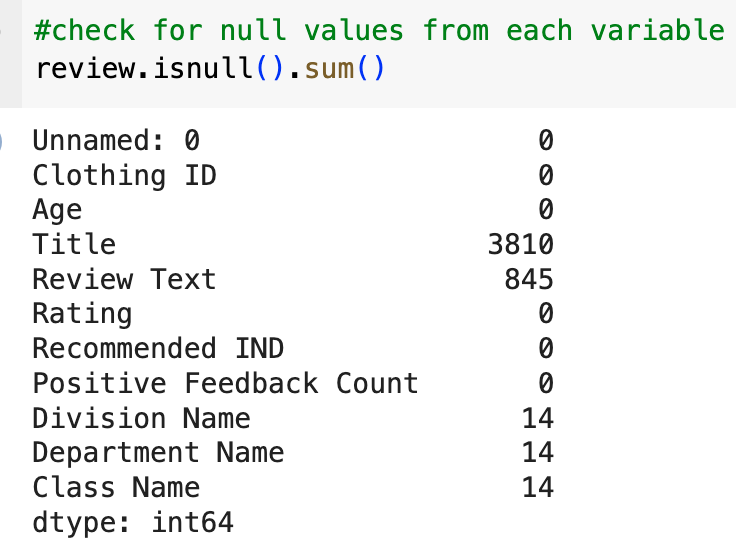
**Team #3: Milestone 2**

**Part 1. Data Cleaning**

There exist null values in multiple variables:

*Title, Review Text, Division Name, Department Name, Class Name*.

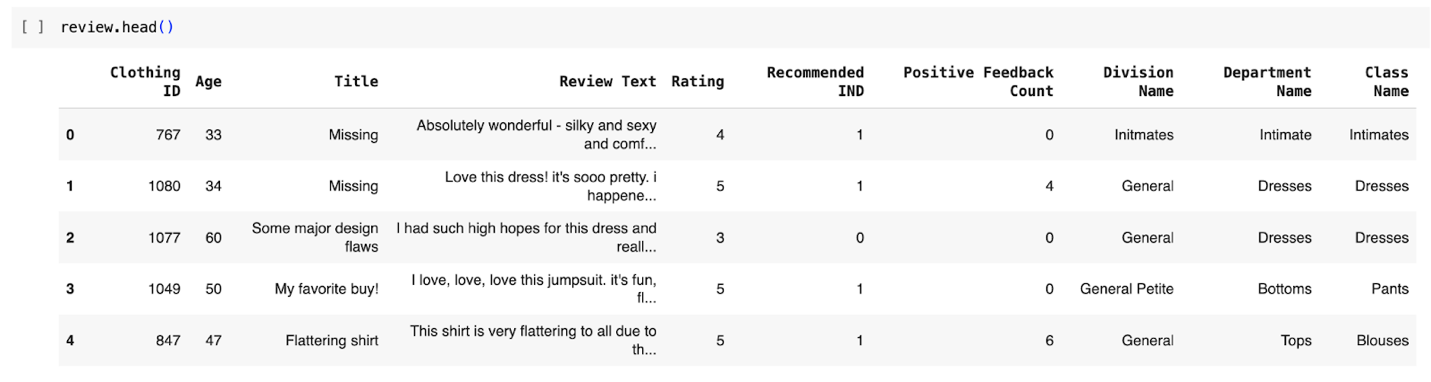


**Part 1.2. Removing Null Values**

To remove the null values from the raw dataset, we start by dropping off columns with missing values in Division Name, Department Name and Class Name using **dropna()**.  Based on examination, the missing value from the three columns was completely removed with only removal of 14 instances, implicating that there exist 14 instances with missing values from all three categories. It seems common to have missing values from Title and review segments, implicating the customers’ negligence to not input any content into their reviews. Therefore, those missing values front text columns can be properly addressed, by re-filling them with a new value: **Missing**.

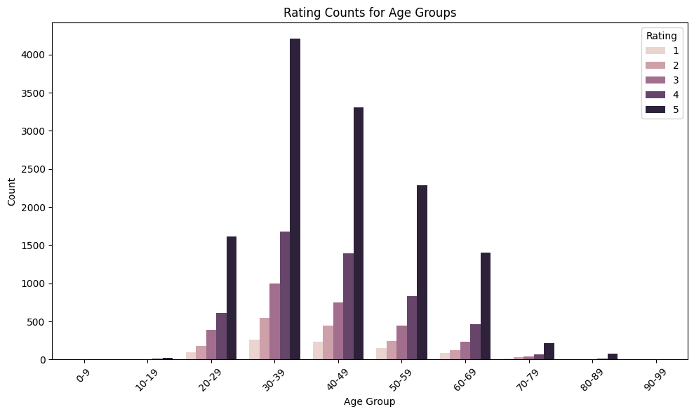


After transformation, the output dataset has all the null values filled and dropped.



**Part 2. EDA (Code Implementation** [***here***](https://colab.research.google.com/drive/1B63HDtqnY2Uve0ZMDR8zGcNWX5Rb7Zxo?usp=sharing)**)**

**EDA #1. Rating counts for age groups**



Among all the age groups, the age of reviewers between 30-39 has the most reviews count, and the most positive review counts as well, they are more generous than other age groups. After that, ages between 40-49 have the second highest number of reviews, slightly lower than 30-39 group. Surprisingly, the younger age groups of (20,29) has similar number of reviews as age group 60-69.

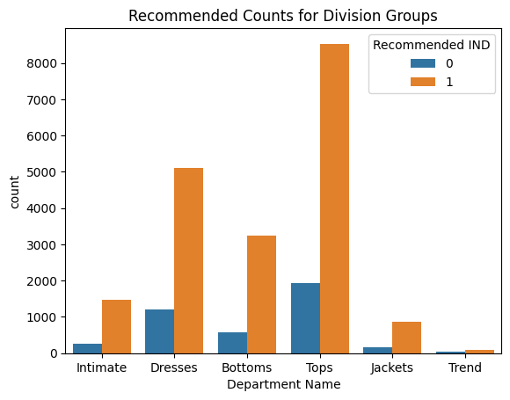
**EDA #2. Customer distribution by department for age groups 30-39**

A pie chart with numbers and text

Description automatically generated

For the age group 30-39, *Tops* department accounts for the most popular department of all divisions with 41.5% of overall reviews. Then *Dresses* takes up 28.9% of all reviews. We next explore all the categories among the most popular age groups.

**EDA #3. Division with most/least recommended counts**



In our age group 30-39, which has the highest number of reviews, indicating the *Tops* and *Dresses* category are the most popular divisions. Inspected deeply into divisions, we use *Recommended IND* to measure the customer satisfaction and their willingness to give high ratings. Above graph clearly shows the *Tops* and *Dresses* arepopular among all age groups.

**EDA #4. Top 10 Popular Items**

A graph of blue bars

Description automatically generated with medium confidence

To determine the popularity of items, we firstly filtered the counts of reviews higher than 4 and picked the top 10 counts of reviews of items. Item 1078 has over 700 reviews with the rating higher than 4, followed by item 862, slightly above 600.

**Part 3. Text Analysis (Code Implementation** [***here***](https://drive.google.com/open?id=1_7WHYVSFNr6wQcKafIMztWTm3NTsQq1R&usp=drive_fs)**)**

**3.1. Text Preprocessing**

* Converting all text to lowercase. This will make sure that all text is consistently formatted across the reviews. So that words can be grouped together in the use case of a clustering use-case.
* Removing punctuation and special characters. This will remove unnecessary noise needed for the classification model used for text analysis.
* Removing numbers and digits from text. This is also to remove noise from the textual data.
* Stemming tokens to obtain the root form of the word using **PortStemmer**. This will further group words written in multiple parts of speech into the same root group to prevent the model from having illusion errors.

**3.2. Text Classification**

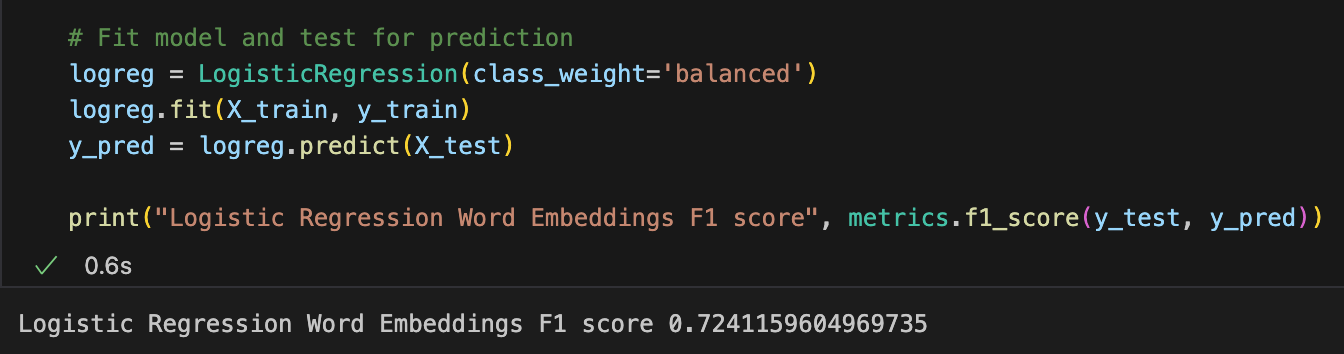
One initial approach and two alternative approaches were tried to classify the review text into a positive or negative sentiment value.

**3.2.1. Initial Approach:**

*Text Representation:* Word Embeddings with Dimension 100

*Classification Model:* Logistic Regression

*Code and Output:*



Word Embedding text representation is delivering an F1-score of 0.72, which is not good enough model performance. Alternative approaches of text representation or model selection will be attempted to achieve a better model performance.

A graph of negative and negative

Description automatically generated

**3.2.2. Alternative Approach #1:**

*Text Representation:* Bag of N-Grams

*Classification Model:* Logistic Regression

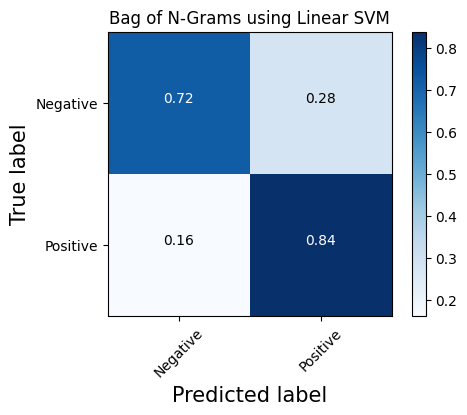
*Code and Output:*

A screen shot of a computer

Description automatically generated

The F-1 score of the text classification using Bag of N-Grams text representation on a Logistic Regression Model is 0.82, which is much higher than the Word Embeddings approach.

A diagram of a positive and negative label

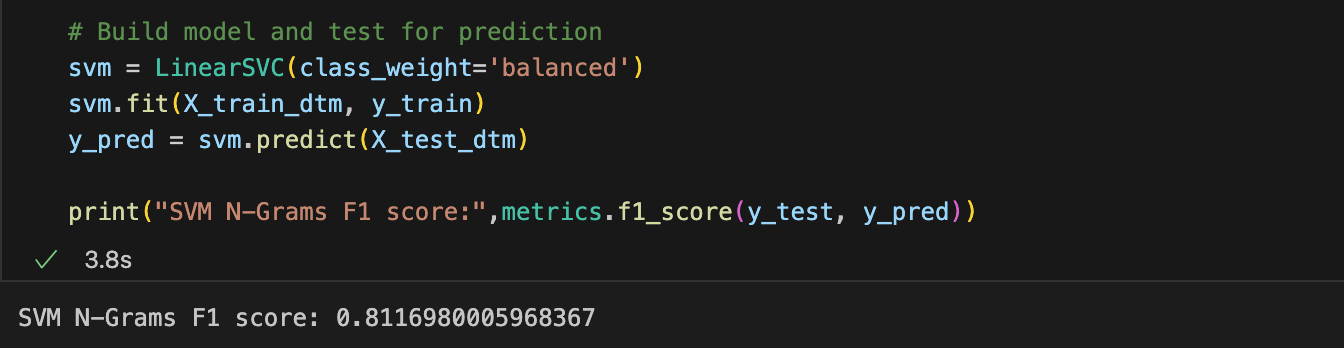
Description automatically generated with medium confidence 

**3.2.3. Alternative Approach #2:**

*Text Representation:* Bag of N-Grams

*Classification Model:* Support Vector Machine (SVM)

*Code and Output:*



The F-1 score of the text classification using Bag of N-Grams text representation on a Linear SVM Model is 0.81, which is much higher than the Word Embeddings approach, but still lower than the performance of the Logistic Regression Model.

**Part 4. Bonus Questions and Exploration**

**Techniques Explored:**

1. CNNs as a classification model (**TensorFlow Keras**) – [Code Implementation: [*here*](https://colab.research.google.com/drive/1Q_NUQ4sAv94lqPZdYpjRsBo93oQpktPb?usp=sharing)]
2. Sentiment Analysis (**Textblob** and **VaderSentiment**) – [Code Implementation: [*here*](https://colab.research.google.com/drive/1_cMIv8UaV_y78PXW_C5v4_17YvOVPJju?usp=sharing)]

**4.1. CNNs as a Classification Model**

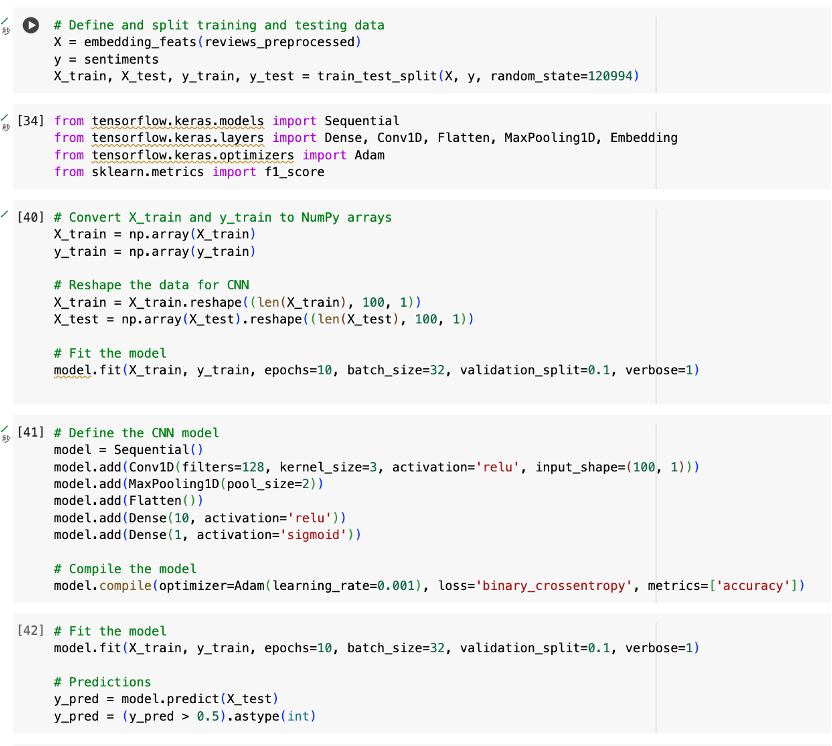
*Why do you apply this new method/tools/packages in this project?*

We applied CNN in this project because we didn’t employ this model in-class and exercises before. Also, CNNs are efficient in text classification tasks as they can automatically and adaptively learn spatial hierarchies of features from text data, which is crucial for understanding the sentiment or classification of the text.

*What is the benefit of using it?*

CNNs are great for classifying text because they're good at picking up on important words or phrases, no matter where they are in the input. They can handle this all at once, which makes them faster and more efficient than models that work step-by-step, like RNNs. Plus, CNNs can deal with texts of different lengths and formats, which is handy since natural language can be messy. Hence, CNNs are quick and accurate at pulling out useful information, which helps a lot in getting good results for classification tasks.

*Screenshot of the Code:*



*Outcome of CNN:*

A screenshot of a computer code

Description automatically generated

The F1 score value of 0.76 shows that by using CNN as a classification model, the overall performance is better when compared to using a Logistic Regression model. But still, it is not good enough.

*Source:*<https://towardsdatascience.com/nlp-with-cnns-a6aa743bdc1e>

**4.2. Sentiment Analysis**

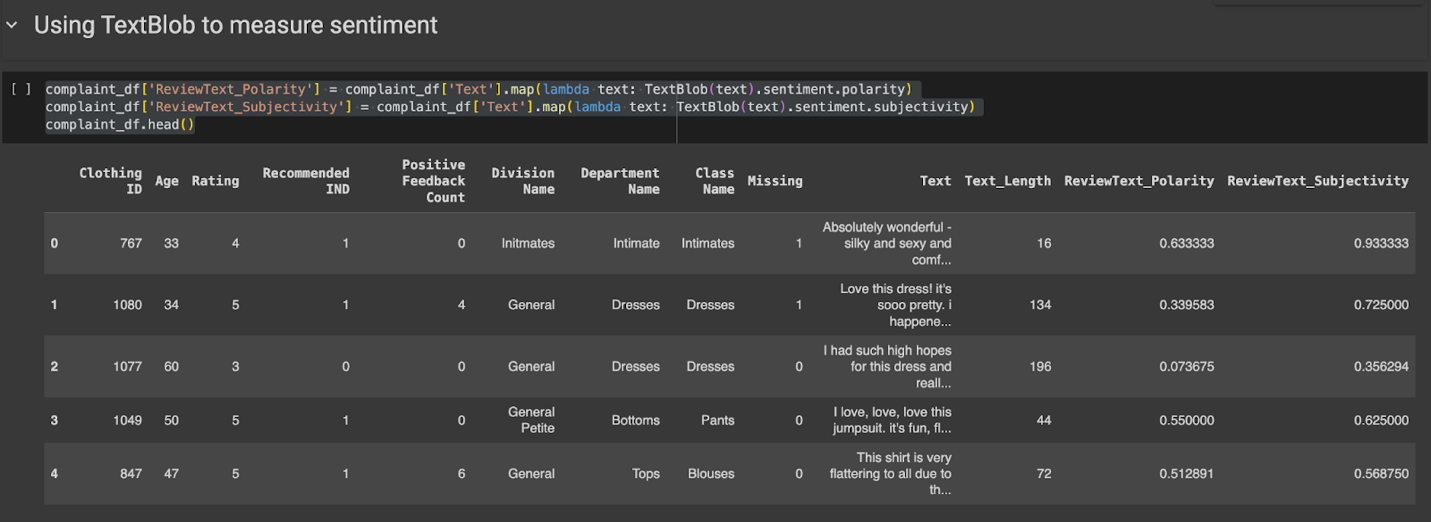
*Why do you apply this new method/tools/packages in this project?*

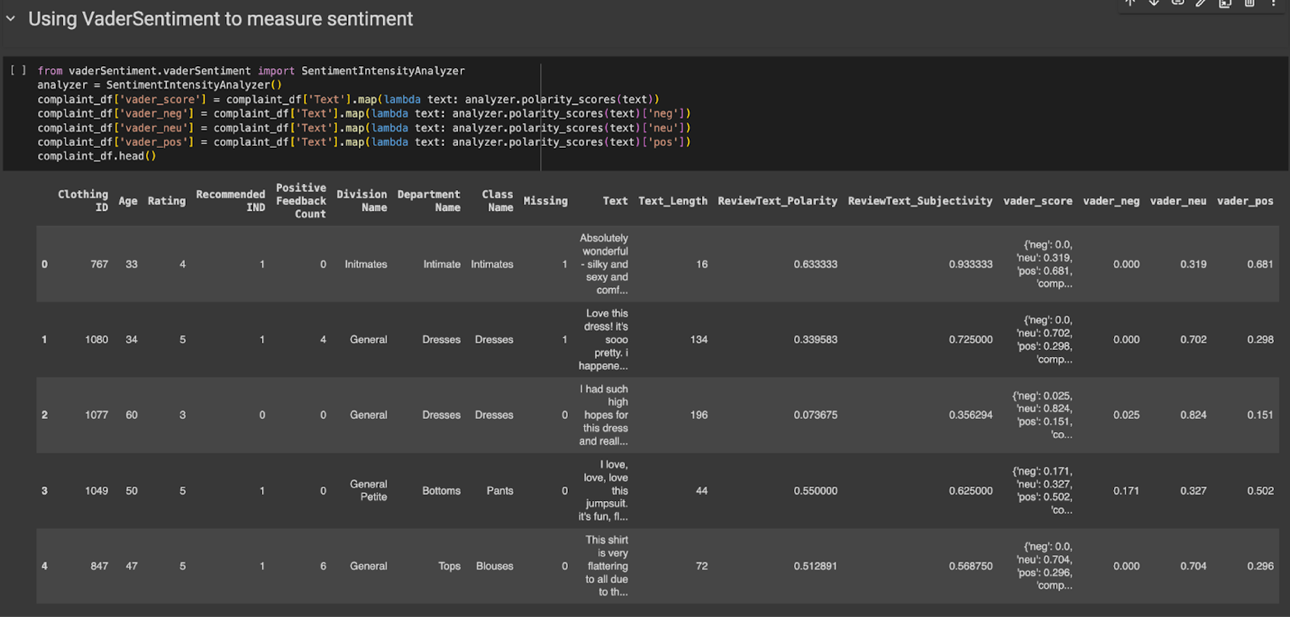
In the realm of E-Commerce, Sentiment Analysis can be useful to reveal key insights about how your brand, product, or company is viewed by your customers and stakeholders. The insights retrieved can be a great asset for the E-retailer to improve average product qualities via the vendor selection/filtering process.

*What is the benefit of using it?*

* **Textblob** is a great library that can detect both the subjectivity and polarity of the words’ context. In such a way we can use this tool to filter out those reviews that tend to be more rational, thus providing higher quality of information.
* **VaderSentiment** is a great library that helps to transfer context’s sentiment into metric scores, which provides a perspective to view the sentiment quantitatively. Scaling the extent of sentiment can be useful when it comes to market segmentations as well as summative statistics.

*Screenshots of the Code:*





*Outcomes:*

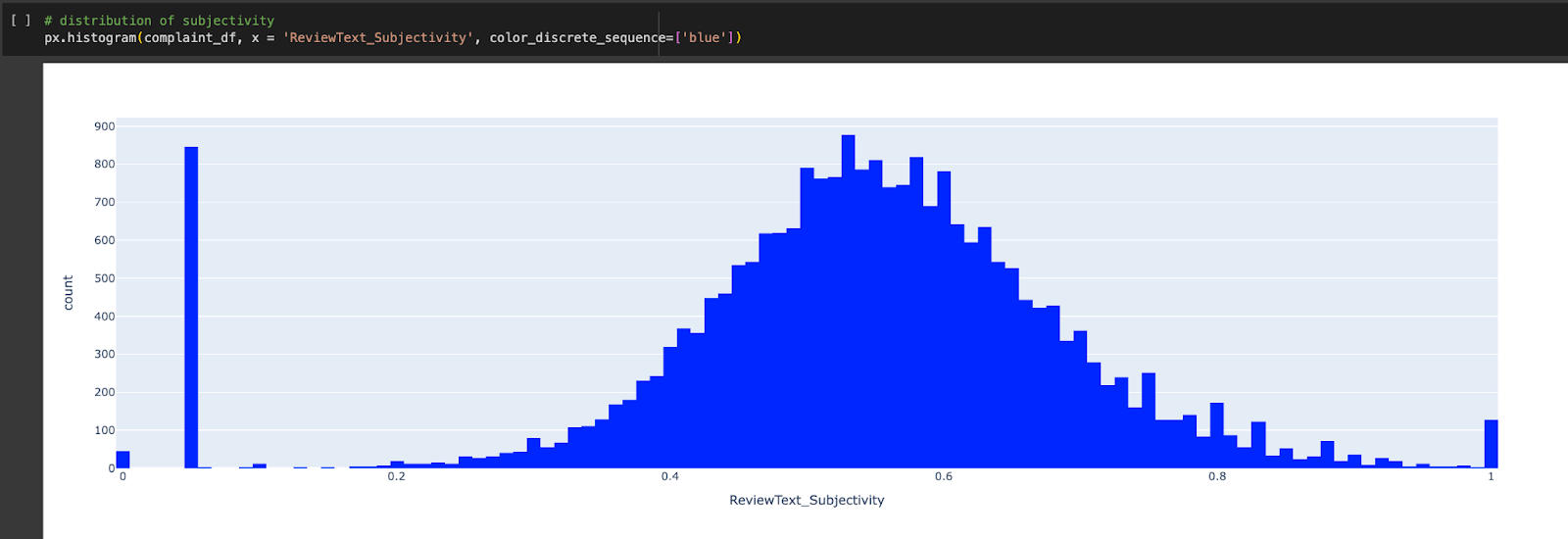
*Output #1: Sentiment Analysis using* **Textblob**

A screenshot of a graph

Description automatically generated

By interpreting the polarity, it can be found that most of the reviews’ comments are centered around -0.2 to 0.2, implicating that most of the customers have slightly positive views of their products, whereas there also exist many customer reviews that are slightly negative.

*Output #2 : Sentiment analysis using* **VaderSentiment**



In terms of the subjectivity, the plot above indicates a breath of bimodal distribution as a major subgroup of customers provided highly objective reviews while the majority of customers just gave very subjective reviews.

**Part 5. Meeting Agenda**

**Meeting #1:**

Prepared by: Niharika Chunduru

Date and time:  02/27/2024, 3 PM

Location: In-person

Team members in attendance: Shi Yang (SY), Yifei Zhao (YZ), Niharika Chunduru (NC), Xiaoxiao Huang (XH)

Meeting objectives: Allocate tasks for Milestone-2 Coding

Agenda: Assigned explicit portions of the Milestone-2 tasks to work on.

Next Actions: Finish coding to compile into a milestone report.

|  |  |  |  |
| --- | --- | --- | --- |
| **Action Item** | **Assigned To** | **Time Spent (approx.)** | **Due Date** |
| Data Cleaning and Preprocessing Code | YZ | 2 hours | 02/29/2024 |
| Exploratory Data Analysis Code | SY | 3 hours | 03/03/2024 |
| Text Representation and Analysis Code | NC | 3 hours | 03/03/2014 |
| Bonus Questions and Exploration Code | XH, YZ | 3 hours each | 03/03/2024 |

Time meeting ended: 3:30PM 02/27/2024.

Date and time of next meeting: 03/03/2024, 7 PM

**Meeting #2:**

Prepared by: Niharika Chunduru

Date and time:  03/03/2024, 7 PM

Location: Virtual (Zoom)

Team members in attendance: Shi Yang (SY), Yifei Zhao (YZ), Niharika Chunduru (NC), Xiaoxiao Huang (XH)

Meeting objectives: Discuss coding outputs and compile milestone report.

Agenda: The findings from EDA and Text Analysis were discussed, and the milestone compilation to begin.

Next Actions: Compile the findings into the milestone report per the requirements mentioned in the milestone description document.

|  |  |  |  |
| --- | --- | --- | --- |
| **Action Item** | **Assigned To** | **Time Spent (approx.)** | **Due Date** |
| Data Cleaning and Preprocessing Report | YZ | 30 mins | 03/04/2024 |
| Exploratory Data Analysis Report | SY | 15 mins | 03/04/2024 |
| Text Representation and Analysis Report | NC | 30 mins | 03/04/2024 |
| Bonus Questions and Exploration Report | XH, YZ | 30 mins each | 03/04/2024 |
| Proof-reading and Editing Report | SY, YZ, NC, XH | 2 hours | 04/04/2024 |

Time meeting ended: 03/03/2024, 7:30 PM

Date and time of next meeting: *TBD*