TASK 6

Data Science Example

Sentiment Analysis

for Movie Reviews Using Python

Name: Nishtha Singh (19BCE0037)

Email: nishtha.singh2019@vitstudent.ac.in

College: VIT VELLORE

About The Project

Sentiment analysis is the process of analyzing and categorizing opinions conveyed in a piece of text as positive, negative, or neutral. In this code, sentiment analysis is performed on a dataset consisting of movie reviews and its executed using Python in jupyter notebook.

The dataset used for this task consists of 20 reviews, with the sentiment labels 'positive' and 'negative'.

The algorithms used in this code are Naive Bayes algorithm and MultinomialNB algorithm.

Naive Bayes algorithm is a probabilistic algorithm that uses Bayes' theorem to classify text.

The MultinomialNB algorithm is then used to train the model on the training data. It's designed for text classification tasks and is commonly used in sentiment analysis.

Steps

1. The first part of the code imports the necessary libraries and loads the dataset. The 'positive' sentiment labels are changed to 1 and the 'negative' labels to 0.

2. The dataset is then split into training and testing sets, with 80% of the data used for training.

3. The text data is then vectorized using the CountVectorizer class from scikit-learn. Then this class is used to convert text data into a matrix of word counts. Stop words are removed from the text data using the 'stop_words' parameter, which is set to a list of commonly occurring words such as 'the', 'a', 'an', etc.

4. The MultinomialNB class is then used to train the model on the training set. The model is then used to predict the sentiment of each review in the dataset.

5. For each review, the first 100 words of the review are printed, along with the top positive or negative words depending on the predicted sentiment.

6. The top words are obtained by sorting the features according to their log probabilities and selecting the top features. The 'reverse' parameter is used to sort the features in descending order. Stop words are excluded from the top words list.

7. At the end, we get the sentiment for every movie review in terms of positive and negative.

Code


```python
In [1]:  1  #Importing all the libraries
         2  import pandas as pd
         3  from sklearn.feature_extraction.text import CountVectorizer
         4  from sklearn.naive_bayes import MultinomialNB
         5  from sklearn.metrics import accuracy_score
         6  from sklearn.model_selection import train_test_split
         7  import numpy as np

In [2]:  1  # Loading the dataset for movie review
         2  # Dataset consists of 20 reviews
         3  data = pd.read_csv('Movie_Review.csv')

In [3]:  1  # Convert 'positive' sentiment labels to 1 and 'negative' sentiment labels to 0
         2  data['sentiment'] = data['sentiment'].map({'positive': 1, 'negative': 0})
         3

In [4]:  1  # Splitting the dataset into training and testing
         2  train_data, test_data, train_labels, test_labels = train_test_split(data['review'], data['sentiment'], test_size=0.
         3

In [5]:  1  # Vectorize the text data
         2
         3  my_stop_words = ['the', 'a', 'an', 'and', 'of', 'to','this','that','too']
         4  vectorizer = CountVectorizer(stop_words=my_stop_words)
         5  train_vectors = vectorizer.fit_transform(train_data)
         6  test_vectors = vectorizer.transform(test_data)

In [6]:  1  # Training the model
         2  clf = MultinomialNB()
         3  clf.fit(train_vectors, train_labels)

Out[6]: ▾ MultinomialNB
        MultinomialNB()
```

```python
        12          print( rositive review.  , review[:100])
        13          log_prob_positive = clf.feature_log_prob_[1]
        14          coefs_with_fns = sorted(list(zip(log_prob_positive, feature_names)), reverse=True)
        15          top_positive = [x for x in coefs_with_fns if x[1] not in my_stop_words and x[1].isalpha() and len(x[1]) >=
        16          top_positive_review = sorted(list(zip(vector.toarray()[0], feature_names)), reverse=True)[:topn]
        17          top_positive_review_words = [x[1] for x in top_positive_review if x[1] not in my_stop_words and x[1].isalph
        18          print("Top positive words:", top_positive_review_words)
        19
        20      else:
        21          print("Negative review: ", review[:100])
        22          log_prob_negative = clf.feature_log_prob_[0]
        23          coefs_with_fns = sorted(list(zip(log_prob_negative, feature_names)))
        24          top_negative_review = sorted(list(zip(vector.toarray()[0], feature_names)), reverse=True)[:topn]
        25          top_negative_review_words = [x[1] for x in top_negative_review if x[1].isalpha() and len(x[1]) > 3 and not
        26          print("Top negative words:", top_negative_review_words)
        27      print("="*80)
        28
```

```
Positive review:  I loved Pathaan movie. One of best movies of SRK after his romance era. Action and plotting everyth
i
Top positive words: ['romance', 'plotting', 'pathaan', 'movies', 'movie', 'loved']
================================================================================
Negative review:  I watch a lot of movies and I like to give them all a chance just in case there is something intere
s
Top negative words: ['just', 'would', 'with', 'wish', 'watch', 'warrant']
================================================================================
Negative review:  Well. What did I just watch ? I'm not sure if I can find words to describe this train wreck.So, a r
a
Top negative words: ['there', 'script', 'because', 'what', 'well', 'watch', 'take', 'sure', 'still']
================================================================================
Positive review:  I thought the cast was great. Brianna and Emma were exceptionally talented in thier characters. Fun
f
Top positive words: ['thought', 'thier', 'talented', 'great', 'exceptionaly']
================================================================================
Negative review:  I rarely stop watching a movie although how crappy it is. Well for this one I made an exception sin
c
Top negative words: ['well', 'watching', 'stop', 'skip', 'since', 'reviews', 'rarely']
```

Results

Positive Review with top positive words

Negative words with top negative words

```
Positive review:  I loved Pathaan movie. One of best movies of SRK after his romance era. Action and plotting everyth
i
Top positive words: ['romance', 'plotting', 'pathaan', 'movies', 'movie', 'loved']
=======================================================================
Negative review:  I watch a lot of movies and I like to give them all a chance just in case there is something intere
s
Top negative words: ['just', 'would', 'with', 'wish', 'watch', 'warrant']
=======================================================================
Negative review:  Well. What did I just watch ? I'm not sure if I can find words to describe this train wreck.So, a r
a
Top negative words: ['there', 'script', 'because', 'what', 'well', 'watch', 'take', 'sure', 'still']
=======================================================================
Positive review:  I thought the cast was great. Brianna and Emma were exceptionaly talented in thier characters. Fun
f
Top positive words: ['thought', 'thier', 'talented', 'great', 'exceptionaly']
=======================================================================
Negative review:  I rarely stop watching a movie although how crappy it is. Well for this one I made an exception sin
c
Top negative words: ['well', 'watching', 'stop', 'skip', 'since', 'reviews', 'rarely']
```

```
i
Top positive words: ['world', 'design', 'worth', 'watch', 'thrillers', 'thoroughly', 'their']
=======================================================================
Negative review:  No grip in story. Pathetic acting by main actress. The guy with her more worse
Top negative words: ['watch', 'such', 'still', 'shot']
=======================================================================
Positive review:  Ayushmann Khurrana is excellent as the leading man as he charms his way throug
Top positive words: ['movie', 'action', 'which', 'through', 'their']
=======================================================================
Negative review:  I'm not sure what accomplished director/producer/cinematographer Joshua Caldwe
g
Top negative words: ['writing', 'what', 'joshua', 'writer', 'would', 'when']
=======================================================================
Positive review:  Easily the best scream movie since the original. In every way (but one - expla
r
Top positive words: ['scream', 'scary', 'intensity', 'things']
=======================================================================
Positive review:  Though the premise of 4 old friends setting out on Everest Base Camp at a very
s
Top positive words: ['would', 'setting', 'middle', 'class']
```