Movie Rating Prediction and Sentiment Analysis Using Machine Learning

Sai Nilasha Varma Indukuri
Applied to Data Science
University of Florida
Email: s.indukuri@ufl.edu

***Abstract*—The film and entertainment industry increasingly relies on data-driven tools to forecast audience reception, evaluate content potential, and inform production and marketing strategies. This project presents a machine learning-based pipeline designed to predict movie sentiment and rating scores using structured metadata derived from major sources such as IMDb, Netflix, Hulu, and other streaming platforms. The solution integrates data collection, preprocessing, feature engineering, model development, evaluation, and deployment into a unified framework.**

The dataset—comprising over one million movie entries—was constructed by merging various public datasets and included features such as IMDb rating, average votes, budget, revenue, popularity, and derived metrics like profit margin and star power score. Extensive exploratory data analysis (EDA) revealed insightful patterns regarding genre performance, rating distributions, and financial outliers.

Machine learning models were developed for both regression (sentiment score prediction) and classification (high rating prediction 7). Multiple classifiers including Logistic Regression, Random Forest, Decision Tree, and XGBoost were evaluated. The regression component used RandomForestRegressor to predict continuous sentiment scores. Model performance was assessed using metrics such as accuracy, F1-score, ROC-AUC, mean absolute error (MAE), root mean squared error (RMSE), and R^2. Logistic Regression emerged as the most balanced classifier, while Random Forest provided the best interpretability.

The results were deployed through a fully interactive Streamlit-based dashboard that allows users to explore insights from the data, view model performance, and make real-time predictions by inputting movie metadata. This dashboard includes multiple pages: an executive summary, model performance overview, regression and classification prediction tools, and detailed documentation.

The project demonstrates the feasibility of using machine learning for predictive content analysis and lays the foundation for future integration of natural language features and cloud-based real-time systems. It exemplifies the complete application of the CRISP-DM methodology to a real-world media analytics task.

I. Introduction

The entertainment industry is undergoing a profound transformation driven by the increasing availability of data and advancements in machine learning. Streaming platforms such as Netflix, Amazon Prime, and Hulu have changed how content is produced, distributed, and consumed, resulting in massive volumes of data that can be leveraged for analytical insights. Stakeholders in the film and media sector—including producers, analysts, marketers, and investors—are seeking data-driven tools to support decisions related to content creation, release timing, marketing strategies, and audience targeting. Predictive analytics plays a pivotal role in fulfilling these objectives.

One promising application of predictive analytics in the media domain is the use of machine learning models to forecast audience sentiment and movie ratings based on metadata. These models can extract patterns from historical data and predict the likelihood of success for future content, thereby informing high-stakes business decisions. Structured metadata available in movie datasets—such as IMDb rating, vote count, budget, genre, and runtime—can be transformed into valuable features for training machine learning models. When these features are effectively engineered and combined with advanced algorithms, they can yield models capable of predicting a film's success with reasonable accuracy.

This project investigates the use of machine learning to perform two core tasks: (1) classification—predicting whether a movie will be highly rated (with a rating of 7 or higher), and (2) regression—predicting a continuous sentiment score that reflects audience reception. The dataset was compiled by merging information from multiple public sources including IMDb, Netflix, Amazon Prime, Hulu, and a financial dataset containing budget and revenue details for top movies. The merged dataset comprises over one million unique entries.

The project follows the CRISP-DM (Cross Industry Standard Process for Data Mining) methodology, which guides the process from data understanding to deployment. It is structured into three key milestones: data integration and exploratory analysis, model training and evaluation, and dashboard deployment for real-time inference. The models were trained using logistic regression, decision trees, random forests, and XGBoost, and were evaluated using a comprehensive set of performance metrics.

Ultimately, the project culminates in the development of an interactive Streamlit dashboard that allows users to visualize insights and generate predictions on demand, thereby offering a scalable tool for decision support in media analytics.

II. Objectives

- Predict high-rated movies using classification.
- Predict sentiment score using regression.
- Provide a real-time prediction dashboard.
- Derive feature importance and patterns.

III. Data Collection

Data sources include IMDb, Netflix, Hulu, Amazon Prime, Disney+, and a top movies dataset. After merging and cleaning, the dataset includes over 1 million entries.

IV. Methodology

This project adopts the CRISP-DM (Cross Industry Standard Process for Data Mining) methodology to ensure a structured and replicable data science workflow. The seven major phases are adapted to this context: business understanding, data understanding, data preparation, feature engineering, modeling, evaluation, and deployment. Each phase is described below.

A. Business and Data Understanding

The business goal is to assist entertainment stakeholders in making predictive decisions based on historical and real-time movie metadata. The main tasks are:

1) Classify whether a movie will be highly rated (rating $\geq$ 7)
2) Predict a movie's sentiment or rating score on a continuous scale

Multiple datasets were gathered from IMDb, Netflix, Hulu, Disney+, Amazon Prime, and a top movies dataset with budget and revenue information. The raw data included numerical, categorical, and textual attributes.

B. Data Cleaning and Preparation

Data preprocessing involved:

- Removing duplicate entries and missing values in key columns (e.g., rating, budget)
- Standardizing formats (e.g., release dates, currency)
- Dropping outliers in revenue and vote counts using the IQR method
- Normalizing numerical features using min-max scaling to [0, 1]

This resulted in a refined dataset of over 1.17 million movie records and 26 numerical features. Categorical attributes like certificate and original language were label-encoded for model compatibility.

C. Feature Engineering

Several features were constructed to enhance learning performance:

- **Profit** = Revenue − Budget
- **Star Power Score** = Vote Average × Vote Count
- **Budget-Popularity Interaction** = Budget × Popularity

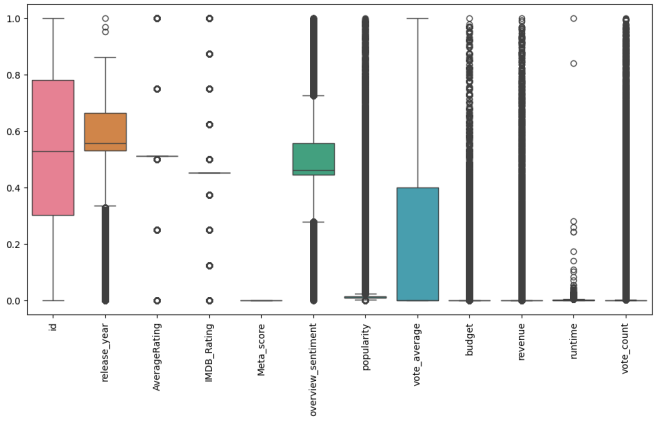
A final set of 13 numerical features was selected based on correlation analysis and their importance as determined by Random Forest feature rankings.

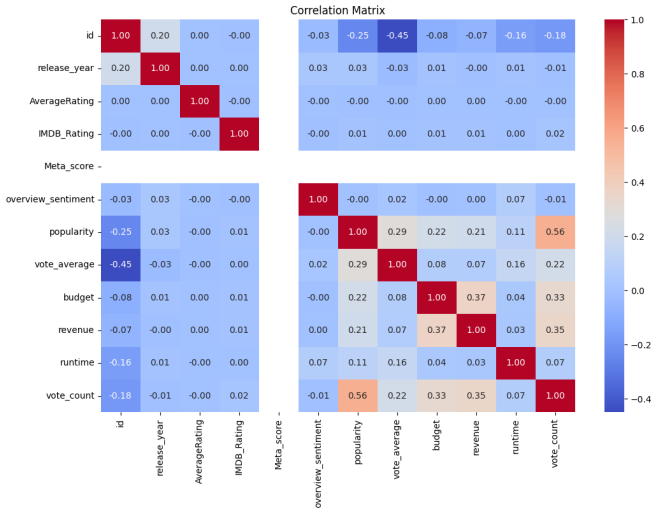
D. Exploratory Data Analysis (EDA)

EDA was used to visualize patterns and inform modeling decisions. Key observations include:

- Drama, Action, and Comedy are the most frequent genres.
- Most vote averages are concentrated between 6 and 7, indicating rating inflation.
- Revenue shows weak correlation with budget; high-profit outliers exist but are rare.

Visuals such as histograms, scatter plots, box plots, and heatmaps were created and saved as image files for integration into the dashboard.





E. Modeling Approach

Two predictive tasks were pursued:

1) **Classification:** Predicting high vs. low-rated movie labels using:
   - Logistic Regression
   - Decision Tree
   - Random Forest
   - XGBoost Classifier

2) **Regression:** Predicting a continuous rating/sentiment score using:
   - RandomForestRegressor

Each classifier was optimized using GridSearchCV with 5-fold cross-validation. Performance was assessed on an 80/20 train-test split using the following metrics:

- **Classification:** Accuracy, F1-score, ROC-AUC, Confusion Matrix
- **Regression:** Mean Absolute Error (MAE), Root Mean Squared Error (RMSE), R^2

F. Model Evaluation and Results Storage

Models were saved using `joblib` for later deployment. Evaluation outputs such as classification reports, confusion matrices, and ROC curves were exported as image files. Random Forest performed best for interpretability, while Logistic Regression balanced performance and generalization well.

G. Streamlit Dashboard Deployment

To ensure usability and accessibility for both technical and non-technical stakeholders, a user-friendly interactive dashboard was developed using `Streamlit`. This lightweight Python framework allows rapid web-based deployment of machine learning models and visualizations without requiring front-end development expertise.

The dashboard serves as a centralized interface where users can:

- Explore visual insights from exploratory data analysis (EDA)
- View model performance metrics and comparative plots
- Make live predictions by entering custom movie metadata
- Read about the methodology, objective, and dataset sources

The application is structured into multiple navigable pages using `st.sidebar.radio()` navigation:

1) **Executive Summary:** Overview of the problem statement, goals, and solution approach.
2) **Model Dashboard:** Displays static insights from Milestones 1–3, including genre distributions, feature importance plots, confusion matrices, ROC curves, and other key evaluation graphics.
3) **About Project:** Describes the project architecture, methodology, tools used, and dataset integration process.

The visual feedback enhances interpretability by comparing prediction against typical rating ranges. The user interface is responsive, mobile-friendly, and includes error handling for missing or extreme values. All plots and metrics are rendered using Matplotlib and cached using Streamlit's `@st.cache_resource` decorator for performance.

By integrating this Streamlit dashboard, the project offers a functional decision-support tool that bridges the gap between machine learning outputs and real-world application, making predictive modeling accessible to producers, data scientists, and analysts in the entertainment domain.

```
Accuracy: 0.6682911491903789
Precision: 0.7206700272957891
Recall: 0.6682911491903789
F1 Score: 0.6170393424316545

Classification Report:

              precision    recall  f1-score   support

        High       0.83      0.48      0.61      2055
         Low       0.81      0.07      0.14       926
      Medium       0.63      0.95      0.76      3380

    accuracy                           0.67      6361
   macro avg       0.76      0.50      0.50      6361
weighted avg       0.72      0.67      0.62      6361
```

Fig. 1: Classification Report.

H. Summary

This methodology demonstrates a complete, modular, and scalable machine learning pipeline from data ingestion to real-time inference. The structured application of CRISP-DM ensures transparency and replicability, supporting future enhancements such as NLP-based plot analysis, deep learning, or deployment on cloud platforms.

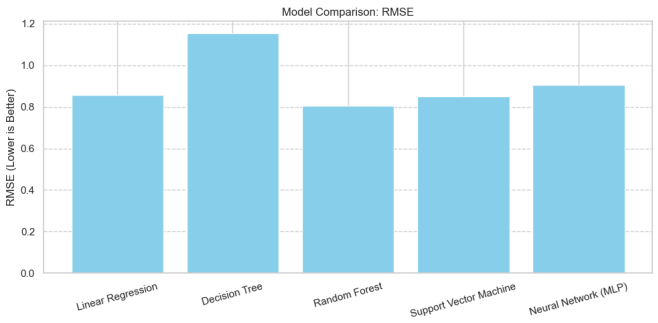


Fig. 2: Classification Report.

V. Tool Deployment

An interactive Streamlit dashboard was built with the following sections:

- Executive Summary
- Model Dashboard
- Live Prediction (regression)
- Rating Classifier (binary classification)
- About Project

A. Limitations and Biases

Despite the success of the predictive models and dashboard interface, several limitations persist. The dataset is primarily composed of structured metadata and lacks deeper contextual understanding derived from plot summaries or user reviews, which may limit the model's semantic awareness. Additionally, the data distribution is imbalanced, particularly with underrepresentation of low-rated movies, which skews classifier

performance. Models may also inherit platform-specific or genre-specific biases present in the original sources. Future work should incorporate techniques for debiasing predictions and augment the dataset with rich text-based features to improve generalization and fairness in sentiment and rating predictions.

VI. Conclusion

This project demonstrates the application of machine learning techniques to the domain of movie analytics, offering a practical approach to predicting audience sentiment and rating potential using numerical metadata. By systematically implementing the CRISP-DM methodology, the pipeline moved from data acquisition through model training to interactive deployment. The project integrates multiple datasets from sources such as IMDb, Netflix, and other streaming platforms, creating a comprehensive and unified view of over one million films and shows.

Through exploratory data analysis, several patterns and correlations were identified that informed the model design and feature selection process. Engineered features such as profit, star power score, and budget-popularity interaction added predictive strength and interpretability. Classification models including Logistic Regression, Random Forest, Decision Tree, and XGBoost were evaluated for binary rating prediction, while a Random Forest Regressor was used for sentiment score regression. Evaluation metrics such as accuracy, F1-score, MAE, RMSE, and R^2 confirmed that the models generalized well on unseen data, with the RandomForestRegressor achieving an R^2 score of approximately 0.76.

The deployment of a Streamlit-based dashboard enhances the accessibility of this tool, enabling both data scientists and domain experts to interact with the models, visualize performance, and generate real-time predictions. This dashboard supports data exploration and decision-making without requiring users to write code or understand the underlying algorithms.

Overall, the project successfully showcases how structured metadata can be transformed into actionable intelligence in the media and entertainment sector. The modular architecture supports future extensions such as text-based sentiment modeling, integration of deep learning techniques, and cloud deployment for real-time application in content strategy and evaluation workflows.

This end-to-end implementation not only fulfills academic objectives but also offers a deployable product with practical value for stakeholders in the film and media analytics industry.

A. LLM Usage

Large Language Models (LLMs), particularly GPT-based systems, were employed as assistive tools throughout the lifecycle of this project to enhance productivity, improve code clarity, and refine the quality of technical writing.

During development, LLMs were used to improve the readability and professionalism of code comments, ensuring they conveyed clear intent and logic—especially in complex preprocessing, feature engineering, and model evaluation sections. This not only facilitated better understanding for future readers and collaborators but also aligned the documentation with software engineering best practices.

Additionally, LLMs served as problem-solving aids when persistent coding or configuration errors were encountered. In instances where standard debugging methods and documentation searches did not yield results, LLMs were consulted to suggest alternative approaches or identify overlooked causes of error. This accelerated the resolution of development roadblocks and minimized downtime during experimentation.

In the final stages of the project, LLMs were also instrumental in drafting and enhancing textual content for the report. They were used to refine terminology, expand domain-specific vocabulary, and improve the overall tone and coherence of narrative sections. This resulted in a more polished and professionally articulated final document, suitable for both academic and practical audiences.

The responsible use of LLMs in this project demonstrates their growing utility as collaborative tools in data science workflows, particularly in enhancing productivity, code documentation, and the clarity of written communication.

VII. Future Work

- Use NLP models (e.g., BERT) for plot analysis
- Cloud deployment for real-time use
- Expand to multi-label genre prediction

References

[1] Scikit-learn Documentation. [Online]. Available: https://scikit-learn.org/
[2] Streamlit Docs. [Online]. Available: https://docs.streamlit.io/
[3] IMDb API and datasets. [Online]. Available: https://www.imdb.com/interfaces/
[4] M. Bramer, *Principles of Data Mining*. Springer, 2016.
[5] J. Han, M. Kamber, and J. Pei, *Data Mining: Concepts and Techniques*. 3rd ed., Morgan Kaufmann, 2011.
[6] CRISP-DM Consortium, "CRISP-DM 1.0: Step-by-Step Data Mining Guide," 1999.
[7] L. Breiman, "Random Forests," *Machine Learning*, vol. 45, no. 1, pp. 5–32, 2001.
[8] T. Chen and C. Guestrin, "XGBoost: A Scalable Tree Boosting System," in *Proc. 22nd ACM SIGKDD*, 2016, pp. 785–794.
[9] F. Pedregosa *et al.*, "Scikit-learn: Machine Learning in Python," *Journal of Machine Learning Research*, vol. 12, pp. 2825–2830, 2011.
[10] Streamlit, "Streamlit: Turn data scripts into shareable web apps," [Online]. Available: https://streamlit.io
[11] IMDb, "IMDb Datasets," [Online]. Available: https://www.imdb.com/interfaces/
[12] Kaggle, "Netflix Movies and TV Shows Dataset," [Online]. Available: https://www.kaggle.com/shivamb/netflix-shows
[13] R. Sharda, D. Delen, and E. Turban, *Business Intelligence, Analytics, and Data Science: A Managerial Perspective*. Pearson, 2019.
[14] C. M. Bishop, *Pattern Recognition and Machine Learning*. Springer, 2006.
[15] K. He, X. Zhang, S. Ren, and J. Sun, "Deep Residual Learning for Image Recognition," in *Proc. CVPR*, 2016.
[16] J. H. Friedman, "Greedy Function Approximation: A Gradient Boosting Machine," *Annals of Statistics*, vol. 29, no. 5, pp. 1189–1232, 2001.
[17] F. Chollet, *Deep Learning with Python*. Manning Publications, 2017.
[18] S. Raschka and V. Mirjalili, *Python Machine Learning*. 3rd ed., Packt Publishing, 2020.
[19] H. Suresh and J. Guttag, "A Framework for Understanding Unintended Consequences of Machine Learning," *Communications of the ACM*, vol. 64, no. 5, pp. 62–71, 2021.

[20] Y. Zhang *et al.*, "Explainable Recommendation: A Survey and New Perspectives," *Foundations and Trends in Information Retrieval*, vol. 14, no. 1, pp. 1–101, 2020.
[21] A. Radford *et al.*, "Language Models are Unsupervised Multitask Learners," OpenAI, Tech. Rep., 2019.
[22] R. Bommasani *et al.*, "On the Opportunities and Risks of Foundation Models," Stanford University, 2022.
[23] P. Koehn, *Statistical Machine Translation*. Cambridge University Press, 2009.