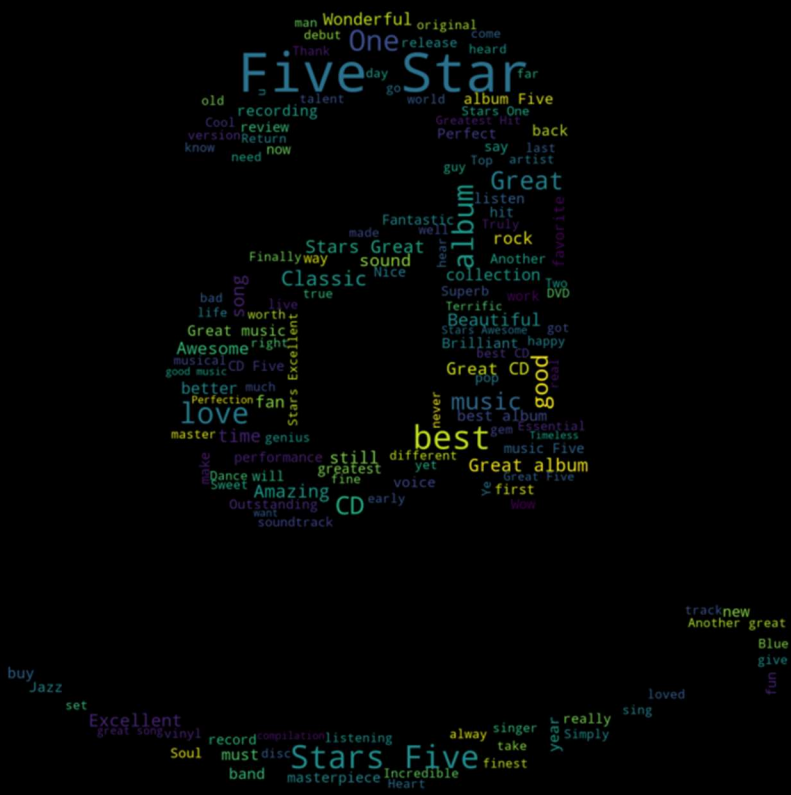


MIE1624H
Introduction to data science and analytics

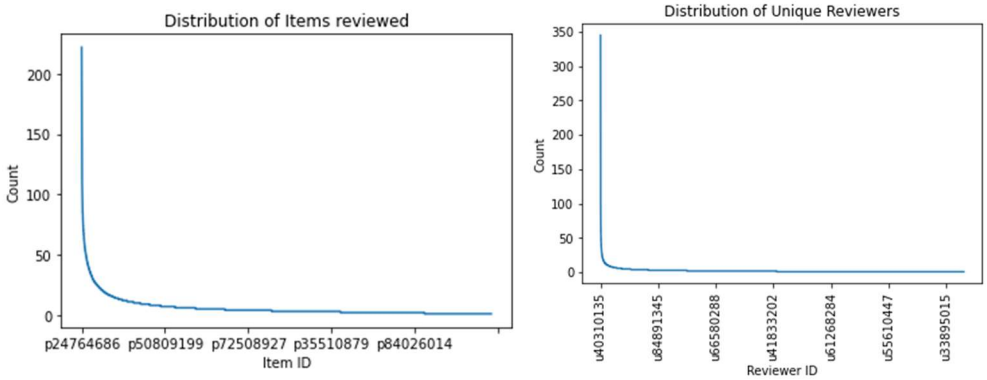
Recommender System Model

Group 21

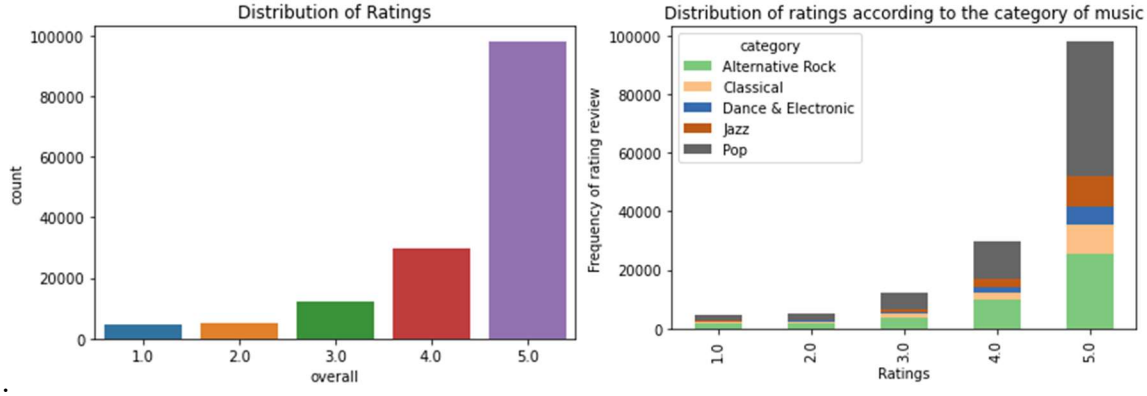
Nehel Malhotra, Ayser Choudhury, Abhi Gandhi, Dingyi Zhang, Hassan Khalid, Ruchit Shah, Siddharth Kulkarni, Syed Shah

Exploratory Analysis and Data Preprocessing

The objective of this Kaggle competition was to build a recommendation system for Amazon music. The dataset included text reviews, summary of the review, price and genre of the music which are used as model features and the overall rating given by the user which is the target variable. Since the data contains textual data, LSTM which is a Recurrent Neural Network model is used for the ratings prediction. Mean squared error is used as the evaluation metrics. There are 24,592 unique items that are reviewed where some items are given reviews multiple times and some of them are given reviews only once.

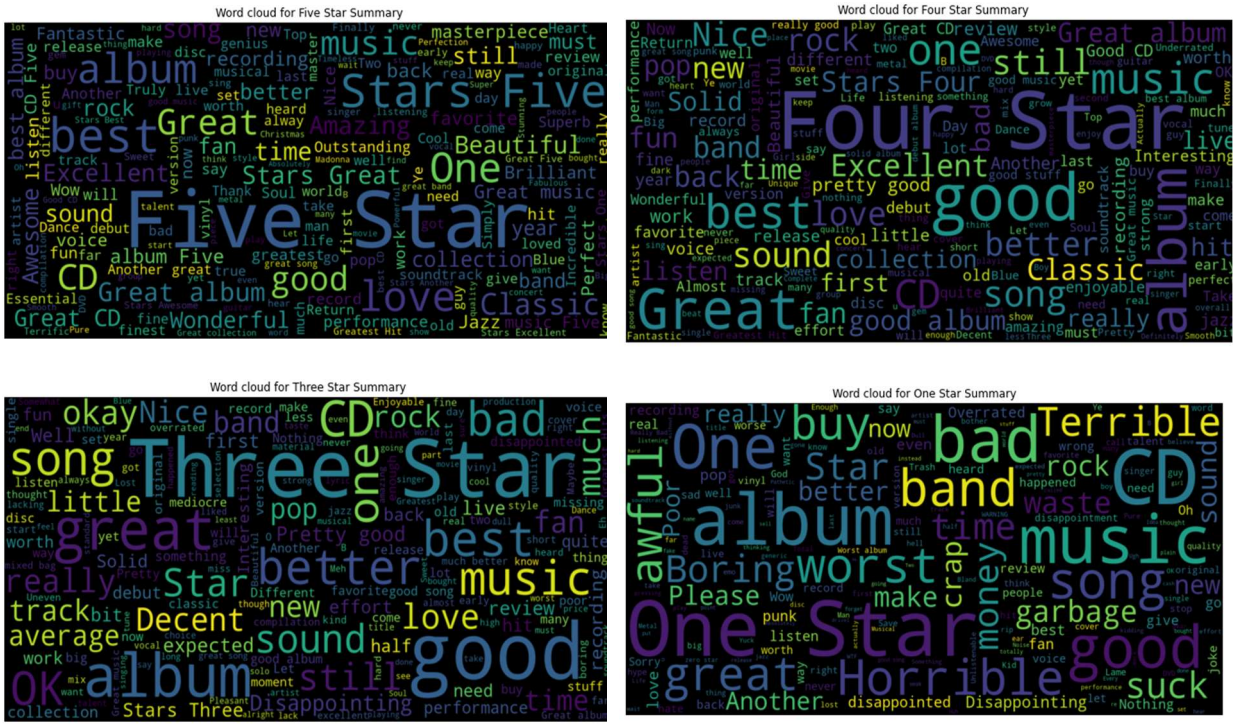


The graph on the left shows an item and their number of reviews. The graph shows that there are a very small number of items that have large numbers of reviews, majority of the items have singular reviews and that is why the curve dips downwards and stretches. Similarly, the graph on the right represents the number of reviews given by a user. A small number of users give multiple reviews, and then most reviews are singular. Since the number of reviews for items are not evenly distributed, and the majority of reviews given for items is only one, this could cause problems for prediction for such items.



The ratings for the reviews range from 1-5. As seen from the graph above, the frequency of the reviews decrease as the rating level decreases. A large number of reviews are given for 5-star review with reviews for 1 star being the lowest. Moreover, from the right figure it is observed that most of the ratings are for the genre Pop and Alternative Rock.

For part of the data cleaning, some features were deemed unimportant and would not assist in predicting the model and therefore, were dropped. These include, reviewTime, unixReivewTime, reviewHash. The reviewText and summary both were joined together as both contained similar information and could be cleaned together. The cleaning steps include; removing any hyperlinks and URLs, converting HTML to ASCII, removing all tags and punctuations, remove most common words along with stop words, (the stop words were imported from the stop words library in python), converting all text to lowercase, and finally lemmatizing the text to group all like words together e.g. playing, plays, played all would be turned to play. A word cloud for all the different ratings was generated as follows:



The word clouds contain similar words even for different ratings. This is because commonly used words like music, song, album, CD occur in all different types of rating. Since there are very few distinct words across different ratings, this could make it harder for a model to predict the sentiment of the review.

Model Implementation

Proposed Model

Long Short-Term Memory (LSTM) network is a type of RNN capable of learning order dependence in sequence prediction problems. LSTM network is capable of learning long-term dependencies. It remembers the information for long period of time eliminating the problem of handling long-term dependencies of the context in RNN.

One of the limitations of the neural network is that there is no memory associated with the model which is the problem for sequential data. The RNN solves this issue by providing feedback loop. LSTM extends the idea by creating short-term and long-term memory component [1]. Therefore, LSTM is the great tool for the time series data and the text having sequences. It stores previous timestep data and learns for prediction.

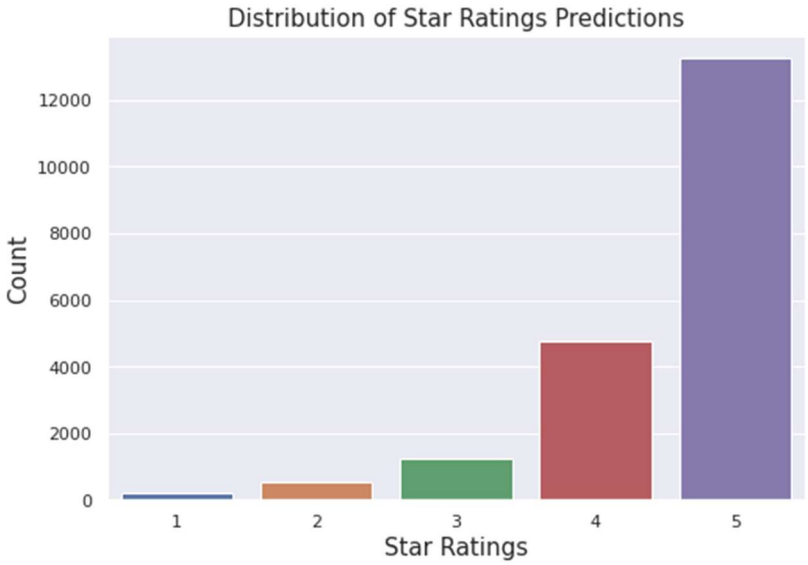
Here, word embedding is used for the model encoding scheme where words are represented by dense vectors, where a vector represents the projection of the word into a continuous vector space [2]. The GloVe method is used for learning word embeddings from text.

Tokenizer class from Keras is used to fit the training data, which converts text to sequences consistently by calling the texts_to_sequences() method on the Tokenizer class. It provides access to the dictionary mapping of words to integers in a word_index attribute. Once the reviews are tokenized, a pre-trained GloVe word embedding file is used to convert the words into vectors. The benefit of using word embeddings is that it is pre-trained and is better able to classify similar words together.

Other than the textual review, price and the genre of the music are also used as features for the neural nets model. Price is standardized using MinMaxScaler which converts it to numbers between 0 and 1. The genres are one-hot encoded.

Model Parameters

The parameters: batch size and epochs are tuned between the value (64, 128) and (10, 20) respectively using the GridSearch algorithm. The best parameters are found to be batch size 128 and epochs 10. The optimized model is fitted on the train dataset and mean squared error is used as the metrics. A final score of 0.42 is obtained on the train data and 0.49 on the validation data. The output prediction for the test dataset is found as shown in the figure.



Hyperparameter Tuning and Overfitting

Training an NLP model using Neural Networks can become very computationally expensive and hard to scale due to the model runtimes. A few of these issues can be resolved by tunning hyperparameters such as learning rate, batch size and number of epochs. Relu activation function is used as it is less expensive to run as compared to tanh or sigmoid activation functions. Hyperparameter tuning was conducted with three cross folds to limit model overfitting. Furthermore, dropout was added in the Neural Net model to randomly drop a percent of neurons to ensure that the model does not memorize the features.

Other Models Considered

Other methods compared were Logistic Regression and Multinomial Naïve Bayes. TF-IDF and word frequency were considered as features for vectorization. Maximum features were set to 3000. The Multinomial Naïve Bayes algorithm with the parameter alpha 0.01 is implemented on the training dataset. The mean squared error is found to be 1.07384. For the logistic regression algorithm, the parameter is selected as C = 1, penalty = l1, solver = saga. The mean squared error is found to be 0.65873 on the train data.

The logistic regression models used with word frequency or TF-IDF failed to provide low errors, with the initial runs showing between 1.4 and 1.8 as the MSE on the test data. Logistic regression was initially considered as it is easy to implement and very efficient to train. It does not make assumptions about distribution of classes in feature space. However, it is limited by the assumption that the dependent and independent variables are linearly correlated. [1] This can often lead to overfitting of data.

The main advantage of comparing different models is the understanding you gain from the different analysis methods underneath each model. The aim is to build a universal model that can allow us to predict values across different data sets for a given task. Some models like linear regression and Naïve Bayes only work well with a few datasets due to their individual limitations.

Results and Discussion

The LSTM model performed well, being able to predict most reviews correctly. This model signifies the ability of a neural network model to correctly predict ratings related to Amazon music reviews. The model can be further improved by using more features or a more balanced dataset.

The model worked due to some of the benefits of using regression analysis, which are:

1. The ability to examine the relative influence of multiple predictor variables by determining the relationship between independent and dependent variables. [2]
2. The ability to identify outliers, which can help fit the data better.
3. Good performance on large datasets.

References

[1] [Online]. Available: https://www.quora.com/What-are-the-advantages-of-LSTM-in-general.

[2] "How to Use Word Embedding Layers for Deep Learning with Keras," Machine Learning Mastery, [Online]. Available: https://machinelearningmastery.com/use-word-embedding-layers-deep-learning-keras/#:~:text=Keras%20offers%20an%20Embedding%20layer,represented%20by%20a%20unique%20integer.&text=The%20Embedding%20layer%20is%20initialized,words%20in%20the%20training%20datase.

[3] "Advantages and Disadvantages of Logistic Regression," geeks for geeks, [Online]. Available: https://www.geeksforgeeks.org/advantages-and-disadvantages-of-logistic-regression/#:~:text=Let's%20discuss%20some%20advantages%20and%20disadvantages%20of%20Linear%20Regression.&text=Logistic%20regression%20is%20easier%20to,it%20may%20lead%20to%20overfitti.

[4] "The Advantages & Disadvantages of a Multiple Regression Model," Sciencing, [Online]. Available: https://sciencing.com/advantages-disadvantages-multiple-regression-model-12070171.html.