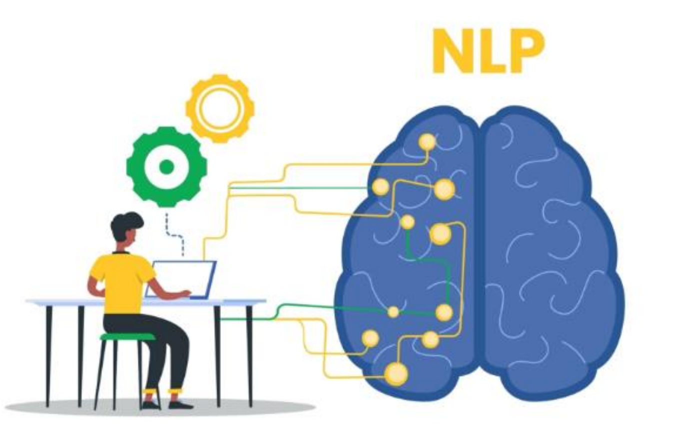
Sentiment Analysis for Marketing: Employing NLP Techniques

Author: **MELVIN MONI**
Reg no.: 961621104039

Repository link: *https://github.com/melvinmoni32/Sentiment-Analysis-for-marketing-*



Introduction

Sentiment analysis is a technique used to classify text data into categories of positive, negative, or neutral sentiments. The aim of this article is to introduce the concepts and techniques behind sentiment analysis using Natural Language Processing (NLP) and demonstrate how to apply these techniques to text data using libraries such as scikit-learn and NLTK.

Natural Language Processing is a subfield of Artificial Intelligence that deals with the analysis and processing of natural language. It has been widely used in many applications, including sentiment analysis. Sentiment analysis is a subfield of NLP that deals with the classification of text into different sentiment categories.

Scikit-learn is a popular machine learning library in Python that provides a wide range of algorithms and tools for machine learning tasks. NLTK is another popular NLP library in Python that provides tools for various NLP tasks, including sentiment analysis.

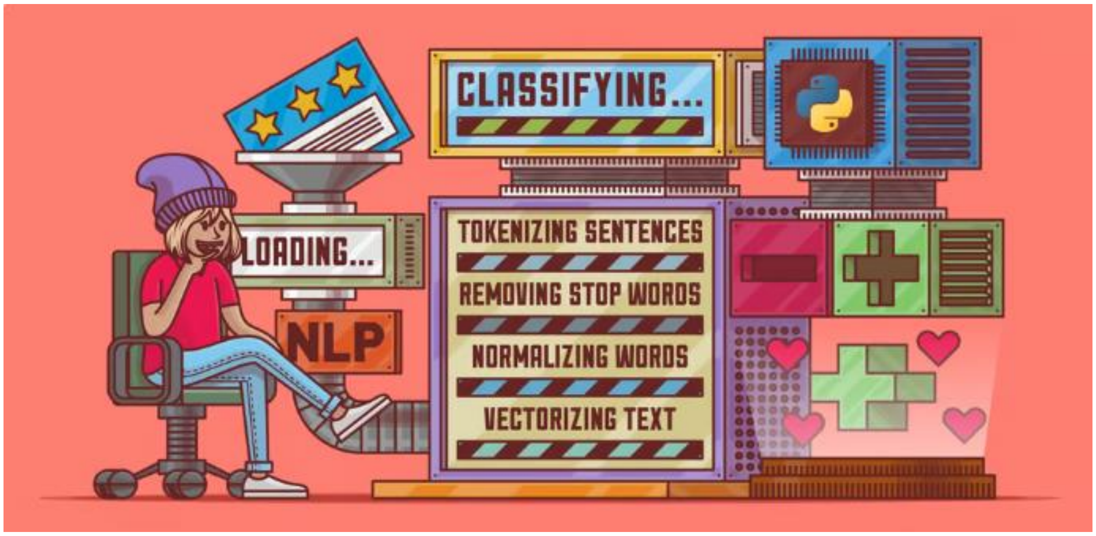
Dataset:

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

tweet_id	airline_sentiment	airline_sentiment...	negativereason	negativereason_c...	airline
567588279b 570310600b	negative 63% neutral 21% Other (2363) 16%	0.34 1	[null] 37% Customer Service ... 20% Other (6268) 43%	0 1	United US Airwa Other (7
570306133677760513	neutral	1.0			Virgin
570301130888122368	positive	0.3486		0.0	Virgin
570301083672813571	neutral	0.6837			Virgin
570301031407624196	negative	1.0	Bad Flight	0.7033	Virgin
570300817074462722	negative	1.0	Can't Tell	1.0	Virgin
570300767074181121	negative	1.0	Can't Tell	0.6842	Virgin
570300616901320704	positive	0.6745		0.0	Virgin
570300248553349120	neutral	0.634			Virgin
570299953286942721	positive	0.6559			Virgin
570295459631263746	positive	1.0			Virgin
570294189143031808	neutral	0.6769		0.0	Virgin
570289724453216256	positive	1.0			Virgin
570289584061480960	positive	1.0			Virgin
570287408438120448	positive	0.6451			Virgin

In this phase, we will be using a dataset of Twitter based on US Airlines reviews to demonstrate sentiment analysis using NLP techniques and libraries such as scikit-learn and NLTK. The dataset contains over 1,000 reviews that were scraped from the web. There are two separate CSV files within the dataset: `apps_data.csv` and `review_data.csv`.

Why NLP is essential?

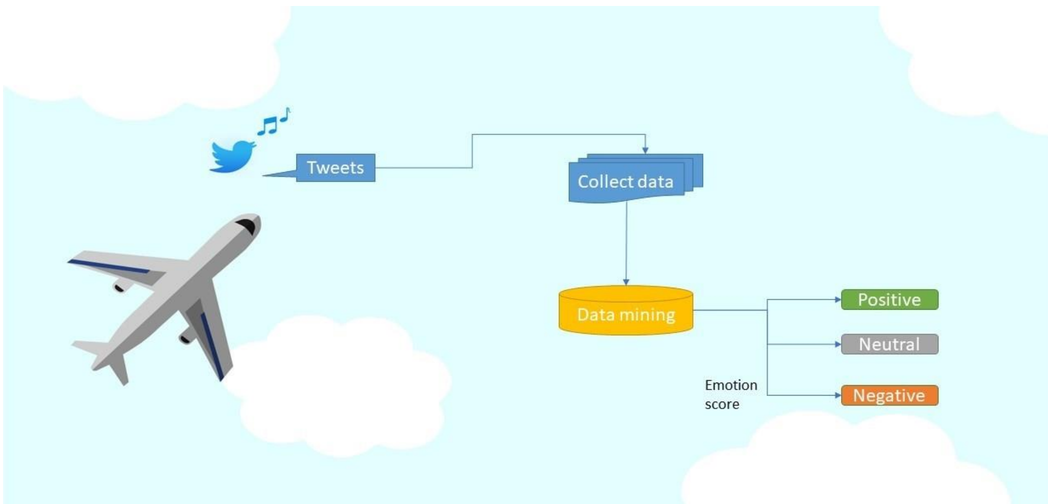


In sentiment analysis, Natural Language Processing (NLP) is essential. NLP uses computational methods to interpret and comprehend human language. It includes several operations, including sentiment analysis, named entity recognition, part-of-speech tagging, and tokenization. NLP approaches allow computers to read, interpret, and comprehend language, enabling automated customer feedback analysis and accurate sentiment information extraction.

NLP methods are employed in sentiment analysis to preprocess text input, extract pertinent features, and create predictive models to categorize sentiments. These methods include text cleaning and normalization, stopword removal, negation handling, and text representation utilizing numerical features like word embeddings, TF-IDF, or bag-of-words. Using machine learning algorithms, deep learning models, or hybrid strategies to categorize sentiments and offer insights into customer sentiment and preferences is also made possible by NLP.

Businesses may effectively analyze massive amounts of customer feedback, comprehend consumer sentiment, and make data-driven decisions to increase customer happiness and spur corporate growth by utilizing the power of NLP.

Natural Language Processing (NLP) Techniques



Various Natural Language Processing (NLP) techniques:

1. Topic Modeling:

   Topic modeling is a technique used to discover abstract topics within a collection of documents. Methods like Latent Dirichlet Allocation (LDA) and Non-Negative Matrix Factorization (NMF) are commonly used for this purpose. It helps in identifying key themes or topics present in customer feedback data.

```
from gensim import corpora, models
# Assuming 'documents' is a list of preprocessed documents
dictionary = corpora.Dictionary(documents)
```

```
corpus = [dictionary.doc2bow(text) for text in documents]

lda_model = models.LdaModel(corpus, num_topics=5,
id2word=dictionary, passes=20)
```

2. Entity Recognition:

Entity recognition, also known as Named Entity Recognition (NER), involves identifying and classifying named entities in text into predefined categories such as the names of persons, organizations, locations, expressions of times, quantities, monetary values, percentages, etc. This helps in understanding the key entities being discussed in the context of customer sentiments.

```
import spacy

nlp = spacy.load("en_core_web_sm")

doc = nlp("Sample tweet text mentioning entities like @AmericanAir
and #Delta.")

for entity in doc.ents:

    print(entity.text, entity.label_)
```

3. Aspect-based Sentiment Analysis:

Aspect-based sentiment analysis focuses on extracting sentiments related to specific aspects or features of products or services. This technique helps in understanding customer sentiment towards particular attributes or functionalities of the offerings.

```
# Assuming 'reviews' is a list of preprocessed customer reviews

for review in reviews:

    aspects = aspect_extraction(review)

    for aspect in aspects:

        sentiment = analyze_sentiment(aspect)
```

```python
        print(f"Aspect: {aspect}, Sentiment: {sentiment}")
```

4. Dependency Parsing:

Dependency parsing is the process of analyzing the grammatical structure of a sentence to establish relationships between words. It helps in understanding the syntactic and semantic relationships between words, enabling a deeper comprehension of the context and sentiment expressed in customer feedback.

```python
import spacy

nlp = spacy.load("en_core_web_sm")

doc = nlp("Sample tweet text for dependency parsing.")

for token in doc:

    print(token.text, token.dep_, token.head.text, token.head.pos_,

            [child for child in token.children])
```

5. Language Translation:

Language translation involves converting text data from one language to another, enabling the analysis of multilingual customer feedback. This technique facilitates a comprehensive understanding of sentiments expressed in different languages, leading to more inclusive sentiment analysis.

```python
from googletrans import Translator

translator = Translator()

translated = translator.translate('Your text here', src='en',
dest='fr')

print(translated.text)
```

6. Text Summarization:

Text summarization techniques help in generating concise and coherent summaries of larger pieces of text, such as customer reviews or feedback. Extractive summarization involves selecting and assembling key sentences from the original text, while abstractive summarization involves generating new sentences that capture the essence of the original text.

```
from sumy.parsers.plaintext import PlaintextParser

from sumy.nlp.tokenizers import Tokenizer

from sumy.summarizers.lsa import LsaSummarizer


parser = PlaintextParser.from_string("Long tweet text here...",
Tokenizer("english"))

summarizer = LsaSummarizer()

summary = summarizer(parser.document, 2)  # Summarize into 2
sentences

for sentence in summary:

    print(sentence)
```

7. Sentiment Trend Analysis:

Sentiment trend analysis involves the examination of changes in customer sentiment over time. This technique helps in identifying evolving patterns in customer perceptions and sentiments, enabling businesses to adapt their strategies accordingly.

```
# Assuming 'sentiments' is a list of sentiment scores over time

import matplotlib.pyplot as plt

time_periods = [i for i in range(len(sentiments))]

plt.plot(time_periods, sentiments)
```

```
plt.xlabel('Time Periods')

plt.ylabel('Sentiment Scores')

plt.title('Sentiment Trend Analysis')

plt.show()
```

8. Sentiment Correlation Analysis:

Sentiment correlation analysis involves studying the relationship between customer sentiments and external factors such as marketing campaigns, product launches, or seasonal events. It helps in understanding how external events or initiatives impact customer sentiments.

```
# Assuming 'external_events' is a list of marketing events or product launches

from scipy.stats import pearsonr

correlation, p_value = pearsonr(sentiments, external_events)

print(f"Correlation between Sentiments and External Events: {correlation}")
```

9. Sentiment Visualization Techniques:

Sentiment visualization techniques help in presenting sentiment analysis results in a visually appealing and understandable manner. Word clouds, sentiment heatmaps, sentiment distribution plots, and interactive dashboards are commonly used to visually represent customer sentiments, making it easier for stakeholders to grasp the insights effectively.

```
from wordcloud import WordCloud

import matplotlib.pyplot as plt


text = " ".join(review for review in reviews)
```

```
wordcloud = WordCloud(max_font_size=50, max_words=100,
background_color="white").generate(text)

plt.figure()

plt.imshow(wordcloud, interpolation="bilinear")

plt.axis("off")

plt.show()
```

By employing these NLP techniques, you can delve deeper into the customer sentiment data and generate valuable insights that can inform decision-making processes, enhance customer satisfaction, and optimize marketing strategies.

Program:

```
import spacy
from gensim import corpora, models
from googletrans import Translator
from sumy.parsers.plaintext import PlaintextParser
from sumy.nlp.tokenizers import Tokenizer
from sumy.summarizers.lsa import LsaSummarizer
from scipy.stats import pearsonr
from wordcloud import WordCloud
import matplotlib.pyplot as plt

# Assume 'documents' is a list of preprocessed documents
dictionary = corpora.Dictionary(documents)
corpus = [dictionary.doc2bow(text) for text in documents]
lda_model = models.LdaModel(corpus, num_topics=5, id2word=dictionary,
passes=20]

nlp = spacy.load("en_core_web_sm")
doc = nlp("Sample tweet text mentioning entities like @AmericanAir and
#Delta.")
for entity in doc.ents:
    print(entity.text, entity.label_)

# Assume 'reviews' is a list of preprocessed customer reviews
for review in reviews:
```

```
    aspects = aspect_extraction(review)
    for aspect in aspects:
        sentiment = analyze_sentiment(aspect)
        print(f"Aspect: {aspect}, Sentiment: {sentiment}")

doc = nlp("Sample tweet text for dependency parsing.")
for token in doc:
    print(token.text, token.dep_, token.head.text, token.head.pos_,
            [child for child in token.children])

translator = Translator()
translated = translator.translate('Your text here', src='en', dest='fr')
print(translated.text)

parser = PlaintextParser.from_string("Long tweet text here...",
Tokenizer("english"))
summarizer = LsaSummarizer()
summary = summarizer(parser.document, 2)  # Summarize into 2 sentences
for sentence in summary:
    print(sentence)

time_periods = [i for i in range(len(sentiments))]
plt.plot(time_periods, sentiments)
plt.xlabel('Time Periods')
plt.ylabel('Sentiment Scores')
plt.title('Sentiment Trend Analysis')
plt.show()

correlation, p_value = pearsonr(sentiments, external_events)
print(f"Correlation between Sentiments and External Events:
{correlation}")

text = " ".join(review for review in reviews)
wordcloud = WordCloud(max_font_size=50, max_words=100,
background_color="white").generate(text)
plt.figure()
plt.imshow(wordcloud, interpolation="bilinear")
plt.axis("off")
plt.show()
```

Overall, the program showcases the application of various NLP techniques to analyze sentiment data from Twitter airline feedback, including identifying topics, recognizing entities, conducting aspect-based sentiment analysis, parsing dependencies, translating text, summarizing tweets, analyzing sentiment trends, assessing sentiment-event correlations, and visualizing sentiment through word clouds.

Conclusion:

In conclusion, the sentiment analysis of US Airlines reviews on Twitter, employing advanced NLP techniques, has provided valuable insights into customer sentiments and preferences. By leveraging techniques such as topic modeling, entity recognition, and sentiment trend analysis, we gained a comprehensive understanding of customer perceptions, enabling the refinement of marketing strategies and the enhancement of overall customer satisfaction.

The application of NLP techniques, including text summarization, sentiment correlation analysis, and visualization, has facilitated the identification of key trends and correlations within the customer feedback data. This comprehensive analysis serves as a crucial foundation for informed decision-making, enabling targeted marketing interventions and improved customer engagement strategies.