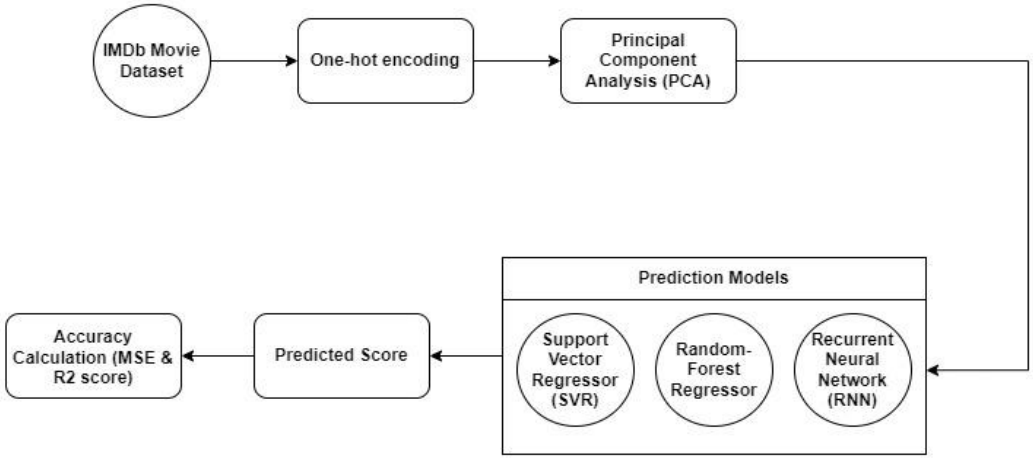

Fig. 3: Architecture for first approach

4.2 With label encoding

Label encoding is a technique used in machine learning to convert categorical data into numerical format. In label encoding, each unique category or label is mapped to an integer. This encoding is particularly useful when working with algorithms that require numerical input, as it allows the model to understand and process categorical variables. Using this technique, we tried three models - RandomForest Regressor, Gradient Boosting Regressor and LightGBM Regressor.

a. Gradient Boosting Regressor - It is an ensemble learning method that builds a strong predictive model by combining the predictions of multiple weak learners, typically decision trees. The term "gradient boosting" refers to the technique of iteratively fitting new models to the residual errors of the previous models. It incorporates regularization techniques to prevent overfitting. Regularization is achieved through hyperparameters that control the learning rate and the depth of the individual trees.
b. LightGBM Regressor - It is based on the gradient boosting framework, which builds an ensemble of weak learners (usually decision trees) sequentially to make accurate predictions. One of the distinctive features of LightGBM is its leaf-wise tree growth strategy. Unlike traditional depth-wise tree growth, LightGBM expands the tree by growing the leaf that reduces the loss the most. This contributes to faster training times and improved accuracy.

Label encoding is used on the dataset and there is no need for PCA. This data is fed to the models - RandomForest Regressor, Gradient Boosting Regressor and LightGBM Regressor that give us a predicted score for the movie. Similar to the previous approach, accuracy is calculated using the metrics such as Mean Squared Error (MSE) and R2 score. Fig-4. shows the architecture diagram of this approach.

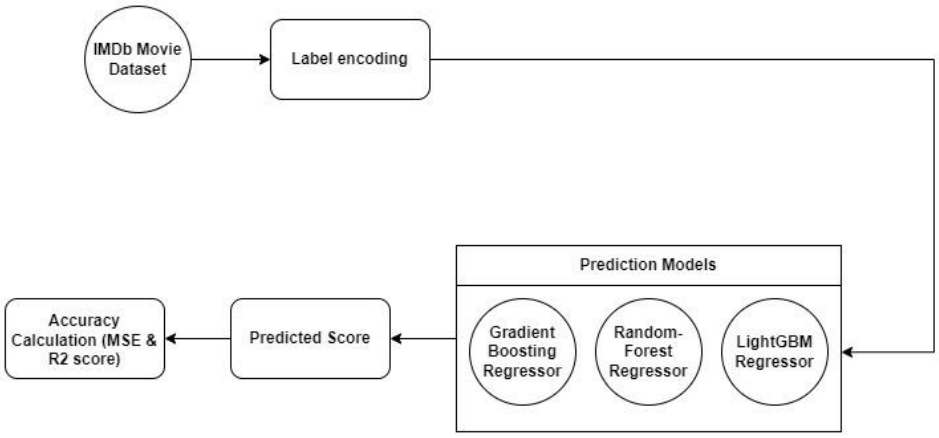

Fig. 4: Architecture for first approach

5. RESULTS

The implementation of our prediction techniques to forecast the IMDb score of a movie based on the given factors was achieved by first preprocessing the data and then dividing it into two different sets. For one dataset, we utilized one-hot encoding to encode the categorical data, significantly increasing the number of features. Subsequently, we performed dimensionality reduction using PCA to reduce the number of features and generate 100 principal components. The resulting dataset was then fed into three models: Support Vector Regressor (SVR), RandomForest Regressor, and Recurrent Neural Network (RNN). For the second dataset, we employed label encoding to convert categorical features to numeric ones and used the resulting dataset for the three models: RandomForest Regressor, Gradient

Boosting Regressor, and LightGBM Regressor. The training and testing of all the models in both approaches were achieved through k-fold cross-validation with 5 folds, evaluating the performance of the model multiple times, and obtaining an average MSE and R2 value for each model.

Mean Squared Error (MSE):
The MSE represents the average squared difference between the actual and predicted values. Lower MSE values are better, indicating that the model's predictions are closer to the actual values.

$$MSE = \frac{1}{N}\sum_{i=1}^{N}(y_i - \hat{y}_i)^2$$

R-squared (R2):
R-squared is a measure of how well the model explains the variance in the target variable. R2 values range from 0 to 1, where 1 indicates a perfect fit and 0 indicates that the model does not explain any variability.

$$R^2 = 1 - \frac{\sum_{i=1}^{n}(Y_i - \hat{Y}_i)^2}{\sum_{i=1}^{n}(Y_i - \bar{Y}_i)^2}$$

It was observed that in the techniques that used one-hot encoding to convert categorical data into multiple features and then applied dimensionality reduction using PCA, SVR had the lowest value of MSE, while RNN had the highest, indicating that deep learning methods are not applicable in our use case. Both SVR and RandomForest Regressor had similar R2 values, with RandomForest Regressor having a slightly better value of 0.337.

The techniques that used label encoding performed better than the models in the first approach. In these models, Light Gradient Boosting Machine (LGBM) performed the best with an MSE of 0.399 and R2 of 0.561. Gradient Boosting Regressor performed similarly, with an MSE of 0.400 and R2 of 0.561. With RandomForest Regressor, we were able to achieve an MSE of 0.428 and an R2 value of 0.530.

Table-1: Resultant accuracy metrics

Approach	Model	Mean Squared Error (MSE)	R-squared (R2)
Using one-hot encoding + PCA	SVR	0.637	0.311
	RandomForest Regressor	0.661	0.337
	RNN	1.029	-0.110
Using categorical labeled data (without encoding/PCA)	RandomForest Regressor	0.428	0.530
	GradientBoosting Regressor	0.400	0.561
	LGBM Regressor	0.399	0.561

6. CONCLUSION

In this paper, we developed and compared various approaches to predict IMDb scores of movies using machine learning. This prediction serves as a valuable tool for individuals in the movie industry, aiding them in understanding the influence that different factors may have on a movie's score. For our case, we considered data spanning four decades, incorporating input features such as movie genre, rating, release date, actor, director, writer, runtime, budget, gross revenue, and profit. Notably, in our approach, we accounted for the inflation rate for each year between 1980 and 2020, appropriately scaling the values for the monetary attributes. We explored two distinct approaches to build our prediction model: one utilizing one-hot encoding and dimensionality reduction using PCA, and another using label encoding for categorical data. We evaluated these two approaches with three different regression models for each approach. We observed that high levels of accuracy (in the form of R2 score) and low error were noted for the techniques utilizing label encoding, with the Light Gradient Boosting Machine (LGBM) achieving the best result. Although high levels of accuracy were attained, we can anticipate that future advancements in machine learning algorithms may further enhance prediction accuracy. Additionally, in future research, we can incorporate users' sentiments into our models.

7. REFERENCES

[1] You, X., Liu, Y., Zhang, M., & Zhang, M. (2021). Movie score predication model based on multiple nonlinear regression. *Tehnicki Vjesnik-technical Gazette*, *28*(3).

[2] Dhir, R., & Ramkumar, A. (2018). Movie Success Prediction using Machine Learning Algorithms and their Comparison. *2018 First International Conference on Secure Cyber Computing and Communication (ICSCCC)*
[3] Bristi, W. R., Zaman, Z., & Sultana, N. (2019). Predicting IMDb Rating of Movies by Machine Learning Techniques. *2019 10th International Conference on Computing, Communication and Networking Technologies (ICCCNT)*
[4] Pramod, S., Joshi, A. & Mary, A.G. (2017). Prediction of movie success for real world movie dataset. *Int. J. of Advance Res., Ideas and Innovations in Technol, 3(3).*
[5] Cizmeci, B., & Öğüdücü, Ş. G. (2018). Predicting IMDb Ratings of Pre-release Movies with Factorization Machines Using Social Media. *2018 3rd International Conference on Computer Science and Engineering (UBMK)*.
[6] Parimi, R., & Caragea, D. (2013). Pre-release Box-Office Success Prediction for Motion Pictures. In *Lecture Notes in Computer Science* (pp. 571–585).
[7] Jotheeswaran, J., Loganathan, R. and Madhu Sudhanan, B. (2012). Feature reduction using principal component analysis for opinion mining. *International Journal of Computer Science and Telecommunications, 3(5), pp.118-121*.
[8] Goyal, A., & Urolagin, S. (2022). Prediction of movie success on IMDb database using machine learning techniques. In *Algorithms for intelligent systems* (pp. 273–288).

