**SENTIMENT ANALYSIS FOR MARKETING**

***INTRODUCTION:***

**Sentiment analysis for marketing** refers to the use of sentiment analysis to understand how customers feel about a brand, product, or service. It can be used to track customer sentiment over time, identify trends, and gain insights into what customers like and dislike.

AI-powered sentiment analysis tools are becoming increasingly sophisticated and accurate. They are now able to identify a wide range of emotions, including sarcasm, irony, and ambiguity. This makes them ideal for use in marketing, where it is important to understand the true meaning of customer feedback.

*NECESSARY STEPS TO FOLLOW:*

1.Import libraries:

Start by importing necessary libraries.

PROGRAM:

import pandas as pd

import numpy as np

from sklearn.feature\_extraction.text import TfidfVectorizer

from sklearn.linear\_model import LogisticRegression

2.Load the dataset:

This code will load the dataset from a CSV file. You can also use other file formats. Once the dataset is loaded, you can inspect it to make sure that it is in the correct format and that it contains the data that you need.

PROGRAM:

import pandas as pd

# Load the dataset

df = pd.read\_csv('sentiment\_analysis\_dataset.csv')

*IMPORTANCE OF LOADING AND PREPROCESSING OF DATASET:*

* **To remove noise and irrelevant information:** The dataset may contain noise and irrelevant information, such as punctuation, stop words, and HTML tags. This information can interfere with the sentiment analysis process and lead to inaccurate results.
* **To convert the data into a consistent format:** The dataset may be in a variety of formats, such as CSV, JSON, or XML. It is important to convert the data into a consistent format so that it can be used by the sentiment analysis model.

**DATASET:**

I have used Twitter US Airline Sentiment from kaggle dataset.

In this dataset the following feature and labes are contained: tweet\_id,airline\_sentiment, airline\_sentiment\_confidence, negative reason, negative reason\_confidence, airline,airline\_sentiment\_gold,name,negative reason\_gold,retweet\_count,text, tweet\_coord,tweet \_created, tweet\_location, user\_timezone

**STEPS:**

1. **Load the dataset.** This can be done using a variety of programming languages and tools. For example, in Python, you can use the pandas library to load the dataset from a CSV file:
2. **Explore the dataset.** This includes checking the data types of the columns, the number of rows and columns, and the distribution of the data. You can also look for any outliers or missing values in the data.
3. **Preprocess the dataset.** This involves cleaning the data and removing any noise or irrelevant information. Some common preprocessing techniques for sentiment analysis include:

* **Converting the text to lowercase:** This helps to normalize the data and makes it easier to compare different pieces of text.
* **Removing punctuation:** Punctuation marks can add noise to the data and make it more difficult to extract sentiment.
* **Removing stop words:** Stop words are common words that do not add much meaning to the text, such as "the", "is", and "of". Removing stop words can help to improve the performance of the sentiment analysis model.
* **Lemmatization:** Lemmatization is the process of converting words to their root form. This helps to ensure that different forms of the same word are treated the same way by the sentiment analysis model

**PROGRAM:**

import seaborn as sns

import matplotlib.pyplot as plt

*# Creating column 'tweet\_length'*

df['tweet\_length'] = df['text'].apply(len)

*# distribution of sentiments*

plt.figure(figsize=(8,6))

sns.countplot(x='airline\_sentiment', data=df)

plt.title('Distribution of Sentiments')

plt.show()

*# Histogram of tweet lengths*

plt.figure(figsize=(8,6))

sns.histplot(df['tweet\_length'], bins=30)

plt.title('Distribution of Tweet Lengths')

plt.show()

*# Boxplot of tweet lengths*

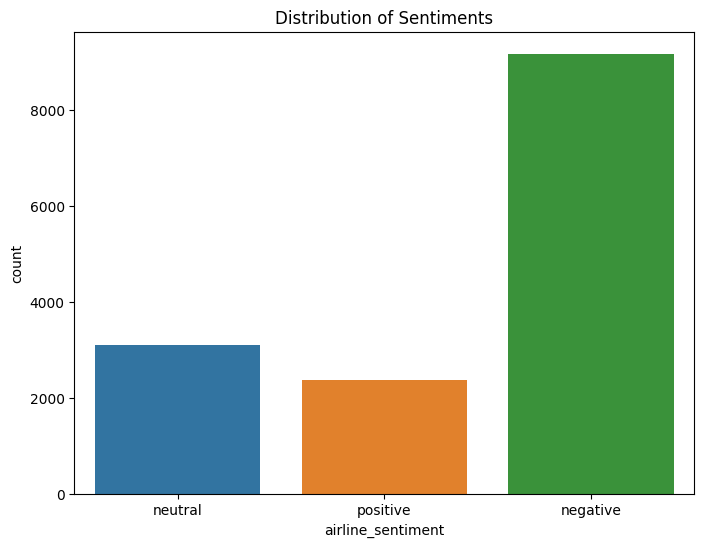
plt.figure(figsize=(8,6))

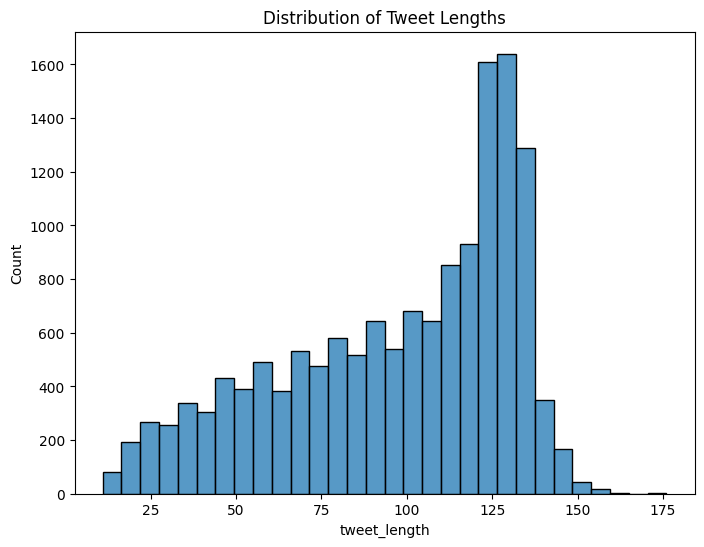
sns.boxplot(x='airline\_sentiment', y='tweet\_length', data=df)

plt.title('Distribution of Tweet Lengths by Sentiment')

plt.show()

**OUTPUT:**



**CONCLUSION:**

Sentiment analysis is a powerful tool that can be used to improve marketing efforts in a variety of ways. By carefully loading and preprocessing the dataset, you can ensure that your sentiment analysis model is able to generate accurate and meaningful results.