Naive_Bayes_Classifier

August 2, 2023

1 Naive Bayes Classifier (NBC) - Email Similarity Project

1.1 What is the Naives Bayes Classifier (NBC)?

-> An NBC is a supervised machine learning algorithm that leverages Bayes' Theorem to make predictions and classifications. It is often used majorly for text classification

-> The Bayes' Theorem is based on a branch of statistics called Bayesian Statistics, where we take prior knowledge into account before calculating new probabilities. The mathematical expression is shown below for two events `A` and `B`

$$P(A|B) = \frac{P(B|A).P(A)}{P(B)}$$

-> In order to use the above equation as a classifier, we replace `B` with the data point and `A` with the class or label.

-> A layman does not need to worry much about the internal workings of the classifier, especially since python has well-developed libraries that can handle all the abstractions. And I have used those libraries in these project

1.2 Project Description

I will be using the Naive Bayes to try to distinguish between different types of emails. For example, figure out which emails are about soccer and which emails are about hockey.

[86]:
```
#First Let's Explore the Dataset
# Step1 - import the email dataset and the neccessary libraries
from sklearn.datasets import fetch_20newsgroups
from sklearn.naive_bayes import MultinomialNB
from sklearn.feature_extraction.text import CountVectorizer
import logging
import sys

#Creating a logging object to help display some outputs
logger = logging.getLogger(__name__)

# Remove existing handlers, if any
logger.handlers = []
```

```
#Creating a stream handler to output messages to the console
stream_handler = logging.StreamHandler(sys.stdout)

#add the stream handler to the logger
logger.addHandler(stream_handler)

#set the logging level to info - to disable the logging use logging.WARNING␣
 ↪instead of logging.INFO
logger.setLevel(logging.INFO)

#See the different email categories
emails = fetch_20newsgroups()
logger.info(emails.target_names)
```

```
['alt.atheism', 'comp.graphics', 'comp.os.ms-windows.misc',
'comp.sys.ibm.pc.hardware', 'comp.sys.mac.hardware', 'comp.windows.x',
'misc.forsale', 'rec.autos', 'rec.motorcycles', 'rec.sport.baseball',
'rec.sport.hockey', 'sci.crypt', 'sci.electronics', 'sci.med', 'sci.space',
'soc.religion.christian', 'talk.politics.guns', 'talk.politics.mideast',
'talk.politics.misc', 'talk.religion.misc']
```

[87]:
```
#Step2 - Let's focus on distinguishing between baseball email and hockey email␣
 ↪using the NBC
emails = fetch_20newsgroups(categories = ['rec.sport.baseball', 'rec.sport.
 ↪hockey'])

#Step3 - View one of the emails, stored in the .data object
logger.info(emails.data[5])
```

```
From: mmb@lamar.ColoState.EDU (Michael Burger)
Subject: More TV Info
Distribution: na
Nntp-Posting-Host: lamar.acns.colostate.edu
Organization: Colorado State University, Fort Collins, CO  80523
Lines: 36

United States Coverage:
Sunday April 18
  N.J./N.Y.I. at Pittsburgh - 1:00 EDT to Eastern Time Zone
  ABC - Gary Thorne and Bill Clement

  St. Louis at Chicago - 12:00 CDT and 11:00 MDT - to Central/Mountain Zones
  ABC - Mike Emerick and Jim Schoenfeld

  Los Angeles at Calgary - 12:00 PDT and 11:00 ADT - to Pacific/Alaskan Zones
  ABC - Al Michaels and John Davidson

Tuesday, April 20
```

```
  N.J./N.Y.I. at Pittsburgh - 7:30 EDT Nationwide
  ESPN - Gary Thorne and Bill Clement

Thursday, April 22 and Saturday April 24
  To Be Announced - 7:30 EDT Nationwide
  ESPN - To Be Announced


Canadian Coverage:

Sunday, April 18
  Buffalo at Boston - 7:30 EDT Nationwide
  TSN - ???

Tuesday, April 20
  N.J.D./N.Y. at Pittsburgh - 7:30 EDT Nationwide
  TSN - ???

Wednesday, April 21
  St. Louis at Chicago - 8:30 EDT Nationwide
  TSN - ???
```

[88]:
```
#Step4 - View the corresponding label of the datapoint at index 5 above
#note that this labels are also called the targets
logger.info('The label of the email datapoint at index 5 is {y}'.
 ↪format(y=emails.target[5]))
logger.info('The name of that label of the email datapoint at index 5 is {y}'.
 ↪format(y=emails.target_names[emails.target[5]]))
```

```
The label of the email datapoint at index 5 is 1
The name of that label of the email datapoint at index 5 is rec.sport.hockey
```

[89]:
```
#Now, Lets Make the Training and Testing Datasets

#Step5 - split the dataset into the training and test sets
#Note that the dataset already has these subsets available. Also by using a␣
 ↪non-zero random_state ensures the data is split the same way everytime we␣
 ↪run the code
train_emails = fetch_20newsgroups(categories = ['rec.sport.baseball', 'rec.
 ↪sport.hockey'], subset = 'train', shuffle = True, random_state = 108)

#Step6 - create the test dataset
test_emails = fetch_20newsgroups(categories = ['rec.sport.baseball', 'rec.sport.
 ↪hockey'], subset = 'test', shuffle = True, random_state = 108)
```

[90]:
```
#Next, lets transform the email datapoints into word counts using the␣
 ↪CountVectorizer class

#Step7 - Create a CountVectorizer Object
counter = CountVectorizer()

#Step8 - Inform the object about the words in our emails by fitting the counter␣
 ↪objects to the words in our emails
counter.fit(train_emails.data+test_emails.data)

#Step9 - Make a list of the counts of words in the training and training dataset
train_counts = counter.transform(train_emails.data)

#Step10 - testing
test_counts = counter.transform(test_emails.data)
```

[91]:
```
#Next, lets Make the Naive Bayes Classifier(NBC)

#Step11 - create an NBC object
classifier = MultinomialNB()

#Step12 - fit the training data to the NBC object
classifier.fit(train_counts,train_emails.target)

#Step13 - Test the NBC by printing the accuracy of the classifier on the test␣
 ↪set
logger.info('The NBC has an accuracy of {score}% on the test dataset'.
 ↪format(score=round(classifier.score(test_counts,test_emails.target)*100,4)))
```

```
The NBC has an accuracy of 97.2362% on the test dataset
```