Sentiment Analysis Report

Description of Dataset

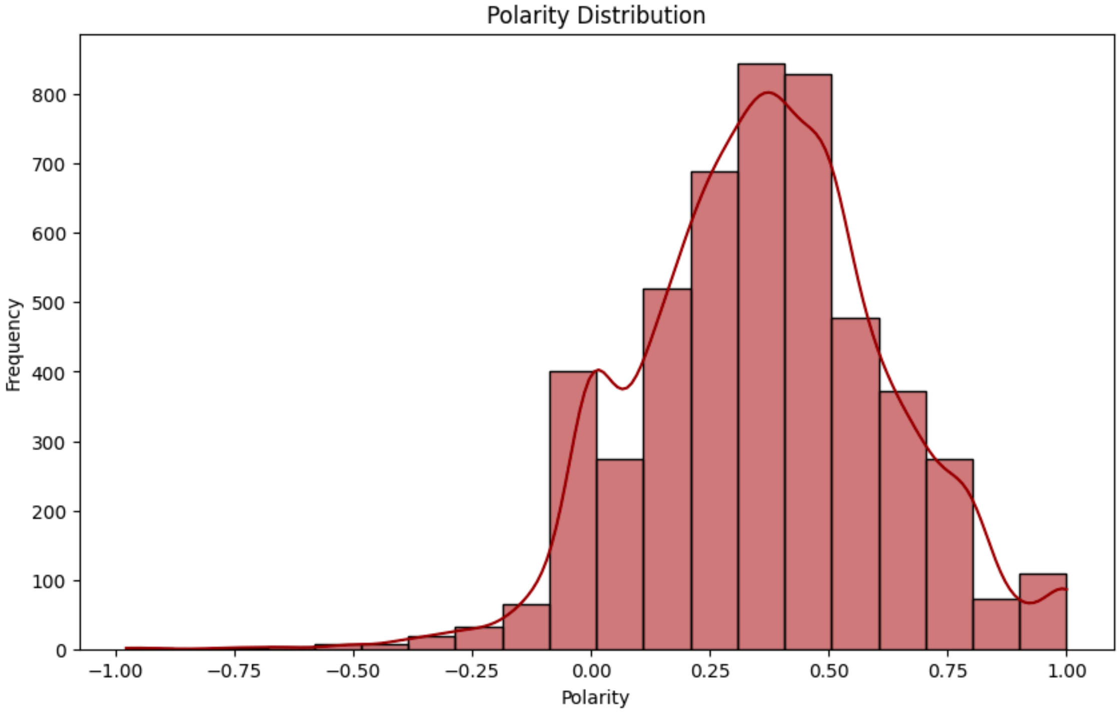
This dataset is string data type. Reviews of products that have been submitted to Amazon.

Details of the preprocessing steps

1. First I imported the necessary libraries to perform the tasks at hand, this included:
   - spacy
   - pandas as pd
   - spacytextblob
   - seaborn as sns
   - matplotlib.pyplot as plt
2. from spacytextblob.spacytextblob, I imported SpacyTextBlob
3. I loaded the en_core_web_sm model from spaCy and added spacytextblob
4. After reading the file, I used the code cleaned_df = df.dropna(subset=['reviews.text']) to drop the empty values
5. I then used user-defined functions to remove stopwords, make the text lowercase, and find the sentiment and polarity and added those values to a separate column.
6. I was then successful in creating a histogram to find the frequency of the values with polarity however I struggled to create it with sentiment as there are two values in the column.
7. I then tested a couple of reviews to see if the polarity and sentiment was accurate which I am confident in.

Evaluation of results

The results can be seen in this histogram below. Here you can see that the majority of reviews are slightly skewed towards being positive. Very few reviews are perfectly positive but they do exist. The average polarity score is 0.36656579180169746, so you can see there is a slight positive opinion.



Insight into the model's strengths and limitations

I find that some of the strengths of this is that it can be used on large datasets to prove insights on trends and overall opinions. It is also able to provide quantifiable figures to qualitative data. The fact that it is able to do this with the small language model from spaCy is a positive as well as it means that it is not too intensive on memory and computing power.

Some of the limitations of this program is that in order to get the most of this model, preparation needs to be done to the dataset prior. This can be time consuming preprocessing and can be rather intensive on the program's memory. This is proven by the fact that took just over 37 seconds for my computer to run the function that applied sentiment analysis and the same for polarity analysis.