# Movie Rating Machine Learning Models

# Introduction

**Problem Description:**

In the digital media and entertainment industry, movie ratings have significant affects for content creators, and platforms aiming to personalize user experiences. Accurate predictions can enhance viewer satisfaction by guiding marketing strategies and recommending viewer with content of their preferences.

The core focus of this research is to evaluate the effectiveness of different machine learning algorithms in predicting movie ratings based on text features compared with numerical features. The specific question guiding this research is: **"How does the utilization of pre-processed text features impact the performance of machine learning models in predicting movie ratings?**

## Data set

In this study, we examined the performance of 3 supervised machine learning models: Logistic Regression (LR), Multinomial Naive Bayes (MNB), and k-Nearest Neighbours (KNN) to predict our movie ratings category.

LR was used as our baseline model trained only on numerical data. MNB and KNN was used as our two models trained on only textual data.

The 2 sets of data that were used for training and evaluating the machine learning models:

1. **Labelled dataset**
2. **TMDB\_train.csv** dataset served as the primary source for training our baseline model.
3. **TMDB\_evaluate.csv** served as the evaluation dataset for our baseline model. It is used to evaluate the performance of the models trained on the `TMDB\_train.csv` data, allowing us to assess their accuracy and generalizability in predicting movie ratings on unseen data.
4. **Pre-processed data**

For MNB and KNN, we utilized the pre-processed text features to investigate our research question. These were provided in the form of TF-IDF vectors stored in sparse row matrix files.

1. **Kaggle Test Data**
2. **TMDB\_test.csv** –This dataset was used for the final testing of our baseline model, to generate predictions on the data which would later be submitted to Kaggle.
3. **test\_concat\_tfidf.npz** - This file is specifically structured to support the MNB and KNN models, which can only make predictions on textual data.

# Literature Review:

The first piece of literature provides a review of text classification using machine learning techniques, emphasizing the utilization of Support Vector Machines (SVM) and Naive Bayes (NB) algorithms. It highlights the critical role of pre-processing methods, such as stop words removal and stemming, essential for refining text data to enhance accuracy.

The study also discusses the challenges inherent in text classification, which includes managing misclassified texts and selecting optimal features, which are crucial for developing efficient machine learning models.   
Parts of the literature where is relevant to our research, is how the research paper highlights the significance of choosing suitable algorithms and pre-processing techniques in text classification, which aligns with our objective to evaluate the effectiveness of different machine learning models in predicting movie ratings from textual versus numerical data.

This comparative analysis approach, as demonstrated in the literature, is particularly important, providing a solid framework for our study to assess and compare the performance of LR, MNB, KNN models in handling complex text data.

This research paper outlines a classification system designed to categorize many academic papers into classes with similar subjects. The system utilizes a combination of Latent Dirichlet Allocation and Term Frequency-Inverse Document Frequency to extract representative keywords and topics from papers, with K-means clustering to organize papers into similar subject groups.

The system's effectiveness is validated through empirical data from published papers in the Future Generation Compute Systems journal. The methodology proposed in this paper aligns well with our research question regarding the effectiveness of machine learning models in classifying text based on features extracted from the text data. By applying text processing techniques like TF-IDF or BoW along with clustering algorithms, this study provides a strong foundation for the viability of using NLP methods to enhance accuracy and supports our exploration of how machine learning models perform in scenarios with textual or numerical data.

# Method: Used Features

Our hypothesis will be:

"The use of text-based features will enhance the accuracy of machine learning models in predicting movie rating categories compared to models that only use numerical data."

To test this hypothesis, we can use a comparative approach using LR, which will use only numerical features comparing with MNB and KNN, which will utilize only text features processed through TF-IDF. The key metrics for comparison will include accuracy, and the confusion matrix and their bias and variances, which will help further visualize the effectiveness of each model across different rating categories.

## Used Features

**Numerical and Text Data**

With the data that we used, we employ a combination of numerical and text data to capture a comprehensive picture of each film. Numerical features such as budget, revenue, and runtime provide quantifiable metrics of a movie's market performance and audience engagement, which are complemented by popularity scores derived from TMDB indicators. These features have been selected based on their established correlation with audience ratings ensuring they are reliable predictors.   
Additionally, text data from the movie’s title, overview, tagline, and production companies enrich our model with detailed descriptions that reveal the thematic and emotional context of the film, capturing nuances that numerical data might miss. Such textual analysis helps in understanding deeper audience reactions influenced by narrative elements, potentially impacting the movie's overall ratings.

# Method: ML Models and Hyperparameters

## Logical Regression

Logistic Regression was selected as the baseline model to measure the linear separability of the features and primarily due to its effectiveness in handling binary and categorical outcomes, which aligns well with the categorical nature of our rate categories. LR operates on a logistic function to predict the probability that a given input belongs to a particular category, making it ideal for binary and multi-class classification problems like movie rating predictions.

### LR - Grid Search for iterations

We utilized Grid Search to determine the optimal number of iterations (max\_iter) required for the logistic regression algorithm to converge. This was necessary to ensure that the model reaches the best possible fit to the training data without overfitting or underfitting.

The max\_iter parameter impacts the convergence of the solver in logistic regression. An insufficient number of iterations might lead to a model that hasn’t yet reached its optimal state, while too many iterations could lead to the model overfitting on noise within the training data.

## Multi Naïve Bayes

Naives Bayes is particularly well-suited for text classification tasks due to its assumption of independence among predictors. MNB is a variant of Naive Bayes designed specifically for multinomially distributed data, making it particularly suitable for text data which is typically represented as word frequency counts or TFIDF vectors.

In the context of text data, which typically involves many unique words, MNB treats each word as independent from the others, simplifying the computations significantly. This assumption, while sometimes an over-simplification of real-world relationships, allows MNB to operate efficiently with very large datasets. These characteristics makes therefore makes MNB an excellent choice for situations like movie reviews and ratings.

### MNB - Smoothing parameter for alpha

This parameter is crucial for smoothing in MNB and is used to handle the problem of zero probabilities in new data, especially in the presence of features not encountered during training.

Setting the smoothing parameter (alpha) helps in controlling the model’s sensitivity to frequent versus rare words. A higher alpha value can prevent the model from giving zero probability to unseen words, thus making the probability estimates more robust.

## K-Nearest Neighbour

KNN was chosen for this study due to its strengths in handling classification tasks where the relationship between feature vectors directly influences the label. This model classifies data based on the proximity to other data points in the feature space, making it highly effective in environments where similar instances generally correspond to similar outputs. This flexibility is crucial in text-related tasks where semantic relationships and contextual similarities play a significant role. This approach aligns well with our research question by allowing use to see how effective a similarity-based classification strategy can leverage text data for predicting movie ratings

**KNN - Changing K clusters**

The number of neighbours (k) in KNN is a critical parameter that directly influences the classification outcome. The choice of k affects the bias-variance trade-off in the model. A smaller k makes the model sensitive to noise in the training data (high variance), while a larger k can smooth out the decision boundaries, reducing the variance but increasing the bias. Optimal k is usually determined through cross-validation to balance this trade-off effectively.

# Method: Evaluation Methods

## Confusion Matrix

The Confusion Matrix provides a detailed breakdown of the correct and incorrect predictions compared against the actual realities in a matrix format, showing true positives, true negatives, false positives, and false negatives.

The matrix provides a detailed view of the model's performance across all categories, rather than just a single accuracy value. It allows us to see the types of errors like false positives and false negatives that are occurring. This is important for recognising imbalanced datasets where some classes might be underrepresented.

By further examining the confusion matrix, we can identify which specific rating categories are often confused with others, leading to possible areas for improvement in feature engineering or tuning our model.  
The Confusion Matrix directly contributes to understanding how effectively each model leverages its respective features whether it is numerical or text. This can result in learning how we can classify movies into accurate rating categories, addressing the research question by highlighting the strengths and weaknesses of each model.

## Bias and Variance

Bias and Variance are metrics used to evaluate the error introduced by a model due to over-simplicity or over-complexity. By analysing bias and variance, we can gauge whether our models are underfitting or overfitting the dataset. High bias indicates underfitting, often a sign that the model is too simplistic and not capturing the underlying patterns which is possibly under-utilizing the informative power of the text data. High variance, conversely, indicates overfitting which is a sign that the model is too complex and capturing noise rather than the signal. While the confusion matrix shows how errors are distributed across different classes, bias and variance provide insights into why these errors might be occurring. Understanding the bias and variance of models trained on different types of data helps answer the research question by indicating which type of data allows the model to generalize better without losing accuracy. This is vital for validating the effectiveness of text processing techniques like TF-IDF in enhancing model performance beyond what is achievable with just numerical data.

# Results

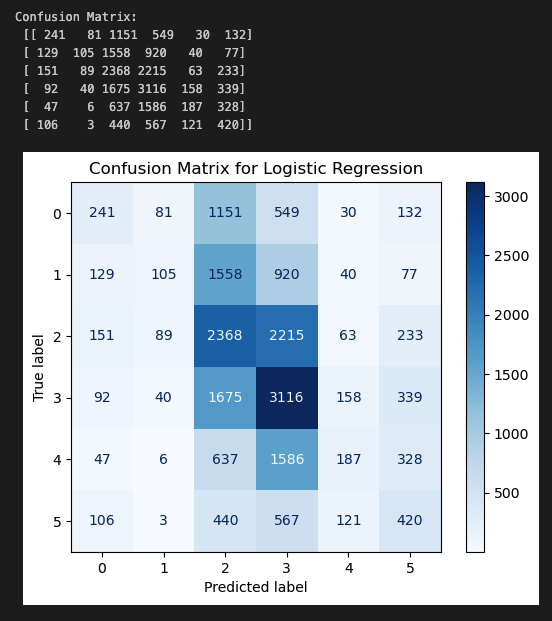
## Accuracy Table:

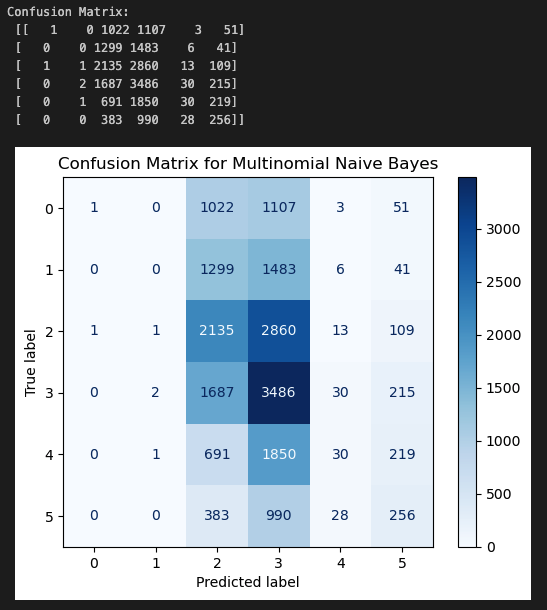
|  |  |  |  |
| --- | --- | --- | --- |
|  | **Logical Regression** | **MNB** | **KNN** |
| **Accuracy on TMDB\_evaluate.csv** | 0.3222 | 0.2959 | 0.3570 |

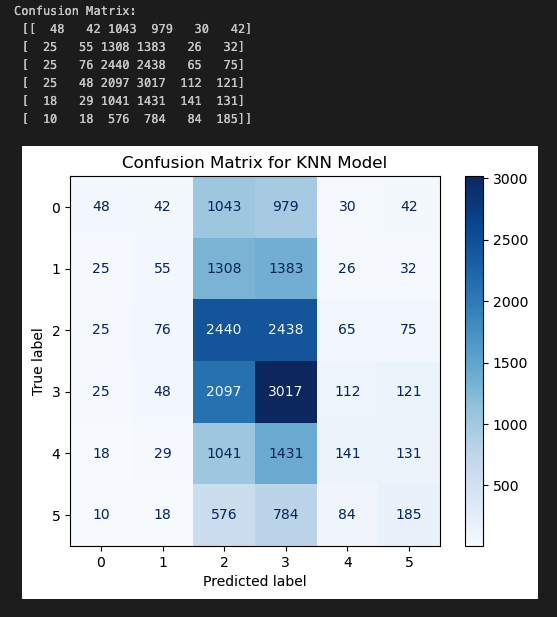
## Bias and Variance Table

|  |  |  |  |
| --- | --- | --- | --- |
|  | **Logical Regression** | **MNB** | **KNN** |
| **Expected Loss** | 0.6800 | 0.7054 | 0.7053 |
| **Bias** | 0.6780 | 0.7031 | 0.7042 |
| **Variance** | 0.0710 | 0.1485 | 0.1485 |

## Confusion Tables



****

****

## **Optimisation Results**

|  |  |  |  |
| --- | --- | --- | --- |
|  | **Logical Regression** | **MNB** | **KNN** |
| **Accuracy w/ Best iteration value = 200** | 0.3182 |  |  |
| **Accuracy w/Best alpha value = 0.1** |  | 0.2954 |  |
| **Accuracy w/ Best cluster value = 3** |  |  | 0.3725 |

# Discussion / Critical Analysis

## **Accuracy: LR vs MNB vs KNN**

Logistic Regression has an accuracy of 0.3222 and served as a baseline trained on only numerical data. As a linear model, LR assumes a linear relationship between input features and the category labels, which may be inadequate for capturing non-linear interactions inherent in complex data like movie ratings and TMDB. The limitations of LR in handling nuanced data interactions were evident, suggesting that numerical features alone might not sufficiently encapsulate the determinants of movie ratings, particularly those elements embedded within textual descriptions. Despite these constraints, LR's simplicity and speed make it a valuable baseline for comparing other machine learning models and for exploring whether textual features alone could provide superior performance.

Multinomial Naive Bayes achieved an accuracy of 0.2959, underperforming relative to the baseline. MNB assumes independence among features within each class, an assumption that rarely holds true in real-world scenarios, especially with correlated text data. This model's lower performance might also be attributed to the limitations of TF-IDF pre-processing, which is effective at weighting terms but does not address issues of data sparsity and the presence of zero frequencies that are more effectively managed by more sophisticated models.

K-Nearest Neighbours (KNN) demonstrated the best performance among the models with an accuracy of 0.357, utilizing the same TF-IDF pre-processed textual data as MNB. This indicates that KNN was better able to leverage the nuanced information within the text, likely due to its methodology of classifying based on the proximity of data points in feature space. This model has shown to effectively capture similarities between movies, suggesting that in scenarios where feature similarities are text based, KNN could outperform simpler probabilistic models like MNB.

In practical applications, however, KNN's reliance on calculating distances across the entire dataset for each prediction introduces significant computational demands, which could be prohibitive with larger datasets from real-world settings. Despite these challenges, the superior performance of KNN supports the exploration of our research question, highlighting the potential for text-based features to outperform numerical data in predicting movie ratings.

## **Confusion Matrix: LR vs MNB vs KNN**

The confusion matrices for the three models provide a detailed view of each model's performance in classifying movie ratings into specific categories. By analysing these matrices, we can extract insights about each model's strengths, weaknesses, and their relevance to the research question.

LR showed moderate effectiveness, managing to classify mid-range ratings with some success but struggled significantly with the lowest and highest ratings. This suggests that while LR can capture general trends, it may lack the nuance needed to deal with the subtleties within extreme categories of text data.

Multinomial Naive Bayes performed poorly across all categories, indicating a potential mismatch between the model assumptions like feature independence. This model's difficulty in handling the interdependent features commonly found in text data underscores its limitations in this experiment.

KNN had a slightly better distribution of correct classifications compared to MNB, indicating that its method of leveraging the structure of the data might be more suitable for text classification. However, the overall accuracy was still limited, suggesting that the high dimensionality or an inappropriate choice of 'k' could be affecting its performance.

The results from the confusion matrix of each of the models provide a significant insight into our research question regarding the effectiveness of machine learning algorithms utilizing text features for classification. These results specifically reveal the challenges and potential of text data in accurately classifying movie ratings. While each model brought a different approach to handling text data, their performances highlighted the inherent complexities of text as a feature type.

## **Bias and Variance: LR vs MNB vs KNN**

LR shows the lowest bias (0.678) among the three models, suggesting it is relatively effective at capturing the general trends in the dataset. However, it also exhibits the lowest variance (0.071), indicating that while the model is stable across different samples of the data, it potentially underfits the dataset. This lower complexity and stability make it a good baseline but perhaps too simplistic to capture the nuances in text data effectively.

MNB and KNN display higher bias (0.70315 and 0.70425, respectively) compared to LR. This higher bias suggests that both MNB and KNN might be missing underlying patterns in the dataset or that the assumptions made by these models are not perfectly suited for the complexity of text data. Both models also show significantly higher variance (0.148582 for MNB and 0.14850525 for KNN) than LR, indicating more sensitivity to fluctuations in the training dataset. This could be due to MNB’s and KNN's higher reliance on the specific distribution and representation of text data, which can vary widely between different samples.

These results are pivotal in answering our research question. The relatively lower bias of LR suggests it can generalize better across different sets of data within the scope of its simpler model structure. In contrast, the higher bias and variance in MNB and KNN indicate these models might be overfitting to nuances in the training data that do not generalize well, or they are not adequately capturing the underlying patterns necessary for more accurate predictions.

The higher variance in MNB and KNN also suggests that while these models are more sensitive to the data's specifics, they might not provide the consistency and reliability needed for practical applications like movie rating predictions.

# Conclusion

The analysis of different machine learning models utilizing numerical and text-based features has highlighted the complex dynamics of predictive modelling in rating prediction. While text-based features processed through TF-IDF demonstrated potential, the results showed the need for models that can effectively integrate and interpret both numerical and textual data.

Moving forward, it is imperative to explore hybrid models that leverage the strengths of both data types. Additionally, further research should focus on optimizing feature engineering and model parameters to enhance the accuracy of predictions. These steps will be crucial in refining our approach to understanding and predicting movie ratings, offering valuable insights for both academic research and practical applications in entertainment industry.

Bottom of Form

# Bibliography

1. Luo, X. (2021). Efficient English text classification using selected Machine Learning Techniques. *Alexandria Engineering Journal*, *60*(3), 3401–3409. https://doi.org/10.1016/j.aej.2021.02.009
2. Kim, S.-W., & Gil, J.-M. (2019). Research paper classification systems based on TF-IDF and LDA schemes. *Human-Centric Computing and Information Sciences*, *9*(1). https://doi.org/10.1186/s13673-019-0192-7