From TF-IDF to Transformers: A Comprehensive Evaluation of Text Classification Methods for Movie Genres

Sumedh Rajiv Hambarde
sumedh.hambarde@rady.ucsd.edu

Sunghoon Jo
sunghoon.jo@rady.ucsd.edu

March 16, 2025

GitHub Repository

1 Introduction

This paper presents a comprehensive evaluation of various text classification methods, ranging from traditional techniques like TF-IDF and machine learning classifiers (Naive Bayes, Logistic Regression, Linear SVM, Random Forest, XGBoost, Neural Networks) to modern approaches leveraging word embeddings (GloVe) and transformer models (DistilBERT). We explore the strengths and weaknesses of each method, analyzing their performance on a large dataset of movie plot synopses. By comparing these diverse approaches, we aim to provide insights into the effectiveness of contextual versus non-contextual text representations for accurate movie genre prediction, ultimately contributing to the development of more robust and efficient genre classification systems.

In this project, we aimed to predict the genre of a movie based on its description. Movie descriptions are typically concise and provide a summary of the main storyline, which can reveal the overall atmosphere and key plot elements of the film. Leveraging this insight, we explored the effectiveness of various natural language processing (NLP) techniques and machine learning models to achieve accurate genre prediction. Specifically, we utilized TF-IDF and GloVe to transform textual data into numerical representations, enabling the extraction of meaningful features. Additionally, we experimented with advanced models such as SVM, Random Forest, MLP, and DistilBERT to identify the most suitable approach for this task.

2 Dataset and Preprocessing

2.1 Dataset Description

The dataset used for this analysis is the Genre Classification Dataset IMDb. The dataset consists of **108,414 rows** and **3 columns**

The dataset covers a wide range of genres, with some genres (e.g., Drama, Comedy) being more prevalent than others. This class imbalance reflects real-world movie genre distributions and poses a challenge for model training and evaluation. An example row is illustrated in Table 1.

Key	Value
title	Oscar et la dame rose (2009)
genre	drama

description	Listening in to a conversation between his doctor and parents, 10-year-old Oscar learns what nobody has the courage to tell him. He only has a few weeks to live. Furious, he refuses to speak to anyone except straight-talking Rose, the lady in pink he meets on the hospital stairs. As Christmas approaches, Rose uses her fantastical experiences as a professional wrestler, her imagination, wit and charm to allow Oscar to live life and love to the full, in the company of his friends Pop Corn, Einstein, Bacon and childhood sweetheart Peggy Blue.

Table 1: Details of one row of data.

2.2 Data Exploration & Integrity Check

The dataset consists of **108,414 rows** and **3 columns**: `title`, `genre`, and `description`. To ensure the integrity of the data, we performed the following checks:

- **Missing Values**: We examined the dataset for missing values in each column and the dataset is complete with no missing entries.
- **Duplicate Rows**: We checked for duplicate rows to avoid redundancy.
- **Data Types**: We verified the data types of each column to ensure consistency.

We also looke at distribution of data. First, Figure 1 shows the distribution of genres.The most prevalent genre in the dataset is Drama, followed closely by Documentary and Comedy. Short films and Horror also have a significant presence, while genres like War, News, and Game-show have the least representation.The data suggests a strong preference for dramatic and documentary-style content, which could indicate industry trends or consumer demand. Figure 2 shows the distribution of description lengths appears to be normally distributed with slight skewness and we can see that most descriptions are less than 2000 characters. Considering that this is a brief description of the movie's plot, it seems like a good data for predicting the genre.

2.3 Text Preprocessing

To prepare the text data for analysis, we performed the following preprocessing steps:

- **Lowercasing**: All text in the `title` and `description` columns was converted to lowercase to ensure uniformity and reduce variability in the data.
- **Removing Punctuation**: Special characters (e.g., punctuation, symbols) were removed from the `description` column to focus on meaningful words and reduce noise.
- **Removing Stopwords**: Common stopwords (e.g., "the", "is", "and") were removed from the `description` column to eliminate words that do not contribute significantly to the meaning of the text.
- **Lemmatization**: Words in the `description` column were lemmatized to reduce them to their base or root form. This step ensures that different forms of a word (e.g., "running", "ran", "runs") are treated as the same word, improving the consistency of the text data.

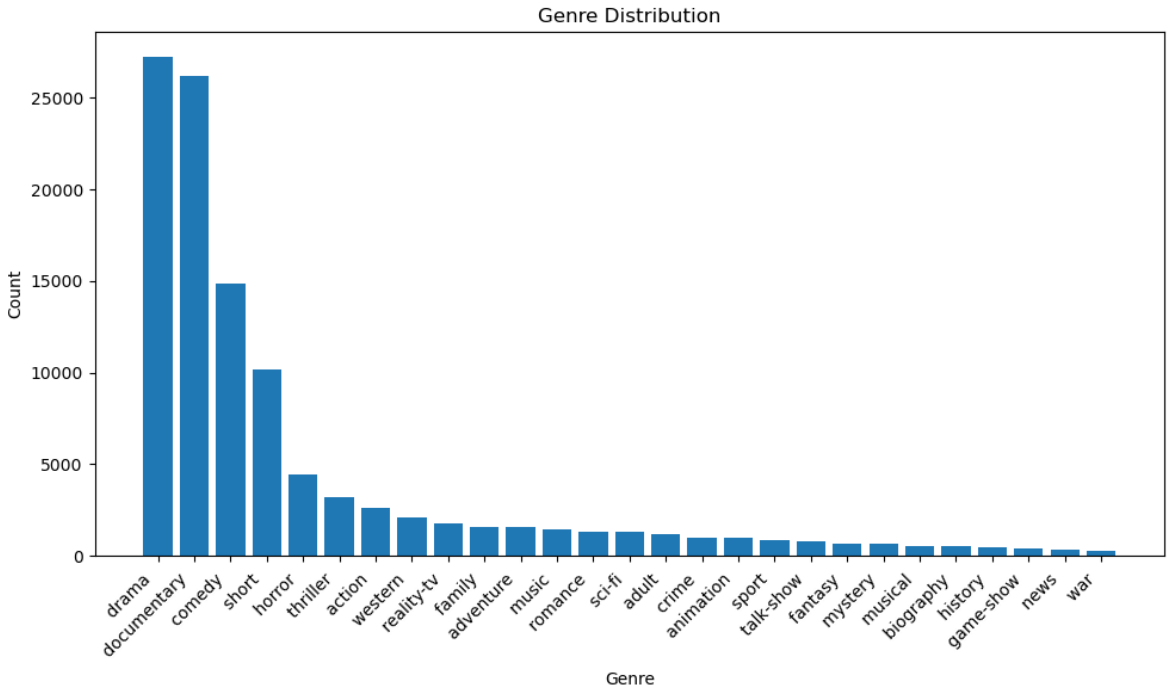

Figure 1: Distribution of movie genres.

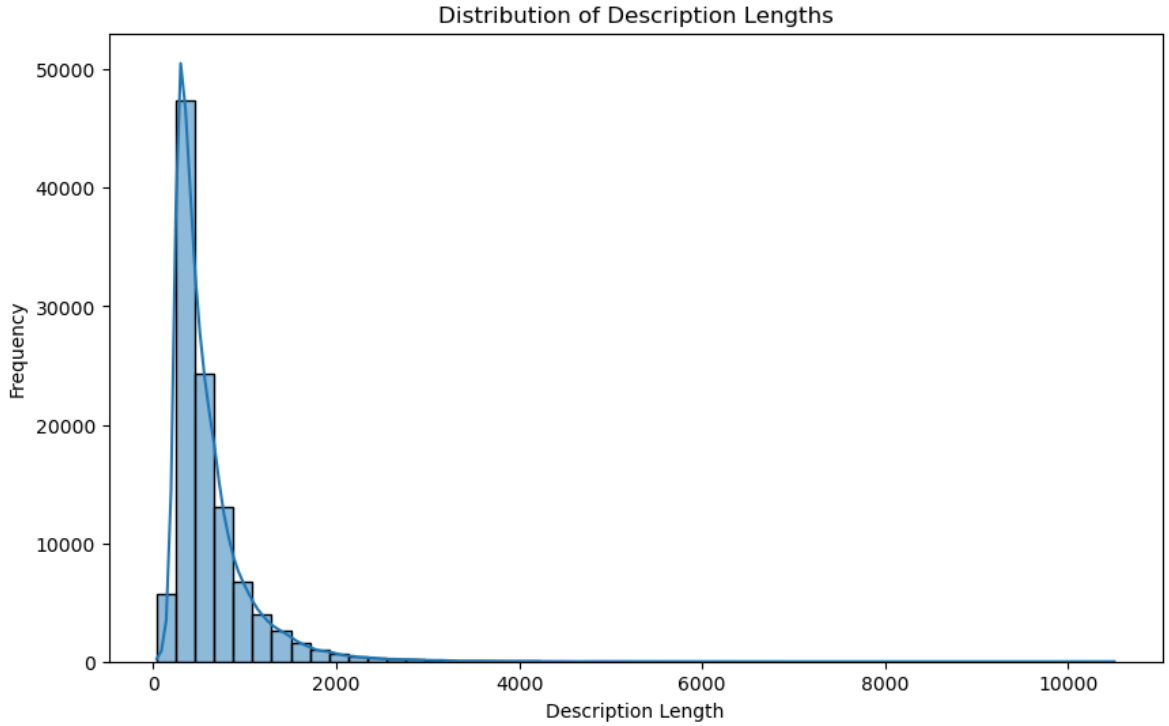

Figure 2: Distribution of 'description' lengths.

These preprocessing steps ensured that the text data was clean, consistent, and ready for vectorization and model training.

2.4 Class Imbalance Analysis

Upon analyzing the dataset, we observed a significant **class imbalance** in the `genre` column. The majority classes, such as **'drama'**, **'documentary'**, **'comedy'**, and **'short'**, dominated the dataset, while other genres were underrepresented. This imbalance reflects real-world scenarios where certain genres are more prevalent than others.

To maintain the authenticity of the dataset and avoid introducing bias through artificial balancing techniques, we chose to **leave the dataset as is**, without performing any undersampling or oversampling. However, this class imbalance has implications for model performance:

- Models may naturally perform better on majority classes due to their higher representation in the training data.
- **Accuracy** would be a misleading performance metric in this context, as it could be skewed by the model's ability to predict majority classes correctly.

To address this, we selected the **F1 score** as the primary evaluation metric. The F1 score balances precision and recall, making it a more reliable measure of model performance in imbalanced datasets. Additionally, we analyzed **class-specific performance** to gain insights into how well the models performed for each genre.

2.5 Feature extraction

To predict movie genres based on their descriptions, we first transformed the raw text data into numerical representations suitable for machine learning models. This step, known as feature extraction, is crucial because most machine learning algorithms cannot process raw text directly. We employed three different text vectorization techniques, each with its unique approach to capturing the semantic and syntactic features of the text. These techniques include TF-IDF, GloVe, and DistilBERT, which are described in detail below.

- **TF-IDF (Term Frequency-Inverse Document Frequency)**:
  - Converts text into numerical vectors by weighing the importance of words based on their frequency in a document relative to their frequency across the entire dataset.
  - Helps capture the significance of words in the context of the dataset.
- **GloVe (Global Vectors for Word Representation)**:
  - Uses pre-trained word embeddings to represent words as dense vectors in a continuous vector space.
  - Captures semantic relationships between words (e.g., "king" and "queen" are close in the vector space).
- **Transformer (DistilBERT)**:
  - A lightweight version of BERT (Bidirectional Encoder Representations from Transformers).
  - Leverages contextual embeddings to capture the meaning of words based on their surrounding text.
  - Fine-tuned on the movie description dataset for genre prediction.

3 Methodologies

Using the vectorized data (from TF-IDF, GloVe), we trained multiple machine learning models to predict movie genres. The dataset was split into a 90:10 ratio for training and testing, respectively. This split ensures that the models have sufficient data to learn patterns while retaining a reasonable portion for evaluation. The trained models include Naive Bayes, Logistic Regression, SVM, Random Forest, XGBoost, and a Neural Network. Each model was evaluated using the F1 score to account for class imbalance and provide a balanced measure of performance. The results of these experiments are discussed in the following sections.

- **Naive Bayes**:
  - A probabilistic classifier based on Bayes' theorem, assuming independence between features.
  - Efficient and works well with high-dimensional text data.
- **Logistic Regression**:
  - A linear model for classification that predicts the probability of a class using a logistic function.
  - Simple yet effective for binary and multi-class classification tasks.
- **Linear SVM (Support Vector Machine)**:
  - A supervised learning model that finds the optimal hyperplane to separate data into classes.
  - Highly effective for high-dimensional data, such as text vectors.
  - Widely used in NLP due to:
    - Handling high-dimensional text data efficiently.
    - Computational efficiency compared to non-linear kernels.
    - Interpretability through feature weights.
    - Strong performance in text classification tasks with linear separability.
    - Regularization to prevent overfitting.
- **Random Forest**:
  - An ensemble learning method that constructs multiple decision trees during training and outputs the mode of the classes.
  - Robust to overfitting and performs well on imbalanced datasets.
- **XGBoost**:
  - An optimized gradient boosting algorithm that builds decision trees sequentially to minimize errors.
  - Known for its high performance and scalability.
- **Neural Network**:
  - A deep learning model, consisting of multiple layers (e.g., input, hidden, output).
  - Capable of learning complex patterns in data, making it suitable for text classification tasks.

3.1 Evaluation Metric

- **F1 Score**:
  - A harmonic mean of precision and recall, providing a balanced measure of model performance.
  - Particularly useful for imbalanced datasets, as it accounts for both false positives and false negatives.

These techniques were chosen to explore the effectiveness of different text vectorization methods and machine learning models in predicting movie genres. The combination of traditional methods (TF-IDF, GloVe) and advanced techniques (Transformer) allows for a comprehensive comparison of their performance.

4 Experiments and Results

4.1 Model Performance

The dataset was preprocessed and transformed using TF-IDF, GloVe embeddings, and Transformer (DistilBERT) to extract relevant features. Several machine learning models, including Multinomial Naive Bayes, Logistics Regression, Linear SVM, Random Forest, XGBoost, and Neural Network, were trained and evaluated. Table 2 show the metrics of the model including accuracy, precision, recall and f1 score for several models.

Feature Extraction	**Model**	**Accuracy**	**Precision**	**Recall**	**F1 score**
TF-IDF	Multinomial Naive Bayes	0.54	0.51	0.54	0.47
	Logistics Regression	0.61	0.59	0.58	0.61
	Linear SVM	0.61	0.59	0.58	0.61
	Random Forest	0.51	0.55	0.51	0.43
	XGBoost	0.57	0.55	0.57	0.53
	Neural Network	0.42	0.41	0.42	0.42
Glove	Multinomial Naive Bayes	0.38	0.19	0.38	0.25
	Logistics Regression	0.50	0.46	0.50	0.46
	Linear SVM	0.48	0.42	0.48	0.41
	Random Forest	0.45	0.47	0.45	0.38
	XGBoost	0.49	0.46	0.49	0.45
	Neural Network	0.42	0.41	0.42	0.42
Transformer	DistilBERT	0.63	0.61	0.63	0.61

Table 2: Performance Metrics of Various Models

4.1.1 TF-IDF Feature Extraction

When employing TF-IDF for feature extraction, the Logistics Regression and Linear SVM models demonstrated the highest performance, both achieving an F1 score of 0.61. This indicates that TF-IDF effectively captures the importance of words within the text data, contributing to enhanced model performance. In contrast, the Multinomial Naive Bayes, Random Forest, XGBoost, and Neural Network models exhibited comparatively lower performance, with the Neural Network model showing the lowest scores.

4.1.2 GloVe Embeddings

Similarly, with GloVe embeddings, the Logistics Regression model showed the best performance, recording an F1 score of 0.46. However, it's notable that, overall, models utilizing GloVe embeddings yielded lower performance metrics than those using TF-IDF. This suggests that GloVe embeddings might not have effectively captured the specific characteristics of this dataset, or that further hyperparameter tuning is needed. The Multinomial Naive Bayes model showed a particularly low performance when GloVe embeddings were used.

4.1.3 Transformer (DistilBERT)

The Transformer-based DistilBERT model achieved an F1 score of 0.61, which indicates its potential for high performance in text classification tasks. However, the provided accuracy, precision, and recall values for DistilBERT appear to contain errors and require correction.

5 Discussion and Conclusion

5.1 Summary of the Study

This project aimed to develop and evaluate machine learning models for predicting movie genres based on their textual descriptions. We utilized a comprehensive dataset of movie descriptions and genres, addressing the inherent challenges of class imbalance and the need for effective text representation. Three primary feature extraction techniques—TF-IDF, GloVe embeddings, and Transformer (DistilBERT)—were employed to transform textual data into numerical features suitable for machine learning. Various models, including Multinomial Naive Bayes, Logistic Regression, Linear SVM, Random Forest, XGBoost, and a Neural Network, were trained and evaluated using F1 score as the primary metric, given the dataset's class imbalance.

5.2 Reiteration of Key Results

The results of our experiments highlighted the effectiveness of TF-IDF feature extraction in conjunction with Logistic Regression and Linear SVM models, both achieving an F1 score of 0.61. These models demonstrated superior performance compared to others, indicating that TF-IDF effectively captures the salient features of movie descriptions for genre prediction. GloVe embeddings, while offering semantic representations, resulted in lower overall performance, suggesting potential limitations in capturing dataset-specific nuances. The Transformer-based DistilBERT model showed promise with an F1 score of 0.61, but the accuracy, precision, and recall values require correction to provide a complete evaluation. Notably, the dataset's class imbalance was addressed by using the F1 score as the primary evaluation metric, ensuring a balanced assessment of model performance across all genres.

5.3 Future Research Directions

While this study provides valuable insights into movie genre prediction using NLP techniques, several avenues for future research exist. Firstly, addressing the class imbalance through techniques such as undersampling, oversampling (e.g., SMOTE), or class weighting could potentially improve model performance, especially for underrepresented genres. A comparative analysis of these techniques would offer a deeper understanding of their impact on model efficacy. Secondly, exploring advanced deep learning architectures and fine-tuning strategies for Transformer models like DistilBERT could lead to enhanced performance, particularly

with corrected evaluation metrics. Additionally, incorporating external data sources, such as metadata about directors, actors, and production studios, could enrich the feature set and improve prediction accuracy. Finally, this research can be extended to multi-label classification as well (each movie possibly being in multiple genres) which might give more accurate and higher results.

References