

All files mentioned in this file are uploaded into the *github* repository.

The \LaTeX code involved in the generation of this document was aided by the example code provided in the links that Dr. Nelson sent out on January 17, 2016[1].

All credit is given to Shawn Jones who wrote the original code used in this assignment[2]. This was found via the following url: <https://github.com/shawnmjones/cs595-f13/tree/master/assignment9> and has been modified to accept inputs from anyone using this code.

This document was compiled on www.sharelatex.com

Problem 1

Generation of the blog-term matrix begins with a script retrieves 98 random blogs. A program written by Shawn Jones called `fetchFeeds.py`, retrieves a list of random blogs[2]. The code was modified to ensure that the same link is not put into the list of links. A data file called `feedlist.txt` will be generated and put into a folder called `producedFiles`. Listing 1 shows the code used to generate a list of random blogs[2]. Listing 2 lists all the uri's used in this assignment.

Listing 1: Retrieving a Unique list of blogs

```
1 import sys
2 import os
3 import os.path
4 import feedparser
5 import urllib2
6 import time
7 import chardet
8 import argparse
9 from bs4 import BeautifulSoup
10
11 def dereferenceUri(uri):
12
13     pagehandle = urllib2.urlopen(uri)
14     pagedata = pagehandle.read()
15     derefurl = pagehandle.geturl()
16     pagehandle.close()
17
18     return pagedata,derefurl
19
```

```
20 def getAtomFeedUri(html):
21
22     soup = BeautifulSoup(html, 'html')
23
24     atomLinks = soup.find_all('link',
25                               attrs = { 'rel' : 'alternate', 'type' : 'application/atom+xml' })
26
27     # we assume there is only one atom link
28     atomURI = atomLinks[0].attrs['href']
29
30     return atomURI
31
32 def meetsCriteria(feedText):
33
34     parsedData = feedparser.parse(feedText)
35
36     # assume we're good to go by default (fail optimistic?)
37     goodToGo = True
38
39     sys.stderr.write("blog has " + str(len(parsedData.entries)) + "
40                       entries\n")
41
42     if (len(parsedData.entries) < 25):
43         goodToGo = False
44
45     #if (charset.detect(feedText)['encoding'] != 'ascii'):
46     #    sys.stderr.write("blog charset is " +
47     #                    charset.detect(feedText)['encoding'] +
48     #                    ", likely won't parse well for feed vector\n")
49     #    goodToGo = False
50
51     return goodToGo
52
53 def getNextUri():
54
55     uri = "http://www.blogger.com/next-blog?navBar=true&blogID=30361063"
56
57     pagehandle = urllib2.urlopen(uri)
58     nexturi = pagehandle.geturl()
59     pagehandle.close()
60
61     return nexturi
62
63 if __name__ == "__main__":
```

```
64 feedlist = []
65 feedlist.append(
66     "http://f-measure.blogspot.com/feeds/posts/default?max-results=200"
67 )
68 feedlist.append(
69     "http://ws-dl.blogspot.com/feeds/posts/default?max-results=200"
70 )
71 feedlist.append(
72     "http://www.russianmachineneverbreaks.com/feed/"
73 )
74
75
76 uriList = []
77 while (len(feedlist) <= 100):
78     out = open('../producedFiles/feedlist.txt','w+')
79     for link in feedlist:
80         out.write(link)
81         out.write('\n')
82     out.close()
83
84     try:
85         uri = getNextUri()
86         print("Getting unique uri")
87         while len(list(filter(lambda x: x == uri, uriList))) != 0:
88             uri = getNextUri()
89             #time.sleep(1)
90             uriList.append(uri)
91         except urllib2.HTTPError as e:
92             sys.stderr.write("failed to acquire next uri, delaying 5
93                 seconds\n")
94             time.sleep(5)
95         else:
96             sys.stderr.write("\n" + "working on URI " + uri + "\n")
97
98         try:
99             # dereference the uri and get text
100             html, derefuri = dereferenceUri(uri)
101         except urllib2.HTTPError as e:
102             sys.stderr.write("failed to dereference " + uri +
103                 ", delaying 5 seconds\n")
104             time.sleep(5)
105         else:
106             try:
107                 # fetch the atom feed URI
108                 feedURI = getAtomFeedUri(html)
109                 sys.stderr.write("acquired feed URI " + feedURI + "\n")
```

```
109         except IndexError as e:
110             sys.stderr.write(
111                 "failed to acquire Atom feed from HTML, delaying 5
                  seconds\n\n")
112         else:
113             try:
114                 # get the atom feed text
115                 feedText, feedURI = dereferenceUri(feedURI)
116             except urllib2.HTTPError as e:
117                 sys.stderr.write("failed to dereference " + feedURI +
118                                 ", delaying 5 seconds\n")
119                 time.sleep(5)
120             else:
121                 # if it meets the criteria, save the file
122                 #toCheck = feedURI + '?max-results=1000\n'
123                 if meetsCriteria(feedText):
124
125                     sys.stderr.write("Saving blog feed " + feedURI +
126                                     "?max-results=1000\n")
127                     #print(feedURI + "?max-results=1000")
128                     feedlist.append(feedURI + "?max-results=1000")
129
130                 # be nice to the site
131                 time.sleep(1)
132
133     for feed in feedlist:
134         print feed
```

Listing 2: List of blog uri's

```
1 http://f-measure.blogspot.com/feeds/posts/default?max-results=200
2 http://ws-dl.blogspot.com/feeds/posts/default?max-results=200
3 http://www.russianmachineneverbreaks.com/feed/
4 http://turnitupjack.blogspot.com/feeds/posts/default?max-results=1000
5 http://ashleyemwarren.blogspot.com/feeds/posts/default?max-results=1000
6 http://www.sonology.com/feeds/posts/default?max-results=1000
7 http://elijace.blogspot.com/feeds/posts/default?max-results=1000
8 http://stonehillsketchbook.blogspot.com/feeds/posts/default?max-results=1000
9 http://mondaywakeup.blogspot.com/feeds/posts/default?max-results=1000
10 http://www.chrisanne-grise.com/feeds/posts/default?max-results=1000
11 http://isyelili.blogspot.com/feeds/posts/default?max-results=1000
12 http://ihatethe90s.blogspot.com/feeds/posts/default?max-results=1000
13 http://flipmpip.blogspot.com/feeds/posts/default?max-results=1000
14 http://timeispoetry.blogspot.com/feeds/posts/default?max-results=1000
15 http://seveninchesisenough.blogspot.com/feeds/posts/default?max-results=1000
16 http://kunstlertreu.blogspot.com/feeds/posts/default?max-results=1000
```

```
17 http://ourpodcastcouldbeyourlife.blogspot.com/feeds/posts/default?max-results=1000
18 http://jamiemclelland.blogspot.com/feeds/posts/default?max-results=1000
19 http://ps-music.blogspot.com/feeds/posts/default?max-results=1000
20 http://theindiefriend.blogspot.com/feeds/posts/default?max-results=1000
21 http://adrianomarquesblog.blogspot.com/feeds/posts/default?max-results=1000
22 http://glipress.blogspot.com/feeds/posts/default?max-results=1000
23 http://feeds.feedburner.com/blogspot/qbENp?max-results=1000
24 http://flowradio.blogspot.com/feeds/posts/default?max-results=1000
25 http://my-name-is-blue-canary.blogspot.com/feeds/posts/default?max-results=1000
26 http://mattsbunker.blogspot.com/feeds/posts/default?max-results=1000
27 http://dinosaursarefun.blogspot.com/feeds/posts/default?max-results=1000
28 http://feeds.feedburner.com/karldrinkwater?max-results=1000
29 http://markeortega.blogspot.com/feeds/posts/default?max-results=1000
30 http://skinnyshoes.blogspot.com/feeds/posts/default?max-results=1000
31 http://marshwiggles.blogspot.com/feeds/posts/default?max-results=1000
32 http://ahtapotunbahcesi.blogspot.com/feeds/posts/default?max-results=1000
33 http://richardwhitten.blogspot.com/feeds/posts/default?max-results=1000
34 http://kidchair.blogspot.com/feeds/posts/default?max-results=1000
35 http://hiiiijaaackie.blogspot.com/feeds/posts/default?max-results=1000
36 http://mo-forgetaboutit.blogspot.com/feeds/posts/default?max-results=1000
37 http://hani-bittersweet.blogspot.com/feeds/posts/default?max-results=1000
38 http://listeningear.blogspot.com/feeds/posts/default?max-results=1000
39 http://theworldsfirstinternetbaby.blogspot.com/feeds/posts/default?max-results=1000
40 http://jbreitling.blogspot.com/feeds/posts/default?max-results=1000
41 http://bogglemethursday.blogspot.com/feeds/posts/default?max-results=1000
42 http://feeds.feedburner.com/AW/sixeyes?max-results=1000
43 http://ngaio1619.blogspot.com/feeds/posts/default?max-results=1000
44 http://rosiegigga2media.blogspot.com/feeds/posts/default?max-results=1000
45 http://blog.spinitron.com/feeds/posts/default?max-results=1000
46 http://floorshimezipperboots.blogspot.com/feeds/posts/default?max-results=1000
47 http://www.samtasticreview.com/feeds/posts/default?max-results=1000
48 http://steel-city-rust.blogspot.com/feeds/posts/default?max-results=1000
49 http://ilovetotaldestruction.blogspot.com/feeds/posts/default?max-results=1000
50 http://onestunningsingleegg.blogspot.com/feeds/posts/default?max-results=1000
51 http://didnotchart.blogspot.com/feeds/posts/default?max-results=1000
52 http://lostintheshuffle899.blogspot.com/feeds/posts/default?max-results=1000
53 http://mesastivromia.blogspot.com/feeds/posts/default?max-results=1000
54 http://laespiraldejosephk.blogspot.com/feeds/posts/default?max-results=1000
55 http://deadbeatfanzine.blogspot.com/feeds/posts/default?max-results=1000
56 http://spicyseatdolphin.blogspot.com/feeds/posts/default?max-results=1000
57 http://dana9morgan.blogspot.com/feeds/posts/default?max-results=1000
58 http://psychfolkmusic.blogspot.com/feeds/posts/default?max-results=1000
59 http://semregrasluispink.blogspot.com/feeds/posts/default?max-results=1000
60 http://out-of-the-swamp.blogspot.com/feeds/posts/default?max-results=1000
61 http://mobbie2.blogspot.com/feeds/posts/default?max-results=1000
62 http://cherryarea.blogspot.com/feeds/posts/default?max-results=1000
```

```
63 http://storiesfromthecityradiovalencia.blogspot.com/feeds/posts/default?max-results=1000
64 http://organmyth.blogspot.com/feeds/posts/default?max-results=1000
65 http://jojobethkatiehannahlcm1516.blogspot.com/feeds/posts/default?max-results=1000
66 http://srkikiblog.blogspot.com/feeds/posts/default?max-results=1000
67 http://thepowerofindependenttrucking.blogspot.com/feeds/posts/default?max-results=1000
68 http://harisphotonet.blogspot.com/feeds/posts/default?max-results=1000
69 http://thefastbreakofchampions.blogspot.com/feeds/posts/default?max-results=1000
70 http://globalgoon.blogspot.com/feeds/posts/default?max-results=1000
71 http://mileinmine.blogspot.com/feeds/posts/default?max-results=1000
72 http://dancingincirclesnow.blogspot.com/feeds/posts/default?max-results=1000
73 http://campusbuzzwsou.blogspot.com/feeds/posts/default?max-results=1000
74 http://rantsfromthepants.blogspot.com/feeds/posts/default?max-results=1000
75 http://fortheotherthings.blogspot.com/feeds/posts/default?max-results=1000
76 http://moontopples.blogspot.com/feeds/posts/default?max-results=1000
77 http://hartsdesire.blogspot.com/feeds/posts/default?max-results=1000
78 http://onebaseonanoverthrow.blogspot.com/feeds/posts/default?max-results=1000
79 http://flatbasset.blogspot.com/feeds/posts/default?max-results=1000
80 http://doginasweatershowreviews.blogspot.com/feeds/posts/default?max-results=1000
81 http://maggotcaviar.blogspot.com/feeds/posts/default?max-results=1000
82 http://encorenorthernireland.blogspot.com/feeds/posts/default?max-results=1000
83 http://superchicken46.blogspot.com/feeds/posts/default?max-results=1000
84 http://castironsongs.blogspot.com/feeds/posts/default?max-results=1000
85 http://feeds.feedburner.com/DavecromwellWrites?max-results=1000
86 http://thekidsarecoming.blogspot.com/feeds/posts/default?max-results=1000
87 http://rodshone.blogspot.com/feeds/posts/default?max-results=1000
88 http://jlmldhlcm1516.blogspot.com/feeds/posts/default?max-results=1000
89 http://intheframefilmreviews.blogspot.com/feeds/posts/default?max-results=1000
90 http://desolationrowrecords.blogspot.com/feeds/posts/default?max-results=1000
91 http://www.symmetrysymmetry.com/feeds/posts/default?max-results=1000
92 http://www.radioshower.com/feeds/posts/default?max-results=1000
93 http://headspacedaily.blogspot.com/feeds/posts/default?max-results=1000
94 http://thebeautifultrashart.blogspot.com/feeds/posts/default?max-results=1000
95 http://www.thejeopardyofcontentment.com/feeds/posts/default?max-results=1000
96 http://beckysharpfashionblog.blogspot.com/feeds/posts/default?max-results=1000
97 http://sweetcrimsonblood.blogspot.com/feeds/posts/default?max-results=1000
98 http://thenightmail.blogspot.com/feeds/posts/default?max-results=1000
99 http://mtjrrantsravesonmusic.blogspot.com/feeds/posts/default?max-results=1000
100 http://chantellesmedia2.blogspot.com/feeds/posts/default?max-results=1000
```

Once this code was written, a blog term matrix was generated by a program called `generatefeedvector.py`[2]. This program generates a matrix with all the popular terms. A script called `eliminateWords.py` creates a new matrix in which no more than 500 terms are present. This matrix is called `blogtermmatrix.txt` and is used to create an ascii dengrogram and jpeg dendrogram used for question 2.

Problem 2

Creating the ascii and jpeg dendrogram is performed by a program called makeDendrogram.py [2]. This code was modified so that any feed-list can be used and the user has the option of generating either an ascii dendrogram or a jpeg dendrogram. This code utilizes the argparse module to do this as seen in listing 3. This code also utilizes code written in a file called clusters.py[3] which runs the hierarchical clustering algorithm as described on page 33 in the book Programming Collective Intelligence[3]. Once the hierarchical clustering algorithm is complete, an ascii dendrogram called asciDendrogram.txt and a jpeg dendrogram called dendrogram.jpg are produced. The jpeg dendrogram can be seen in figure 2.

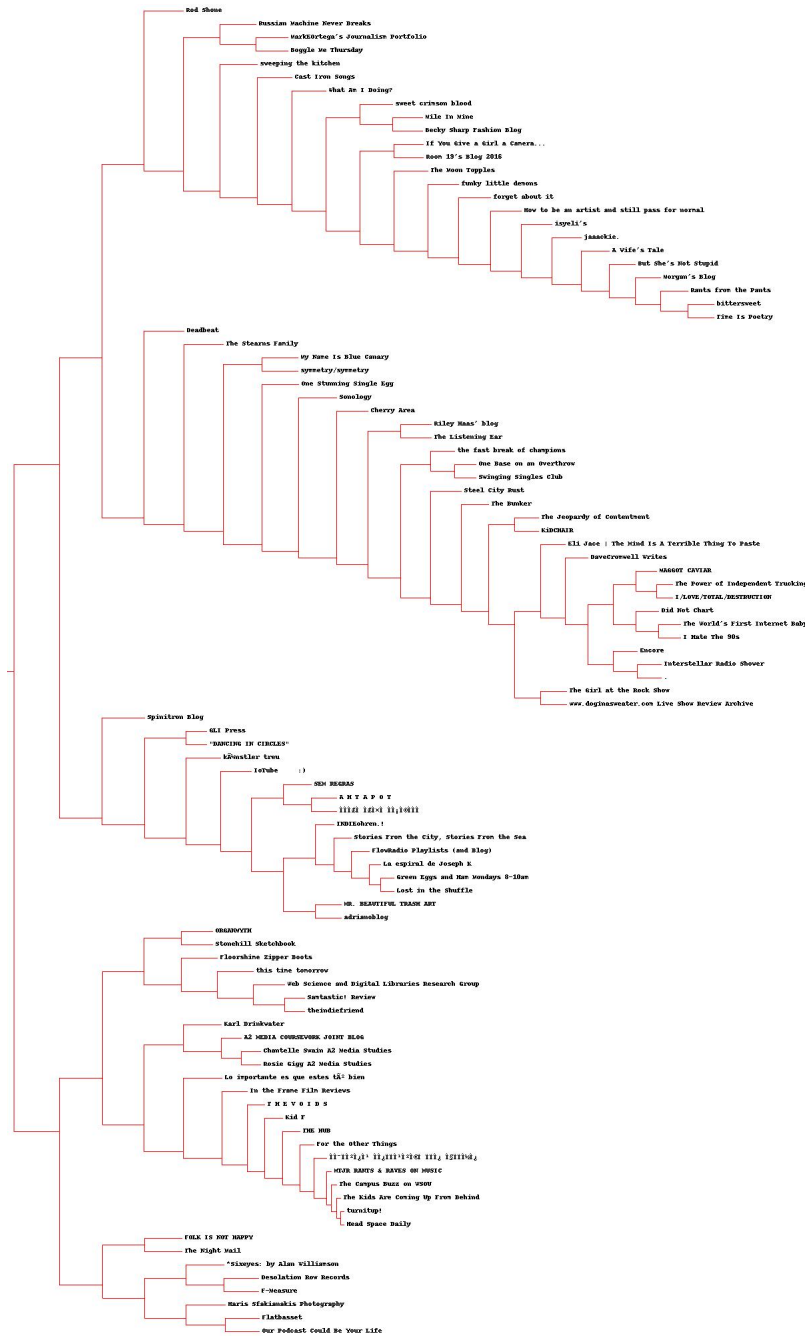


Figure 1: Dendrogram created for the blog links in listing 2

Problem 3

To cluster the blogs using K-Means, a program was written which used the function `kcluster` which was supplied from the book *Programming Collective Intelligence*[3]. The program takes in two command line arguments, `-f` file containing the blog term matrix and `-k` values of `k` to use. This program will allow the user to supply any number of values for `k` and will print the results to the standard output. Listing 3 contains the output acquired from the `k`-clustering algorithm.

Listing 3: Results for K clustering

```
1 For k= 5
2 Iteration 0
3 Iteration 1
4 Iteration 2
5 Iteration 3
6 Iteration 4
7 *****
8 ["Riley Haas' blog", 'SEM REGRAS', 'My Name Is Blue Canary', 'ORGANMYTH',
   "MarkEOrtega's Journalism Portfolio", 'Green Eggs and Ham Mondays
   8-10am', 'Stories From the City, Stories From the Sea', 'Lost in the
   Shuffle', 'How to be an artist and still pass for normal', 'IoTube :)',
   'forget about it', 'jaaackie.', 'The Girl at the Rock Show', 'The
   Listening Ear', 'FlowRadio Playlists (and Blog)', 'INDIEehren.!',
   'Russian Machine Never Breaks', "Morgan's Blog", 'www.doginasweater.com
   Live Show Review Archive', 'La espiral de Joseph K', 'funky little
   demons', 'Boggle Me Thursday', 'Mile In Mine', 'The Stearns Family', 'If
   You Give a Girl a Camera...', 'What Am I Doing?', 'Becky Sharp Fashion
   Blog', 'Rod Shone', 'KiDCHAIR', 'Sonology', 'bittersweet', "Room 19's
   Blog 2016", 'sweeping the kitchen', 'One Stunning Single Egg', 'Rants
   from the Pants', 'Time Is Poetry', "A Wife's Tale", 'sweet crimson
   blood', 'the fast break of champions', 'The Moon Topples', "isyeli's",
   "But She's Not Stupid", 'Haris Sfakianakis Photography', 'Cast Iron
   Songs']
9 *****
10 ['MTJR RANTS & RAVES ON MUSIC', 'turnitup!', 'adrianoblog', 'For the Other
    Things', 'T H E V O I D S', 'Head Space Daily', 'The Campus Buzz on
    WSOU', 'Spinitron Blog', 'k\xc3\xbcnstler treu', 'THE HUB', 'Lo
    importante es que estes t\xc3\xba bien', 'Kid F', 'The Kids Are Coming
    Up From Behind', 'In the Frame Film Reviews', 'Cherry Area',
    '\xce\x94\xce\xaf\xcf\x83\xce\xba\xce\xbf\xce\xb9
    \xce\x9c\xce\xbf\xcf\x85\xcf\x83\xce\xb9\xce\xba\xce\xae\xcf\x82
    \xcf\x83\xcf\x84\xce\xbf \xce\xa7\xcf\x81\xcf\x8c\xce\xbd\xce\xbf']
11 *****
12 ['Steel City Rust', 'One Base on an Overthrow', 'Did Not Chart',
    'Interstellar Radio Shower', 'The Power of Independent Trucking',
```

```
'DaveCromwell Writes', 'Rosie Gigg A2 Media Studies', 'FOLK IS NOT
HAPPY', 'MAGGOT CAVIAR', 'Desolation Row Records', "The World's First
Internet Baby", 'Eli Jace | The Mind Is A Terrible Thing To Paste', 'The
Jeopardy of Contentment', '.', '*Sixeyes: by Alan Williamson', 'Swinging
Singles Club', 'I/LOVE/TOTAL/DESTRUCTION', 'F-Measure', 'Encore', 'I
Hate The 90s']
13 *****
14 ['MR. BEAUTIFUL TRASH ART', 'A H T A P O T', 'Floorshime Zipper Boots',
    'Stonehill Sketchbook', 'Samtastic! Review', 'The Bunker', 'Deadbeat',
    'theindiefriend', 'symmetry/symmetry', 'this time tomorrow',
    '\xce\x9c\xce\x95\xce\xa3\xce\x91 \xce\xa3\xce\xa4\xce\x97
    \xce\x92\xce\xa1\xce\xa9\xce\x9c\xce\x99\xce\x91']
15 *****
16 ['Flatbasset', 'Web Science and Digital Libraries Research Group', 'GLI
    Press', 'Chantelle Swain A2 Media Studies', 'A2 MEDIA COURSEWORK JOINT
    BLOG', 'Our Podcast Could Be Your Life', 'The Night Mail', 'Karl
    Drinkwater', '"DANCING IN CIRCLES"']
17 For k= 10
18 Iteration 0
19 Iteration 1
20 Iteration 2
21 Iteration 3
22 Iteration 4
23 *****
24 ['Steel City Rust', 'DaveCromwell Writes', "The World's First Internet
    Baby", '.', 'The Night Mail']
25 *****
26 ['ORGANMYTH', 'How to be an artist and still pass for normal', 'forget about
    it', 'A2 MEDIA COURSEWORK JOINT BLOG', 'Russian Machine Never Breaks',
    "Morgan's Blog", 'Boggle Me Thursday', 'Mile In Mine', 'What Am I
    Doing?', 'Becky Sharp Fashion Blog', 'bittersweet', "Room 19's Blog
    2016", 'One Stunning Single Egg', 'Time Is Poetry', "A Wife's Tale",
    'Karl Drinkwater', 'In the Frame Film Reviews', 'Cherry Area', 'The Moon
    Topples', "isyeli's", "But She's Not Stupid"]
27 *****
28 ['MTJR RANTS & RAVES ON MUSIC', 'turnitup!', 'For the Other Things', 'T H E
    V O I D S', 'Head Space Daily', 'The Campus Buzz on WSOU', 'THE HUB',
    'Lo importante es que estes t\xc3\xba bien', 'Kid F', 'The Kids Are
    Coming Up From Behind',
    '\xce\x94\xce\xaf\xcf\x83\xce\xba\xce\xbf\xce\xb9
    \xce\x9c\xce\xbf\xcf\x85\xcf\x83\xce\xb9\xce\xba\xce\xae\xcf\x82
    \xcf\x83\xcf\x84\xce\xbf \xce\xa7\xcf\x81\xcf\x8c\xce\xbd\xce\xbf']
29 *****
30 ['MR. BEAUTIFUL TRASH ART', 'Stonehill Sketchbook', 'Samtastic! Review',
    'theindiefriend', 'symmetry/symmetry', 'Rants from the Pants', 'this
    time tomorrow', '\xce\x9c\xce\x95\xce\xa3\xce\x91
```

```
\x3\x4\x7
\x92\x1\x9\x9c\x99\x91']
31 *****
32 ['One Base on an Overthrow', 'Interstellar Radio Shower', 'The Power of
    Independent Trucking', 'MAGGOT CAVIAR', 'Swinging Singles Club',
    'I/LOVE/TOTAL/DESTRUCTION', 'F-Measure']
33 *****
34 ['SEM REGRAS', 'Green Eggs and Ham Mondays 8-10am', 'GLI Press', 'Stories
    From the City, Stories From the Sea', 'Lost in the Shuffle', 'A H T A P
    O T', 'adrianoblog', 'IoTube :)', 'FlowRadio Playlists (and Blog)',
    'INDIEehren.!', 'La espiral de Joseph K', 'If You Give a Girl a
    Camera...', 'sweeping the kitchen', 'sweet crimson blood', '"DANCING IN
    CIRCLES"']
35 *****
36 ['Flatbasset', "Riley Haas' blog", 'The Girl at the Rock Show', 'The
    Listening Ear', 'www.doginasweater.com Live Show Review Archive', 'Eli
    Jace | The Mind Is A Terrible Thing To Paste', 'The Bunker', 'Deadbeat',
    'KiDCHAIR', 'Our Podcast Could Be Your Life', 'the fast break of
    champions', 'Encore']
37 *****
38 ["MarkEOrtega's Journalism Portfolio", 'Did Not Chart', 'Rosie Gigg A2 Media
    Studies', 'FOLK IS NOT HAPPY', 'funky little demons', 'The Jeopardy of
    Contentment', '*Sixeyes: by Alan Williamson', 'Rod Shone', 'Sonology',
    'Haris Sfakianakis Photography', 'I Hate The 90s', 'Cast Iron Songs']
39 *****
40 ['My Name Is Blue Canary', 'jaaackie.', 'Spinitron Blog', 'Desolation Row
    Records', 'k\x3\xbcnstler treu']
41 *****
42 ['Web Science and Digital Libraries Research Group', 'Floorshime Zipper
    Boots', 'Chantelle Swain A2 Media Studies', 'The Stearns Family']
43 For k= 20
44 Iteration 0
45 Iteration 1
46 Iteration 2
47 Iteration 3
48 Iteration 4
49 Iteration 5
50 Iteration 6
51 *****
52 ['Mile In Mine', 'Deadbeat', 'What Am I Doing?', 'Becky Sharp Fashion Blog',
    'Karl Drinkwater', 'The Moon Topples']
53 *****
54 ['SEM REGRAS', 'Lo importante es que estes t\x3\xba bien']
55 *****
56 ["MarkEOrtega's Journalism Portfolio"]
57 *****
```

```
58 ['MR. BEAUTIFUL TRASH ART', 'A H T A P O T', 'IoTube :)',  
   '\xc9\x95\xa3\x91 \xa3\xa4\x97  
   \x92\xa1\xa9\x9c\x99\x91', 'Haris Sfakianakis  
   Photography']  
59 *****  
60 ['Web Science and Digital Libraries Research Group']  
61 *****  
62 []  
63 *****  
64 ['Green Eggs and Ham Mondays 8-10am', 'Stories From the City, Stories From  
   the Sea', 'Lost in the Shuffle', 'adrianoblog', 'FlowRadio Playlists  
   (and Blog)', 'INDIEehren.!', 'La espiral de Joseph K', 'sweeping the  
   kitchen']  
65 *****  
66 ['MTJR RANTS & RAVES ON MUSIC']  
67 *****  
68 ['Steel City Rust', 'One Base on an Overthrow', 'Floorshime Zipper Boots',  
   'Did Not Chart', 'Interstellar Radio Shower', 'The Power of Independent  
   Trucking', 'DaveCromwell Writes', 'FOLK IS NOT HAPPY', 'Desolation Row  
   Records', 'The World's First Internet Baby', 'Eli Jace | The Mind Is A  
   Terrible Thing To Paste', 'The Jeopardy of Contentment', '.', '*Sixeyes:  
   by Alan Williamson', 'Swinging Singles Club',  
   'I/LOVE/TOTAL/DESTRUCTION', 'F-Measure', 'Encore', 'I Hate The 90s',  
   'Cast Iron Songs']  
69 *****  
70 ['forget about it', 'jaaackie.', 'k\xc3\xbcnstler treu', 'funky little  
   demons', 'Rod Shone', 'symmetry/symmetry', 'bittersweet', 'One Stunning  
   Single Egg', 'Time Is Poetry', 'A Wife's Tale', 'sweet crimson blood']  
71 *****  
72 ['Riley Haas' blog', 'The Listening Ear', 'Russian Machine Never Breaks']  
73 *****  
74 ['My Name Is Blue Canary', 'The Girl at the Rock Show',  
   'www.doginasweater.com Live Show Review Archive', 'KiDCHAIR', 'the fast  
   break of champions']  
75 *****  
76 ['Stonehill Sketchbook', 'Samtastic! Review', 'theindiefriend', 'this time  
   tomorrow']  
77 *****  
78 ['Boggle Me Thursday', 'The Night Mail']  
79 *****  
80 ['turnitup!', 'For the Other Things', 'T H E V O I D S', 'Head Space Daily',  
   'The Campus Buzz on WSOU', 'Spinitron Blog', 'THE HUB', 'Kid F', 'The  
   Kids Are Coming Up From Behind',  
   '\x94\xaf\xcf\x83\xba\xbf\x9  
   \x9c\xbf\xcf\x85\xcf\x83\x9\xba\xae\xcf\x82  
   \xcf\x83\xcf\x84\xbf \xa7\xcf\x81\xcf\x8c\xbd\xbf']
```

```
81 *****
82 ['Flatbasset', 'Our Podcast Could Be Your Life']
83 *****
84 ['ORGANMYTH', 'How to be an artist and still pass for normal', "Morgan's
   Blog", 'If You Give a Girl a Camera...', 'Sonology', "Room 19's Blog
   2016", 'Rants from the Pants', "isyeli's", "But She's Not Stupid"]
85 *****
86 ['GLI Press', 'MAGGOT CAVIAR', 'The Stearns Family', '"DANCING IN CIRCLES"']
87 *****
88 ['Chantelle Swain A2 Media Studies', 'A2 MEDIA COURSEWORK JOINT BLOG',
   'Rosie Gigg A2 Media Studies']
89 *****
90 ['The Bunker', 'In the Frame Film Reviews', 'Cherry Area']
```

Problem 4

To create a 2D representation of the blog space for the links in listing 2, a program called `makeMDS.py`[2] was used. The program utilizes the scaledown algorithm from the book *Programming Collective Intelligence* and draws a 2D representation in a file called `2dBlogSpace.jpg`, of the blog space[3]. Figure 2 shows the 2D representation of the blog space and listing 4 shows the number iterations needed to scale the blog space.

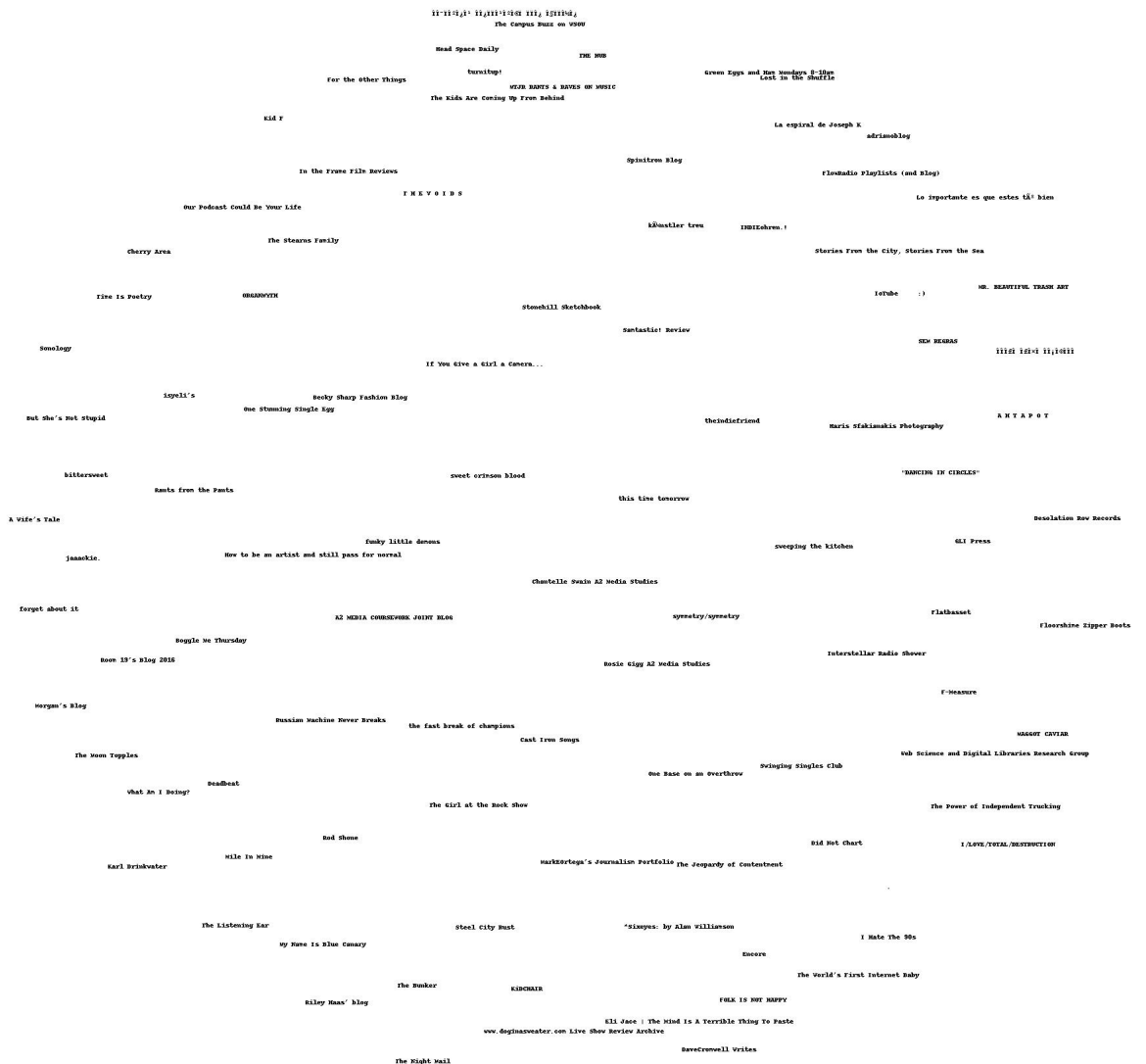


Figure 2: 2D Representation of the blog space

Listing 4: Iterations required for MDS

```

1 4718.51649633
2 3641.97350222
3 3576.96217617
4 3539.60253906

```

```
5 3512.67702734
6 3494.99028129
7 3483.80107693
8 3475.57121843
9 3469.52965368
10 3463.27561441
11 3456.7835787
12 3450.27551531
13 3444.08961094
14 3437.54006712
15 3431.50292669
16 3426.17372458
17 3421.10982328
18 3415.98606711
19 3411.1559161
20 3406.32027765
21 3401.83641255
22 3398.12688863
23 3394.79579743
24 3391.49244857
25 3388.49506945
26 3384.64294071
27 3380.75208309
28 3377.59262979
29 3375.00512136
30 3372.68827107
31 3370.96246612
32 3369.34268541
33 3368.08545575
34 3366.85898541
35 3365.49677511
36 3364.02518928
37 3362.78501271
38 3361.43977691
39 3359.8537339
40 3358.40389381
41 3356.87464499
42 3355.1467079
43 3353.38305176
44 3351.87112389
45 3350.2260748
46 3348.35168706
47 3346.37604568
48 3344.27519227
49 3342.18341596
50 3340.13615412
```

```
51 3338.05365006
52 3335.87144199
53 3333.41969356
54 3330.83221337
55 3327.95127669
56 3324.84257802
57 3322.0416319
58 3319.16234253
59 3316.3473465
60 3313.37295427
61 3310.49165346
62 3307.28385959
63 3304.00323659
64 3300.82293052
65 3297.99970987
66 3295.75455656
67 3293.81779719
68 3291.97788891
69 3290.49808839
70 3288.82862761
71 3287.55682522
72 3286.63368372
73 3285.99950652
74 3285.31668922
75 3284.45321373
76 3283.23801208
77 3281.78382236
78 3280.24517945
79 3278.84454485
80 3277.25266183
81 3275.51790068
82 3273.67190647
83 3271.81521812
84 3269.9431567
85 3267.96365612
86 3265.72522362
87 3263.17232414
88 3260.51327019
89 3258.24482526
90 3256.3866625
91 3254.44642971
92 3252.24169692
93 3249.82006317
94 3247.35481361
95 3244.81039009
96 3242.20922239
```



```
97 3239.72801646
98 3237.40627784
99 3234.94280414
100 3232.77566797
101 3230.69377997
102 3228.47989115
103 3226.02924869
104 3223.31669341
105 3220.71667971
106 3218.04554086
107 3215.45717012
108 3213.09564425
109 3210.93683631
110 3208.59910428
111 3206.2020228
112 3203.69862518
113 3201.29193291
114 3198.72893757
115 3196.29244053
116 3193.78519901
117 3191.20431635
118 3188.26711971
119 3185.12243763
120 3182.16120409
121 3179.18510355
122 3176.48397672
123 3174.20406002
124 3172.03853225
125 3170.24383412
126 3168.58344488
127 3166.87999717
128 3165.09531244
129 3163.31661213
130 3161.42268393
131 3159.40143804
132 3157.5876041
133 3155.71798425
134 3154.00147959
135 3152.59721439
136 3151.14431428
137 3149.67417281
138 3148.35140712
139 3147.08168744
140 3145.79569992
141 3144.4006539
142 3143.04111531
```

```
143 3141.64944648
144 3140.43713912
145 3139.25935606
146 3137.99021553
147 3136.85180613
148 3136.06412775
149 3135.27798404
150 3134.53058866
151 3133.80460728
152 3132.87800713
153 3131.99527854
154 3131.01588024
155 3130.09252017
156 3129.28375933
157 3128.60271563
158 3128.06204592
159 3127.55347909
160 3127.1236509
161 3126.69595292
162 3126.23457725
163 3125.78166362
164 3125.40652465
165 3125.08931784
166 3124.85547976
167 3124.59595213
168 3124.31859554
169 3123.98557577
170 3123.65897693
171 3123.19598904
172 3122.68951641
173 3122.25382081
174 3121.80650338
175 3121.37520038
176 3121.08536914
177 3120.70361954
178 3120.33402496
179 3119.92323731
180 3119.53582426
181 3119.06714439
182 3118.59252274
183 3118.07158124
184 3117.57261216
185 3117.03673574
186 3116.49588772
187 3115.93681307
188 3115.34079945
```

```
189 3114.65744936
190 3113.91109348
191 3113.22003957
192 3112.64482366
193 3112.01945422
194 3111.3382306
195 3110.71302774
196 3110.04597428
197 3109.45294195
198 3109.10198756
199 3108.87170342
200 3108.60347398
201 3108.22979951
202 3107.74561401
203 3107.26572759
204 3106.75330697
205 3106.23290917
206 3105.69647949
207 3105.20698392
208 3104.77487977
209 3104.37557331
210 3104.01187388
211 3103.69034947
212 3103.37304116
213 3103.08534168
214 3102.73296415
215 3102.32582518
216 3101.89548411
217 3101.39862864
218 3100.91169405
219 3100.57661925
220 3100.25039012
221 3099.90928657
222 3099.58402974
223 3099.29819089
224 3099.06248655
225 3098.80589936
226 3098.51908921
227 3098.1828542
228 3097.79177092
229 3097.44975923
230 3097.09948512
231 3096.73779302
232 3096.4186162
233 3096.13915026
234 3095.87259575
```

```
235 3095.60801025
236 3095.39126726
237 3095.15259978
238 3094.90146284
239 3094.68612559
240 3094.45551959
241 3094.25702271
242 3094.05871604
243 3093.87283206
244 3093.67792791
245 3093.43898554
246 3093.14471602
247 3092.82933052
248 3092.56720895
249 3092.34146051
250 3092.12097417
251 3091.91494931
252 3091.67683608
253 3091.47605608
254 3091.29069908
255 3091.11613855
256 3090.95214451
257 3090.77552722
258 3090.63105552
259 3090.50208367
260 3090.36575605
261 3090.306496
262 3090.23548283
263 3090.15826187
264 3090.10753242
265 3090.09813437
266 3090.07926905
267 3090.03551191
268 3089.94110065
269 3089.83812378
270 3089.6771371
271 3089.45384228
272 3089.20636229
273 3088.94872664
274 3088.68614012
275 3088.41624462
276 3088.15176819
277 3087.85913983
278 3087.54890355
279 3087.26278425
280 3087.00674136
```

```
281 3086.76313181
282 3086.52667818
283 3086.30999801
284 3086.11442809
285 3085.94162213
286 3085.8722325
287 3085.77796736
288 3085.657085
289 3085.52392424
290 3085.40790028
291 3085.27818146
292 3085.1999139
293 3085.21398912
```

References

- [1] Mohammed Aturban. *Assignment 9*. 2013. URL: https://github.com/maturban/cs595-f13/blob/master/assignment9/latex/Mohamed_Aturban_Assignment_9.tex.
- [2] Shawn Jones. *Assignment 9*. 2013. URL: <https://github.com/shawnmjones/cs595-f13/tree/master/assignment9>.
- [3] Toby Segaran. *Programming Collective Intelligence: Building Smart Web 2.0 Applications*. O'Reilly, 2007.