Vivek Poojari
2019130050
TE COMPS
AIML Lab

Experiment 4

Aim: To train and test machine learning models using naive bayes algorithm.

Theory:

- The Bayes' Theorem is used to create a collection of classification algorithms known as Naive Bayes classifiers. It is a family of algorithms that share a similar idea, namely that each pair of features being classified is independent of the others. - The Naive Bayes assumption is that each feature contributes equally and independently to the outcome. - The Bayes' Theorem calculates the likelihood of an event occurring given the probability of a previous event.

Code:

```
##Loading all the required libraries
import pandas as pd
import numpy as np

from sklearn.naive_bayes import MultinomialNB

from sklearn.feature_extraction.text import CountVectorizer
from sklearn.model_selection import train_test_split

from sklearn.metrics import accuracy_score
import time



#library for regular expretion
import re



#Loading the data which is from Kaggle
#This dataset contains four news class -b for business, e- entertainment, -t for technol
data = pd.read_csv("first_batch.csv",encoding='latin-1')

data.tail()
```

```
data['TITLE'].replace('', np.nan, inplace=True)
data.tail()

data = data[['TITLE','CATEGORY']]

#Printing all different types of categories
data.CATEGORY.unique()

#converting category column into numeric target NUM_CATEGORY column
data['NUM_CATEGORY']=data.CATEGORY.map({'b':0,'e':1,'t':2})
data.head()

#Splitting dataset into 60% training set and 40% test set
x_train, x_test, y_train, y_test = train_test_split(data.TITLE, data.NUM_CATEGORY, rando

vect = CountVectorizer(ngram_range=(2,2))
#converting traning features into numeric vector
X_train = vect.fit_transform(x_train)
#converting training labels into numeric vector
X_test = vect.transform(x_test)

#Training and Predicting the data
mnb = MultinomialNB(alpha =0.2)
```

```
#Training and Predicting the data
mnb = MultinomialNB(alpha =0.2)

mnb.fit(X_train,y_train)

result= mnb.predict(X_test)
print(result)

#Printing accuracy of the our model
accuracy_score(result,y_test)
```

Output:

```
0.9411764705882353
```

Conclusion:

The motive of this experiment was to implement the Naïve Bayes Algorithm and train a machine learning model which helps to classify news into 4 categories namely – business, entertainment, technology, and medical. The likelihood of an event occurring in relation to any condition is described by Bayes' theorem. In the naive bayes method, we calculate the probability of each output category and choose the one with the highest probability. The dataset which was used in this model was taken from kaggle, named the news aggregator dataset. To train the model, the data set was split into an 60:40 ratio and was trained on the 60% of data and the rest was kept as test data. With this model we can predict the news category with an accuracy around 94 percent.