Netflix Content Popularity Prediction

Abstract

In the digital streaming landscape, predicting content popularity can transform how platforms like Netflix recommend, promote, and produce new content. This project explores the use of machine learning models to predict whether a Netflix title is likely to be popular based on metadata like genre, description, release year, and content type. Using a combination of text vectorization (TF-IDF) and classification algorithms (Random Forest, XGBoost, SVM), the system accurately identifies patterns in data that correlate with content popularity. The best-performing model, XGBoost, achieved ~85% accuracy, showing strong predictive capability based on available features.

Introduction

Netflix has revolutionized content consumption by offering an expansive library of movies and TV shows. However, with thousands of titles available, recommending the right content to the right audience is a growing challenge. Understanding what makes a title popular is crucial for enhancing user experience, content acquisition, and personalization strategies.

This project aims to build a supervised machine learning model that predicts the popularity of Netflix content based on its metadata. It not only explores structured features like content type, genre, and year but also processes unstructured data (like descriptions) using natural language processing.

Methodology

The project follows a typical machine learning pipeline:

1. **Data-Collection**
   Netflix metadata including title, type (Movie/TV), release year, duration, description, country, and genres.
2. **Data Preprocessing**
   - Cleaned missing or null values.
   - Text cleaning and tokenization of descriptions.
   - One-hot encoding for categorical variables like type and genre.

3. **Feature Engineering**
    - TF-IDF vectorization for the description field.
    - Combined numerical, categorical, and text-based features into a single feature matrix.
4. **Model Building & Training**
    - Multiple classification models tested: Logistic Regression, Random Forest, XGBoost, SVM.
    - Grid Search and cross-validation for hyperparameter tuning.
5. **Model Evaluation**
    - Metrics: Accuracy, Precision, Recall, F1-score, ROC-AUC.
    - Visualization of genre trends, word frequency (WordCloud), and class distributions.
6. **Model Selection**
    - XGBoost performed best with ~85% accuracy and balanced precision-recall.

Process Flow

Data Loading

↓

Data Cleaning & Preprocessing

↓

Text Feature Engineering (TF-IDF for Descriptions)

↓

Categorical Encoding (Genre, Type)

↓

Combine All Features → Final Feature Matrix

↓

Train-Test Split

↓

Model Training (XGBoost, RF, SVM, etc.)

↓

Evaluation (Accuracy, F1, ROC-AUC)

↓

Best Model Selection & Interpretation

Technology Used

Category	Tools / Libraries
Language	Python
Environment	Jupyter Notebook
Data Manipulation	Pandas, NumPy
Visualization	Matplotlib, Seaborn, WordCloud
Text Processing	Scikit-learn's TF-IDF Vectorizer
Machine Learning	Scikit-learn, XGBoost, RandomForest
Model Evaluation	Classification Report, Confusion Matrix
Deployment (optional)	Flask / Streamlit (planned)

Conclusion

This project demonstrates that Netflix content popularity can be reasonably predicted using metadata and simple machine learning techniques. Key insights:

- **Textual descriptions** play a major role in determining popularity.
- **Genres** like Drama, Action, and Documentaries tend to have higher popularity.
- **Recent content** shows more popularity trends.
- **XGBoost** outperforms other models, validating its efficiency in structured and sparse datasets.

With the right features and preprocessing, even limited metadata can provide strong predictive signals. Future work can enhance this system by integrating user ratings, watch history, and deep learning-based NLP (e.g., BERT) for even better performance.

References

1. Netflix Movies and TV Shows Dataset
   Source: Kaggle
   DATASET
2. Géron, Aurélien. *Hands-On Machine Learning with Scikit-Learn, Keras, and TensorFlow* (2nd Edition). O'Reilly Media, 2019.
   [ISBN: 9781492032649]
3. Scikit-learn Documentation
   https://scikit-learn.org/stable/user_guide.html
4. XGBoost: A Scalable Tree Boosting System
   Tianqi Chen, Carlos Guestrin. *Proceedings of the 22nd ACM SIGKDD*, 2016.
   https://arxiv.org/abs/1603.02754
5. TF-IDF Vectorizer – Scikit-learn
6. WordCloud Library in Python
7. Netflix Official Site – Content Categories and Recommendations
   https://www.netflix.com
8. Google Colab Link:
   https://colab.research.google.com/drive/1YRe1y00pEJFOY9ahFFmq1tDslMzUrZWl?usp=sharing