SENTIMENT ANALYSIS FOR MARKETING

TEAM MEMBER

NAME- HARISH S M

PHASE 3

PROJECT DEVELOPMENT

PART 1

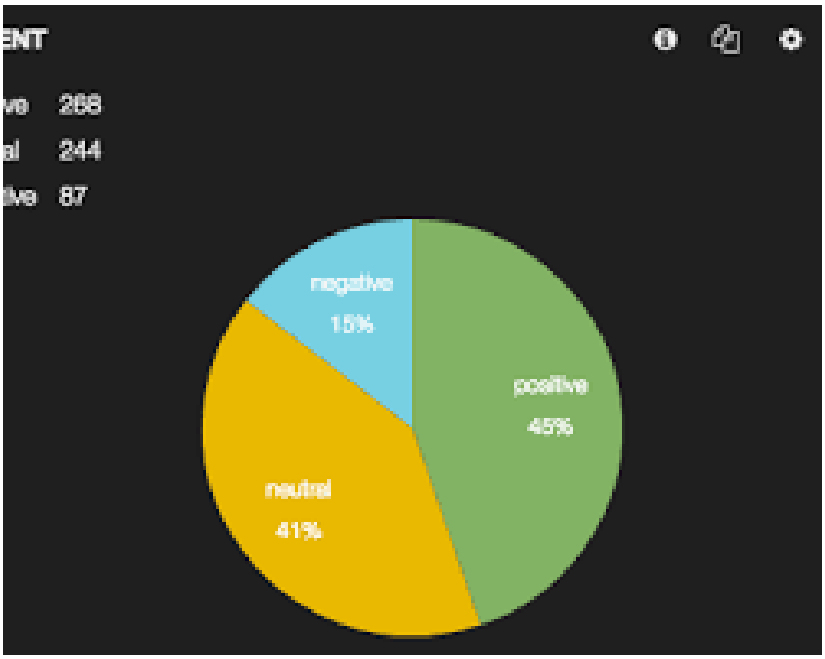
TITLE-**SENTIMENT ANALYSIS FOR MARKETING**

Two important aspects in sentiment analysis for marketing are:

Understanding Customer Emotions:

Sentiment analysis helps marketers gauge customer emotions towards products, services, or marketing campaigns. It's crucial to not only identify whether a sentiment is positive, negative, or neutral

but also to comprehend the underlying emotions. Understanding the emotional tone, such as happiness, frustration, excitement, or disappointment, provides nuanced insights.



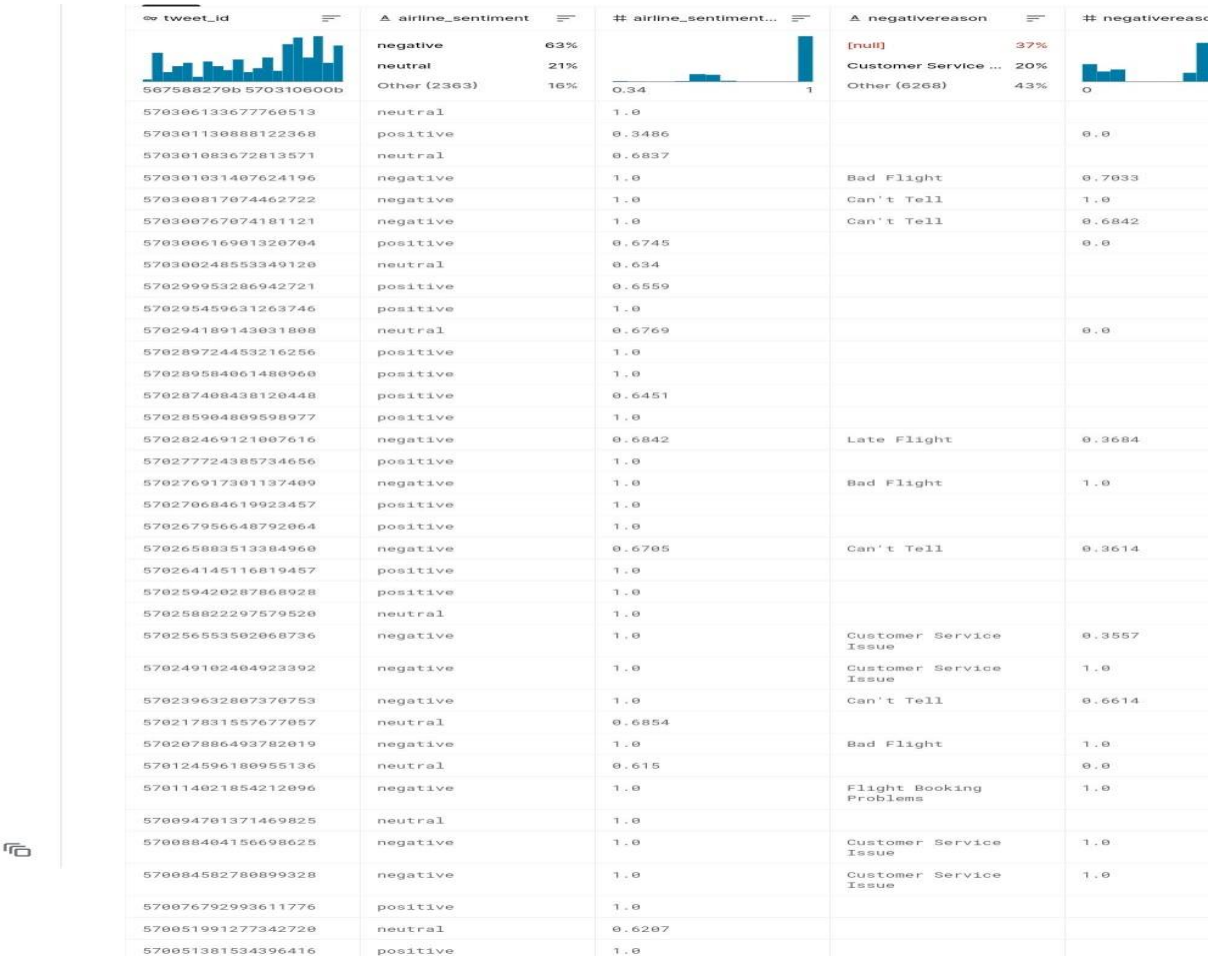
Contextual Analysis:

Context is paramount in sentiment analysis for marketing. The same phrase or word can carry different sentiments based on the context it's used in. Analyzing the context helps in accurate sentiment interpretation. For instance, the phrase "small size" might be positive when referring to portable gadgets but negative when describing a product meant to be large. Contextual analysis

involves understanding the industry-specific jargon, sarcasm, idiomatic expressions, and cultural nuances.

- DATASET:

Dataset Link:https://www.kaggle.com/datasets/crowdflower/twitter-airline-sentiment



tweet_id	airline_sentiment	airline_sentiment...	negativereason	negativereasc
567588279b 570310600b	negative 63% neutral 21% Other (2363) 16%	0.34 1	[null] 37% Customer Service ... 20% Other (6268) 43%	0
570306133677760513	neutral	1.0		
570301130888122368	positive	0.3486		0.0
570301083672813571	neutral	0.6837		
570301031407624196	negative	1.0	Bad Flight	0.7033
570300817074462722	negative	1.0	Can't Tell	1.0
570300767074181121	negative	1.0	Can't Tell	0.6842
570300616901320704	positive	0.6745		0.0
570300248553349120	neutral	0.634		
570299953286942721	positive	0.6559		
570295459631263746	positive	1.0		
570294189143031808	neutral	0.6769		0.0
570289724453216256	positive	1.0		
570289584061480960	positive	1.0		
570287408438120448	positive	0.6451		
570285904809598977	positive	1.0		
570282469121007616	negative	0.6842	Late Flight	0.3684
570277724385734656	positive	1.0		
570276917301137409	negative	1.0	Bad Flight	1.0
570270684619923457	positive	1.0		
570267956648792064	positive	1.0		
570265883513384960	negative	0.6705	Can't Tell	0.3614
570264145116819457	positive	1.0		
570259420287868928	positive	1.0		
570258822297579520	neutral	1.0		
570256553502068736	negative	1.0	Customer Service Issue	0.3557
570249102404923392	negative	1.0	Customer Service Issue	1.0
570239632807370753	negative	1.0	Can't Tell	0.6614
570217831557677057	neutral	0.6854		
570207886493782019	negative	1.0	Bad Flight	1.0
570124596180955136	neutral	0.615		0.0
570114021854212096	negative	1.0	Flight Booking Problems	1.0
570094701371469825	neutral	1.0		
570088404156698625	negative	1.0	Customer Service Issue	1.0
570084582780899328	negative	1.0	Customer Service Issue	1.0
570076792993611776	positive	1.0		
570051991277342720	neutral	0.6207		
570051381534396416	positive	1.0		

To perform sentiment analysis on the Twitter US Airline dataset, you'll need to follow these general steps:

1. **Collect the Dataset:** First, you need to obtain the dataset. You can find various datasets related to Twitter sentiment analysis on platforms like Kaggle or directly from research institutions.

2. **Import Libraries**: Use Python and libraries like Pandas, NumPy, and Scikit-Learn for data manipulation and machine learning tasks.

```python
import pandas as pd
from sklearn.model_selection import train_test_split
```

```
from sklearn.feature_extraction.text import CountVectorizer
from sklearn.naive_bayes import MultinomialNB
from sklearn.metrics import accuracy_score, classification_report
```

3. **Load and Explore the Data**: Load the dataset into a Pandas DataFrame and explore the data to understand its structure.

```python
# Load the dataset into a Pandas DataFrame
df = pd.read_csv('twitter_airline_sentiment.csv')

# Explore the dataset
print(df.head())
```

PREPROCESSING THE DATASET:

1. **Removing Special Characters and Numbers:** Remove symbols, special characters, and numbers as they usually don't contribute much to the sentiment.

   Import re

   Def clean_text(text):
   # Remove special characters and numbers using regular expression
   Cleaned_text = re.sub(r'[^a-zA-Z\s]', '', text)
   Return cleaned_text

2. **Tokenization**: Split the text into words or smaller sub-texts (tokens). In English, this is generally easy because words are usually separated by spaces.

3. **Lowercasing**: Convert all the words to lowercase. This ensures that the algorithm

treats words like “Text” and “text” the same way.

Text = text.lower() # Convert to lowercase

4. **Removing Stopwords**: Stopwords like ‘and’, ‘the’, ‘is’, etc., don’t contribute much to the sentiment. Removing them can help in focusing on the important words.

   Stop_words = set(stopwords.words(‘english’))

   BPs = PorterStemmer()

5. **Stemming or Lemmatization**: Reduce words to their base or root form. For instance, ‘running’ becomes ‘run’. Stemming is faster but might not always result in a real word, whereas lemmatization is slower but gives you a valid word.

   Import nltk

   From nltk.stem import PorterStemmer

   From nltk.stem import WordNetLemmatizer

From nltk.tokenize import word_tokenize

6. **Handling Emojis and Emoticons**: Emojis and emoticons are a significant part of tweets. You might want to replace them with text equivalents or remove them, depending on you're anyou're analysis.

   # Extract emojis from the text
   emojis_list = [c for c in text_with_emojis if c in emoji.UNICODE_EMOJI]

7. **Handling URLs:** Remove or replace URLs with a generic word like "URL" as they don't provide meaningful information for sentiment analysis.
   Text = re.sub(r'http\S+|www\S+|https\S+', 'URL', text, flags=re.MULTILINE) # Replace URLs

8. **Handling User Mentions (@username)**:Replace user mentions with a generic word like “USER” as they usually don’t carry sentiment information.

   Def preprocess_text(text):

      Text = re.sub(r’@[A-Za-z0-9]+’, ‘USER’, text) # Replace user mentions

9. **Handling Hashtags (#hashtag):** Remove the ‘#’ symbol and keep the hashtag word. Hashtags might indicate a specific topic or sentiment.
   Text = re.sub(r’#’, ‘’, text) # Remove hashtags