An Exploration of the Scripts from the TV Series 'Friends' and their Impact on IMDb Ratings

Group 13: Brandon Downer, Patrick Page, Ruiston Dsouza

Introduction

Friends is an American television sitcom that aired ten seasons on NBC from the year 1994 to 2004. The show starred some popular and upcoming talent including Jennifer Aniston, Courteney Cox, Lisa Kudrow, Matt LeBlanc, Matthew Perry and David Schwimmer. The show revolves around the day to day life of six individuals in their 20s and 30s living in Manhattan, New York City. The series was produced By Bright/Kauffman/Crane Productions, in association with Warner Bros. Television.

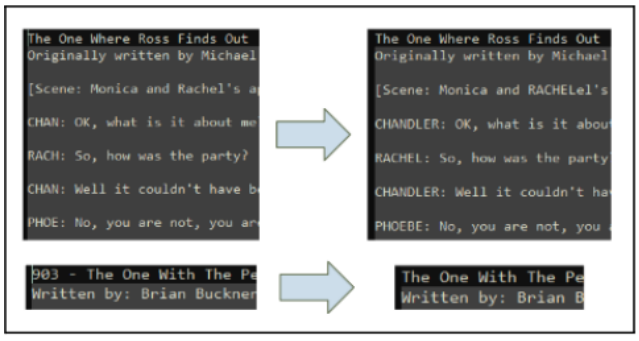
Friends ran for ten seasons, spanning a decade, and gained widespread popularity, representing a cultural phenomenon with its relatable characters, witty humor, and memorable moments that continue to resonate with viewers around the world, making it a timeless classic in the realm of television. Friends ran for ten seasons, spanning a decade, and gained widespread popularity, representing a cultural phenomenon with its relatable characters, witty humor, and memorable moments that continue to resonate with viewers around the world, making it a timeless classic in the realm of television.

Problem Description

With the availability of datasets similar to the Friends TV series, there is an opportunity for TV ad agencies to improve their marketing strategies and content generation by analyzing the popularity of certain characters, episodes, and seasons. By understanding these factors that contributed to the success of Friends, these agencies can apply these insights to other TV series of similar genre and make informed decisions about how to allocate marketing dollars for future shows, in order to keep viewers engaged in the long term.

Data Description

This Dataset contains the 228 text file of all the Screenplay Scripts and Dialogue for each episode in the FRIENDS TV Show. Each text file includes title, writer, character, change of screens and dialogue. We also used the IMDB rating dataset for the series Friends for all episode that contain the IMDB rating, Summary of the episode and total number of votes by episode



Methodology

Before any analysis could be performed on the datasets, data cleaning and preprocessing needed to be performed. Inconsistencies existed in the scripts dataset regarding how the files were formatted and how the different characters were listed. Because these inconsistencies were localized to only a few scripts, these issues were

resolved manually in the files themselves rather than in the python notebook workflow.

The data was imported and parallelized into an RDD and then converted into a PySpark dataframe. PySpark SQL was then used to obtain various features from the dataset. The original questions of the project were: *Does the amount of lines each character says in an episode have an impact on the IMDB rating of Friends? And if so, which character line quantity has the biggest impact?* Because of these questions, the line totals that each character speaks in each episode were the primary features that were extracted.

Other features were then determined that could potentially have positive or negative relationships with the IMDB score. These features include: episode director name, episode writer name, season of episode, episode number, number of scene changes, number of unique characters that feature in each episode, and the total lines overall that were spoken in each episode.

Features such as episode writer and director were hypothesized to have strong associations with IMDB scores since writers and directors often vary in talent. The scene change feature was to see if IMDB scores were associated with how often the episode changed scene locations. The unique character feature was to determine if episodes that had a lot of side characters score better than episodes that only feature the main cast.

Feature selection was performed on the set of features to reduce the amount of redundant and irrelevant variables that will later be fed into the prediction model. A combination of ANOVA and correlation tests were performed.

Feature	sum_sq	F-statistic	p-value	Correlation
scene_changes	0.824	6.388	0.0122	-0.039
unique_characters	1.783	13.826	0.0003	-0.173
total_lines	2.059	15.968	0.0001	0.210
ross_pct	0.377	2.920	0.0889	0.165
rachel_pct	0.245	1.897	0.1698	
chandler_pct	0.281	2.181	0.1412	
duration	0.139	1.078	0.3003	
phoebe_pct	0.064	0.499	0.4806	
joey_pct	0.131	1.019	0.3138	
monica_pct	0.000	0.000	0.9953	

Table 1: Feature selection performed by ANOVA and Pearson correlation test

The most significant features included the amount of scene changes in each episode, the quantity of unique characters, the total number of lines spoken, and the percentage of lines that belonged to Ross.

Once the features were narrowed down for the model, a preprocessing pipeline was created to handle the labeled data. The preprocessor consisted of the following three transformers: an indexer that converted the factor to a numeric value, a One-Hot encoder that converted the numeric value to a binary vector, and an assembler that combined the all the data together into a single feature vector.

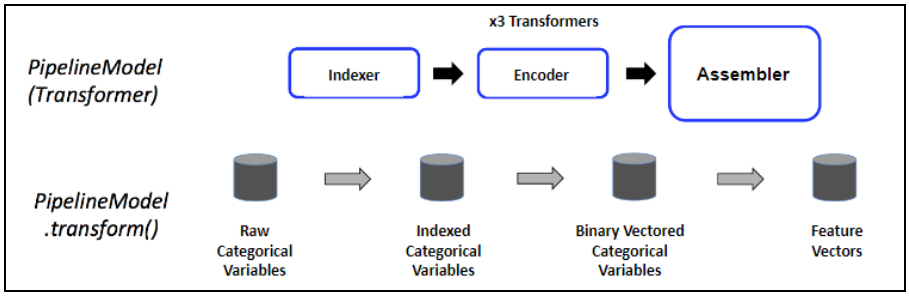


Figure 2: Preprocessor Pipeline

A final dataframe was created that contained the feature vectors and the IMDB scores. This dataframe was split into training and test data sets with 75% of the data making up the training set and the remaining 25% making up the test data set.

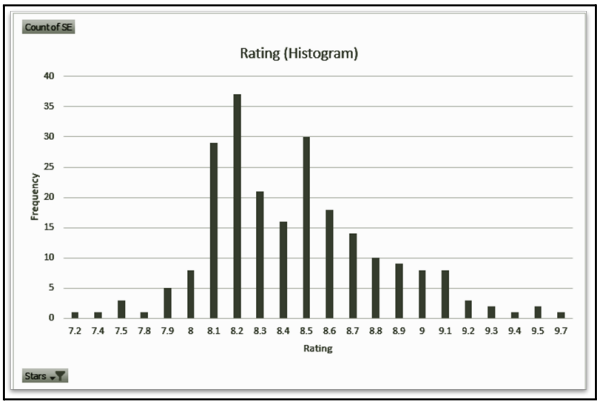
Three different PySpark MLlib machine learning models were tested in order to obtain the best IMDB prediction model. The three models tested were linear regression, random forest regression, and gradient boosted tree regression models.

Results

The GBT Regression model underperformed the linear regression model and the random forest regression model. The Linear regression model slightly outperformed the random forest regression model by having a slightly lower mean squared error and a slightly higher coefficient of determination. That being said, no model performed "well" which could be attributed to the low variation of Friends IMDB scores.

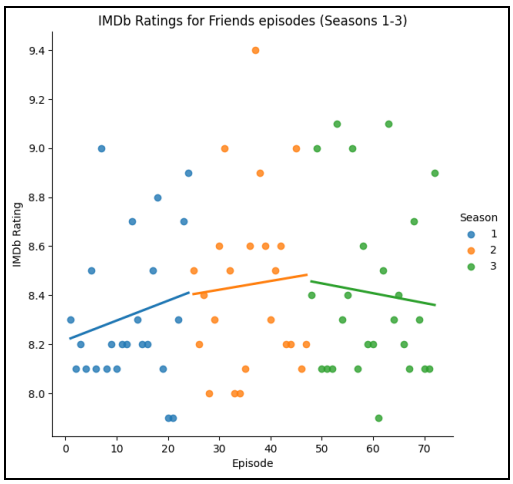
Model	**MSE**	**R-Squared**
Linear Regression	0.117	0.136
Random Forest Regression	0.118	0.128
GBT Regression	0.185	-0.367

Table 2: Evaluations of the machine learning models

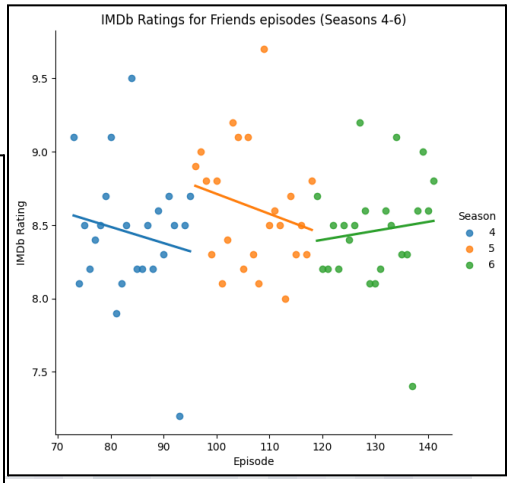


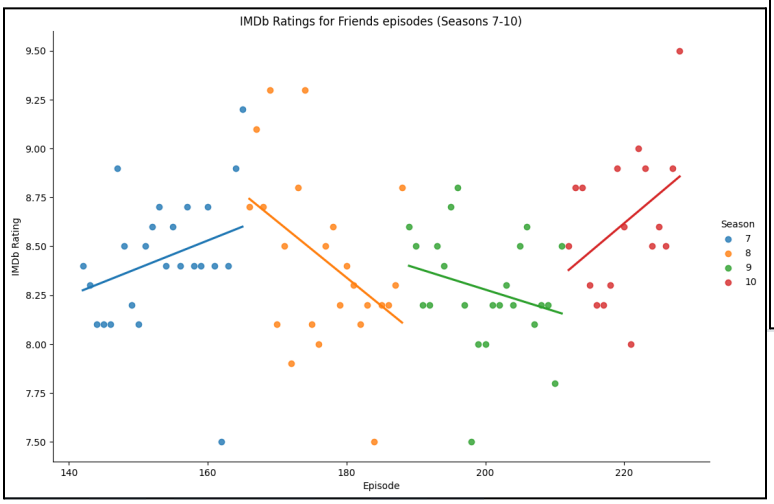
The preliminary findings of this study showed that the mean and median of the IMDB ratings for the TV show Friends were the same, indicating a nearly symmetrical distribution of ratings. However, we wanted to gain a deeper understanding of the popularity trend of the show by analyzing the IMDB ratings by episode and season. Using the seaborn[6] function, we were able to fit for regression by season and produce readable and interpretable results. The results showed that there was a trend of higher ratings for episodes towards the end of the season, and the finale episode had the highest rating, which was expected considering the popularity of the show.

The analysis also answered additional questions, such as the most popular character in Friends, which was found to be Rachel based on the summary field for most mentions by character for all seasons. However, this varied by season, indicating a dynamic popularity trend for the characters. The study also found that Kevin Bright was the most popular writer and director based on the weighted average of the number of scripts written and rating received on IMDB. Finally, the analysis revealed that S05 E14 was the most popular episode, and the series finale (S10E17) had the highest IMDB rating.



Overall, the results of this study provide valuable insights into the popularity trend of the TV show Friends and shed light on the most popular characters, episodes, writers, and directors. These findings can be used to inform future research on popular TV shows and their audience preferences.



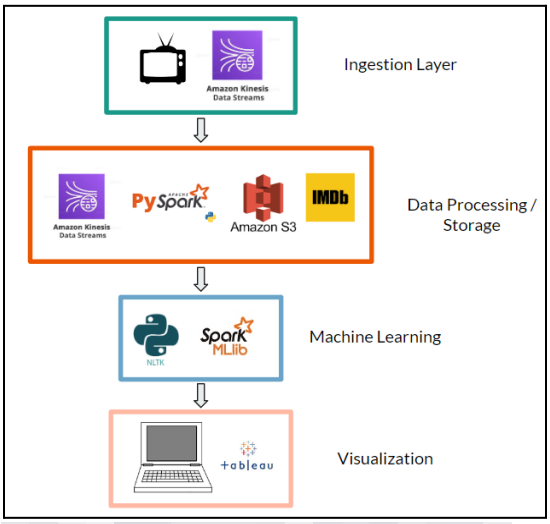


Discussion

Spark SQL is able to process large amounts of data in parallel across multiple nodes in a cluster. Additionally, Spark SQL provides an SQL interface to work with structured data in dataframes, making it easy to manipulate data using standard SQL queries. That allowed us to perform complex queries on large datasets without having to write custom code, reducing the time and effort required to perform data analysis. We learnt Spark SQL can connect data output with various data sources like HDFS, Apache Cassandra, and Amazon S3. This allows the flexibility to process data from different sources and conduct swift comprehensive analyses. Lastly, scalability of the Spark SQL to hypothetically handle live stream closed captioning during live broadcasting of episodes, the ability to process data from different sources makes it a perfect case in point in this analysis and future scope.

Additionally, Spark offers a native functionality for integrating and comparing multiple different Machine Learning algorithms into the Spark code previously described. This functionality enables users to not only process and analyze large datasets, but also extract meaningful insights and predictive capabilities, while allowing additional data to be added and evaluated without the need for further code modification.

The diagram below illustrates this theoretical cloud-based architecture for processing live-streamed closed captioned TV shows and generating insights using machine learning. It starts with the Ingestion Layer where TV shows are streamed and ingested into Apache Kafka. The Data Processing/Storage layer processes the data using PySpark, stores it in Amazon S3, and fetches additional information from IMDb. The Machine Learning layer utilizes PySpark and Spark MLlib to analyze the data and generate insights. Finally, the insights are visualized using Tableau in the Visualization layer.



References

1. Sharma, N. B. (2019). Role of OneHotEncoder and Pipelines in PySpark ML Feature - Part 2. Medium. Retrieved from https://medium.com/@nutanbhogendrasharma/role-of-onehotencoder-and-pipelines-in-pyspark-ml-feature-part-2-3275767e74f0
2. Apache Spark. (n.d.). ML Pipelines. Retrieved April 1, 2023, from https://spark.apache.org/docs/latest/ml-pipeline.html
3. B. Cruise. (2021). Friends Episode Data Analysis. Kaggle. Retrieved from https://www.kaggle.com/code/bcruise/friends-episode-data-analysis/notebook
4. Rezaghari, A. (n.d.). Friends Series Dataset. Kaggle. Retrieved April 1, 2023, from https://www.kaggle.com/datasets/rezaghari/friends-series-dataset
5. Blesson Densil, S. (n.d.). Friends TV Series Screenplay Script. Kaggle. Retrieved April 1, 2023, from https://www.kaggle.com/datasets/blessondensil294/friends-tv-series-screenplay-script
6. Seaborn Development Team. (n.d.). seaborn.lmplot. Retrieved April 1, 2023, from https://seaborn.pydata.org/generated/seaborn.lmplot.html

Appendix

Contributions

Name	**Contribution**
Brandon Downer	All areas, but with a focus on data processing and ML models
Patrick Page	All areas, but with a focus on data processing and ML models
Ruiston Dsouza	All areas, but with a focus Spark SQL and interpretation of results