**ABSTRACT**

**Amazon Alexa Reviews with Sentiment Analysis**

**INTRODUCTION**

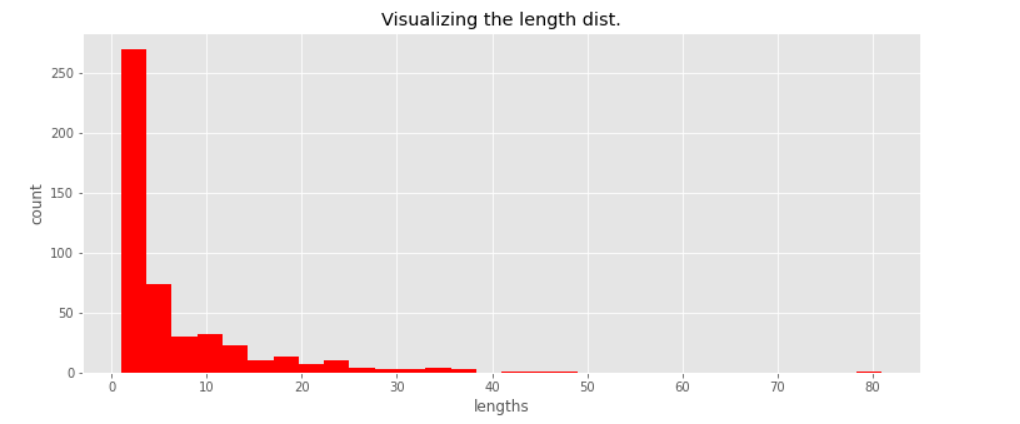
Using the dataset consists of a nearly 3000 Amazon customer reviews (input text), star ratings, date of review, variant and feedback of various amazon Alexa products like Alexa Echo, Echo dots, Alexa Firesticks etc. for learning how to train Machine for sentiment analysis. We trained a model to classify reviews into positive, negative, or neutral categories based on the text content, which can help understand customer satisfaction levels and identify areas for improvement. Also we implemented a model to predict the helpfulness of a review based on the input content. With valuable insights, we want to providing a deeper understanding of users' emotions and sentiments towards the product. And we want to help product team better get their products upgrade from the reality.

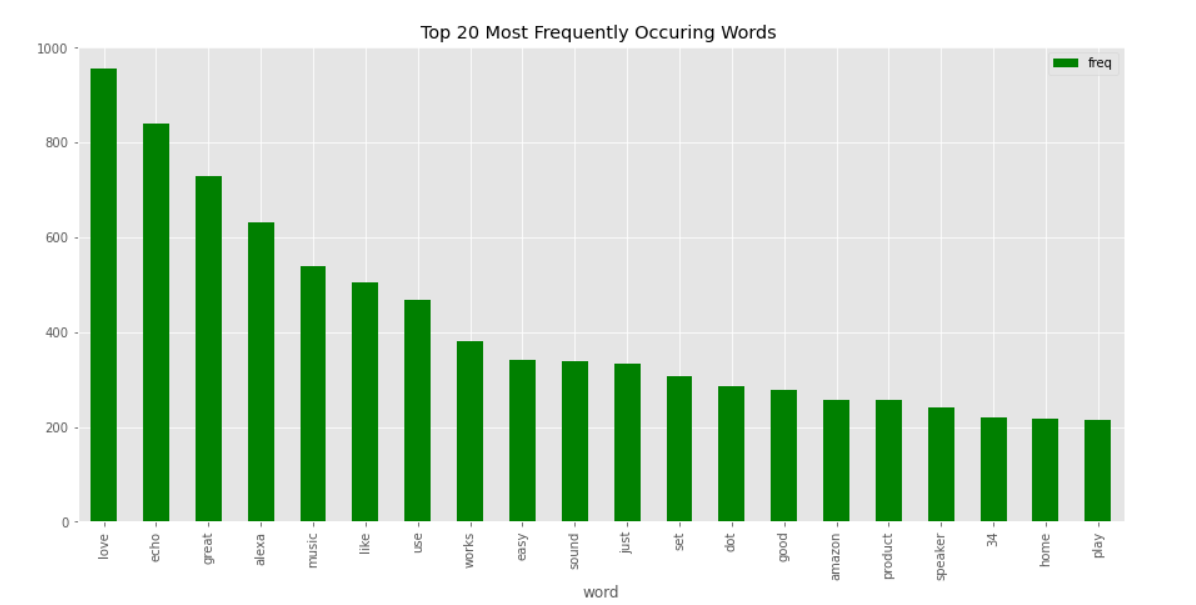
**3. DATA**

The dataset is from Kaggle, the source is extracted from Amazon's website, with the columns of ratings, verified reviews, date and feedback. We cleaned and preprocessed the dataset using the Python nltk package to remove stop words and punctuations.

**3. VISUALIZATION**

In our research, we finished statistics of our data and classified it into a visualized format. We designed the count distributions for each different attribute such as variations, reviews length and more. In addition, we made a distribution of Top 20 frequency words to visualize possible words occurring in both positive or negative reviews by customers.





**4. ANALYSIS**

The Vectorization part, we used CountVectorizer to create a matrix of token counts for the 'verified\_reviews' column in the dataframe. The text is converted to lowercase, and only unigrams are considered. Then split the data into training and testing sets. Created a logistic regression model object and fit the model with the training data, made predictions on the test data. Calculated and printed the accuracy, precision, recall, and F1-score of the logistic regression model. Then replaced the CountVectorizer with a TfidfVectorizer, which considers the importance of words in the dataset rather than just frequency, which improved the performance of the logistic regression model.

The prediction part, we applied 3 classification models, RandomForestClassifier, MultinomialNB (Naive Bayes), and LogisticRegression, comparing the performance of them, using 5-fold cross-validation to evaluate the accuracy of each model. Finally, we predicted the class of a new sample text using models above.

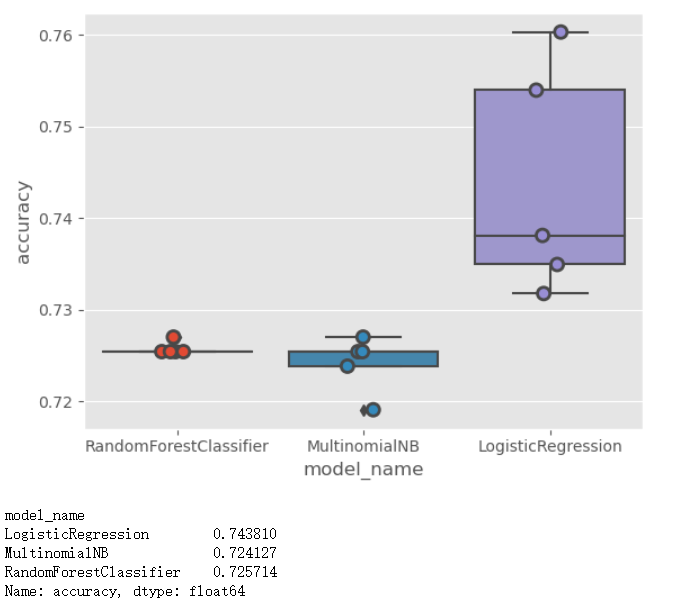
The measurement part, we used the Confusion Matrix based on the RandomForestClassifier model to visualize performance of our model. In this matrix, we calculate the correctly predicted number of pre-trained samples and try to figure out any bias or mistakes of the model.

Finally, we used Word cloud to visualize the positive and negative reviews to emphasize our research and findings.

**SUMMARY**

The result of Vectorizer part showed that CountVectorizer had the better accuracy and precision with 95% and 96%, but TF-IDF Vectorizer had better recall 1.0, the model can identify all actual positive reviews, but might also indicate that the model has some bias, tending to predict more instances as positive reviews. Overall the CountVectorizer got the better F1-score 97%, the model has achieved a good balance between precision and recall.

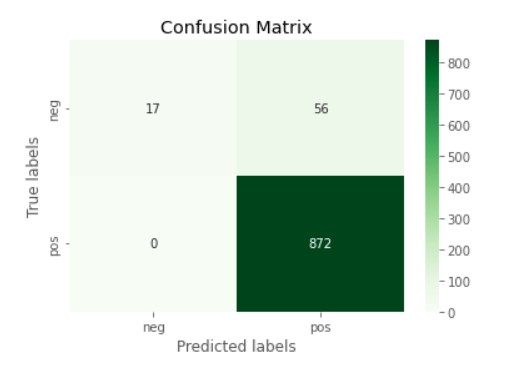
The prediction part’s result output with seaborn boxplot, showing that LogisticRegression had the best performance with 75%.

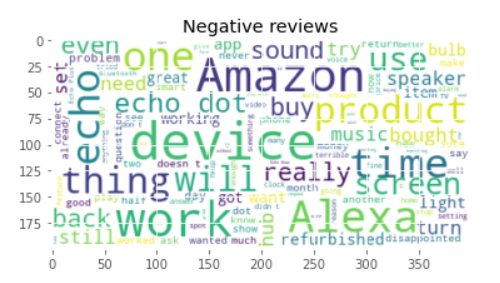
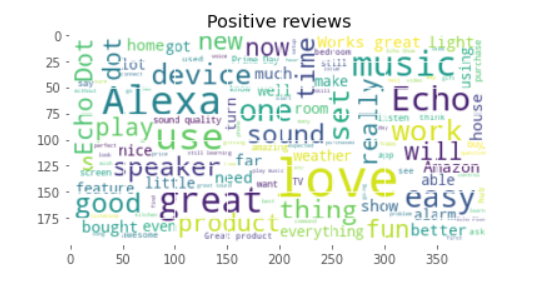




The model's ability to predict positive and negative reviews can provide valuable insights into the aspects of the product that customers like or dislike, which can guide product improvement and development. By identifying negative reviews, customer service teams can proactively address customer complaints and resolve issues, leading to increased customer satisfaction and loyalty. Understanding the factors that drive positive reviews can help marketing teams design more effective promotional campaigns, emphasizing the features and benefits that customers value the most.

However, from our measurement part Confusion Matrix analysis, we discovered that the dataset is highly imbalanced, with a significantly higher number of positive reviews, it affects the model's performance in identifying negative reviews. Also the high recall score might indicate that the model has some bias towards predicting positive reviews, which could lead to misclassification of some negative reviews. Simultaneously, our Word Cloud visualization emphasizes the conclusion, we can clearly find many subjective positive words in positive reviews but only neutral words in negative reviews.





In the end, we want to develop an interactive window with customers for more detailed feedback. And investigate more advanced text processing techniques, such as incorporating bigrams or trigrams, using word embeddings, or incorporating sentiment analysis to improve the model's performance.