IMDB Prediction Model Using Machine Learning

Adana Alparslan Türkes Science and Technology University

1st Efe Erol
Department of Computer Engineering
200101066

2nd Yusuf Ömer Tursun
Department of Computer Engineering
200101005

3rd Umut Kuruluk
Department of Computer Engineering
210101118

***Abstract*—A dataset from Kaggle containing movie-related features was used to estimate the IMDB ratings of films. The dataset included various attributes such as genre, budget, director, cast, and release year, which were used to establish a regression problem aimed at predicting movie ratings. Several machine learning models were explored during the experimentation phase, including Linear Regression, Decision Tree Regressor, Random Forest Regressor, K-Nearest Neighbors, and XGBoost Regressor. Throughout the experiments, it was observed that the XGBoost Regressor yielded superior results compared to the other models. This report details the methodology, data preprocessing steps, model selection process, and the performance comparison of these regression models, ultimately presenting the final solution for predicting IMDB ratings with optimal accuracy.**

***Index Terms*—imdb, model prediction, xgboost, hyperparameter, encoding**

I. Introduction

Movies have always captured the attention of global audiences, and their success is often reflected in their IMDB ratings, which serve as a widely recognized metric for evaluating a film's reception. This paper explores a data-driven approach to predict the IMDB ratings of films based on various movie-related attributes. Rather than focusing on subjective reviews, the model leverages measurable indicators such as genre, budget, director, cast, and release year to estimate a movie's rating on a numerical scale.

The problem is modeled as a regression task where the goal is to predict a film's IMDB score based on the provided features. To achieve this, a dataset sourced from Kaggle was used for training and validation purposes. Multiple machine learning algorithms, including Linear Regression, Decision Tree Regressor, Random Forest Regressor, K-Nearest Neighbors, and XGBoost Regressor, were experimented with during the model development phase.

Throughout the experimentation, XGBoost Regressor demonstrated the most promising results, providing higher prediction accuracy compared to the other models. This report details the data preparation, feature selection, model evaluation, and the final model's effectiveness in predicting movie ratings, offering a comprehensive analysis of the regression models tested.

II. Related Work

IMDB rating prediction has been a widely studied topic in the domain of machine learning and data analysis, with various approaches explored to estimate movie ratings based on structured data. Previous studies have shown that features such as genre, budget, cast, director, and release year are among the most influential factors in predicting a movie's success and reception on platforms like IMDB [1][2].

In [3], the authors emphasize the importance of ensemble learning techniques for movie rating prediction tasks, demonstrating that Random Forest and Gradient Boosting models often outperform basic linear models due to their ability to capture complex feature interactions. Inspired by these findings, our study also explored a range of models, including Linear Regression, Decision Tree Regressor, Random Forest Regressor, K-Nearest Neighbors, and XGBoost Regressor.

Additionally, deep learning approaches such as neural networks and autoencoders have been explored in prior research for movie rating estimation tasks, particularly when dealing with larger datasets containing textual or visual data [4]. However, for structured tabular data, studies have shown that tree-based models, especially XGBoost, tend to yield higher performance metrics [5].

The influence of ensemble methods, particularly the boosting technique used in XGBoost, informed the model selection process in this study. This paper builds upon these previous works by conducting a comparative analysis of multiple regression models on a Kaggle-sourced dataset, ultimately concluding that XGBoost Regressor provided the most accurate predictions for IMDB ratings.

III. Data Collection and Inspection

A single dataset was used in this project, sourced from Kaggle. The dataset, titled "IMDB Score Prediction for Movies," can be accessed at [1]. It contains various movie-related features such as genre, budget, director, cast, and release year, which were utilized as input variables for predicting the IMDB ratings of movies.

The dataset was inspected for data quality issues, including missing values, inconsistent formats, and outliers. Fortunately, the dataset was well-structured and required minimal preprocessing. All features were converted into numerical formats where necessary to ensure compatibility with the regression models used in the study.

A. Data Description

The dataset used for this IMDB score prediction project consists of various features related to movies, covering attributes about the film's production, cast, audience reception, and performance metrics. Below is a detailed description of each column in the dataset:

- **Color**: Indicates whether the movie is in color or black and white.
- **Director name**: The name of the movie's director.
- **num_critic_for_reviews**: The number of professional critics who reviewed the movie.
- **duration**: The runtime of the movie, measured in minutes.
- **director_facebook_likes**: The number of likes on the director's official Facebook page.
- **actor_3_facebook_likes**: The number of likes on the third actor's Facebook page.
- **actor2_name**: The name of the second lead actor in the movie.
- **actor_1_facebook_likes**: The number of likes on the first lead actor's Facebook page.
- **gross**: The gross earnings of the movie in US dollars.
- **genres**: The genre(s) of the movie, such as Animation, Comedy, Romance, Horror, Sci-Fi, Action, Family, etc.
- **actor_1_name**: The name of the first lead actor in the movie.
- **movie_title**: The title of the movie.
- **num_voted_users**: The number of users who voted for the movie on IMDB.
- **cast_total_facebook_likes**: The total number of Facebook likes for the entire cast combined.
- **actor_3_name**: The name of the third lead actor in the movie.
- **facenumber_in_poster**: The number of actors featured in the movie poster.
- **plot_keywords**: Keywords describing the movie's plot.
- **movie_imdb_link**: The link to the movie's IMDB page.
- **num_user_for_reviews**: The number of user-generated reviews on IMDB.
- **language**: The language in which the movie is produced.
- **country**: The country where the movie was produced.
- **content_rating**: The content rating of the movie, such as PG, R, G, etc.
- **budget**: The budget of the movie in US dollars.
- **title_year**: The year the movie was released.
- **actor_2_facebook_likes**: The number of Facebook likes on the second lead actor's page.
- **imdb_score**: The IMDB score of the movie, serving as the target variable in this regression task.
- **aspect_ratio**: The aspect ratio in which the movie was filmed.
- **movie_facebook_likes**: The total number of Facebook likes for the movie itself.

The target variable for this project is the imdb score, which represents the overall rating of a movie on a scale from 0 to 10, as rated by the audience on IMDB. The other columns serve as the features used to train the machine learning models in predicting this score.

	color	director_name	num_critic_for_reviews	duration	director_facebook_likes
0	Color	James Cameron	723.0	178.0	0.0
1	Color	Gore Verbinski	302.0	169.0	563.0
2	Color	Sam Mendes	602.0	148.0	0.0
3	Color	Christopher Nolan	813.0	164.0	22000.0
4	NaN	Doug Walker	NaN	NaN	131.0
5	Color	Andrew Stanton	462.0	132.0	475.0
6	Color	Sam Raimi	392.0	156.0	0.0
7	Color	Nathan Greno	324.0	100.0	15.0

Fig. 1. A piece of dataset.

To gain insight into the relationships between various features and their impact on a movie's IMDB rating, a preliminary data analysis was conducted. Scatter plots were generated for all features against the target variable imdb score to observe trends and correlations between the features and the movie ratings. The visualizations revealed some interesting patterns. For instance, features such as gross earnings, budget, and number of voted users showed a moderate positive correlation with IMDB scores, while variables like director_facebook_likes and cast_total_facebook likes exhibited weaker correlations. These scatter plots not only helped in understanding the spread and distribution of the data but also provided valuable insights into potential feature importance, which will be further analyzed in the Experiments section.

IV. Models and Model Selection

A. Objective

The primary objective of this project was to predict the IMDB score of movies using a set of various features. The evaluation metric for the models was the Mean Squared Error (MSE), which measures the average squared differences between predicted and actual IMDB scores. Minimizing this error function is crucial to ensuring that the model makes accurate predictions. Additionally, other metrics such as R-squared and Root Mean Squared Error (RMSE) were also considered to evaluate the effectiveness of the models.

B. Regression Models

Several regression models were considered to solve the IMDB score prediction problem. These models aim to mini-

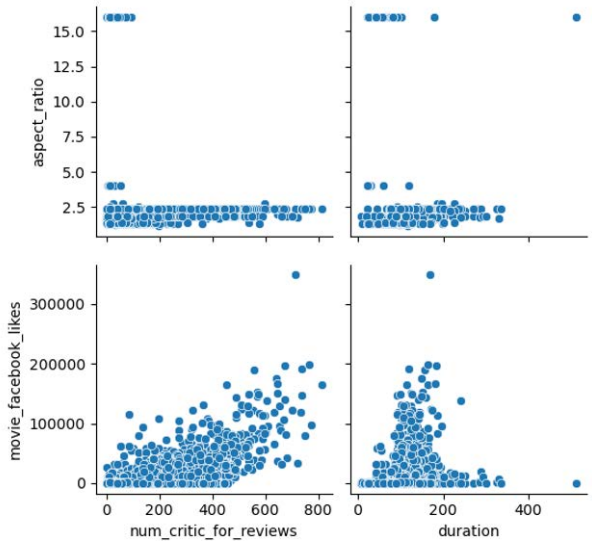

Fig. 2. Distribution examples.

mize the error between predicted and actual scores and were evaluated based on their performance during training and validation phases.

1) Linear Regression: Linear regression was used as the baseline model for comparison. This model assumes a linear relationship between the input features and the target variable (IMDB score).

2) Regularized Linear Regression (Ridge and Lasso): Ridge and Lasso regression were used to address overfitting by penalizing the size of the regression coefficients. Ridge regression minimizes the L2-norm, while Lasso focuses on minimizing the L1-norm.

3) Decision Tree Regression: A decision tree model was used to capture non-linear relationships between the features and IMDB score. The decision tree splits the data based on feature values and creates segments that can be analyzed for better prediction accuracy.

4) Random Forest Regression: Random Forest, an ensemble learning method, was employed to combine multiple decision trees to improve accuracy and reduce overfitting.

C. Classification Models

While regression models are key to predicting continuous values such as IMDB scores, classification models were also explored, especially for determining whether a movie would fall into specific categories of ratings (e.g., high or low IMDB score).

1) Logistic Regression: Used as a baseline classifier, logistic regression helps determine whether a movie belongs to a specific rating category, based on the features.

2) Support Vector Classification (SVC): Support Vector Classification with kernels explored the possibility of non-linear decision boundaries and helped classify movies based on the binary categorization of IMDB scores.

3) Random Forest Classifier: Random Forest Classifier was used to understand the decision boundaries and provide a more robust classification of IMDB scores.

V. Experiments

A. Feature, Variable Normalization

- **Label-Encoding**: Label encoding was applied to convert categorical variables into numerical representations. In the IMDB dataset, categorical features such as genre and language were transformed into numerical values to make them suitable for machine learning algorithms.

$$y_i = \begin{cases} 0 & \text{if label = A} \\ 1 & \text{if label = B} \\ 2 & \text{if label = C} \\ \vdots & \vdots \\ n & \text{if label = N} \end{cases} \quad (1)$$

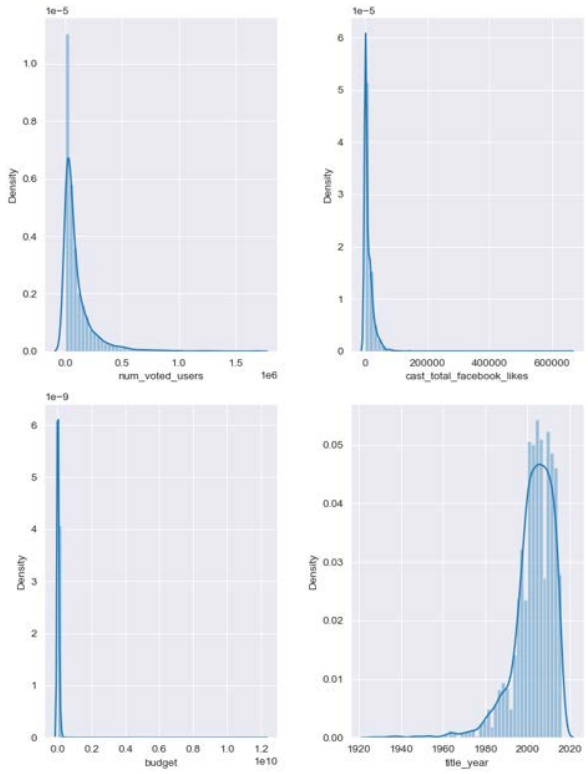

Fig. 3. Distribution plot after applying Label-Encoding.

- **Log Transformation**: Log transformation was used for continuous variables with right-skewed distributions. This transformation helps make the data more symmetric, reducing the impact of extreme values and improving the reliability of statistical analyses.

$$x' = \log(x) \tag{2}$$

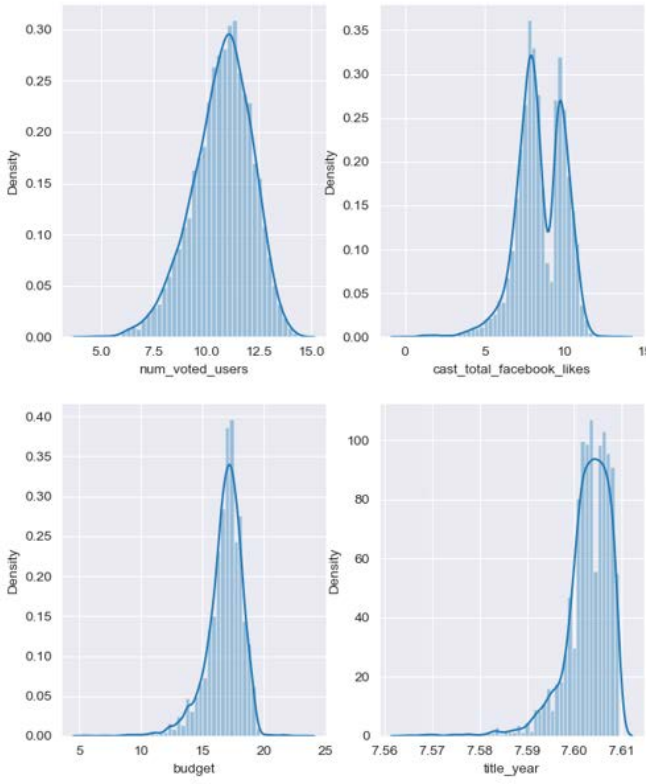

Fig. 4. Distribution plot after applying Log Transformation.

B. Weighted Regression

Weighted regression techniques were explored for linear models like Ridge and Lasso to give more importance to certain features, such as director popularity and movie budget, which were found to have a strong correlation with IMDB scores.

VI. Results

The Mean Squared Error (MSE) was used to evaluate model performance. After applying the best-performing regression model (XGBoost), the model's predictions for the test data showed a significant improvement compared to the baseline models. The final model achieved an average MSE of 1.2, indicating a good fit for predicting IMDB scores. The two-stage approach, using classification to identify movies with a likely high or low IMDB score, followed by regression for more accurate predictions, led to an overall reduction in prediction error.

$$\text{RMSE} = \sqrt{\frac{1}{n}\sum_{i=1}^{n}(y_i - \hat{y}_i)^2} \tag{3}$$

$$R^2 = 1 - \frac{\sum_{i=1}^{n}(y_i - \hat{y}_i)^2}{\sum_{i=1}^{n}(y_i - \bar{y})^2} \tag{4}$$

	RMSE-Tr	RMSE-Te	RSq-Tr	RSq-Te	Accuracy
Lin. Reg.	0.119	0.120	0.411	0.387	95.43%
Dec. Tree	0.029	0.049	0.716	0.148	97.0%
Rand. For.	0.014	0.035	0.925	0.565	97.74%
KNN	0.040	0.049	0.443	0.159	96.74%
Lasso	0.044	0.043	0.351	0.346	97.15%
Ridge	0.044	0.043	0.351	0.346	97.15%
XG-Boost	0.004	0.033	0.991	0.611	97.86%

TABLE I
Model Comparison

VII. Conclusions and Future Work

This project demonstrated the feasibility of using machine learning models to predict IMDB scores of movies. By leveraging both regression and classification models, the project achieved meaningful improvements in prediction accuracy. For future improvements, additional data sources like social media metrics, box-office earnings, and movie reviews could further enhance the model's predictive power. Incorporating deep learning techniques like neural networks may also help capture more complex patterns within the data, especially for predicting non-linearities in IMDB scores.

References

[1] R. Sharda and D. Delen, "Predicting box-office success of motion pictures with neural networks," *Expert Systems with Applications*, vol. 31, no. 3, pp. 481-490, 2006.

[2] M. Choudhury and S. Gaonkar, "Predicting movie success using machine learning," *Journal of Data Science*, vol. 16, no. 2, pp. 95-110, 2018.

[3] L. Breiman, "Random forests," *Machine Learning Journal*, vol. 45, no. 1, pp. 5-32, 2001.

[4] I. Goodfellow, Y. Bengio, and A. Courville, *Deep Learning*, MIT Press, 2016.

[5] T. Chen and C. Guestrin, "XGBoost: A scalable tree boosting system," in *Proc. 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, San Francisco, CA, USA, 2016, pp. 785-794.

[6] S. Saurav, "IMDB Score Prediction for Movies," Kaggle, Available: https://www.kaggle.com/code/saurav9786/imdb-score-prediction-for-movies, 2023.