Pilot Study

Farida Bey
900212071

&

Youssef Ghaleb
900211976

INTRODUCTION

Predicting movie genres is a complex multi-label classification problem. The goal is to classify movies based on a variety of features, including textual features such as the overview and keywords features. In addition to numerical data (such as runtime and vote average) and lastly categorical data such as genres. The availability of multiple supervised learning models offers different trade-offs in terms of accuracy, computational efficiency, and generalization.

This report provides an in-depth evaluation of six supervised learning models implemented on the preprocessed TMDB dataset, including: **Decision Trees, K-Nearest Neighbors (KNN),Logistic Regression,Naive Bayes,Neural Networks,** and **Random Forest.** The objective is to compare the models' performance using key metrics such as the precision, recall, macro f1 score, micro f1 score, and weighted F1-scores are critical metrics also to be used to assess the different model performance across genres with varying frequencies. The best model will be selected based on performance, suitability for our classification problem, and efficiency.

1. MODEL EVALUATION

1.1 Performance Comparison Methodology

Given the multi-label nature of the problem, multiple metrics are used to capture different aspects of performance. These include precision, recall, macro F1 score, micro f1 score, and weighted F1-scores. Which will allow us to see how each model balances frequent and infrequent genres, assess their ability to generalize across genres, and highlight areas for improvement.

1.2 Decision Tree

1.2.1 Setup and Parameters

We had three parameters for the Decisions Tree; max_depth which limits the depth of the tree to avoid overfitting. criterion='gini' which is an impurity used to measure the quality of splits by minimizing impurity at each node, and lastly min_samples_split which specifies the minimum number of samples required to split a node. These parameters controlled the tree size, balancing model complexity and generalization.

1.2.2 Performance Evaluation

Table 1 Decision Tree Report

Class	Precision	Recall	F1-Score	Support
Action	0.19	0.21	0.2	2988
Adventure	0.1	0.11	0.11	1681
Animation	0.15	0.15	0.15	2603
Comedy	0.3	0.29	0.29	8162
Crime	0.11	0.12	0.12	2220
Documentary	0.28	0.27	0.27	5998
Drama	0.42	0.42	0.42	12017
Family	0.09	0.1	0.09	1683
Fantasy	0.07	0.08	0.07	1434
History	0.05	0.05	0.05	1002
Horror	0.14	0.15	0.14	2961
Music	0.09	0.09	0.09	1993
Mystery	0.06	0.07	0.06	1279
Romance	0.15	0.16	0.15	3280
Science Fiction	0.08	0.1	0.09	1360
TV Movie	0.12	0.12	0.12	1483
Thriller	0.16	0.18	0.17	3297
War	0.05	0.05	0.05	706
Western	0.05	0.05	0.05	494
Micro Avg	0.22	0.23	0.23	56641
Macro Avg	0.14	0.14	0.14	56641
Weighted Avg	0.23	0.23	0.23	56641
Samples Avg	0.24	0.24	0.22	56641

As seen in Table 1 the precision and recall values indicate a trade-off between the two metrics. For instance, the Comedy class has a precision of 0.30 and a recall of 0.29, showing a reasonably balanced performance. However, classes like Action (precision 0.19, recall 0.21) and Adventure (precision 0.10, recall 0.11) exhibit lower values, indicating a challenge in correctly identifying relevant instances while minimizing false positives. While the support column highlights potential class imbalance, as classes like Comedy and Drama have significantly more instances compared to War and Western.

Table 1.1 Decision Tree F1 Scores

Metric	Value
Macro F1 Score	0.14
Weighted F-1 Score	0.23
Micro F-1 Score	0.23

The low Macro F1 Score of 0.14 is indicative that the model struggles with underrepresented genres, while the higher weighted F1-score of 0.23 reflects better performance on frequent genres. Finally The micro F1-score of 0.23 shows the model performed reasonably well on the more frequent genres.

Pros:

- Interpretability: Easy to understand and explain.
- Handles categorical data well and provides feature importance insights.

Cons:

- Overfitting risk: Even with depth restriction, some genres are predicted poorly.
- Poor generalization: Struggled with niche genres (War, Western).

1.3 K-Nearest Neighbors (KNN)

1.3.1 Setup and Parameters

We had two parameters for the KNN algorithm; n_neighbors which uses the five closest neighbors to determine the genre, and metric='minkowski'. Minkowski distance (with p=2) generalizes to Euclidean distance. Which helped compute proximity between movies based on feature embeddings.

1.3.2 Performance Evaluation

Table 2 KNN Report

Class	Precision	Recall	F1-Score	Support
Action	0.41	0.01	0.02	2988
Adventure	0.73	0.01	0.03	1681
Animation	0.25	0	0	2603
Comedy	0.33	0.05	0.08	8162
Crime	0	0	0	2220
Documentary	0.27	0.02	0.04	5998
Drama	0.4	0.1	0.16	12017
Family	0.29	0	0	1683
Fantasy	0.4	0	0	1434
History	0	0	0	1002
Horror	0.29	0	0.01	2961
Music	0	0	0	1993
Mystery	0	0	0	1279
Romance	0.1	0	0	3280
Science Fiction	0.73	0.01	0.01	1360
TV Movie	0	0	0	1483
Thriller	0.27	0.01	0.01	3297
War	0	0	0	706
Western	0	0	0	494
Micro Avg	0.37	0.03	0.06	56641
Macro Avg	0.23	0.01	0.02	56641
Weighted Avg	0.29	0.03	0.05	56641
Samples Avg	0.05	0.03	0.04	56641

As seen in table 2 the precision and recall values for KNN show a similar trade-off as observed in the Decision Trees model. For example, the Comedy class has a precision of 0.33 and a recall of 0.05, indicating a high rate of false positives compared to true positives. Conversely, classes like Horror and Crime show very low precision and recall (both 0.00 for several metrics), suggesting significant difficulty in accurately identifying these categories.

Table 2.1 KNN F1 Scores

Metric	Value
Macro F1 Score	0.02
Weighted F-1 Score	0.05
Micro F-1 Score	0.06

The significantly low Macro F1-score of 0.02 indicates that the model performed poorly across all genres. In addition, the Weighted F1-score of 0.05 reflects poor performance even on frequent genres. Moreover, the Micro F1-score of 0.06 shows very low predictive power across the entire dataset.

Pros:

- Simple to implement and understand.
- Does not require explicit training.

Cons:

- High computational cost during inference.
- Sensitive to irrelevant features and suffers from the curse of dimensionality.

1.4 Logistic Regression

1.4.1 Setup and Parameters

A one-vs-rest strategy is used to handle the multi-label classification task, where Each class is treated as a separate binary classification problem. We have three parameters, two of which are hyperparameters. Our Hyperparameters are solver='saga' (which is optimized for large datasets with sparse inputs), max_ite (which ensures convergence of the optimization algorithm), and multiclass='ovr' (which ensures each class is modeled independently).

1.4.2 Performance Evaluation

Table 3 Logistic Regression Report

Class	Precision	Recall	F1-Score	Support
0	0.63	0.04	0.07	2988
1	0.64	0.02	0.04	1681
2	0	0	0	2603
3	0.13	0	0	8162
4	0	0	0	2220
5	0	0	0	5998
6	0.57	0.21	0.3	12017
7	0	0	0	1683
8	0.23	0	0	1434
9	0	0	0	1002
10	0.33	0	0	2961
11	0	0	0	1993
12	0	0	0	1279
13	0.43	0	0	3280
14	0.55	0.01	0.02	1360
15	0	0	0	1483
16	0.31	0	0	3297
17	0	0	0	706
18	0	0	0	494
Micro Avg	0.56	0.05	0.09	56641
Macro Avg	0.2	0.01	0.02	56641
Weighted Avg	0.27	0.05	0.07	56641
Samples Avg	0.08	0.05	0.06	56641

As seen in table 3 the precision values vary significantly across the different classes. Some classes have relatively high precision, indicating that when they are predicted as positive, they are mostly correct. In contrast, many classes have a precision of 0.00, indicating that the model fails to correctly predict any instances for these classes, leading to a high number of false positives. In addition, recall values are generally low across most classes. Suggesting that the model struggles to identify most actual positive instances.

Table 3.1 Logistic Regression F1 Scores

Metric	Value
Macro F1 Score	0.02
Weighted F-1 Score	0.07
Micro F-1 Score	0.09

Table 3.1 shows that: the model struggles with rare genres due to its low Macro F1 score, while the Weighted F1 score shows slightly better performance in predicting frequent genres the Micro F1 score indicated that most predictions were incorrect.

Pros:

- Simple and efficient for linear problems.
- Scalable to large datasets.

Cons:

- Limited by linear assumptions, failing to model non-linear relationships.
- Poor performance on rare genres.

1.5 Naive Bayes

1.5.1 Setup and Parameters

We used the Bernoulli Naive Bayes model to handle binary inputs since our genres have been one hot encoded. With a parameter of alpha=1.0, which applies laplace smoothing to avoid zero probabilities for unseen feature-genre pairs.

1.5.2 Performance Evaluation

Table 4 Naive Bayes Report

Class	Precision	Recall	F1-Score	Support
Action	0.1	0.97	0.18	2988
Adventure	0.05	0.97	0.1	1681
Animation	0.09	0.97	0.16	2603
Comedy	0.25	0.98	0.4	8162
Crime	0.07	0.98	0.13	2220
Documentar	0.23	0.87	0.36	5998
Drama	0.54	0.16	0.25	12017
Family	0.05	0.98	0.1	1683
Fantasy	0.04	0.99	0.09	1434
History	0.03	0.98	0.06	1002
Horror	0.1	0.98	0.18	2961
Music	0.06	0.98	0.12	1993
Mystery	0.04	0.98	0.08	1279
Romance	0.1	0.98	0.19	3280
Science Ficti	0.04	0.97	0.08	1360
TV Movie	0.05	0.97	0.1	1483
Thriller	0.11	0.98	0.19	3297
War	0.02	0.96	0.04	706
Western	0.02	0.99	0.03	494
Micro Avg	0.08	0.79	0.15	56641
Macro Avg	0.11	0.93	0.15	56641
Weighted A	0.22	0.79	0.22	56641
Samples Avg	0.09	0.77	0.15	56641

As portrayed in Table 4 Precision values are generally low across most classes,indicating that the model frequently misclassifies instances as positive. In addition, recall values are notably high for most classes, suggesting that the model is effective at identifying most actual positive instances, although it evidently does so at the expense of precision.

Table 4.1 Naive Bayes F1 Scores

Metric	Value
Macro F1 Score	0.15
Weighted F-1 Score	0.22
Micro F-1 Score	0.15

According to Table 4.1 it is evident that this model performed slightly better than previous models with regards to niche genres due to having a Macro F1 score of 0.15. The Weighted F1 score also shows decent performance on more frequent genres and the Micro F1 score is reflective of a better overall performance compared to logistic regression. That being said, the model still struggles with imbalanced data.

Pros:

- Fast and computationally efficient.
- Works well with high-dimensional data.

Cons:

- Unrealistic independence assumption limits its accuracy.
- Poor at modeling complex relationships between features.

1.6 Neural Network

1.6.1 Setup and Parameters

Our Neural Network contained two hidden layers with 128 and 64 neurons, using both the relu and sigmoid activation over 50 epochs.

1.6.2 Performance Evaluation

Table 5 Neural Network Report

Class	Precision	Recall	F1-Score	Support
Action	0.68	0.04	0.08	2988
Adventure	0.6	0.01	0.01	1681
Animation	0.63	0.03	0.05	2603
Comedy	0.71	0.01	0.02	8162
Crime	0	0	0	2220
Documentary	0.6	0.03	0.06	5998
Drama	0.58	0.25	0.35	12017
Family	0	0	0	1683
Fantasy	0	0	0	1434
History	0	0	0	1002
Horror	0	0	0	2961
Music	0	0	0	1993
Mystery	0	0	0	1279
Romance	0	0	0	3280
Science Fiction	1	0	0	1360
TV Movie	0.8	0	0.01	1483
Thriller	0.8	0	0	3297
War	0.5	0	0.01	706
Western	0	0	0	494
Micro Avg	0.59	0.06	0.11	56641
Macro Avg	0.36	0.02	0.03	56641
Weighted Avg	0.47	0.06	0.09	56641
Samples Avg	0.11	0.07	0.08	56641

Table 5 shows precision values to be inconsistent, with some classes performing well while many others, such as have a precision of 0.00. Indicating a failure to predict these classes, resulting in high false positive rates. Which is reiterated by the recall values being generally low, also indicating the model struggles with identifying true positives.

Table 5.1 Neural Network F1 Scores

Metric	Value
Macro F1 Score	0.03
Weighted F-1 Score	0.09
Micro F-1 Score	0.11

Strengthening the findings from table 5, table 5.1 further highlights the models failure due to its poor macro F1 score and weighted scores, showing its struggle to classify most genres even though the micro f1 score shows few correct predictions with a score of 0.11.

Pros:

- Can model non-linear relationships.
- Scalable to large datasets with sufficient tuning.

Cons:

- Overfitting is a common issue without enough data.
- Computationally expensive.

1.7 Random Forest

1.7.1 Setup and Parameters

We initialized the forest to contain 100 trees, with no restriction to tree depth and used the same gini criterion as previously done for our decision tree test.

Table 6 Random Forest Report

Class	Precision	Recall	F1-Score	Support
Action	0	0	0	2988
Adventure	0	0	0	1681
Animation	0	0	0	2603
Comedy	0	0	0	8162
Crime	0	0	0	2220
Documentary	0	0	0	5998
Drama	0.66	0.01	0.02	12017
Family	0	0	0	1683
Fantasy	0	0	0	1434
History	0	0	0	1002
Horror	0	0	0	2961
Music	0	0	0	1993
Mystery	0	0	0	1279
Romance	0	0	0	3280
Science Fiction	0	0	0	1360
TV Movie	0	0	0	1483
Thriller	0	0	0	3297
War	0	0	0	706
Western	0	0	0	494
Micro Avg	0.66	0	0.01	56641
Macro Avg	0.03	0	0	56641
Weighted Avg	0.14	0	0	56641

Table 6 confirms without a doubt that the Random Forest model was a complete failure for our problem. With almost all precision and recall values being 0 with the exception of the drama class. It is evident the model was unable to correctly classify or predict true positives for the majority of the genres in the dataset. Which is supported by the finding in table 6.1 below.

Table 6.1 Random Forest F1 Scores

Metric	Value
Macro F1 Score	0
Weighted F-1 Score	0
Micro F-1 Score	0.01

2. TOP THREE MODEL CHOICES

After analyzing the results of the models, we identified **Decision Trees (DT)**, **K-Nearest Neighbors (KNN)**, and **Neural Networks (NN)** as the top three performers. Decision Trees provided a relatively balanced performance across genres, achieving moderate recall and precision, with a weighted average F1-score of 0.23. However, the model exhibited some limitations in handling infrequent genres in addition to being prone to overfitting. KNN showed higher precision for certain genres but its recall and F1-scores were generally poor, indicating challenges in making correct predictions across most genres. While KNN is easy to implement, its scalability is poor for larger datasets, as evidenced by low recall and F1-scores. However, the Neural Network outperformed the other models in terms of weighted precision having a value of 0.47 and showed potential for handling complex datasets, given its compatibility with Word2Vec embeddings. That being said the neural Network struggled with infrequent genres, and certain classes had zero precision, indicating difficulty in confidently predicting those genres.

Weighted Averages:

- **Decision Trees:** Precision: 0.23, Recall: 0.23, F1-score: 0.23
- **KNN:** Precision: 0.29, Recall: 0.03, F1-score: 0.05
- **Neural Networks:** Precision: 0.47, Recall: 0.06, F1-score: 0.09

3. TOP MODEL

After analyzing the performance metrics and the nature of the dataset, *Neural Networks* emerged as the most suitable model due to its compatibility with text-based embeddings and its ability to capture non-linear patterns effectively. However, to prevent overfitting, a more efficient split of the training and testing datasets will be implemented in the next phase. This will ensure that the model generalizes better and improves performance across both frequent and infrequent genres.