ASSIGNMENT 8

CS 432 Spring 2017

APRIL 12, 2017

Michelle Graham

Q1:

To begin with, I created a set to contain the default url: https://www.blogger.com/next-blog?navBar=true&blogID=3471633091411211117. This is used to generate random blogs. I attempted to create a counter alongside each new blog to ensure that it was working properly. I also made an attempt at making use of the "r.status_code" function in the requests library to verify that each link was in fact "OK" or returned 200. However, this proved to be quite finicky so I kind of just left it alone for the time being. In any case, I went on to print each element of the set, along with the two links specified in the assignment description: http://f-measure.blogspot.com and http://ws-dl.blogspot.com. I appended the generated list of blogs to include these desired links. Also, some of my links were bad links. I simply re-ran my code. I deleted the bad links manually and inserted the newly created ones in their place.

In order to create a blog term matrix, I made use of the generatefeedvector.py file from the PCI book. I hope to find time to make the layout nicer by adjusting some things. For the moment, I have left it as it is due to time constraints. I also had to cover a Unicode error for some of my links.

getblogs.py:

```
import requests

f = open('urls.txt', 'w')
s = set()
while (len(s) < 100):
    url = "http://www.blogger.com/next-blog?navBar=true&blogID=3471633091411211117"
    r = requests.get(url)
    r.status_code == requests.codes.ok
    print('good link #') + str(r.status_code)
    update = r.url.strip('/?expref=next-blog')+('/feeds/posts/default?alt=rss'+'\n')
    s.add(update)
for element in s:
    print element
    f.write(element + '\n')
f.write('http://f-measure.blogspot.com' + '\n' + '\n')
f.write('http://ws-dl.blogspot.com')
```

Unicode error fix in generatefeedvector.py:

```
import re
import sys

reload(sys)
sys.setdefaultencoding('utf-8')
```

blogdata1.txt:

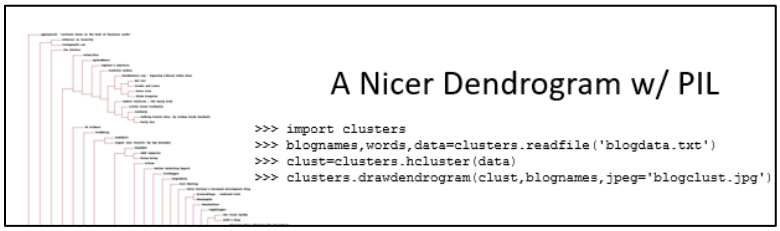
Blog	kids	golden	catchy	travel	wrong	fit	songwriter	service	needed	feeling
g	weekend	father	okay	filled	box	boy	bob	teenage	making	performance
eeping	science	beautiful		sense	imagine	number	relationship	video	play	plan
n	drums	island	videos	field	starting		fall	town	none	blood	late
face	fact	bring	decade	should	listened		means	packed	ends	pool	ghost
n	obvious	february		apparently		mid	mix	taking	add	although	
haven	rough	familiar		lucky	h	wire	etc	upon	less	paul	five
general	appearance		helped	thought	soft	alive	noise	brand	j	sleep	changed
bittersweet		1	0	0	0	5	1	0	0	2	2
0	4	1	1	0	5	1	2	1	0	0	0
12	3	2	0	0	0	1	8	0	0	2	0
adrianoblog		0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0
tumbleweed		1	0	0	0	1	0	0	1	4	1
0	0	0	0	0	0	0	0	0	1	0	0
0	0	0	1	0	0	0	0	0	0	1	2
the traveling neighborhood				0	0	0	1	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0
0	1	0	1	0	0	0	0	0	0	0	0
LOST	PLACES	0	0	0	0	1	0	0	0	0	0
15	0	0	3	1	0	1	0	1	0	0	0
1	2	0	1	0	0	0	1	1	0	0	0
the roofy leak		0	0	3	0	1	0	2	0	0	0
0	0	1	0	0	0	0	0	0	0	0	2

Q2:

The clustering PowerPoint from class was very helpful for this question. In particular, slides 12 and 13 were very helpful!

PPT code:

```
>>> import clusters
>>> blognames,words,data=clusters.readfile('blogdata.txt') # returns blog
titles, words in blog (10%-50% boundaries), list of frequency info
>>> clust=clusters.hcluster(data) # returns a tree of foo.id, foo.left,
foo.right
>>> clusters.printclust(clust,labels=blognames) # walks tree and prints
ascii approximation of a dendogram; distance measure is Pearson's r
-
 gapingvoid: "cartoons drawn on the back of business cards"
  -
    -
      Schneier on Security
      Instapundit.com
    -
      The Blotter
      -
```



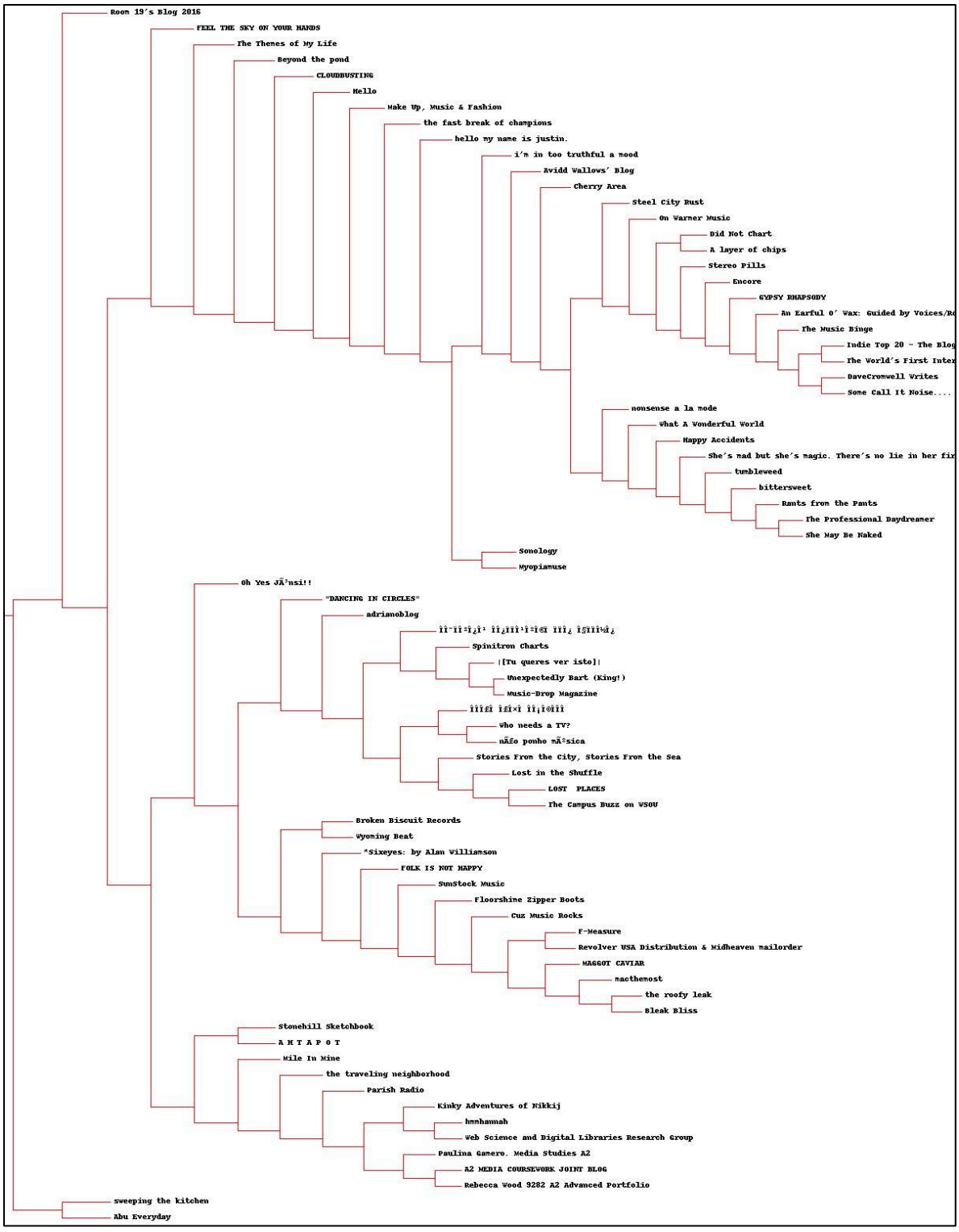
dendo.py:

```
import clusters
import sys
import codecs

blognames, words, data = clusters.readfile('blogdata1.txt')

clust = clusters.hcluster(data)

with codecs.open('ascii.text', 'w+', encoding ="ascii") as fout:
    clusters.printclust(clust, labels=blognames)

clusters.drawdendrogram(clust, blognames, jpeg='dendrogram.jpg')
```

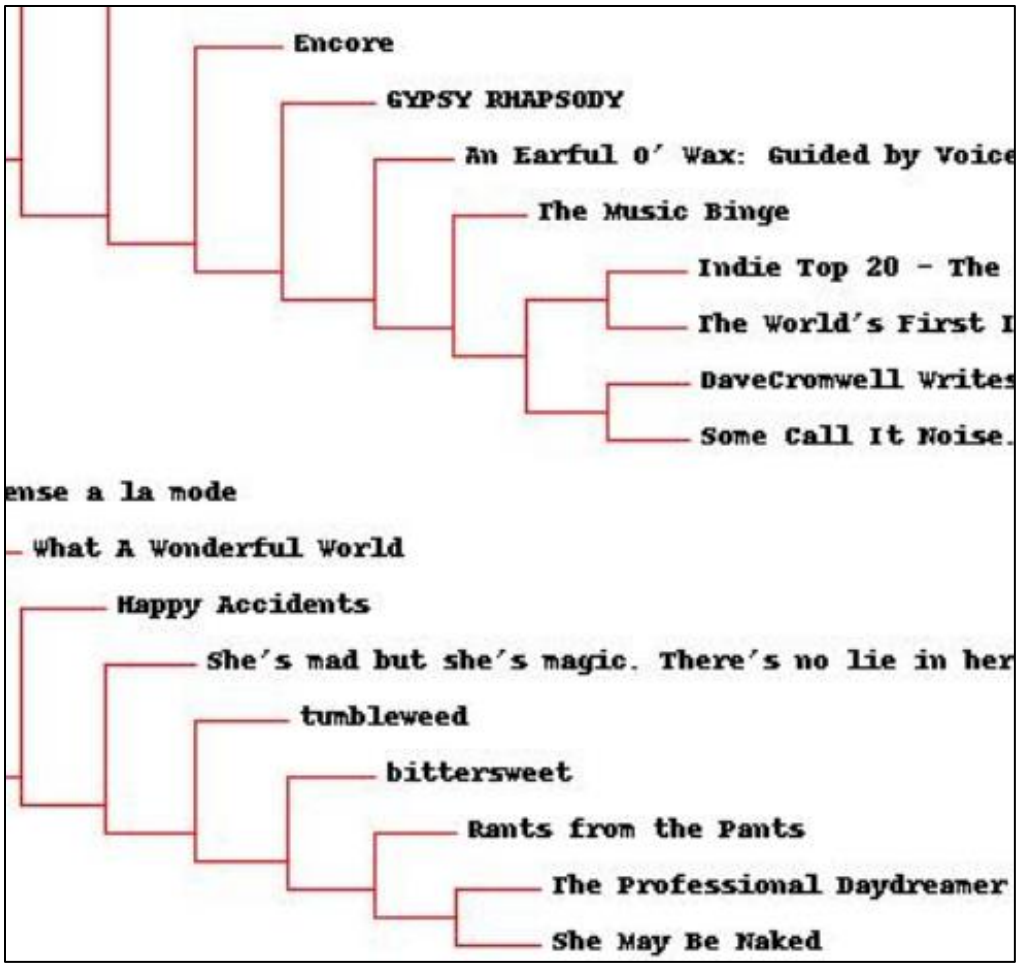
<u>Ascii output</u>:

```
                          The Music Binge
                          -
                            -
                              Indie Top 20 - The Blog!
                              The World's First Internet Baby
                            -
                              DaveCromwell Writes
                              Some Call It Noise....
            -
              nonsense a la mode
              -
                What A Wonderful World
                -
                  Happy Accidents
                  -
                    She's mad but she's magic. There's no lie in her fir
                    -
                      tumbleweed
                      -
                        bittersweet
                        -
                          Rants from the Pants
```

Dendrogram:

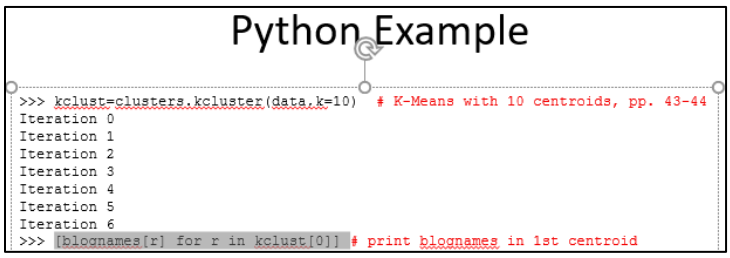

Room 19's Blog 2016
FEEL THE SKY ON YOUR HANDS
The Themes of My Life
Beyond the pond
CLOUDBUSTING
Hello
Wake Up, Music & Fashion
the fast break of champions
hello my name is justin.
i'm in too truthful a mood
Avidd Wallows' Blog
Cherry Area
Steel City Rust
On Warmer Music
Did Not Chart
A layer of chips
Stereo Pills
Encore
GYPSY RHAPSODY
An Earful O' Wax: Guided by Voices/R
The Music Binge
Indie Top 20 - The Blog
The World's First Intex
DaveCromwell Writes
Some Call It Noise....
nonsense a la mode
What A Wonderful World
Happy Accidents
She's mad but she's magic. There's no lie in her fix
tumbleweed
bittersweet
Rants from the Pants
The Professional Daydreamer
She May Be Naked
Sonology
Myopiamuse
Oh Yes JÃ²nsi!!
"DANCING IN CIRCLES"
adriamoblog
Spinitron Charts
|[Tu queres ver isto]|
Unexpectedly Bart (King!)
Music-Drop Magazine
Who needs a TV?
nÃ£o ponho mÃ²sica
Stories From the City, Stories From the Sea
Lost in the Shuffle
LOST PLACES
The Campus Buzz on WSOU
Broken Biscuit Records
Wyoming Beat
^Sixeyes: by Alan Williamson
FOLK IS NOT HAPPY
SumStock Music
Floorshime Zipper Boots
Cuz Music Rocks
F-Measure
Revolver USA Distribution & Midheaven mailorder
MAGGOT CAVIAR
macthemost
the roofy leak
Bleak Bliss
Stonehill Sketchbook
A M T A P O T
Mile In Mine
the traveling neighborhood
Parish Radio
Kinky Adventures of Nikkij
hmmhannah
Web Science and Digital Libraries Research Group
Paulina Gamero. Media Studies A2
A2 MEDIA COURSEWORK JOINT BLOG
Rebecca Wood 9282 A2 Advanced Portfolio
sweeping the kitchen
Abu Everyday

Zoomed In:


Encore
GYPSY RHAPSODY
An Earful O' Wax: Guided by Voice
The Music Binge
Indie Top 20 - The
The World's First I
DaveCromwell Writes
Some Call It Noise.
ense a la mode
What A Wonderful World
Happy Accidents
She's mad but she's magic. There's no lie in her
tumbleweed
bittersweet
Rants from the Pants
The Professional Daydreamer
She May Be Naked

Q3:

Slide 18 of the clustering PowerPoint from class was helpful for this question.



k.py:

```
import clusters

blognames, words, data = clusters.readfile('blogdata1.txt')

print('k = 5')
kclust = clusters.kcluster(data, k = 5)
k= [blognames[r] for r in kclust[0]]
print str(k) + '\n'

print('k = 10')
kclust = clusters.kcluster(data, k = 10)
print str(k) + '\n'

print('k = 20')
kclust = clusters.kcluster(data, k = 20)
print str(k)
```

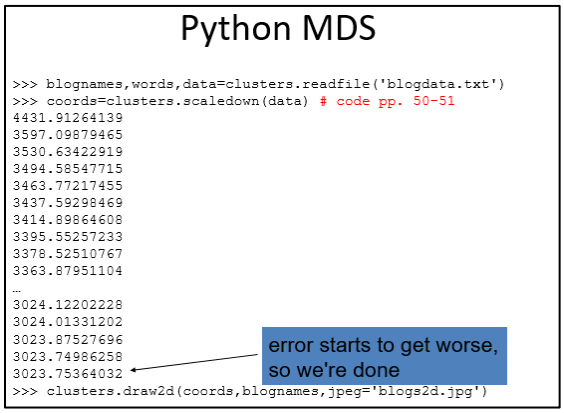
Output:

```
mgrah@DESKTOP-3OIR4AC MINGW64 /c/
$ python k.py
k = 5
Iteration 0
Iteration 1
Iteration 2
Iteration 3
Iteration 4
Iteration 5
['Bleak Bliss', 'On Warmer Music'
es/Robert Pollard Song by Song Re

k = 10
Iteration 0
Iteration 1
Iteration 2
Iteration 3
Iteration 4
['Bleak Bliss', 'On Warmer Music'
es/Robert Pollard Song by Song Re

k = 20
Iteration 0
Iteration 1
Iteration 2
Iteration 3
Iteration 4
Iteration 5
Iteration 6
Iteration 7
Iteration 8
['Bleak Bliss', 'On Warmer Music'
es/Robert Pollard Song by Song Re
```

Q4:

For this question, I referenced slide 28 of the clustering PowerPoint from class again.

Slide 28:

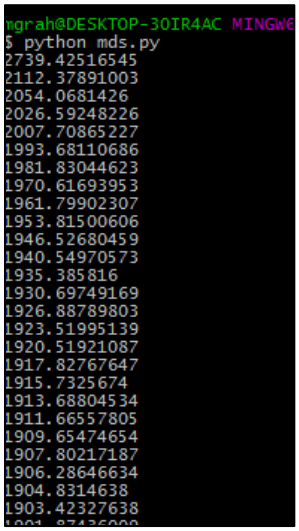


Mds.py:

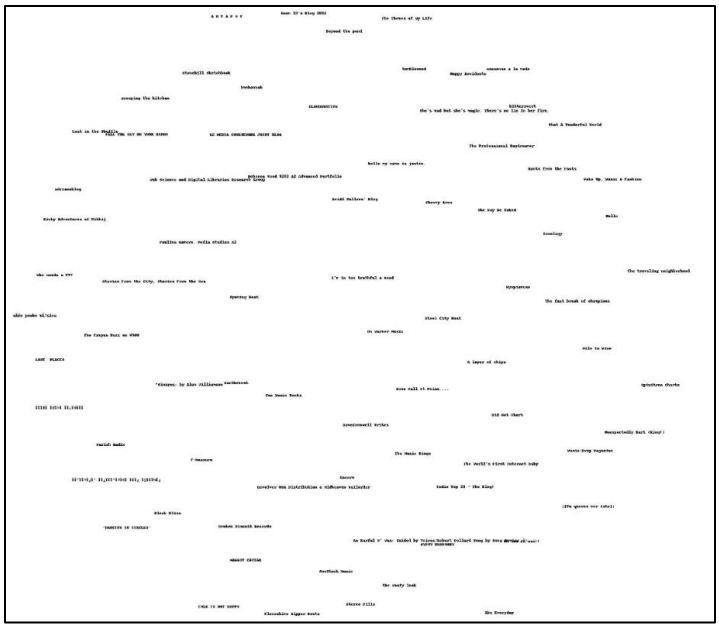
```
import clusters

blognames, words, data = clusters.readfile('blogdata1.txt')

c = clusters.scaledown(data)

clusters.draw2d(c, blognames, jpeg='mds.jpg')
```

Output:

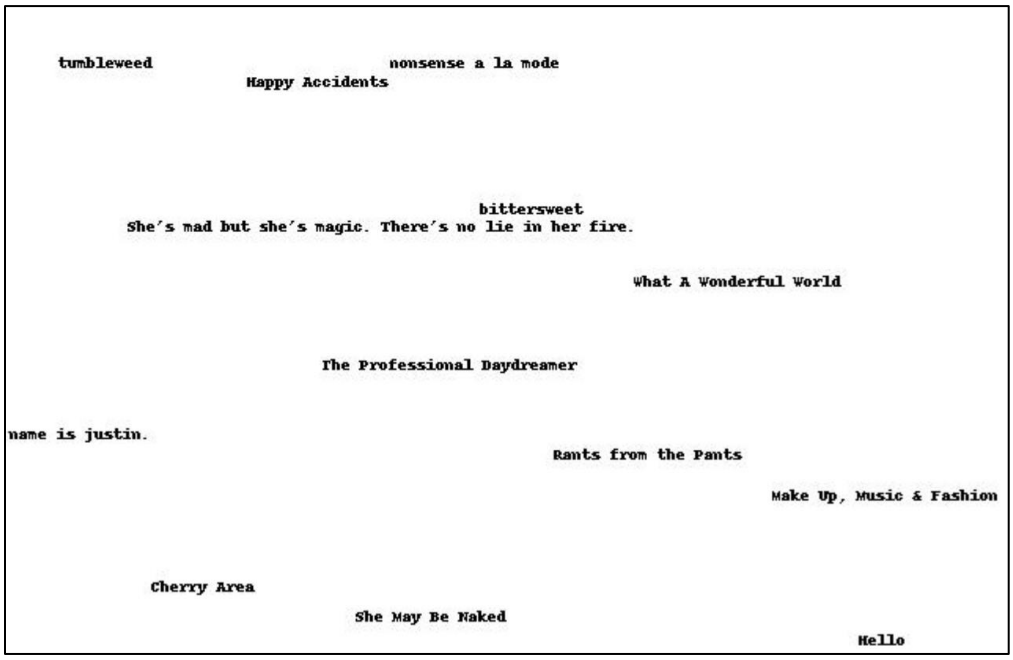
There were 399 lines in total for the output, therefore it took 399 iterations to complete.



Jpeg:



Zoomed in:



Resources:

https://github.com/uolter/PCI/tree/master/chapter3

http://stackoverflow.com/questions/4706499/how-do-you-append-to-a-file

http://stackoverflow.com/questions/14852480/about-handling-a-redirection-in-python