SENTIMENT ANALYSIS FOR MARKETING

Problem Definition:

The problem at hand is to perform sentiment analysis on customer feedback to gain insights into competitor products. Understanding customer sentiments is crucial for companies to identify strengths and weaknesses in competing products, thereby enhancing their offerings. This project involves the utilization of various Natural Language Processing (NLP) methods to extract valuable insights from customer feedback

Design Thinking:

1. Data Collection:

Identify and gather a dataset containing customer reviews and sentiments about competitor products.
Dataset Link: Twitter Airline Sentiment Dataset

2. **Data Preprocessing**:

Clean and preprocess the textual data for analysis. Steps include:
Removing HTML tags, special characters, and irrelevant symbols.
Tokenization: Splitting text into words or tokens.
Lowercasing: Ensuring uniformity by converting text to lowercase.
Removing Stopwords: Eliminating common words that don't carry significant meaning.

3. **Sentiment Analysis Techniques:**

Utilize various NLP techniques for sentiment analysis, such as:
Bag of Words (BoW): Creating a document-term matrix based on word frequency.
Word Embeddings (e.g., Word2Vec, GloVe): Representing words as dense vectors.
Transformer models (e.g., BERT, GPT-3): Leveraging pre-trained models for context-aware sentiment analysis.

Feature Extraction:

Extract features and sentiments from the preprocessed text data. Features may include:
Sentiment scores (positive, negative, neutral).
Key phrases or entities.
Document-level sentiment.

5. **Visualization**:

Create visualizations to depict the sentiment distribution and analyze trends. Visualization tools may include:
Bar charts or pie charts to represent sentiment proportions.
Time series plots to observe sentiment changes over time.
Word clouds to highlight frequently mentioned words.

6. **Insights Generation**:

Extract meaningful insights from the sentiment analysis results to guide business decisions.
Insights may include:
Identifying common pain points mentioned by customers.
Highlighting areas where competitor products excel.
Discovering potential opportunities for product improvement.

Conclusion:
This project aims to leverage NLP techniques to analyze customer feedback on competitor products, helping companies make data-driven decisions. The outlined design thinking process encompasses data collection, preprocessing, sentiment analysis, feature extraction, visualization, and insights generation. This approach will enable businesses to gain a deeper understanding of customer sentiments and enhance their competitive edg

PROGRAM:

```
import pandas as pd
import numpy as np
import nltk
from nltk.corpus import stopwords
from nltk.tokenize import word_tokenize
from sklearn.feature_extraction.text import
TfidfVectorizer
from sklearn.model_selection import train_test_
split
from sklearn.ensemble import
RandomForestClassifier
```

```
from sklearn.metrics import accuracy_score,
classification_report, confusion_matrix
import matplotlib.pyplot as plt
import seaborn as sns

# Download NLTK data
nltk.download('stopwords')
nltk.download('punkt')

# Load the dataset
df = pd.read_csv('Tweets.csv')

# Display basic statistics of the dataset
print("Dataset Statistics:")
print(df.describe())

# Display class distribution
class_distribution = df['airline_sentiment'].value_
counts()
print("\nClass Distribution:")
print(class_distribution)

# Preprocess the data
stop_words = set(stopwords.words('english'))

def preprocess_text(text):
    # Tokenize the text
    words = word_tokenize(text)
    # Remove stopwords and convert to lowercase
    filtered_words = [word.lower() for word in
words if word.isalnum() and word.lower() not in
stop_words]
    return ' '.join(filtered_words)
```

```
df['text'] = df['text'].apply(preprocess_text)

# Split the dataset into training and testing sets X
= df['text']
y = df['airline_sentiment']

X_train, X_test, y_train, y_test = train_test_split (X,
 y, test_size=0.2, random_state=42)

# Vectorize the text data using TF-IDF (Term
Frequency-Inverse Document Frequency)
tfidf_vectorizer = TfidfVectorizer(max_features=
5000)  # Limit the number of features
X_train_tfidf = tfidf_vectorizer.fit_transform(X_
train)
X_test_tfidf = tfidf_vectorizer.transform(X_test)#

Train a Random Forest classifier
rf_classifier = RandomForestClassifier(n_
estimators=100, random_state=42)
rf_classifier.fit(X_train_tfidf, y_train)
# Make predictions on the test data
y_pred = rf_classifier.predict(X_test_tfidf)
```

```
# Evaluate the classifier
accuracy = accuracy_score(y_test, y_pred)
print(f'\nAccuracy: {accuracy:.2f}')
print('\nClassification Report:')
print(classification_report(y_test, y_pred))
print('\nConfusion Matrix:')
conf_matrix = confusion_matrix(y_test, y_pred)
print(conf_matrix)

# Visualize the confusion matrix
plt.figure(figsize=(8, 6))
sns.heatmap(conf_matrix, annot=True, fmt='d',
cmap='Blues', xticklabels=['Negative', 'Neutral', '
Positive'], yticklabels=['Negative', 'Neutral', 'Positive
'])
plt.xlabel('Predicted Labels')
plt.ylabel('True Labels')
plt.title('Confusion Matrix')
plt.show()
```

OUTPUT:

```
Dataset Statistics:
       airline_sentiment_confidence
negativereason_confidence  retweet_count
count                  14640.000000
10522.000000   14640.000000
mean                       0.900169                  0
.638298       0.082650
std                       0.162830                 0.
330440       0.745778
min                       0.335000                  0.
000000       0.000000
25%                        0.692300                  0
.360600       0.000000
50%                        1.000000                  0
.670600       0.000000
75%                        1.000000                  1.
000000       0.000000
max                       1.000000                  1.
000000      44.000000

Class Distribution:
negative    9178
neutral     3099
positive    2363
Name: airline_sentiment, dtype: int64

Accuracy: 0.75

Classification Report:
              precision    recall  f1-score   support

    negative       0.82      0.87     Dataset
```

```
Statistics:
       airline_sentiment_confidence
negativereason_confidence  retweet_count
count                     14640.000000
10522.000000   14640.000000
mean                          0.900169
 0.638298       0.082650
std                           0.162830
0.330440       0.745778
min                           0.335000
0.000000       0.000000
25%                           0.692300
 0.360600       0.000000
50%                           1.000000
 0.670600       0.000000
75%                           1.000000
 1.000000       0.000000
max                           1.000000
1.000000      44.000000


Class Distribution:
negative    9178
neutral     3099
positive    2363
Name: airline_sentiment, dtype: int64

Accuracy: 0.75
```

```
Classification Report:
              precision    recall  f1-score   support

    negative       0.82      0.87      0.85
1864
     neutral       0.60      0.54      0.57       607
    positive       0.68      0.60      0.64       464

    accuracy                           0.75      2935
   macro avg       0.70      0.67      0.68
2935
weighted avg       0.74      0.75      0.75
2935

Confusion Matr
 [ 253  331   23]
 [ 116   55  293]] 0.85      1864
     neutral       0.60      0.54      0.57       607
    positive       0.68      0.60      0.64       464

    accuracy                           0.75      2935
   macro avg       0.70      0.67      0.68
2935
weighted avg       0.74      0.75      0.75
2935


Confusion Matrix:
[[1626  149   89]
 [ 253  331   23]
 [ 116   55  293]]
```