email_filter

September 14, 2021

1 Email Classifier

In this project we will make a classifier to see if we can predict the email type based on its content.

1.1 Imports

We only need sklearn for this functionality and dataset.

```
[1]: from sklearn.datasets import fetch_20newsgroups
     from sklearn.naive_bayes import MultinomialNB
     from sklearn.feature_extraction.text import CountVectorizer
```

1.2 Investigating the Data

We'll take look at the available newsgroups and emails.

```
[2]: # checking the available groupings
     emails = fetch_20newsgroups()
     print(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']
```

Now lets see if based on the content, we can tell the difference between an email about baseball or hockey.

```
[9]: # selecting the categories
     bh_email = fetch_20newsgroups(categories=['rec.sport.baseball', 'rec.sport.
      ↪hockey'])

     # browsing an email
     print('Example Email:\n')
     print(bh_email.data[5])
```

```
Example Email:
```

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 - ???

1.3 Building the Model

Now we can make our classifier and split our data. We will make a set to train off of, and a set to test off of. We can set a random state to maintain the same outputs across runs.

[4]:

```
# splitting data into training and test sets
train_emails = fetch_20newsgroups(categories=['rec.sport.baseball', 'rec.sport.
 ↪hockey'], subset='train', shuffle = True, random_state=1)
test_emails = fetch_20newsgroups(categories=['rec.sport.baseball', 'rec.sport.
 ↪hockey'], subset='test', shuffle = True, random_state=1)

# creating a CountVectorizer object
counter = CountVectorizer()

# telling counter what can exist in the emails
counter.fit(test_emails.data + train_emails.data)

train_counts = counter.transform(train_emails.data)
test_counts = counter.transform(test_emails.data)

# making the Naive Bayes classifier
classifier = MultinomialNB()
classifier.fit(train_counts, train_emails.target)
```

[4]:

```
MultinomialNB()
```

Now that we split our data and trained our model, let's see how it performs.

[10]:

```
# testing the accuracy
print(classifier.score(test_counts, test_emails.target))
```

```
0.9723618090452262
```

97% is pretty good!

We can apply the same methodology to any of these newgroup categories, or to make a spam filter for email or text messages.

Data Sources Data was provided by the sklearn package.