Assignment 2 COMP30027 Report

Anonymous

1. Introduction

The report aims to investigate how certain features influence the perceived success of a movie or TV show, as measured by its IMDb score. Our analysis leverages a train and test datasets to create Machine Learning models to predict these binned IMDB scores. These datasets provide a variety of information regarding the characteristics of movie content, as well as the actors and directors cast for each of these. After extensive exploratory data analysis, all instances were deemed valuable to incorporate into the predictions of binned IMDB scores.

This report delves into the implementation of two classification models, the K-nearest Neighbours (KNN) and Bagging with Decision Trees.

2. Methodology

2.1 Text Pre-processing

It was noted that several issues with the datasets provided, including unclean strings and missing values. Initially, the 'genres' and 'plot_keywords' columns were cleaned by removing '|' and splitting the values into lists. In relation to the missing values, these seemed to be missing completely at random, examples include missing empty genre and plot keyword columns. In which, rows were removed from the training data to enhance data quality.

Features that contained numerical values greater than a range of 10000 were also selected to be normalised. This increases interpretability of the data and prevents features with larger scales dominating the learning algorithm's process. In addition to this, since a KNN classifier is being used as one of the Machine Learning Models, this model is extremely sensitive to the scale of features as it utilises distance metrics to appropriately reflect similarities between instances.

2.2 Label Encoding

The 'genres' and 'plot keywords' columns were also labelled encoded as part of text pre-processing before analysis. It converts the categorical data provided into meaningful numerical form while preserving ordinal information to maintain machine learning workflow. Each unique keyword in respective columns had value attached to it. This encoded feature was used in the KNN model.

2.3 Feature Selection

The SelectKBest and ANOVA F-test method were implemented to optimise the results of the K-NN model. It evaluates the significance of each feature in relation to the "ratio of two variances, or technically, two mean squares." (Frost, 2024) The combination of these two methods allows the best K-value to be chosen for its respective features being trained.

2.4 Exploratory Data Analysis

Visuals were created to better understand the dataset at hand.

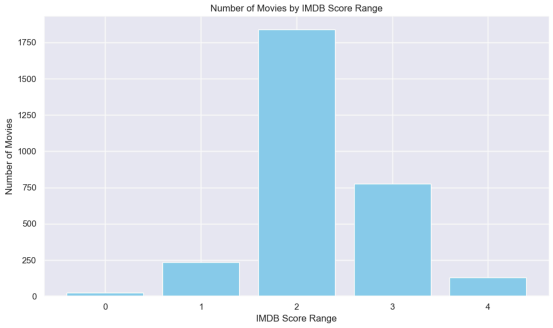

Figure 1- Bar graph depicting the number of movies and their binned IMDB score representation.

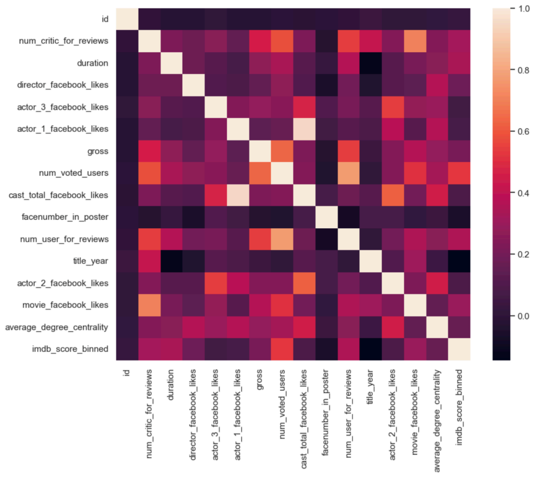

Figure 2- Heatmap demonstrating the correlation between features in the dataset.

3. Results

The K-NN model created which used label encoded genres, movie_facebook_likes, facenumber_in_poster, num_critic_for_reviews and gross revenue features performed at an average accuracy of 62.42%. This was compared to the Zero-R model which used the most frequent label in our training set to predict the label for all instances. The accuracy of this model was 61.21%. The table below displays a summary of all performance metrics ran on the K-NN model built.

Performance Metric	Average Score
Accuracy	0.6242
Precision	0.5211
Recall	0.6242
F1 Score	0.5252

Table 1- Performance Metrics for the KNN algorithm after 5-fold cross-validation

The bagging model that utilises a Decision Tree base classifier was shown to have an average score of 71.05% (Kaggle submission was slightly lower). Normalised features were only used to create this model. In comparison with the KNN model, this model showed superiority in all performance metrics demonstrating that it was a better choice model to predict based on the dataset.

Performance Metric	Average Score
Accuracy	0.7105
Precision	0.6918
Recall	0.7105
F1 Score	0.6964

Table 2- Performance Metrics for the Bagging Model

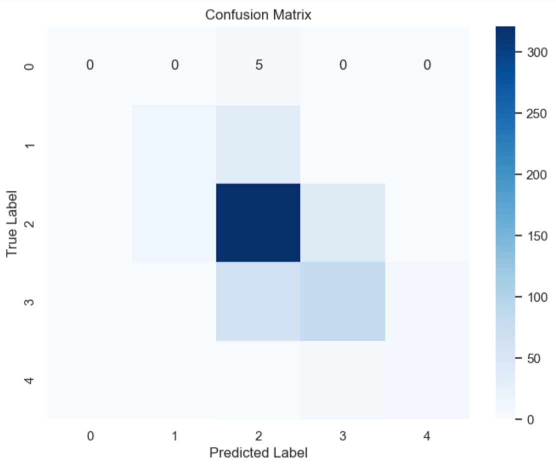

Figure 3- Confusion Matrix based on the Bagging Model

4. Discussion

Based on these results, the Bagging Model using a Decision Tree base classifier outperformed the K-NN model across various performance metrics.

4.1 The K-NN model

The K-nearest neighbour model was designed to be a classification model based on initial data evaluation. “the algorithm assigns the most common class label among the K neighbours as the predicted label for the input data point” (Srivastava, 2024). This simplicity and effectiveness allow easy, interpretable results allowing in model transparency. Note the features selected and k value of 28 was through SelectKBest and the ANOVA F-test, optimising the K-NN model further. This was proven to be even more important due to the high dimensional dataset provided where irrelevant features can negatively impact model performance. However, based on performance metrics the K-NN model did not perform well. An accuracy improvement of 1% from the Zero-R model and F1 Score of 52.5% was the lowlight of results. This could be explained through two main factors. First, feature engineering could be improved through the method of principal component analysis (PCA).

By reducing the dimensionality further, it can help by capturing the most variance in data with fewer features. In addition to this, a better model choice would probably be more suited to the characteristics of this data.

4.2 The Bagging Model

An ensemble model created through decision trees as its base classifier handles classification tasks by averaging multiple decision tree predictions trained on different bootstrap samples of the data. It is "particularly effective in reducing variance and overfitting, making the model more robust and accurate, especially in cases where the individual models are prone to high variability" (Awan, 2023). Normalised features were selected as features for this model, particularly to increase numerical stability. Although decision trees individually may not benefit as much to this change, the bagging method receives the same scale of data that reduces complexity and lead to cohesive predictions.

In comparison to the K-NN model, this model created had an improvement of 9% accuracy from the Zero-R model of K-NN. There were also all-round improvements in the performance metrics, and a significant difference of 20% in the models' F1 Score. This can be further supported through a confusion matrix where true labels were predicted most of the time. Bootstrap Aggregating leverages the strength of base classifiers like Decision Trees to capture complex relationships in the data, making it a versatile and powerful technique in ensemble learning. This also allows for parallel processing and efficient use of computational resources making it realistic to utilise in a real-world environment. On the other hand, overfitting with deep trees is the main issue at hand. Individual trees may overfit their respective bootstrap samples, which decreases maximum accuracy. Greater amounts of decision trees can also impact direct interpretability of the data. Limiting maximum depth of decision trees, introducing pruning techniques to remove less important branches are some future improvements to be considered to reduce the risk of overfitting.

5. Conclusion

The K-nearest neighbours and a Bagging ensemble model with decision trees used as base classifiers has been evaluated for its effectiveness to predict binned IMDB rating scores. The K-NN model that leveraged SelectKBest and ANOVA F-test demonstrated simplicity and transparency in its results. But the Bagging model's ability to generalise unseen data well was evident through its improvement in all performance metrics. In the future, further enhancements for the K-NN model can utilise principal component analysis to capture essential data variance. Limiting decision tree depth and pruning techniques are also strategies to implement for the Bagging ensemble model to address overfitting concerns in the future. In essence, due to the high dimensionality of the provided data the Bagging model is the preferred choice to achieve optimal and predictive accuracy of binned IMDB score rating.

References

Srivastava, T. (2024, January 4). *A complete guide to K-Nearest Neighbors (updated 2024)*. Analytics Vidhya. https://www.analyticsvidhya.com/blog/2018/03/introduction-k-neighbours-algorithm-clustering/

Singh, S. (2023, July 19). *Title: Label encoding and one-hot encoding for data preprocessing*. LinkedIn. https://www.linkedin.com/pulse/title-label-encoding-one-hot-data-preprocessing-shivani-singh#:~:text=To%20prevent%20biases%20from%20being,also%20be%20taken%20into%20account.

Awan, A. A. (2023, November 20). *A guide to bagging in machine learning: Ensemble method to reduce variance and improve accuracy*. DataCamp. https://www.datacamp.com/tutorial/what-bagging-in-machine-learning-a-guide-with-examples

Bajaj, A. (2023, December 8). *Performance metrics in machine learning [complete guide]*. neptune.ai. https://neptune.ai/blog/performance-metrics-in-machine-learning-complete-guide

Frost, J. (2024, April 7). *How F-tests work in analysis of variance (ANOVA)*. Statistics By Jim. https://statisticsbyjim.com/anova/f-tests-anova/