Spoiler Detection in Movie Reviews

1. Introduction

In today's digital age, user-generated reviews play a crucial role in shaping our decisions, especially when it comes to selecting movies or TV shows. However, these reviews often contain unintended spoilers that reveal critical plot points, diminishing the viewer's enjoyment and experience. This project aims to address this issue by developing a robust and accurate model to identify and flag spoilers in IMDb reviews.

By leveraging advanced Natural Language Processing (NLP) techniques and deep learning models, we can create a system that effectively distinguishes between spoiler and non-spoiler content. This not only enhances user satisfaction by preserving the element of surprise but also increases user loyalty and engagement on review platforms like IMDb.

Our approach involves comprehensive data analysis, model training, and evaluation to ensure the highest accuracy in spoiler detection. The key steps in this project include detailed exploratory data analysis, the implementation of various machine learning models, hyperparameter tuning, and thorough model evaluation. Through this process, we aim to build a model that can accurately classify reviews and help users make informed choices while avoiding spoilers.

2. Details of the Dataset:

The dataset consists of IMDb movie reviews, totaling 573,913 entries. Each entry includes seven features: review_date, movie_id, user_id, is_spoiler, review_text, rating, and review_summary. The target variable is is_spoiler, indicating whether the review contains spoilers (True) or not (False). The data types include boolean for is_spoiler and object types for the remaining features.

Data Types

- review_date: Object (String)
- movie_id: Object (String)
- user_id: Object (String)
- is_spoiler: Boolean
- review_text: Object (String)
- rating: Object (String)
- review_summary: Object (String)

Statistical Summary

- Total Reviews: 573,913
- Unique Movies: 1,572
- Unique Users: 263,407
- Reviews Containing Spoilers: 150,924 (26.3%)
- Reviews Not Containing Spoilers: 422,989 (73.7%)

3. Exploratory Data Analysis:

A thorough exploratory data analysis (EDA) was conducted to uncover patterns, anomalies, and relationships within the dataset. Key findings include:

1. **Distribution of Spoilers in Reviews**:
   - There are significantly more non-spoiler reviews (422,989) compared to spoiler reviews (150,924), indicating class imbalance.
2. **Distribution of Review Lengths**:
   - Review lengths vary widely, with most reviews between 500 and 2000 characters long.
3. **Distribution of Ratings**:
   - Ratings are distributed across the scale, with a peak at the highest rating of 10. Ratings of 8 and 9 are also common, while lower ratings are less frequent.
4. **Word Cloud Analysis**:
   - Spoiler reviews often mention specific plot details and characters, whereas non-spoiler reviews focus on general aspects like performances and overall experience.
5. **Review Length vs. Spoiler Status**:
   - Spoiler reviews tend to be slightly longer on average compared to non-spoiler reviews.
6. **Correlation Analysis**:
   - A moderate positive correlation (0.23) exists between review length and spoiler status, indicating longer reviews are more likely to contain spoilers. A slight negative correlation (-0.088) between rating and spoiler status suggests spoiler reviews tend to have slightly lower ratings.

Conclusion

From the EDA, we observe the following key insights:

- **Imbalance in Target Variable**: There are significantly more non-spoiler reviews compared to spoiler reviews, indicating a potential class imbalance issue.
- **Review Lengths**: Reviews vary greatly in length, with spoiler reviews generally being longer.
- **Ratings Distribution**: Higher ratings (8, 9, and 10) are more common in the dataset.
- **Common Words**: Word clouds reveal that spoiler reviews often mention specific plot details, whereas non-spoiler reviews are more general.
- **Correlations**: Longer reviews are more likely to contain spoilers, and spoiler reviews tend to have slightly lower ratings.

These insights will help guide the subsequent steps in data preprocessing and model building. For example, we might need to address class imbalance, consider review length as a feature, and account for common words in text analysis.

4. Model Training and Hyperparameter Search:

Three different machine learning models were selected and trained: LSTM, Feedforward Neural Network (FNN), and BERT. Hyperparameter optimization techniques were employed to fine-tune model performance.

1. **LSTM Model**:
    - **Architecture**: Embedding layer, LSTM layer with 128 units, and Dense output layer.
    - **Best Parameters**: Learning rate = 0.001, Batch size = 32, Epochs = 2.
    - **Validation Accuracy**: 80.35%
2. **FNN Model**:
    - **Architecture**: Two Dense layers with 512 and 256 units, followed by a Dense output layer.
    - **Best Parameters**: Learning rate = 0.001, Batch size = 32, Epochs = 2.
    - **Validation Accuracy**: 73.70%
3. **BERT Model**:
    - **Architecture**: Pre-trained BERT with a classification head.
    - **Best Parameters**: Learning rate = 1e-5, Batch size = 16, Epochs = 5.
    - **Validation Accuracy**: 98.36%

5. Model Evaluation:

The performance of the trained models was assessed using various metrics and validation techniques:

1. **LSTM Model**:
    - **Validation Loss**: 0.437
    - **Validation Accuracy**: 80.35%
    - **ROC AUC**: 0.68
    - **Confusion Matrix**: High true negatives, moderate true positives, indicating decent performance in predicting spoilers.
2. **FNN Model**:
    - **Validation Loss**: 0.576
    - **Validation Accuracy**: 73.70%
    - **ROC AUC**: 0.50
    - **Confusion Matrix**: High true negatives, zero true positives, suggesting poor performance in predicting spoilers.
3. **BERT Model**:
    - **Validation Loss**: 0.047
    - **Validation Accuracy**: 98.36%
    - **ROC AUC**: 0.98
    - **Confusion Matrix**: High true positives and negatives, indicating excellent performance.

6. Result Analysis/Future Work

Analysis

- **BERT Model**: The BERT model achieved high accuracy and AUC, indicating strong performance in identifying spoilers. It significantly outperforms the other models.
- **LSTM Model**: The LSTM model shows moderate performance with an AUC of 0.68. It struggles with recall for spoilers, indicating many missed spoiler reviews.
- **FFNN Model**: The FFNN model performs poorly, with an AUC of 0.50, suggesting it is no better than random guessing. It fails to identify any spoiler reviews.

7. Future Work

- **Address Class Imbalance**: Implement techniques such as oversampling, undersampling, or class weighting to handle the class imbalance between spoiler and non-spoiler reviews.
- **Feature Engineering**: Consider additional features such as user and movie metadata, sentiment scores, and linguistic features to improve model performance.
- **Advanced Models**: Explore more advanced models like transformers (e.g., BERT) and ensemble methods to enhance predictive accuracy.
- **Hyperparameter Tuning**: Conduct more extensive hyperparameter tuning using grid search or randomized search to optimize model performance further.
- **Cross-Validation**: Implement cross-validation techniques to ensure the model's robustness and generalization capability.