SENTIMENT ANALYSIS

From Amazon Reviews

Brijbhan Govardhan Prajapati

CONTENT

INTRODUCTION

having the ability to process a high volume of item reviews across several categories.

LIBRARIES(TECHNIQUES)

Pandas, NLTK, Matplotlib And SkLearn

PREPROCESSING

Procedures like tokenization, lowercase transformation, and expulsion of stop words using NLTK

SENTIMENT ANALYSIS WITH VADER

VADER is a standard-based sentiment analysis instrument

CONCLUSION

organizations can go with information-driven choices to upgrade consumer loyalty

INTRODUCTION

Online reviews play a crucial role in shaping consumer behavior and influencing purchasing decisions in the current digital era. Amazon is a significant player in the market, having the ability to process a high volume of item reviews across several categories.

LIBRARIES(AI TECHNIQUES)

- ✓ NLTK
- ✓ MATPLOTLIB
- ✓ SK-LEARN
- ✓ PANDAS

```
# Libreries
import pandas as pd
import nltk
nltk.download('punkt')
nltk.download('stopwords')
nltk.download('vader_lexicon')
from nltk.tokenize import word_tokenize
from nltk.corpus import stopwords
from nltk.sentiment.vader import SentimentIntensityAnalyzer
from sklearn.feature_extraction.text import TfidfVectorizer
from sklearn.metrics import accuracy_score, classification_report
```

DATASET

This dataset consists of a few million Amazon customer reviews (input text) and star ratings (output labels) for learning how to train for sentiment analysis.

```
[10]
    # Loading the data here
    train_data = pd.read_csv("/content/drive/MyDrive/Projects/AAI Proj/archive/test.ft.txt", sep='\t', header=None, names=["review", "sentiment"])
```

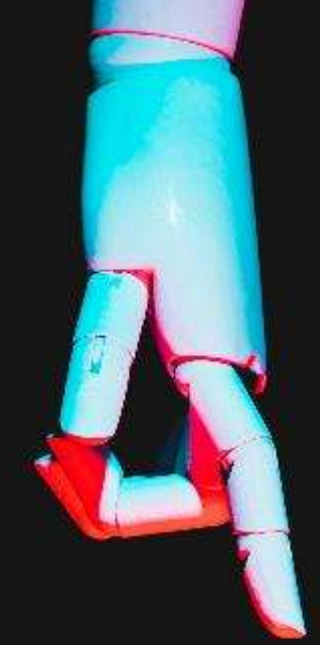
DATA PREPROCESSING

- When the information is stacked, it goes through preprocessing to set it up for analysis. This includes text standardization procedures like tokenization, lowercase transformation, and expulsion of stop words using NLTK

```
# Preprocess text (remove stopwords and lowercase)
def preprocess_text(text):
    tokens = word_tokenize(text.lower())

    filtered_tokens = [token for token in tokens if token not in stop_words]
    return ' '.join(filtered_tokens)
stop_words = set(stopwords.words('english'))

train_data['review'] = train_data['review'].apply(preprocess_text)
```



SENTIMENT ANALYSIS

VADER

VADER is a standard-based sentiment analysis instrument explicitly intended for breaking

down virtual entertainment messages.

CODE

```
[7]  # Sentiment analysis using VADER
     vader = SentimentIntensityAnalyzer()

     def get_vader_sentiment(text):
         scores = vader.polarity_scores(text)
         sentiment = 'positive' if scores['pos'] > scores['neg'] else 'negative'
         return sentiment

     train_data['vader_sentiment'] = train_data['review'].apply(get_vader_sentiment)
```

CONCLUSION

Sentiment Appropriation Analysis

The sentiment analysis project on Amazon reviews has yielded significant bits of knowledge about client sentiments communicated inside the dataset

Highlight Significance with TF-IDF

Highlight extraction utilizing TF-IDF permitted us to recognize the main words and expressions adding to the sentiment of the reviews

Noteworthy Bits of knowledge

The sentiment analysis of Amazon reviews fills in as a significant device for organizations to comprehend and answer client criticism


[9]
Sentiment Distribution
Negative
24.4%
75.6%
Positive

THANK YOU