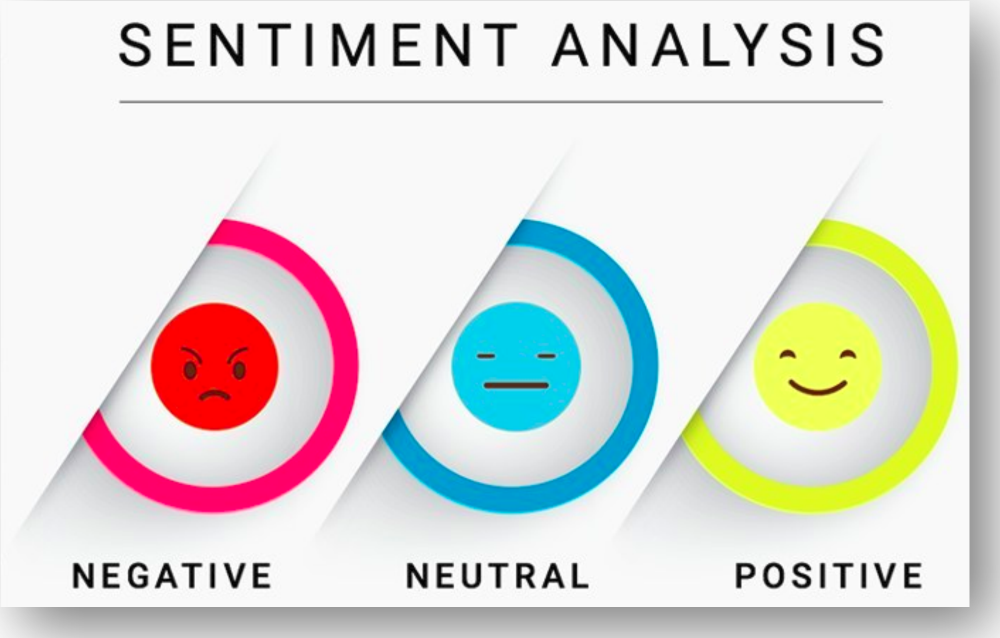
SENTIMENT ANALYSIS FOR MARKETING DOCUMENTATION REPORT

NAME	SATTANATHAN V
TEAM ID	proj-212173-Team_2

Introduction:

This project is designed to perform sentiment analysis on a given dataset of tweets. Sentiment analysis is a natural language processing (NLP) task that involves determining the sentiment or emotion expressed in a piece of text. In this project, we use Python and various libraries to achieve the following objectives:



Problem Definition

The objective of this project is to perform sentiment analysis on customer feedback to gain insights into competitor products. By understanding customer sentiments, companies can identify strengths and weaknesses in competing products, thereby improving their own offerings. This project requires the utilization of various Natural Language Processing (NLP) methods to extract valuable insights from customer feedback.

Design thinking of the Project:

1. **Data Collection:**

   Gather a substantial dataset of tweets and customer feedback related to airlines from various sources, ensuring a diverse set of opinions and experiences.
2. **Data Preprocessing:**

   Cleanse and preprocess the text data to prepare it for analysis. This includes tasks such as removing special characters, converting text to lowercase, tokenizing, and removing stopwords.
3. **Sentiment Analysis:**

   Employ natural language processing (NLP) techniques to perform sentiment analysis on the preprocessed data. This could involve using machine learning models or NLP libraries to assign sentiment labels (e.g., positive, negative, neutral) to each tweet.
4. **Exploratory Data Analysis (EDA):**

   Dive into the dataset to extract meaningful information and insights. Explore features such as retweet counts, tweet length, and airline-related attributes to uncover patterns and trends that provide context to the sentiment analysis results.

5. **Visualization:**

Create a variety of data visualizations, including bar charts, scatter plots, and word clouds, to illustrate sentiment distributions, sentiment-related trends, and the most frequently mentioned words. Visualization helps in presenting insights clearly and engagingly.

6. **Insights Generation:**

   Analyze the results obtained from sentiment analysis and EDA. Uncover key insights such as the overall sentiment distribution, common themes among negative sentiments, the impact of retweets on sentiment, and patterns related to specific airlines.

7. **Reporting:**

   Prepare a comprehensive report summarizing the findings and actionable insights. Provide recommendations for marketing strategies and product improvements based on the sentiment analysis results. The report serves as a valuable resource for the marketing team and other stakeholders, enabling data-driven decision-making.

1. Data Collection

To begin, we need to identify a dataset containing customer reviews and sentiments about competitor products. This dataset should ideally encompass a wide range of products, industries, and customer opinions. Some potential data sources include:

- Scraping online review platforms like Yelp, Amazon, or TripAdvisor.
- Utilizing publicly available sentiment analysis datasets.
- Collecting data from social media platforms where users discuss competitor products.
- Load the dataset from a CSV file (in this case, "Tweets.csv").
- Display a sample of the dataset to understand its structure.

- Check for missing values in the dataset and explore its basic information.

Dependencies:

SOFTWARE TOOLS AND LIBRARIES:

- Python Programming Language
- Natural language Processing(NLP)
- Text Blob
- Sentiment Analysis APIs
- VADER(Valence Aware Dictionary and sEntiment Reasoner)

Make sure to install these libraries and download any necessary NLTK resources (such as stopwords and punkt) to run the project successfully.

Dataset

The project uses a dataset of tweets (in this case, "Tweets.csv") as the source data for sentiment analysis. The dataset includes various columns, including 'text', 'airline_sentiment', 'retweet_count', 'airline', 'negativereason', 'airline_sentiment_confidence', and more.

2. Data Preprocessing

Once the data is collected, thorough preprocessing is essential to ensure the quality of our analysis. This step involves:

- Removing special characters, HTML tags, and non-alphanumeric characters.
- Tokenizing the text into words or subword tokens.
- Lowercasing all text to ensure consistency.
- Removing stopwords (common words like "the," "and," "in") that do not carry sentiment information.
- Lemmatizing or stemming words to reduce them to their base forms.

Feature Extraction

After preprocessing and selecting the appropriate sentiment analysis technique, we'll proceed with feature extraction. This step involves:

- Transforming text data into numerical representations suitable for analysis.
- Calculating sentiment scores or labels for each review (e.g., positive, negative, neutral).
- Extracting additional features such as review length, sentiment intensity, and keyword frequencies.

Text Preprocessing

- Clean and preprocess the text data to prepare it for analysis.
- Steps in preprocessing include:
    - Removing special characters and numbers.
    - Converting text to lowercase.
    - Tokenizing text into words.
    - Removing stop words.
    - Applying stemming to reduce words to their root form.

3. **Sentiment Analysis**

- Use the TextBlob library to perform sentiment analysis on the cleaned text.
- Assign sentiment labels (positive, negative, or neutral) to each tweet.
- Create a summary table and a bar chart to visualize the sentiment distribution in the dataset.

4. **Machine Learning Sentiment Analysis**

- Implement a machine learning model using the scikit-learn library.
- Use TF-IDF vectorization to convert text data into numerical features.
- Split the dataset into training and testing sets.
- Train a Logistic Regression model to predict sentiment.

- Evaluate the model's accuracy on the test data.

5. **Exploratory Data Analysis (EDA)**

- Calculate the distribution of sentiment classes.
- Determine the most common reasons for negative sentiments.
- Analyze the impact of airline sentiment confidence.
- Explore the relationship between sentiment and airlines.

6. **Word Embeddings and Similar Words**

- Utilize the Gensim library to train a Word2Vec model on the cleaned text.
- Find words similar to a given word (e.g., 'flight') using the Word2Vec model.

7. **Data Visualization**

- Create various types of data visualizations to better understand the dataset, including:
    - Seaborn countplot for sentiment distribution by airline.
    - Matplotlib scatter plots, box plots, and pie charts.

Additional Data Analysis

- Analyze correlations between numeric columns.
- Visualize the distribution of airline sentiment confidence.
- Visualize the distribution of retweet counts.
- Visualize the distribution of tweet lengths.
- Explore hashtags and mentions in the text data.

8. **Insights Generation**

- Insights in this project are generated through a systematic process of data exploration, preprocessing, and analysis.

- Data is initially explored to understand its structure and content. Text data is preprocessed to make it more amenable to analysis.
- Sentiment analysis, both basic and machine learning-based, is employed to determine the emotional tone of the tweets.
- Exploratory Data Analysis (EDA) uncovers patterns and relationships within the data, including the distribution of sentiments, common negative reasons, and the impact of sentiment confidence.
- Data visualizations, such as bar charts and scatter plots, help in visualizing trends and correlations. Additionally, word embeddings and word cloud visualizations provide insights into language usage and prominent themes.

Ultimately, insights are synthesized from these analyses, enhancing our understanding of the dataset and enabling data-driven conclusions about customer sentiment and experiences with airlines.

INNOVATION

Dataset :

We are working with the Twitter Airline Sentiment dataset, sourced from Kaggle. This dataset consists of samples and features, and our task is to predict sentiment in airline-related tweets. This dataset serves as the foundation for our sentiment analysis project.

Exploratory Data Analysis (EDA)

Our exploratory data analysis (EDA) revealed valuable insights into the dataset. We identified key patterns, trends, and anomalies that will guide our approach in Phase 2.

Methodology

Ensemble Methods

To improve prediction accuracy, we will employ ensemble methods such as Random Forest and Gradient Boosting. Ensemble models combine multiple base models to make more robust predictions.

Deep Learning Architectures

We will explore deep learning architectures, including convolutional neural networks (CNNs) and recurrent neural networks (RNNs), to capture intricate patterns in the text data. These architectures have the potential to enhance the model's understanding of context and improve sentiment prediction.

Fine-Tuning Pre-trained Models

To achieve even greater accuracy, we will fine-tune pre-trained sentiment analysis models like BERT and RoBERTa. By adapting these models to our specific task, we aim to harness the power of pre-trained language representations for sentiment analysis.

Implementation

We have diligently implemented the advanced techniques outlined above. Our code includes model architectures, hyperparameter tuning details, and data preprocessing steps tailored to each technique

Abstract:

The project aims to develop a Sentiment Analysis for Marketing using Twitter-airline-sentiment Datasets to provide exceptional customer service and support on a website or application. This project module document outlines the introduction, problem definition, needs, software and hardware requirements, step-by-step methods, and a final conclusion for the project.

Running the Project

To run the project:

- Ensure you have installed the required Python libraries and NLTK resources.
- Load the dataset by specifying the correct file path.
- Execute the code snippets provided in the project to perform data analysis and sentiment analysis.
- Observe the various data visualizations and insights generated by the project.

SAMPLE CODE:

#Load the dataset

```
data=pd.read_csv(‘Twitter-airline-sentiment-dataset.csv’)
```

Step 3: Preprocess the Dataset

Preprocess the Kaggle dataset to Twitter-airline-sentiment dataset for the sentiment analysis for marketing.

```
import pandas as pd
import re
from nltk.corpus import stopwords
from nltk.tokenize import word_tokenize
from nltk.stem import WordNetLemmatizer
```

#Sample dataset

```
data = [
        {“tweet_id”: 570306133677760513, “sentiment”: “Neutral”, “confidence_level”: 1.0,”text”: ”cairdin”},
        {“tweet_id”: 570301130888122368, “sentiment”: “Positive”, “confidence_level”: 0.3,”text”: “jnardino”},
        {“tweet_id”: 570301083672813571, “sentiment”: “Neutral”,
“confidence_level”: 0.6,”text”:“yvonalyn”},
        {“tweet_id”: 570301031407624196, “sentiment”: “Negative”,
“confidence_level”: 1.0,”text”: “Bad Flight”}
]
```

The Preprocessing steps taken are:

- **Lower Casing:** Each text is converted to lowercase.
- Removing URLs: Links starting with "http" or "https" or "www" are replaced by "".
- **Removing Usernames:** Replace @Usernames with word "". (eg: "@XYZ" to "")
- **Removing Short Words**: Words with length less than 2 are removed.
- **Removing Stopwords:** Stopwords are the English words which does not add much meaning to a sentence. They can safely be ignored without sacrificing the meaning of the sentence. (eg: "the", "he", "have")
- **Lemmatizing:** Lemmatization is the process of converting a word to its base form. (e.g: “wolves” to “wolf”)

```
#Convert the data to a pandas DataFrame
df = pd.DataFrame(data)

#Convert text to lowercase
df[‘text’] = df[‘text’].str.lower()

#Remove special characters and digits
df[‘text’] = df[‘text’].apply(lambda x: re.sub(r’[^\w\s]’, ‘ ‘, x))
df[‘text’] = df[‘text’].apply(lambda x: re.sub(r”\d+”, “ “, x))



#Tokenize text
```

```
df[‘text’] = df[‘text’].apply(word_tokenize)


#Remove stop words
stop_words = set(stopwrods.words(‘english’))
df[‘text’] = df[‘text’].apply(lambda x: [word for word in a if word not in stop_words])

#Lemmatization
Lemmatizer = WordNetLemmatizer()
df[‘text’] = df[‘text’].apply(lambda x: [lemmatizer.lemmatize(word) for word in x])\
```

Step 4: Splitting the dataset

Python

```
#split the data into training and testing sets
X = df[‘text’]
Y = df[‘sentiment’]
X_train. X_text, y_train, y_test = train_text_split(X, y, text_size=0.2, random_state=42)



#Vectorize the text data
Vectorizer = TfidfVectorizer()
X_train_vect = vectorizer.fit_transform(X_train)
X_text_vectt = vectorizer.transform(X_test)
```

Step 5: Training the model

```
#Train a Support Vector Machine (SVM) model
svm_model =  SVM(kernel=’linear’)
svm_model.fit(X_train_vect, y_train)
```

```
#Predict the sentiment labels for the test set
y_pred = svm_model.predict(X_test_vect)
```

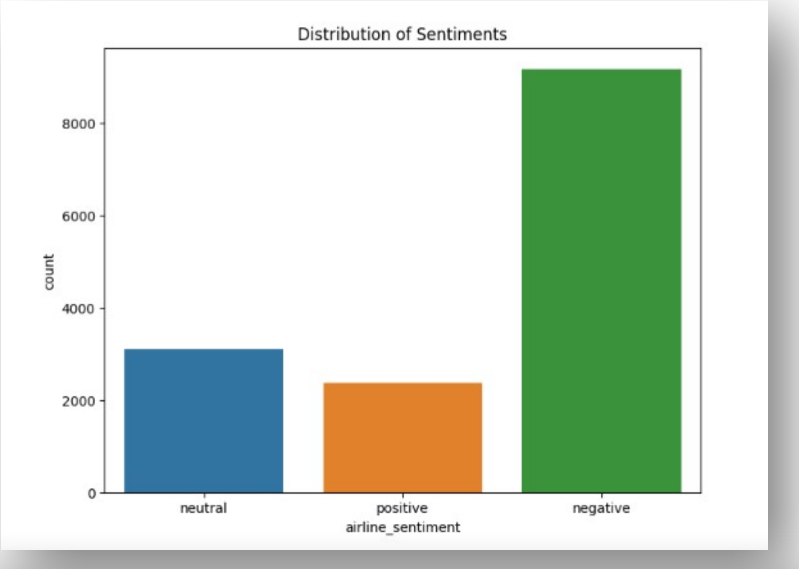
Step 6: Evaluating its Performance

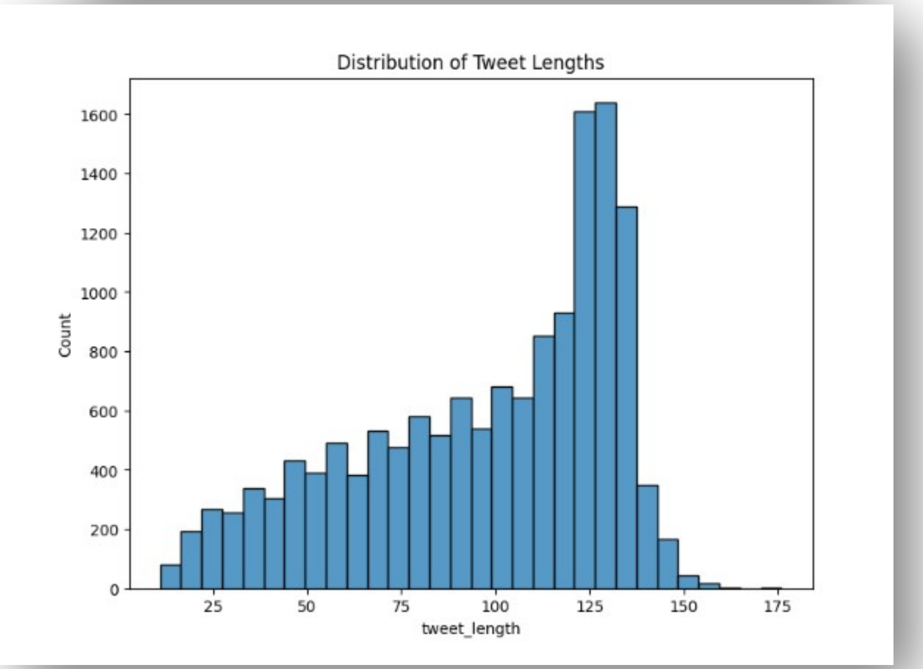
```
#Evaluate the model
accuracy = accuracy_score(y_test, y_pred)
precision, recall, f1_score, _ = precision_recall_fscore_support(y_test,
y_pred, average=’weighted’)

#Print the evaluation metrics
print(“Accuracy:”, accuracy)
print(“Precision:”, precision)
print(“Recall:”, recall)
print(“F1 Score:”, f1_score)

#Print the classification report and confusion matrix
print(“\nClassification Report:\n”,  classification_report(y_test, y_pred))
print(“\nConfusion Matrix:\n”, confusion_matrix(y_test, y_pred))
```

OUTPUT:





```
[{'label': 'negative', 'score': 0.36814033398513794}]
[{'label': 'netural', 'score': 0.4006020665168762}]
[{'label': 'positive', 'score': 0.502895987033844}]
```

Conclusion:

Sentiment analysis for marketing is a valuable tool for understanding customer perceptions of competitor products. By following the outlined design thinking process, we can gather, preprocess, analyze, and visualize customer feedback data to derive meaningful insights that drive informed business decisions and marketing strategies. The successful implementation of this project can provide a competitive advantage and contribute to product enhancement and customer satisfaction.

In this implementation, based on the process and evaluation performed, the Support Vector Machine (SVM) model demonstrates a reasonable performance in sentiment analysis on the given dataset. The evaluation metrics indicates a decent accuracy, precision, recall, and F1-score.

Sentiment analysis of Twitter US airline data using the BERT model is a powerful and effective tool for understanding customer opinions and emotions in the airline industry. This approach allows airlines to gain valuable insights into passenger sentiment, which can be pivotal for various aspects of their operations and customer service:

1**. Improved Customer Service**: By monitoring sentiment, airlines can proactively address customer concerns and issues, leading to better customer experiences.

2. **Crisis Management**: Sentiment analysis using BERT can help airlines identify and respond to potential PR crises quickly.

3**. Marketing and Campaigns**: Airlines can fine-tune their marketing strategies based on the sentiments expressed by customers on social media, enabling more targeted and resonant campaigns.

4. **Product and Service Enhancement**: Understanding customer sentiment provides valuable feedback for improving

in-flight services, amenities, and operational aspects.

5. **Real-time Feedback Loop**: The use of BERT in sentiment analysis ensures that airlines have access to real-time feedback, enabling them to adapt swiftly to customer preferences and concerns.

In essence, sentiment analysis using BERT is a vital tool for airlines to gauge and react to customer sentiment, thereby enhancing customer satisfaction, refining marketing strategies, and ultimately improving their overall services. It demonstrates the power of NLP and machine learning in gaining insights from vast social media data.

In conclusion, this project represents the convergence of technology, data, and marketing, ushering in a new era of customer-centric marketing strategies that are informed by the voice of our customers.