CS 3430: SciComp with Py
Assignment 11
Finding Relevant USENET News Group Posts

Vladimir Kulyukin
Department of Computer Science
Utah State University

April 8, 2017

1 Learning Objectives

1. Vector Space Model of Information Retrieval
2. Texts as Bags of Words
3. Finding Relevant Posts
4. Text Similarity Metrics
5. Vocabulary Normalization

2 Introduction

In this assignment, we will build a mini search engine to retrieve related posts from 20 USENET newsgroups from `sklearn.datasets`. These are real, unfiltered posts. The language in some posts may strike some as offensive. But, on the bright side, this is real-world, raw data.

3 Indexing USENET Data

There are several steps we need to complete to index the data so that it can be used for finding posts relevant to user queries. First, we need to load the USENET data, which is done as follows.

```
import sklearn.datasets
usenet_data = sklearn.datasets.fetch_20newsgroups()
```

Here is how we can get the raw text of the 10th post.

```
>>> usenet_data.data[10]
u'From: irwin@cmptrc.lonestar.org (Irwin Arnstein)\nSubject: Re: Recommendation on Duc\n
Summary: What\'s it worth?\n
Distribution: usa\n
Expires: Sat, 1 May 1993 05:00:00 GMT\n
Organization: CompuTrac Inc., Richardson TX\n
Keywords: Ducati, GTS, How much? \n
Lines: 13\n\n
I have a line on a Ducati 900GTS 1978 model with 17k on the clock.  Runs\n
very well, paint is the bronze/brown/orange faded out, leaks a bit of oil\n
and pops out of 1st with hard accel. The shop will fix trans and oil \n
leak.  They sold the bike to the 1 and only owner.  They want $3495, and\n
I am thinking more like $3K.  Any opinions out there?  Please email me.\n
Thanks.  It would be a nice stable mate to the Beemer.  Then I\'ll get\na jap bike and call myself
Axis Motors!\n\n-- \n
--------------------------------------------------------------------\n
"Tuba" (Irwin)      "I honk therefore I am"     CompuTrac-Richardson,Tx\nirwin@cmptrc.lonestar.org
DoD #0826          (R75/6)\n
--------------------------------------------------------------------\n'
```

Here is how we can get the names of the USENET groups.

```
>>> usenet_data.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']
```

Second, we need to define a class to normalize the vocabulary with both stoplisting and stemming. The class `StemmedCountVectorizer` is what we can use for this job.

```
from sklearn.feature_extraction.text import CountVectorizer
import argparse

import nltk.stem
english_stemmer = nltk.stem.SnowballStemmer('english')
class StemmedCountVectorizer(CountVectorizer):
    def build_analyzer(self):
        analyzer = super(StemmedCountVectorizer, self).build_analyzer()
        return lambda doc: (english_stemmer.stem(w) for w in analyzer(doc))
```

Third, we construct the counter vectorizer object and compute its feature matrix.

```
stemmed_vectorizer = StemmedCountVectorizer(min_df=1, stop_words='english')
usenet_data_feat_mat = stemmed_vectorizer.fit_transform(usenet_data.data)
num_samples, num_features = usenet_data_feat_mat.shape
print('#number of posts: %d, number of features: %d' % (num_samples, num_features))
```

Running the above produces the output below, which tells us that there are 11,314 posts and 110,992 features, i.e., terms.

```
#number of posts: 11314, number of features: 110992
```

You may have noticed that it takes a while to compute the fature map of 11,314 posts. So, it makes sense for us to persist the feature map and the vectorizer in pickle files whose names are given on the command line.

```
if __name__ == '__main__':
  ap = argparse.ArgumentParser()
  ap.add_argument('-feat_mat', '--feat_mat', required=True, help='pickle feat mat file')
  ap.add_argument('-vectorizer', '--vectorizer', required=True, help='pickle vectorizer file')
  args = vars(ap.parse_args())
  with open(args['feat_mat'], 'wb') as feat_mat_pck:
      pickle.dump(usenet_data_feat_mat , feat_mat_pck)
  with open(args['vectorizer'], 'wb') as vectorizer_pck:
      pickle.dump(stemmed_vectorizer, vectorizer_pck)
  print('indexing finished')
```

If we save the above code fragments in `index_usenet_posts.py`, we can index the USENET groups with the following call.

```
$ python index_usenet_posts.py -feat_mat usenet_feat_mat.pck -vectorizer usenet_vectorizer.pck
```

4 Finding Posts Relevant to User Queries

Implement the program `find_usenet_groups.py` that takes the paths to the persisted feature mat and the vectorizer, the user query, and the number of top matching posts to retrieve and prints out the top matching posts. For each top matching post, its number is printed along with its matching distance and its raw text. Note that the call below is directed into the text file `fuel_injector_query.txt`.

```
$ python find_usenet_posts.py -feat_mat usenet_feat_mat.pck -query 'is fuel injector cleaning necessary?'
-top_n 5 -vectorizer usenet_vectorizer.pck > fuel_injector_query.txt
```

Below is the beginning of `fuel_injector_query.txt`. The intial lines ending with `Searching over...` are just diagnostic messages. The most relevant post, if the normalized euclidean distance is used as a metric, is post number 8668 with the matcing distance of 1.016719. The raw text of this post follows.

```
Loading Usenet data
Usenet data loaded...
#num_posts: 11314, #features: 110992
Searching for usenet posts
Searching over...
Post #8668, matching distance=1.016719
Post text:
From: jwg@sedv1.acd.com ( Jim Grey)
Subject: Re: Necessity of fuel injector cleaning by dealership
Organization: Hell
Lines: 19

In article <1993Apr2.174850.6289@cbnewsl.cb.att.com> prm@cbnewsl.cb.att.com (paul.r.mount) writes:
>
>In your experience, how true is it that a fuel injector cleaning
>will do much more good than just using detergent gas.   While I
>agree that a clogged fuel injector would darken my day, how clogged
>do they get, and is $59 a good price (or can I do it myself by buying
>a can of ____ (what?) and doing ___ what?


A "fuel injector cleaning" at the dealer is probably little more than
them opening your gas tank, dumping in a bottle of fuel injector cleaner,
and sending you on your merry way $59 poorer.  Go to KMart and buy the
cleaner yourself for $1.29.

Just because you dealer sez you need it, don't mean it's necessarily so.
Be suspicious.

jim grey
jwg@acd4.acd.com

--------------------
Post #11185, matching distance=1.028127
Post text:
From: sekell@bb1t.monsanto.com
Subject: Re: Necessity of fuel injector cleaning by dealership
Article-I.D.: bb1t.1993Apr6.125537.1
Organization: Monsanto Company, St. Louis, MO
Lines: 29

In article <1993Apr6.131018.12873@acd4.acd.com>, jwg@sedv1.acd.com ( Jim Grey) writes:
> In article <1993Apr2.174850.6289@cbnewsl.cb.att.com> prm@cbnewsl.cb.att.com (paul.r.mount) writes:
>>
>>In your experience, how true is it that a fuel injector cleaning
>>will do much more good than just using detergent gas.   While I
>
> A "fuel injector cleaning" at the dealer is probably little more than
> them opening your gas tank, dumping in a bottle of fuel injector cleaner,
> and sending you on your merry way $59 poorer.  Go to KMart and buy the
> cleaner yourself for $1.29.

This should not be the case if they are at all reputable. Fuel injector
cleaning is done properly with a can of injector cleaner solvent which is
hooked up to the fuel system under high pressure. The car is actually run on
the solvent during the cleaning process. The equipment to properly do this is
pricey, and generally not something the average home mechanic has. The solvent
itself is not very expensive ($5-$8) and you could probably make up a hose to
fit your system and do it yourself, but I didn't tell you that... :-)

Not many in-tank cleaners are worth wasting your money on. There has been a
discussion of these products on here from time to time, and Chevron Techron
(not Pro-Gard with Techron) is generally regarded as the best. It is, however,
a bit more than $1.29 a bottle. IMHO, it will not substitute for proper
```

```
injector cleaning if they are really crudded up. You'll have to decide if the
$59 price is a better deal than spending your time and/or buying equipment to
do it.
  ------------------------------------------------------------------------
  Scott Keller +1 314 537 6317     The Agricultural Group of Monsanto Company
  sekell@bb1t.monsanto.com   KA0WCH packet: ka0wch@k0pfx.mo.usa.na

--------------------
```

The starter code in `find_usenet_posts.py` contains the stub for the function `find_top_n_closest_posts` that you should implement.

5 What To Submit

The zip archive `hw11.zip` contains two files `index_usenet_posts.py` and `find_usenet_posts.py`. You should not have to modify anything in `index_usenet_posts.py`. In `find_usenet_posts.py`, you should write your code for `find_top_n_closest_posts`. Submit your `find_usenet_posts.py` via Canvas. The files `diesel_engines_query.txt`, `fuel_injector_query.txt`, and `nhl_query.txt` contain the posts closest to my three sample queries. State in the comments at the head of your file how long it takes you to find related posts and your thoughts on speeding it up.

Happy Hacking!