**Sentiment analysis for marketing**

**MALLI PRAVEENKUMAR**

**au723921244029**

[**praveenkumarmalli757@gmail.com**](mailto:praveenkumarmalli757@gmail.com)

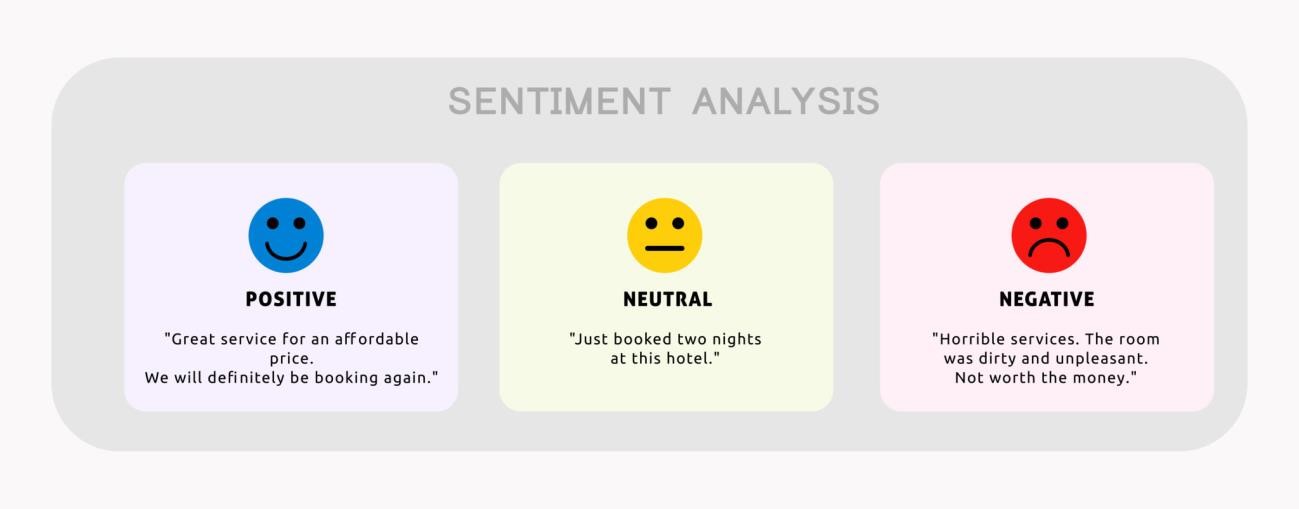
**Phase-3 Document**

**Submission**

**Project : Sentiment analysis for marketing**

**Phase 3: Development Part 1**

**Topic : In this part you will begin building your project by loading and preprocessing the dataset. Start building the sentiment analysis solution by loading dataset and preprocessing the data.**



**Introduction :**

Sentiment analysis is a powerful tool that can be used to understand customer sentiment and make better marketing decisions. It uses natural language processing (NLP) and machine learning to extract opinions and emotions from text data. This data can come from a variety of sources, such as social media posts, customer reviews, and survey responses.

Sentiment analysis can be used to answer a variety of marketing questions, such as:

* What do customers think of our products and services?
* What are the strengths and weaknesses of our brand?
* What are customers saying about our competitors?
* How effective are our marketing campaigns?
* What can we do to improve our customer satisfaction?

Sentiment analysis can be used to improve marketing campaigns in a number of ways. For example, it can be used to:

* Identify target audiences and develop more relevant marketing messages.
* Monitor the performance of marketing campaigns and make adjustments as needed.
* Respond to customer feedback in a timely and effective manner.
* Identify and address customer pain points.
* Improve product and service development.

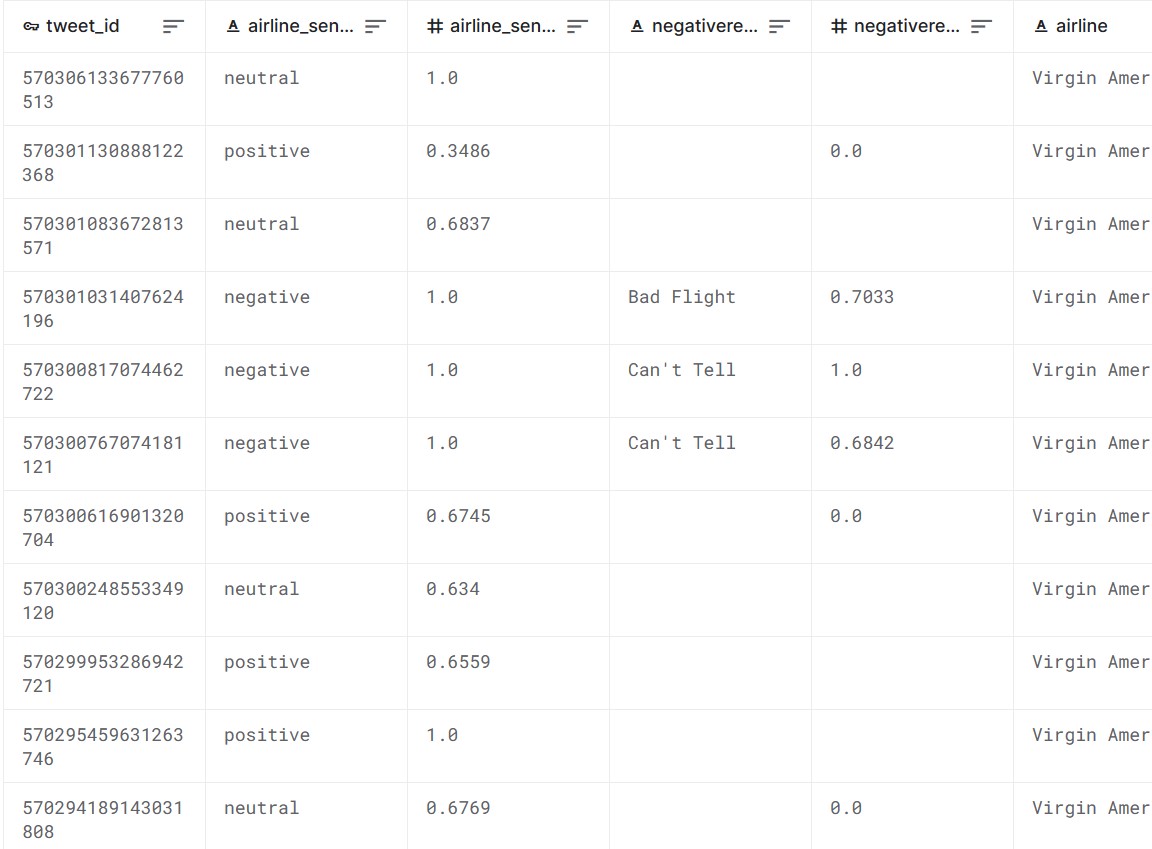
Overall, sentiment analysis is a valuable tool for marketers who want to better understand their customers and improve their marketing efforts.

Here are some specific examples of how sentiment analysis can be used for marketing projects:

* A company can use sentiment analysis to analyze customer reviews on social media to identify common complaints and areas for improvement.
* A business can use sentiment analysis to track the performance of a new marketing campaign and see how customers are responding to it.
* A brand can use sentiment analysis to compare itself to its competitors and see how it fares in terms of customer satisfaction.
* A product manager can use sentiment analysis to gather feedback from customers about a new product design.
* A marketing team can use sentiment analysis to segment its email list and send more targeted messages to different groups of customers.

Sentiment analysis is a powerful tool that can be used to improve marketing campaigns in a variety of ways. By understanding what customers are saying and feeling about a brand, businesses can make better decisions about how to reach and engage with them.

**Given dataset :**



**Importance of loading and processing dataset :**

Loading and processing datasets are fundamental steps in data analysis, machine learning, and many other data-driven applications. Here's why these steps are so important:

**Data Accessibility:**

Before any analysis can be performed, data must be loaded into an environment where it can be manipulated. This often means reading from databases, files, or external sources.

**Data Quality:**

Real-world data can be messy. It might contain missing values, duplicates, and outliers.By processing the data, you can clean and transform it, ensuring its quality and reliability for subsequent analysis or modeling. **Feature Engineering:**

Once data is loaded, often you'll need to create new features from the existing ones to better capture the underlying patterns in the data.For example, from a timestamp, you might extract the hour of the day, day of the week, or even whether it's a holiday.

**Data Scaling and Normalization:**

Many machine learning algorithms work better when numerical features have the same scale. By processing the data, you can apply scaling or normalization to ensure that all features have values in a similar range.

**Data Integration:**

Often, data comes from multiple sources. Loading and processing allow you to integrate these diverse datasets, ensuring consistent and unified information.

**Ensuring Data Privacy:**

When processing data, especially personally identifiable information (PII), you may need to anonymize or encrypt certain fields to ensure privacy and compliance with regulations.

**Efficiency and Performance:**

Large datasets can be unwieldy and slow down analysis or training. By loading data efficiently (e.g., using appropriate data structures) and processing it (e.g., by filtering irrelevant records), you can ensure more timely and responsive operations.

**Challenges involved in loading and preprocessing sentiment analysis dataset :**

Loading and preprocessing sentiment analysis datasets involve some unique challenges. While many of the general challenges associated with data loading and preprocessing apply here as well, sentiment analysis has its peculiarities:

**Language and Slang:** Natural languages evolve, and new slang words emerge. A sentiment analysis model trained on older data may struggle to understand newer phrases or words.

**Handling Ambiguity:** Natural language is often ambiguous. For instance, sarcasm is a form of speech where the intended sentiment might be opposite to the literal words.

**Imbalanced Datasets:** Often, sentiment analysis datasets are imbalanced, with more instances of one sentiment class than others. This imbalance can lead to biased model training if not addressed.

**Multilingual Data:** If the dataset contains multiple languages, preprocessing steps like tokenization or stemming might require languagespecific tools.

**Data Annotation Consistency:** Manual annotations can be inconsistent due to the subjective nature of sentiment. Two annotators might label the same text differently based on their interpretation.

**Short Texts:** Tweets or other short text snippets may not provide a lot of contextual information, making preprocessing and analysis more challenging.

**Noise in the Data:** User-generated content, which is often the source for sentiment analysis datasets, contains noise like URLs, usernames, emojis, and misspellings.

**Context Dependency:** The sentiment of a statement can be contextdependent. For instance, "This is sick!" can be positive in one context and negative in another.

**Data Privacy:** User-generated content might contain personally identifiable information, and this poses privacy concerns. Such data must be anonymized or removed.

**Tokenization Challenges:** Especially in languages where whitespace does not separate words, tokenization can be complex.

**1.Loading the dataset:**

Loading the dataset using machine learning is the process of bringing the data into the machine learning environment so that it can be used to train and evaluate a model.

The specific steps involved in loading the dataset will vary depending on the machine learning library or framework that is being used. However, there are some general steps that are common to most machine learning frameworks:

**a.Identify the dataset:**

The first step is to identify the dataset that you want to load. This dataset may be stored in a local file, in a database, or in a cloud storage service.

**b.Load the dataset:**

Once you have identified the dataset, you need to load it into the machine learning environment. This may involve using a built-in function in the machine learning library, or it may involve writing your own code.

**c.Preprocess the dataset:**

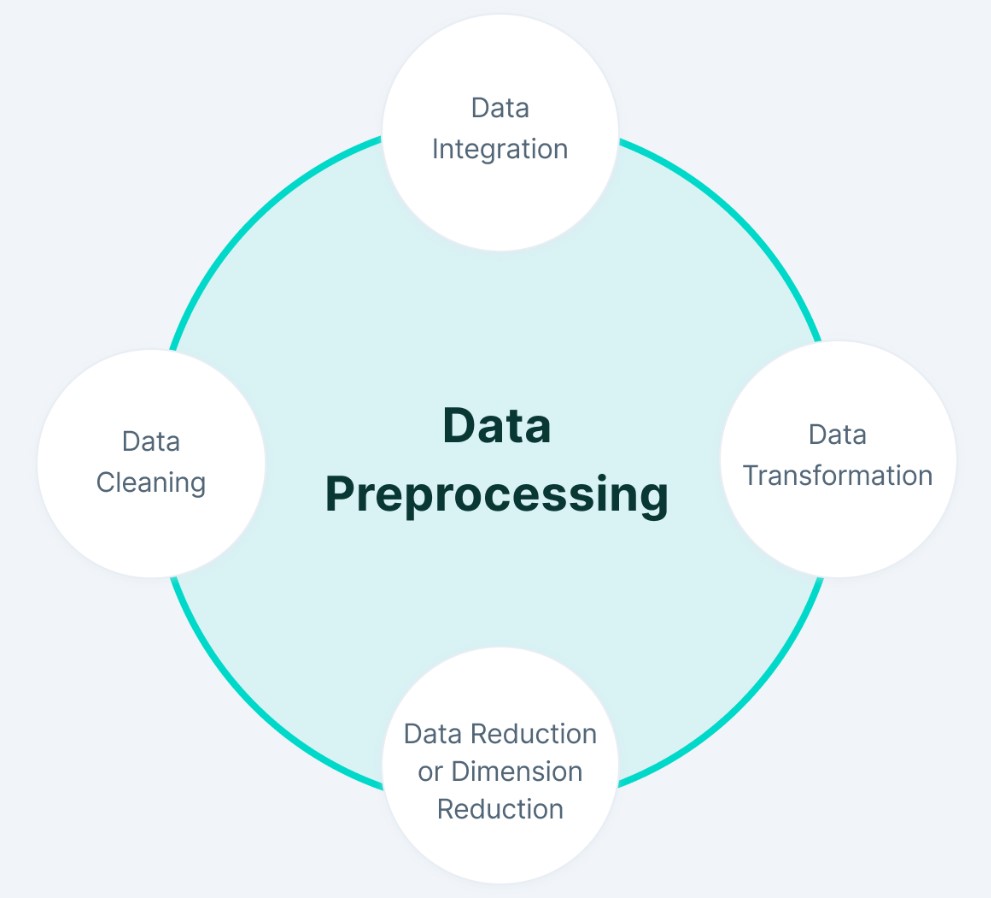
Once the dataset is loaded into the machine learning environment, you may need to preprocess it before you can start training and

evaluating your model. This may involve cleaning the data, transforming the data into a suitable format and splitting the data into training and test sets.

**2.Data Preprocessing**

Data preprocessing is the critical first step in any machine learning project.

It involves cleaning the data, removing outliers, and handling missing values to prepare the dataset for model training. In the context of the house price prediction project, let's elaborate on the specific steps:



**Python Program :**

*# Import necessary libraries* *import* numpy as np import pandas as pd import matplotlib.pyplot as plt import seaborn as sns import missingno as msno import warnings warnings.filterwarnings(action='ignore')

# Import NLTK and download required resources import nltk

from nltk.corpus import stopwords from nltk.tokenize import word\_tokenize, sent\_tokenize from nltk.stem import LancasterStemmer, WordNetLemmatizer

nltk.download('stopwords') nltk.download('punkt')

nltk.download('wordnet')

# Import other libraries import re import string import unicodedata import contractions

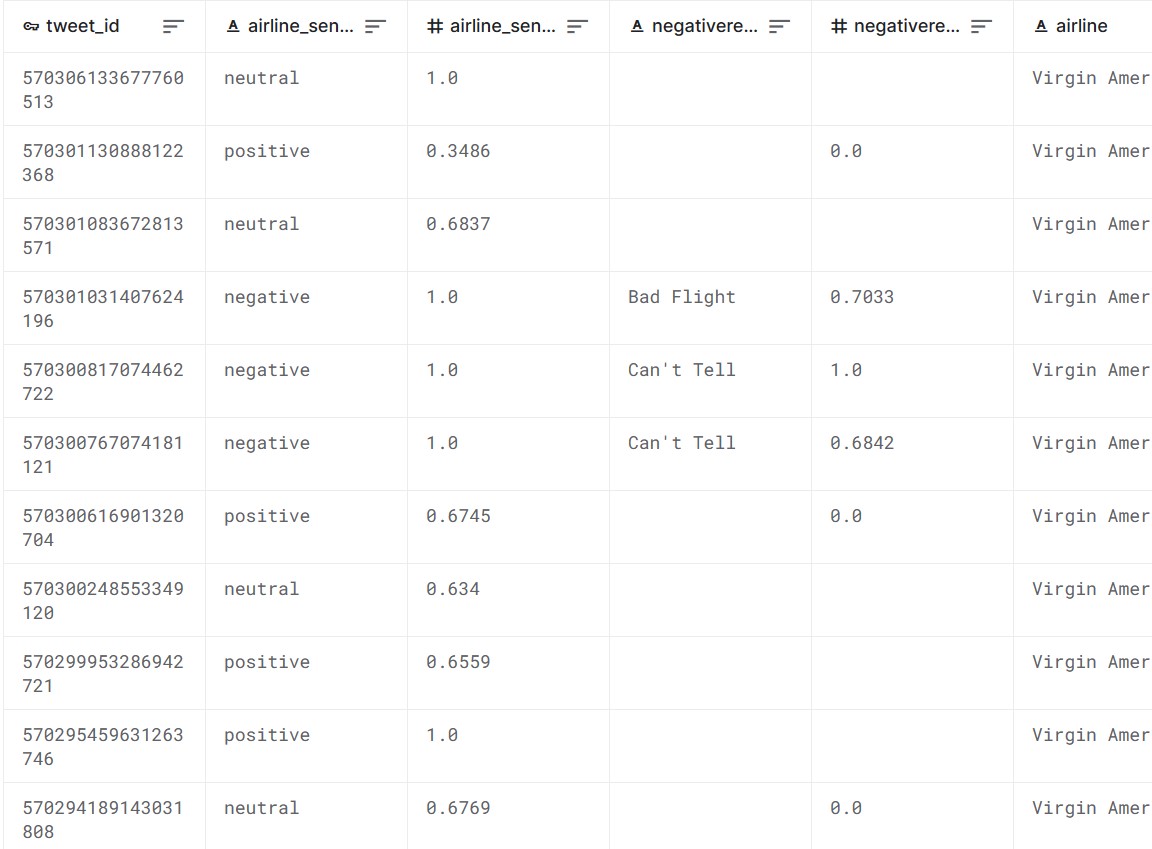
from sklearn.feature\_extraction.text import CountVectorizer, TfidfVectorizer import wordcloud

from wordcloud import STOPWORDS, WordCloud import pandas as pd from sklearn.model\_selection import train\_test\_split, StratifiedKFold from sklearn.svm import LinearSVC from sklearn.ensemble import RandomForestClassifier from sklearn.model\_selection import cross\_val\_score from sklearn.metrics import ( recall\_score, accuracy\_score, confusion\_matrix, classification\_report, f1\_score, precision\_score, precision\_recall\_fscore\_support

)

# Set options for displaying data pd.set\_option("display.max\_columns", None) pd.set\_option("display.max\_rows", 200)

**Output :**



**Observation :**

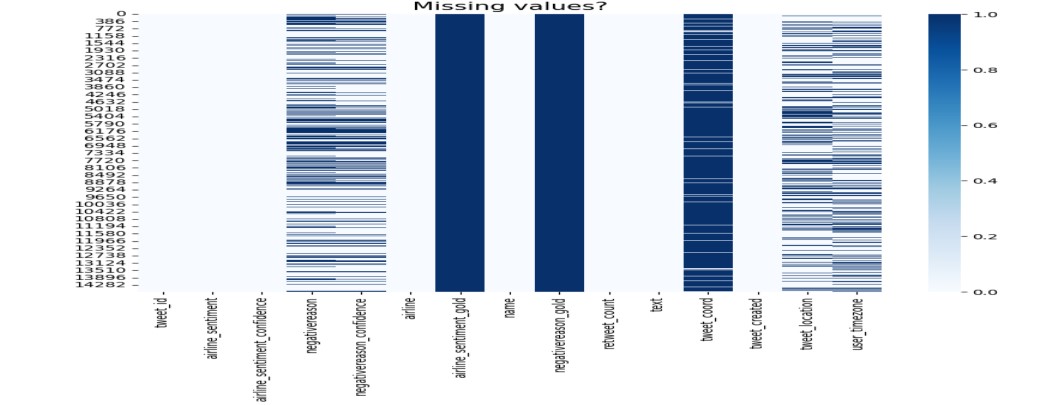
There are 15 columns in the dataset. Half of the columns have null values. Considering both dependent and independent variables not having any null values, we will not do any null value processing. Most columns in the dataset are of object type. airline\_sentiment is our dependent / target variable. text column is our independent variable that we will use for analysis. All other columns will be dropped at a later stage.

df.isnull().sum()

tweet\_id 0 airline\_sentiment 0 airline\_sentiment\_confidence 0 negativereason 5462 negativereason\_confidence 4118 airline 0 airline\_sentiment\_gold 14600 name 0 negativereason\_gold 14608 retweet\_count 0 text 0 tweet\_coord 13621 tweet\_created 0 tweet\_location 4733 user\_timezone 4820 dtype: int64

*#Visualization of missing value using heatmap* plt.figure(figsize=(10,7))

sns.heatmap(df.isnull(), cmap = "Blues") plt.title("Missing values?", fontsize = 15) plt.show()



print("Percentage null or na values in df")

((df.isnull() | df.isna()).sum() \* 100 / df.index.size).round(2)

Percentage null or na values in df

tweet\_id 0.00 airline\_sentiment 0.00 airline\_sentiment\_confidence 0.00 negativereason 37.31 negativereason\_confidence 28.13 airline 0.00 airline\_sentiment\_gold 99.73 name 0.00 negativereason\_gold 99.78 retweet\_count 0.00 text 0.00 tweet\_coord 93.04 tweet\_created 0.00 tweet\_location 32.33 user\_timezone 32.92 dtype: float64

linkcode

df.drop(["tweet\_coord", "airline\_sentiment\_gold", "negativereason\_gold"], axis=1, inp lace=True)

linkcode df.head()

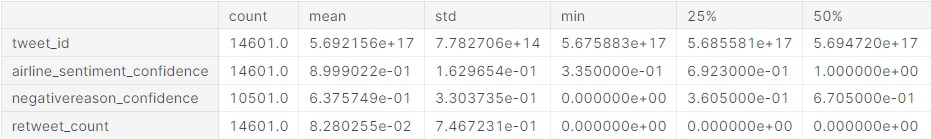


freq = df.groupby("negativereason").size()

df.duplicated().sum()

*# Dropping duplicates* df.drop\_duplicates(inplace = True)

df.duplicated().sum() linkcode df.describe().T



**EDA :**

df.nunique()

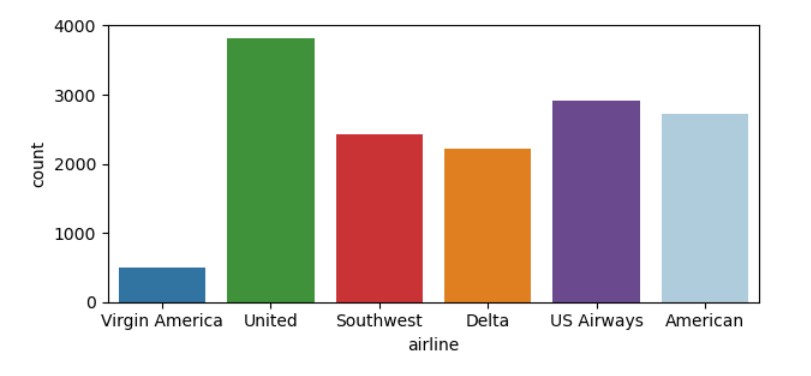
tweet\_id 14485 airline\_sentiment 3 airline\_sentiment\_confidence 1023 negativereason 10

negativereason\_confidence 1410 airline 6 name 7701 retweet\_count 18 text 14427 tweet\_created 14247 tweet\_location 3081 user\_timezone 85 dtype: int64

*# Checking the distribution of airlines* plt.figure(figsize=(7,3))

sns.countplot(data=df,x='airline', palette=['#1f78b4', '#33a02c', '#e31a1c', '#ff7f00

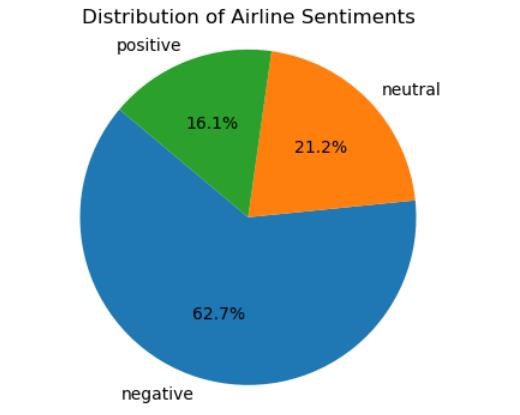
', '#6a3d9a', '#a6cee3']) plt.show()



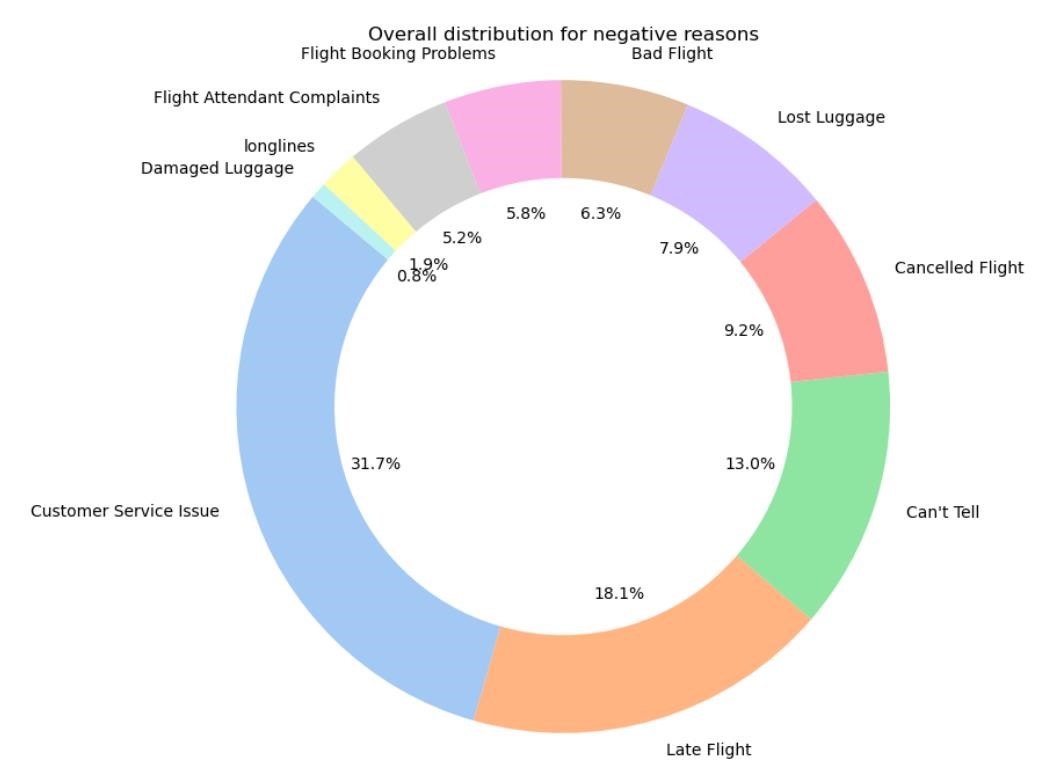
*# Visualize the distribution of airline sentiments using a pie chart* sentiment\_counts = df['airline\_sentiment'].value\_counts() plt.figure(figsize=(6, 4))

plt.pie(sentiment\_counts, labels=sentiment\_counts.index, autopct='**%1.1f%%**', start angle=140) plt.title('Distribution of Airline Sentiments') plt.axis('equal') *# Equal aspect ratio ensures that pie is drawn as a circle.*

plt.show()



|  |
| --- |
| *# Calculate the value counts for each negative reason* value\_counts = df['negativereason'].value\_counts()  *# Create a donut-like pie chart using matplotlib and seaborn* plt.figure(figsize=(8, 8)) labels = value\_counts.index values = value\_counts.values  colors = sns.color\_palette('pastel')[0:len(labels)] *# Use pastel colors for the chart*  plt.pie(values, labels=labels, colors=colors, autopct='**%1.1f%%**', startangle=140, wedgeprops=dict(width=0.3))  plt.title('Overall distribution for negative reasons')  plt.axis('equal') *# Equal aspect ratio ensures the pie chart is drawn as a circl e.*  plt.show() |



**Conclusion :**

The sentiment analysis project can be a valuable tool for businesses to gain insights into customer sentiment towards competitor products. By understanding customer sentiments, businesses can identify strengths and weaknesses in competing products, thereby improving their own offerings.