Khaja Masroor Ahmed

Assignment 4

CS 751: Introduction to Digital Libraries
Dr. Michael Nelson
Spring 2015

May 2, 2015

Contents

1

Question 1

1.1 Question

- Using the pages from A3 that boilerpipe successfully processed, download those representations again & reprocess them with boilerpipe.
- Document the time difference (e.g., Time(A4) Time(A3)).
- Compute the Jaccard Distance x for each pair of pages (i.e., P(A3) & P(A4) for:
  - Unique terms (i.e., unigrams)
  - Bigrams
  - Trigrams
- See: `http://en.wikipedia.org/wiki/Jaccard_index`
- For each of the 3 cases (i.e., 1-, 2-, 3-grams) build a Cumulative Distribution Function that shows the % change on the x-axis & the % of the population on the x-axis
- See: `http://en.wikipedia.org/wiki/Cumulative_distribution_function`
- Give 3-4 examples illustrating the range of change that you have measured.

1.2 Solution

- The time difference between the time since I ran the boilerpipe for the URIs for assignment three and assignment four was 30 days.
- I used the same boilerpipe library as I used in assignment three.
- From the 10000 URIs I ran through boilerpipe I was able to successfully retrieve the data from 6086 URIs.
- I used the 'ngrams' python library for fetching the unigrams, bigrams and trigrams. I used the set data structure to ensure that only the unique terms were included in the comparison.
- The Jaccard Distance can be calculated by finding the union and the intersection. It is the ratio of the difference of the union and intersection by the union of the set.

- Below are the graphs for unigrams, bigrams and trigrams.

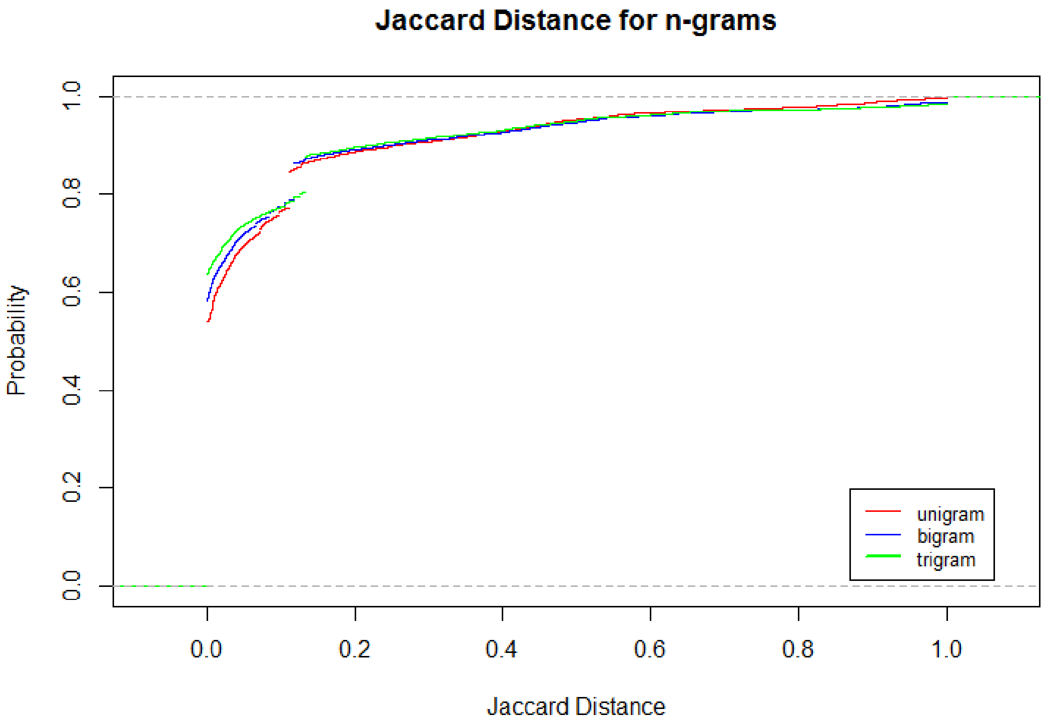


Fig. 1.1. CDF - Jaccard Distance vs. Probability

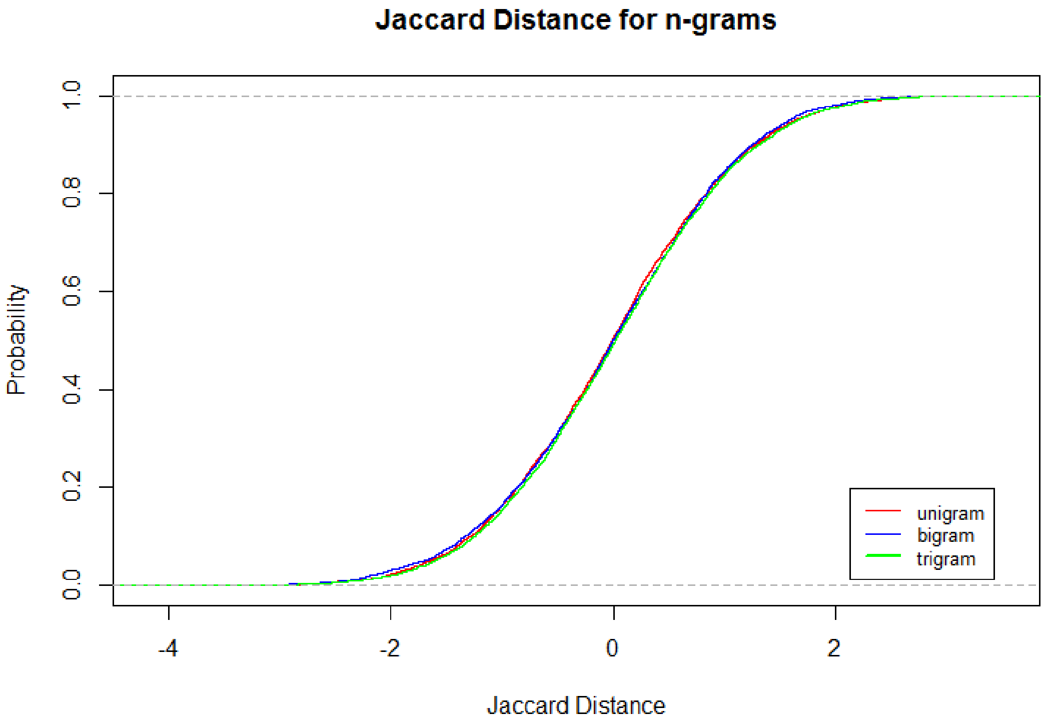


Fig. 1.2. CDF normalized - Jaccard Distance vs. Probability

1.3 Code Listing

```python
import json
from boilerpipe.extract import Extractor
import datetime

f = open('status.txt', 'r+')
print datetime.datetime.now()
for line in f:
        data = json.loads(line)
        try:
                if data['index'] == 10001:
                        print 'Program Executed'
                        break
                finalURL=data['tweetURLData'][0]['finalURL']
                extractor = Extractor(extractor='DefaultExtractor', url=finalURL)
                extracted_text = extractor.getText()
                link = str(data['index']) + '.txt'
                g = open(link, 'w')
                g.write(extracted_text.encode('utf-8'))
                g.close()
        except:
                print datetime.datetime.now()
                print data['index']
                continue
```

Listing 1.1. Python program for fetching the boilerpipe content from the URIs.

```
from nltk.util import ngrams
import os

MAX_NGRAMS=3

def jaccardDistance(list1, list2):
        union = set(list1).union(list2)
        intersect = set(list1).intersection(list2)
        if len(union) == 0:
                return 0
        dist = (len(union) - len(intersect)) * 1.0 / len(
            union)
        return dist

def getNGrams(fileName):
        f = open(fileName, 'r+')
        unigramList=[]
        bigramList=[]
        trigramList=[]

        for line in f:
                unigram = ngrams(line.split(), MAX_NGRAMS-2)
                bigram = ngrams(line.split(), MAX_NGRAMS-1)
                trigram = ngrams(line.split(), MAX_NGRAMS)
                for grams in unigram:
                        unigramList.append(grams)
                #print len(unigramList)
                #print unigramList
                if len(unigramList) == 0:
                        print fileName
                        return unigramList,bigramList,
                            trigramList,True
                for grams in bigram:
                        bigramList.append(grams)
                for grams in trigram:
                        trigramList.append(grams)
        f.close()
        return unigramList, bigramList, trigramList, False

def fileExists(fileName):
        return os.path.isfile(fileName)
count = 1
uni = open('unigramOutput.txt','w')
bi = open('bigramOutput.txt','w')
tri = open('trigramOutput.txt','w')
while count < 10001:
        fileName1 = 'a3/' + str(count) + '.txt'
        fileName2 = 'a4/' + str(count) + '.txt'
```

```
    if fileExists(fileName1) and fileExists(fileName2):
        unigramListA3=[]
        bigramListA3=[]
        trigramListA3=[]
        shouldExit = False
        unigramListA3, bigramListA3, trigramListA3, shouldExit = getNGrams(fileName1)

        if shouldExit:
            print 'exit'
            count += 1
            continue

        unigramListA4=[]
        bigramListA4=[]
        trigramListA4=[]
        unigramListA4, bigramListA4, trigramListA4, shouldExit = getNGrams(fileName2)

        if shouldExit:
            print 'exit2'
            count += 1
            continue
        #print str(jaccardDistance(unigramListA3, unigramListA4))
        uni.write(str(jaccardDistance(unigramListA3, unigramListA4)))
        uni.write('\n')
        bi.write(str(jaccardDistance(bigramListA3, bigramListA4)))
        bi.write('\n')
        tri.write(str(jaccardDistance(trigramListA3, trigramListA4)))
        tri.write('\n')
    count += 1
uni.close()
bi.close()
tri.close()
print 'Program Executed'
```

Listing 1.2. Python program for calculating the Jaccard Distance.

```
dp1 <- read.table(’c:/users/kahmed/desktop/unigramOutput.txt’, header=FALSE)
dp2 <- read.table(’c:/users/kahmed/desktop/bigramOutput.txt’, header=FALSE)
dp3 <- read.table(’c:/users/kahmed/desktop/trigramOutput.txt’, header=FALSE)
datapoint1 <- dp1[,1]
datapoint2 <- dp2[,1]
datapoint3 <- dp3[,1]
X1 = rnorm(sort(datapoint1))
X2 = rnorm(datapoint2)
X3 = rnorm(datapoint3)
P1 = ecdf(datapoint1)
P2 = ecdf(X2)
P3 = ecdf(X3)
plot(P1,  col=”red”, xlab=”Number of Mementos”, ylab=”Probability”, main = ”Number of Mementos for each URI”)
lines(P2, col=”blue”)
lines(P3, col=”green”)
legend(”bottomright”,inset = 0.05, c(”unigram”,”bigram”, ”trigram”),
     cex=.8, col=c(”red”,”blue”, ”green”), lwd=c(1,1.5,2))#pch=c(1,3))
```

Listing 1.3. R program for generating the CDF for Jaccard Distance.

2

Question 2

2.1 Question

- Using the pages from Q1 (A4), download all TimeMaps (including TimeMaps with 404 responses, i.e. empty or null TimeMaps)
  - Upload all the TimeMaps to github
- Build a CDF for # of mementos for each original URI (i.e., x-axis = # of mementos, y-axis = % of links)
- See: `http://timetravel.mementoweb.org/guide/api/`

2.2 Solution

- For downloading the TimeMaps I modified my script from the first assignment to retrieve the html pages. I just had to add a prefix to the URL to include the memento aggregator URL. But I wasn't able to successfully download the TimeMaps for most of the files.
- I then started using the script as provided in the mailing group by Alexander Nwala for retrieving the TimeMaps, but I noticed that the script was taking too long to run.
- Soon after I got back to using my script with a modified memento aggregator URL to point to the following URL `http://labs.mementoweb.org/timemap/json/`. I was successfully able to retrieve in JSON format which would make my life simpler in processing the data.
- Soon after I committed the TimeMaps to github.
- I wrote a script to find the number of mementos in each of the URIs

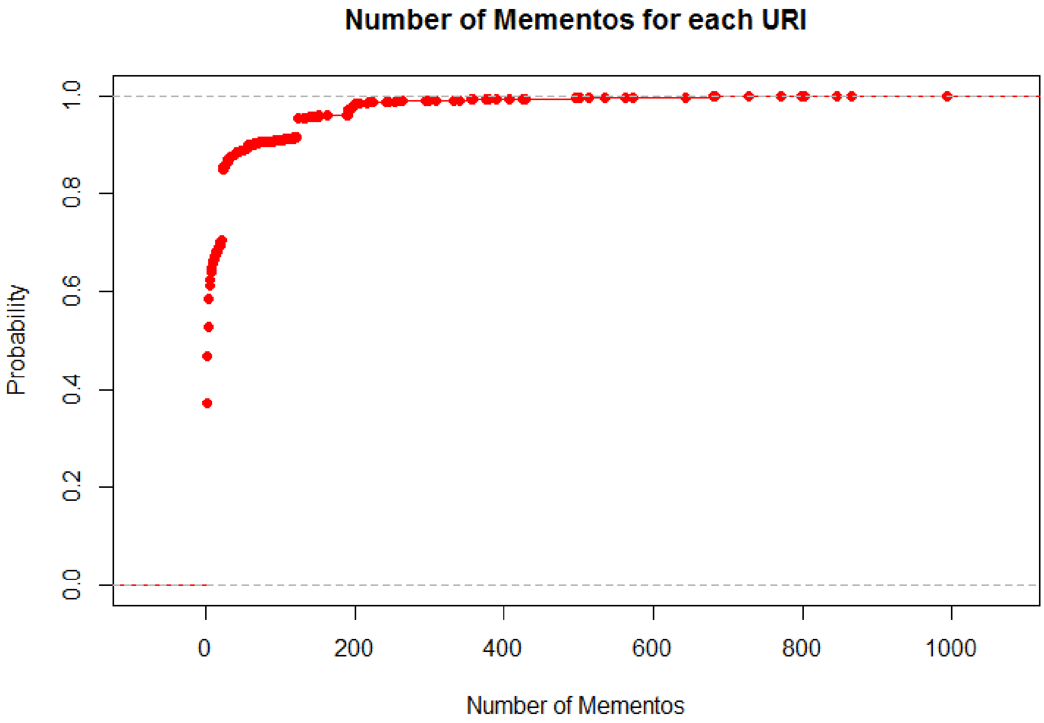


Fig. 2.1. CDF - Number of Mementos for each URI

2.3 Code Listing

```
import os
import datetime
import json

TIME_MAP_URL = "http://labs.mementoweb.org/timemap/json/"

f = open('status.txt', 'r+')
print datetime.datetime.now()
for line in f:
        data = json.loads(line)
        try:
                if data['index'] == 10001:
                        print 'Program Executed'
                        break
                finalURL=data['tweetURLData'][0]['finalURL']
                print '------------------------------------------------------------'
                os.system("wget --output-document=" + str(data['index']) + " " + TIME_MAP_URL + finalURL)
        except:
                continue
print datetime.datetime.now()
```

Listing 2.1. Python program for fetching the TimeMaps.

```
import json
import os

w = open('mementoCount.txt', 'w+')
count = 1
while count < 10001:
        path = 'labs/' + str(count)

        if os.path.isfile(path):
                f = open(path, 'r+')
                if os.stat(path).st_size:
                        try:
                                data = json.loads(f.read())
                                w.write(str(len(data['mementos']['list'])))
                                w.write('\n')
                                f.close()
                        except KeyError:
                                w.write(str(len(data['timemap_index'])))
                                w.write('\n')
                                f.close()
        count += 1
        print count
w.close()
```

Listing 2.2. Python program for finding the number of mementos in each of the TimeMaps.

```
dp1 <- read.table(’c:/users/kahmed/desktop/mementoCount.txt’
    , header=FALSE)
datapoint1 <- dp1[,1]
X1 = rnorm(sort(datapoint1))
P1 = ecdf(datapoint1)
plot(P1,  col=”red”, xlab=”Number of Mementos”, ylab=”
    Probability”, main = ”Number of Mementos for each URI”)
```

Listing 2.3. R program for generating the CDF for number of mementos

3

Question 3

3.1 Question

- Using 20 links that have TimeMaps
  - With ¿= 20 mementos
  - Have existed >= 2 years (i.e., Memento-Datetime of first memento is April XX, 2013 or older)
  - Note: select from Q1/Q2 links, else choose them by hand
- For each link, create a graph that shows Jaccard Distance, relative to the first memento, through time
  - x-axis: continuous time, y-axis: Jaccard Distance relative to the first memento

3.2 Solution

- I manually searched for the 20 URIs that satisfied the condition.
- Then, I ran each of the URIs in the 20 TimeMaps through boilerpipe to retrieve the text. This would help me fetch the unigrams that I would then use to calculate the Jaccard Distance with respect to the first memento.
- I modified the script from question one to calculate the Jaccard Distance.
- Below are the graphs illustrating the Jaccard Distance relative to the first memento.

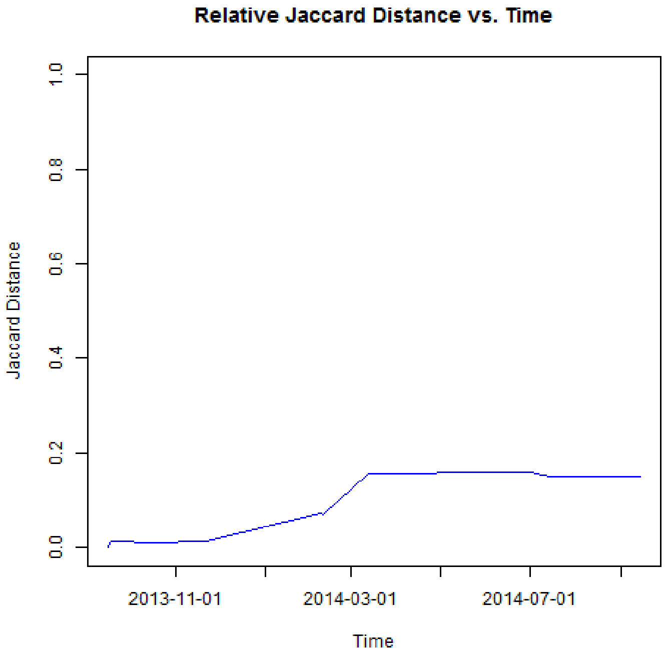


Fig. 3.1. Jaccard Distance relative to first memento for URI# 1

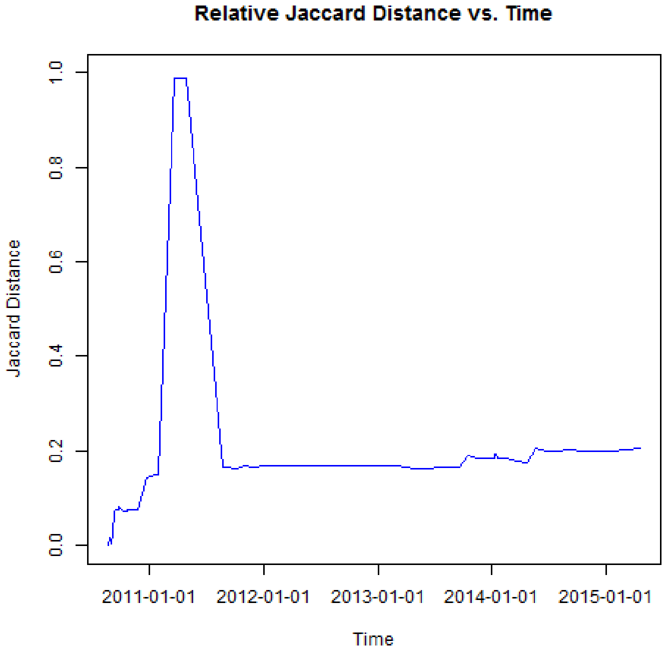


Fig. 3.2. Jaccard Distance relative to first memento for URI# 2

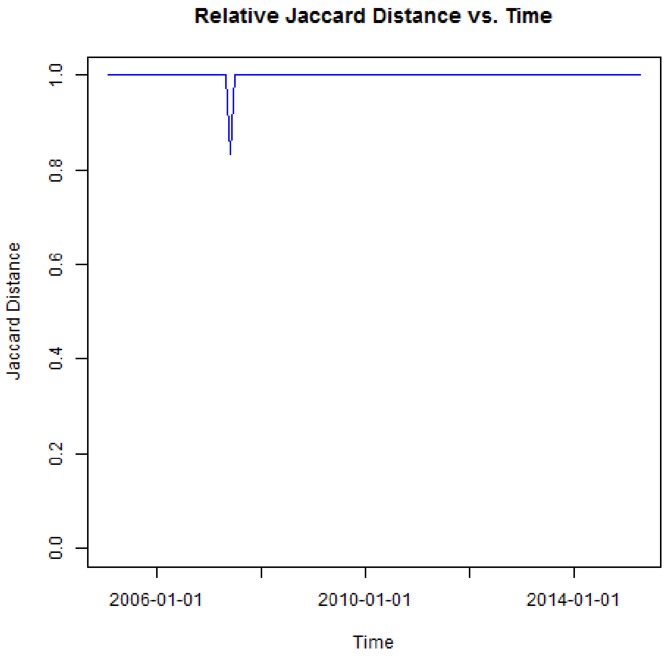


Fig. 3.3. Jaccard Distance relative to first memento for URI# 3

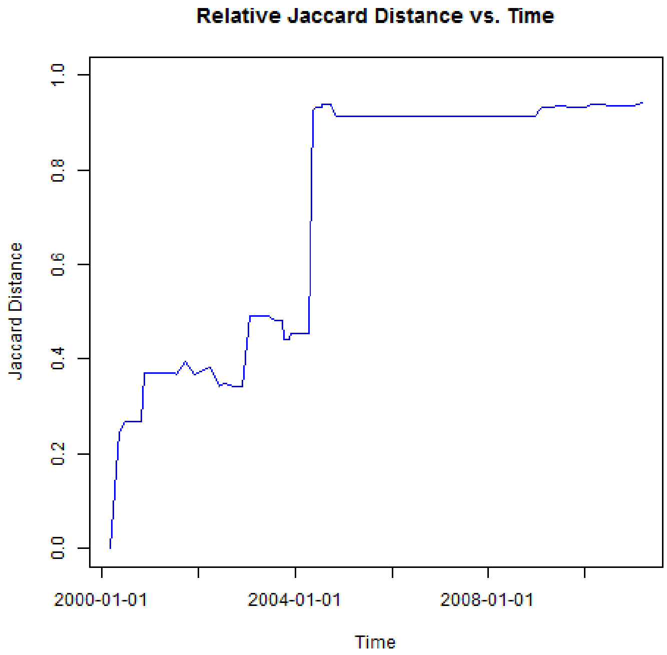


Fig. 3.4. Jaccard Distance relative to first memento for URI# 4

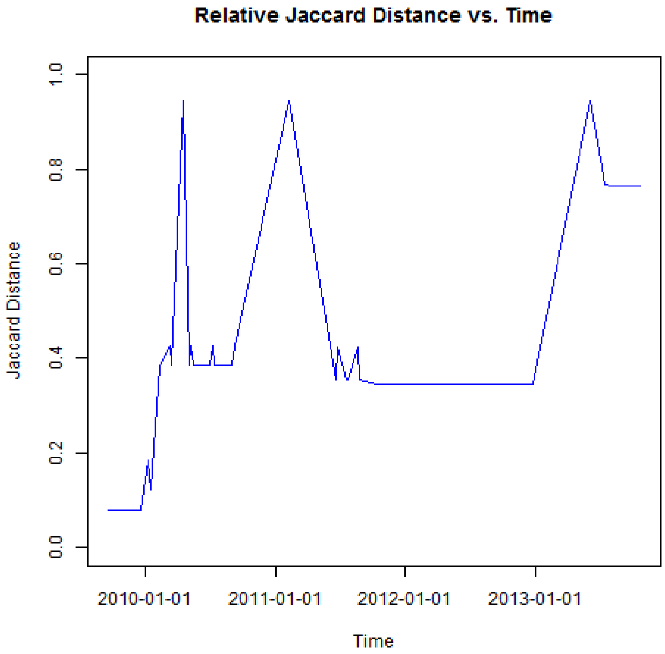


Fig. 3.5. Jaccard Distance relative to first memento for URI# 5

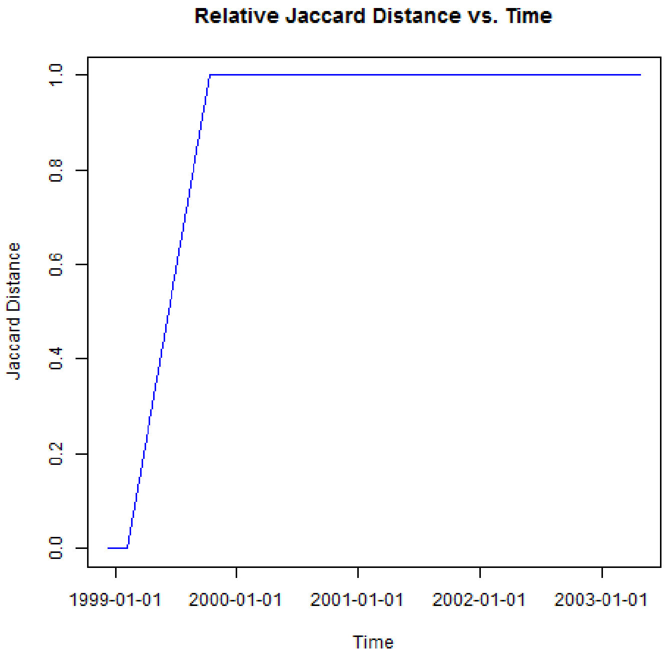


Fig. 3.6. Jaccard Distance relative to first memento for URI# 6

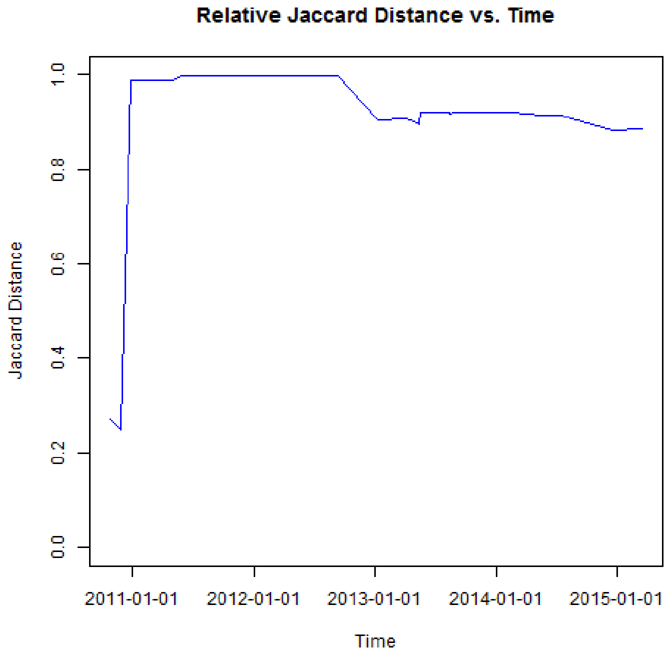


Fig. 3.7. Jaccard Distance relative to first memento for URI# 7

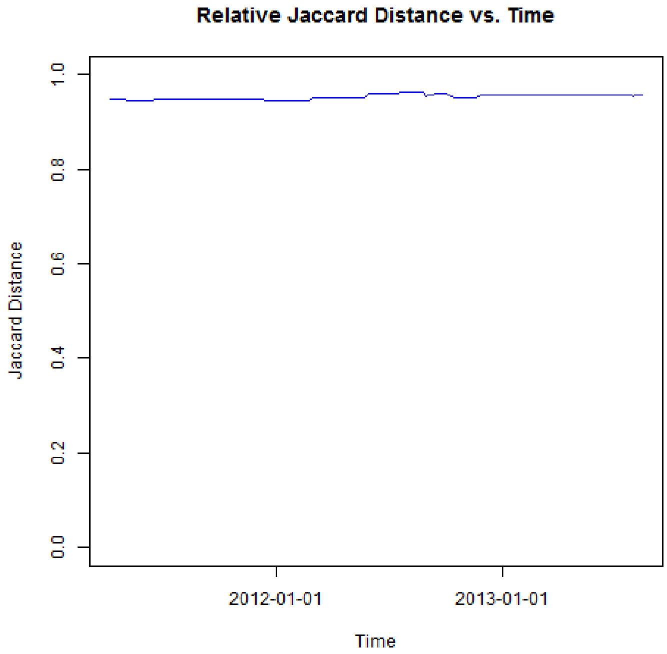


Fig. 3.8. Jaccard Distance relative to first memento for URI# 8

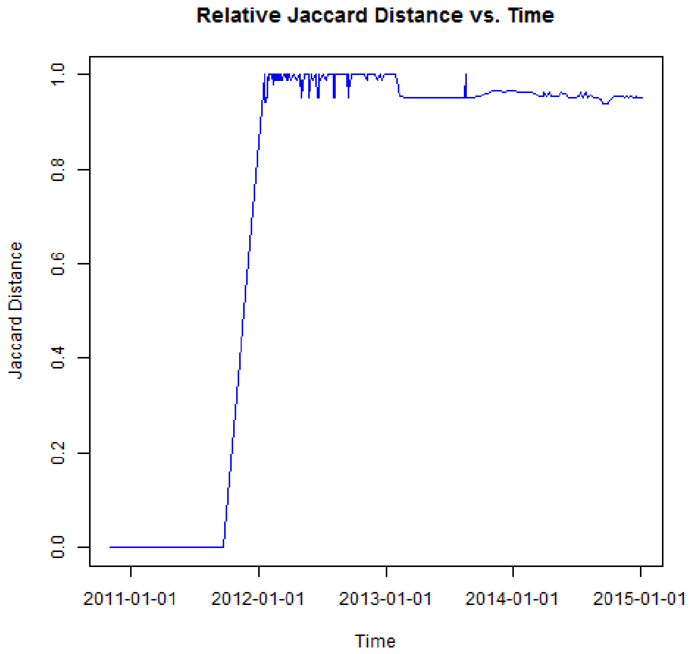


Fig. 3.9. Jaccard Distance relative to first memento for URI# 9

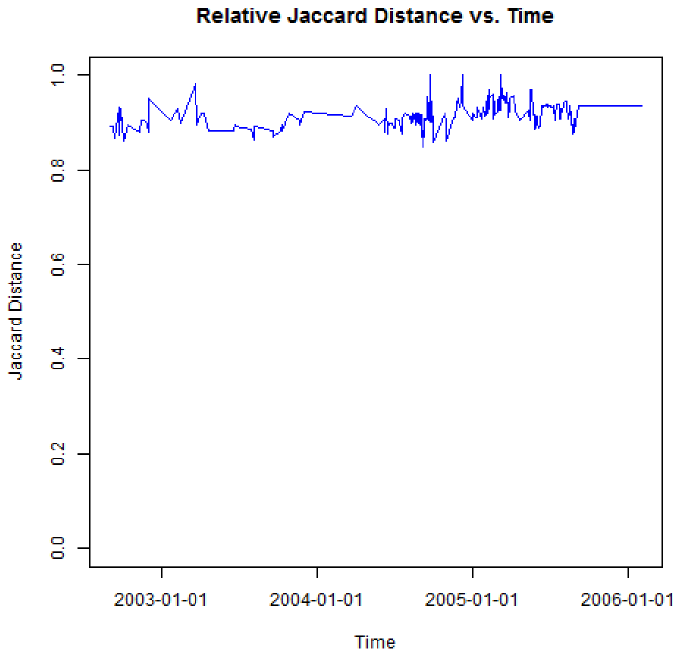


Fig. 3.10. Jaccard Distance relative to first memento for URI# 10

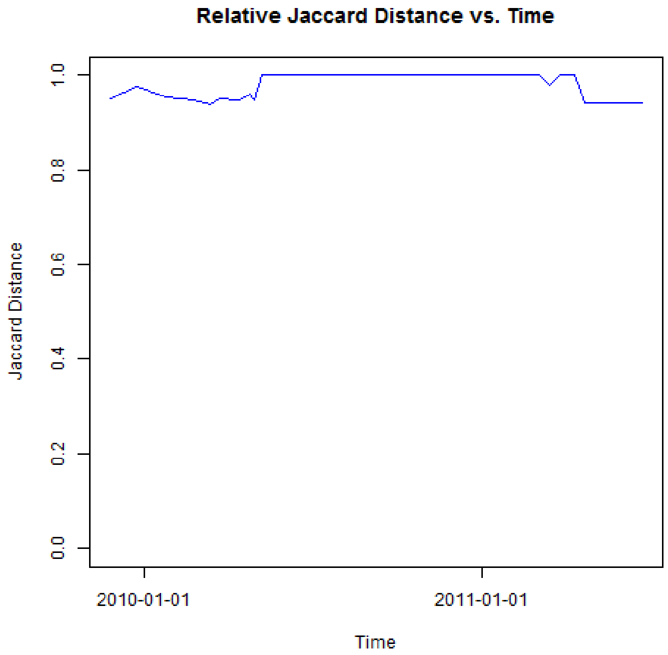


Fig. 3.11. Jaccard Distance relative to first memento for URI# 11

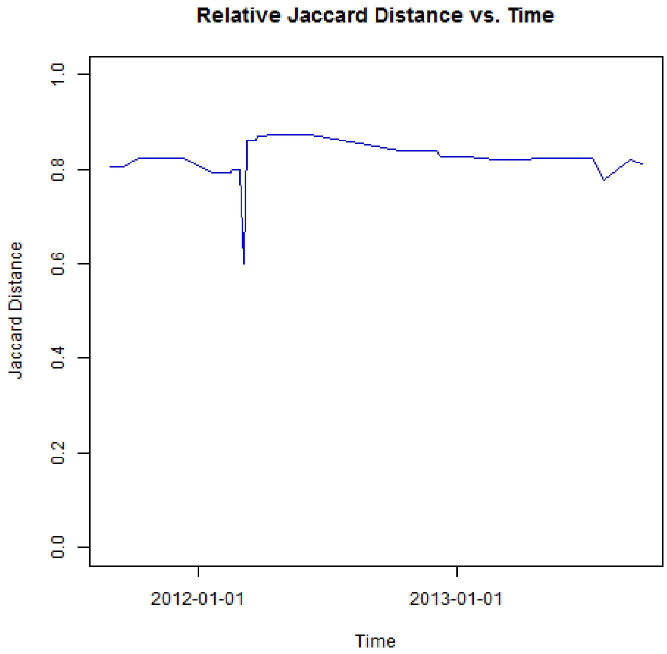


Fig. 3.12. Jaccard Distance relative to first memento for URI# 12

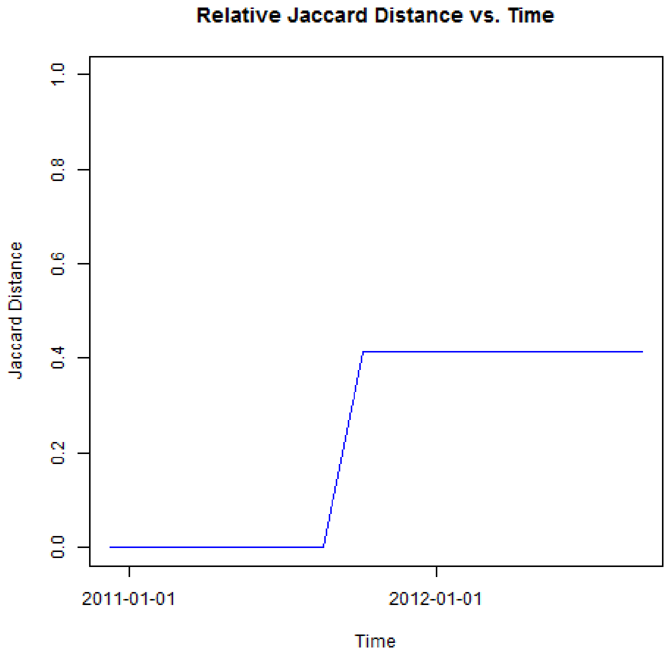


Fig. 3.13. Jaccard Distance relative to first memento for URI# 13

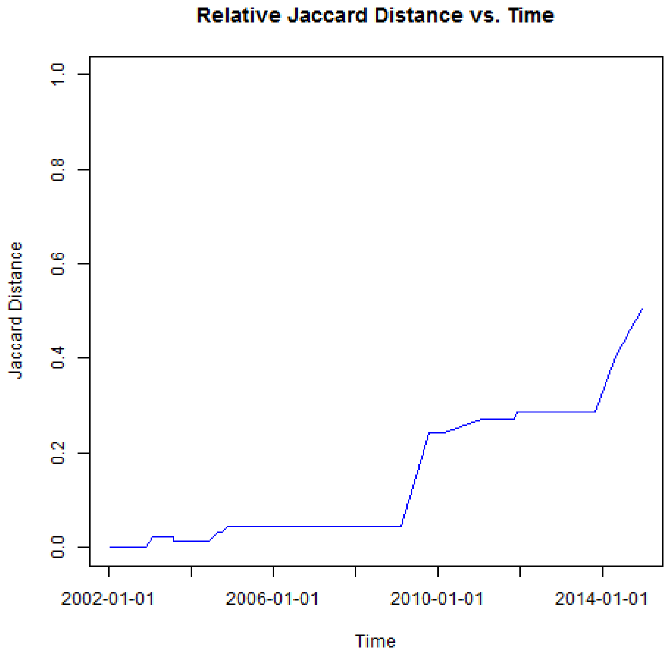


Fig. 3.14. Jaccard Distance relative to first memento for URI# 14

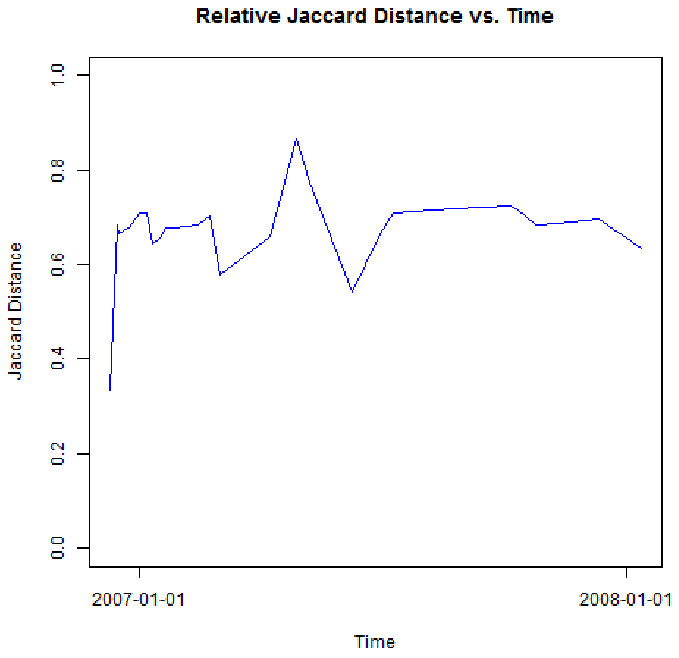


Fig. 3.15. Jaccard Distance relative to first memento for URI# 15

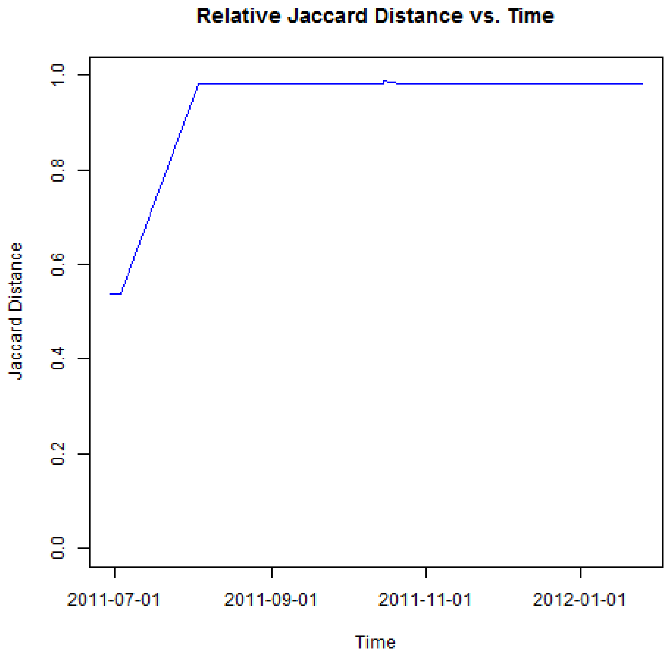


Fig. 3.16. Jaccard Distance relative to first memento for URI# 16

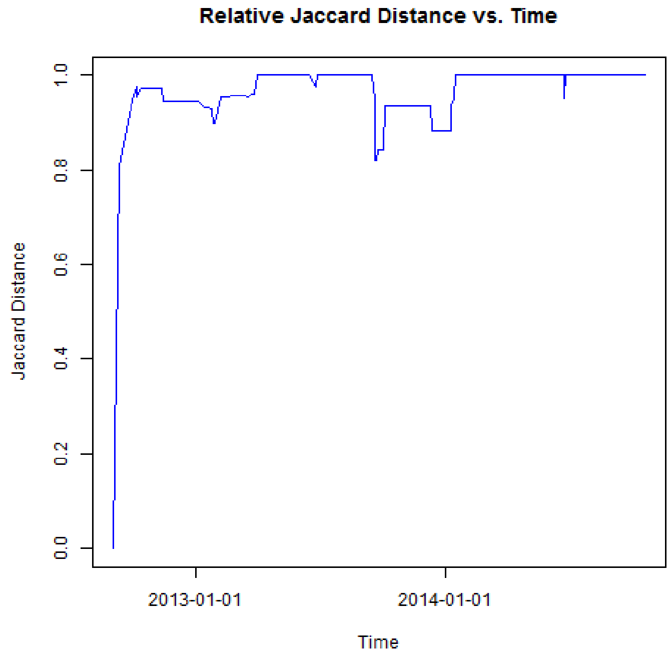


Fig. 3.17. Jaccard Distance relative to first memento for URI# 17

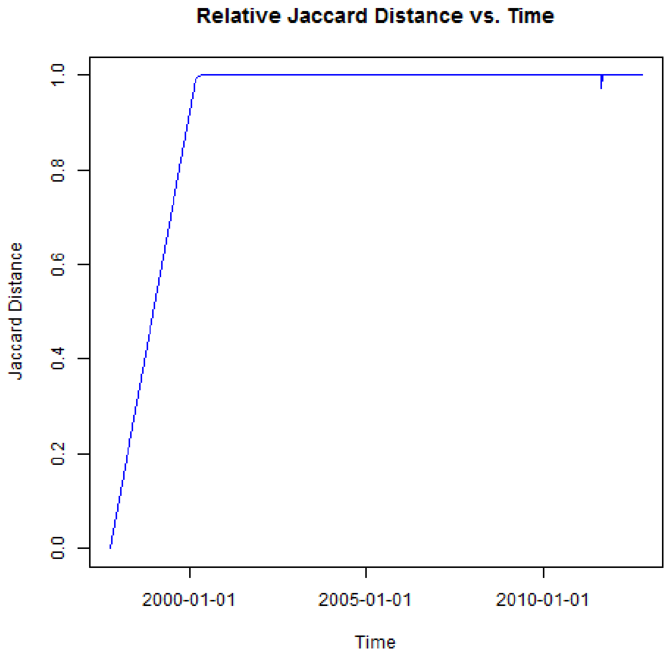


Fig. 3.18. Jaccard Distance relative to first memento for URI# 18

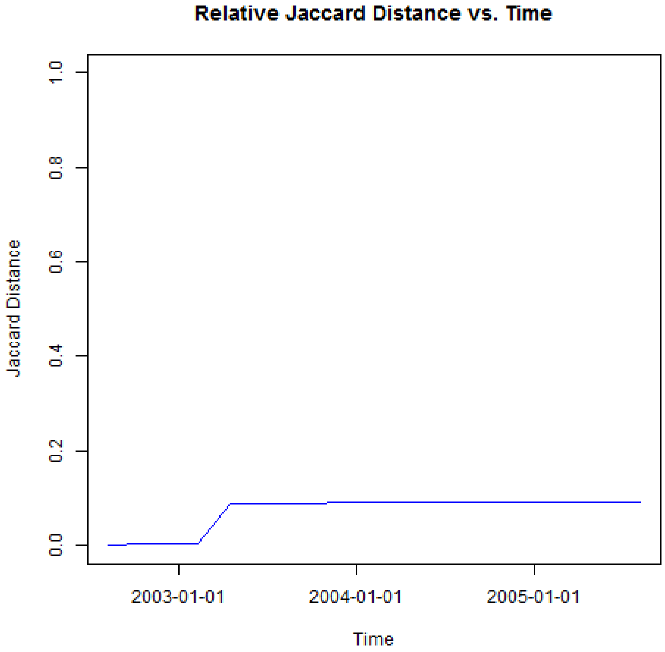


Fig. 3.19. Jaccard Distance relative to first memento for URI# 19

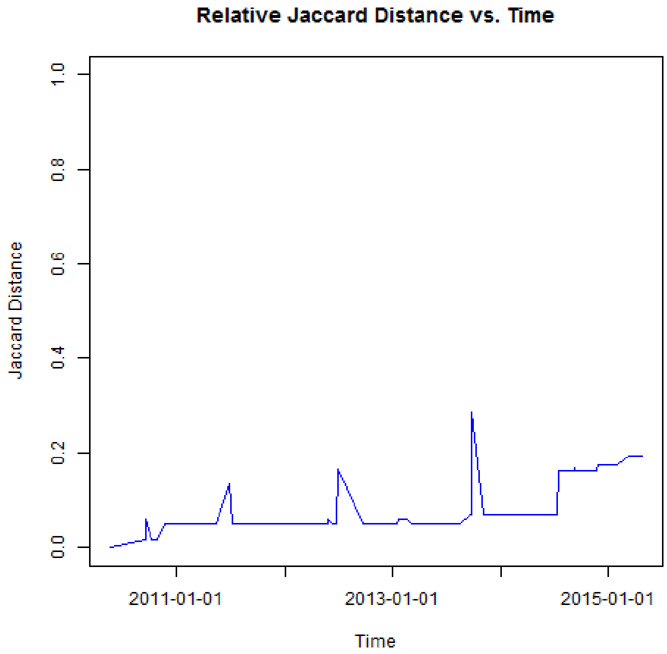


Fig. 3.20. Jaccard Distance relative to first memento for URI# 20

3.3 Code Listing

```
import json
import os
from boilerpipe.extract import Extractor

count = 1
while count < 10001:
    if count == 182 or count == 714 or count == 1106 or count == 1200 or count == 1417 or count == 2077 or count == 2168 or count == 2235 or count == 2338 or count == 2604 or count == 5209 or count == 5986 or count == 6591 or count == 6969 or count == 7861 or count == 8145 or count == 8827 or count == 9093 or count == 9548 or count == 9613:
        print count
        path = 'labs/' + str(count)

        if os.path.isfile(path):
            f = open(path, 'r+')
            if os.stat(path).st_size:
                data = json.loads(f.read())
                if not os.path.exists(str(count)):
                    os.makedirs(str(count))

                filecounter = 1
                link = str(count) + '_datetime.txt'
                g = open(link, 'w')
                for url in data['mementos']['list']:
                    extractor = Extractor(extractor='DefaultExtractor', url=url['uri'])
                    extracted_text = extractor.getText()
                    g.write(str(url['datetime']))
                    g.write('\n')
                g.close()
                filecounter += 1
```

```
                        f.close()
        count += 1
```

Listing 3.1. Python program for fetching boilerpipe content for each memento in the TimeMap.

```
from nltk.util import ngrams
import os

MAX_NGRAMS=1

def jaccardDistance(list1, list2):
        union = set(list1).union(list2)
        intersect = set(list1).intersection(list2)
        if len(union) == 0:
                return 0
        dist = (len(union) - len(intersect)) * 1.0 / len(union)
        return dist

def getNGrams(fileName):
        f = open(fileName, 'r+')
        unigramList=[]
        notEntered = True
        #print fileName
        for line in f:
                notEntered = False
                unigram = ngrams(line.split(), MAX_NGRAMS)
                for grams in unigram:
                        unigramList.append(grams)
                #print len(unigramList)
                #print unigramList
                if len(unigramList) == 0:
                        print fileName
                        return unigramList,True
        f.close()
        if notEntered:
                return unigramList, True
        return unigramList, False

def fileExists(fileName):
        return os.path.isfile(fileName)
count = 1

while count < 10001:
        if os.path.exists(str(count)):
                print str(count)
                fileCounter = 1
                fileName1 = str(count) + '/' + str(fileCounter) + '.txt'
                if fileExists(fileName1):
                        unigramListBaseline=[]
                        shouldExit = False
```

```
            unigramListBaseline, shouldExit = getNGrams(fileName1)
            #print shouldExit
            if shouldExit:
                fileCounter += 1
                fileName1 = str(count) + '/' + str(fileCounter) + '.txt'
                unigramListBaseline=[]
                shouldExit = False
                unigramListBaseline, shouldExit = getNGrams(fileName1)
                print 'exit1'
            #print unigramListBaseline
            uni = open(str(count) + '_jaccard.txt','w')
            while fileCounter < 1000:
                fileCounter += 1
                fileName2 = str(count) + '/' + str(fileCounter) + '.txt'
                if fileExists(fileName2):
                    unigramListCurrent=[]
                    shouldExit = False
                    unigramListCurrent, shouldExit = getNGrams(fileName2)
                    #print shouldExit
                    if shouldExit:
                        print 'exit2'
                        #count += 1
                        uni.write(str(1.0))
                        uni.write('\n')
                        continue
                    #print str(len(unigramListBaseline)) + '\t' + str(len(unigramListCurrent))

```

```
                                        uni.write(str(jaccardDistance(unigramListBaseline, unigramListCurrent)))
                                        uni.write('\n')

                        uni.close()
        count += 1

print 'Program Executed'
```

Listing 3.2. Python program for calculating the Jaccard Distance relative to first memento.

```
data <- read.table(’C:/Users/kahmed/Desktop/q3/uri20.txt’,
    sep=”\t”, colClasses=c(”POSIXct”, ”numeric”))
png(’C:/Users/kahmed/Desktop/q3/uri20.png’)
p1 <- plot(data, type=”l”, col=’blue’, main=”Relative
    Jaccard Distance vs. Time”, xlab=”Time”, ylab=”Jaccard
    Distance”, xaxt=”n”, ylim=c(0,1))
axis.POSIXct(side=1, data$V1, format=”%Y-%m-%d”)
```

Listing 3.3. R program for generating plot for Jaccard Distance vs. Time

4

Question 4

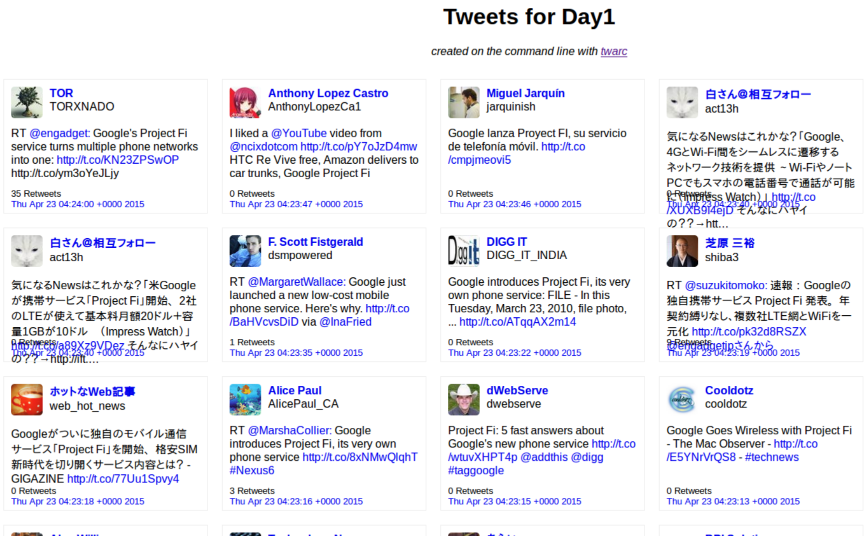
4.1 Question

- Choose a news-related event
- Use twarc.py to collect 1000 tweets, every day for 5 different days
  - See: `https://github.com/edsu/twarc`
- For each day:
  - Create a wall
  - Build a tag/word cloud for each day
  - Create a map using GeoJSON & Github
    - `https://help.github.com/articles/mapping-geojson-files-on-github/`
- Discuss in detail lessons learned, experiences, etc.

4.2 Solution

- I chose ‘Google Fi’ as the topic for this question.
- I installed the twarc package using ‘sudo pip install twarc’ on my ubuntu virtual machine.
- I created a script to fetch 1000 tweets and ran this script for five consecutive days.
- At the end of each day I followed the instructions of the author of the library to create the wall, word cloud, GeoJSON.
- As I progressed to fetch the tweets for the fourth day I noticed it took longer to fetch tweets. I suspected this could be attributed to the reduction in the discussion about this topic.
- Some of the tools provided by twarc are powerful.
- The wall displays the tweets as a html which I didn’t think was of much help as there are multitude of websites that provide this facility of displaying tweets based on search parameters.

- I felt that the word cloud utility was a powerful feature that illustrates the words used in the tweet and the size of the word changes based on the frequency of usage in each of the tweets.
- The geojson utility would be a good tool to visualize the contributors and their location if the geo-location is shared along with the tweet. But due to the limited tweets that had the co-ordinates it would be too premature to make more sense of the feature.
- The html page generated by the wall utility is a good example of a file which would be boilerpipe successful. It has body content which can be extracted.
- The html page generated by the wordcloud utility is a perfect example of a file which would not be boilerpipe successful because it has no body content but receives all its data through scripts.
- I committed the files on github and the GeoJSON can be viewed using github pages from the following URL `https://github.com/maxbizarre/cs851-s15/blob/master/assignment4/twarc/tweetDay1.geojson` and `https://github.com/maxbizarre/cs851-s15/blob/master/assignment4/twarc/tweetDay2.geojson`.



Tweets for Day1

created on the command line with twarc

TOR
TORXNADO
RT @engadget: Google's Project Fi service turns multiple phone networks into one: http://t.co/KN23ZPSwOP http://t.co/ym3oYeJLjy
35 Retweets
Thu Apr 23 04:24:00 +0000 2015

Anthony Lopez Castro
AnthonyLopezCa1
I liked a @YouTube video from @ncixdotcom http://t.co/pY7oJzD4mw HTC Re Vive free, Amazon delivers to car trunks, Google Project Fi
0 Retweets
Thu Apr 23 04:23:47 +0000 2015

Miguel Jarquín
jarquinish
Google lanza Proyect FI, su servicio de telefonía móvil. http://t.co/cmpjmeovi5
0 Retweets
Thu Apr 23 04:23:46 +0000 2015

白さん@相互フォロー
act13h
気になるNewsはこれかな?「Google、4GとWi-Fi間をシームレスに遷移するネットワーク技術を提供 ～Wi-FiやノートPCでもスマホの電話番号で通話が可能に（Impress Watch）」http://t.co/XUXB9I4ejD そんなにハヤイの??→htt...
0 Retweets
Thu Apr 23 04:23:40 +0000 2015

白さん@相互フォロー
act13h
気になるNewsはこれかな?「米Googleが携帯サービス「Project Fi」開始、2社のLTEが使えて基本料月額20ドル+容量1GBが10ドル （Impress Watch）」http://t.co/a89Xz9VDez そんなにハヤイの??→http://ift....
0 Retweets
Thu Apr 23 04:23:40 +0000 2015

F. Scott Fistgerald
dsmpowered
RT @MargaretWallace: Google just launched a new low-cost mobile phone service. Here's why. http://t.co/BaHVcvsDiD via @InaFried
1 Retweets
Thu Apr 23 04:23:35 +0000 2015

DIGG IT
DIGG_IT_INDIA
Google introduces Project Fi, its very own phone service: FILE - In this Tuesday, March 23, 2010, file photo, ... http://t.co/ATqqAX2m14
0 Retweets
Thu Apr 23 04:23:22 +0000 2015

芝原 三裕
shiba3
RT @suzukitomoko: 速報：Googleの独自携帯サービス Project Fi 発表。年契約縛りなし、複数社LTE網とWiFiを一元化 http://t.co/pk32d8RSZX @engadgetjpさんから
9 Retweets
Thu Apr 23 04:23:19 +0000 2015

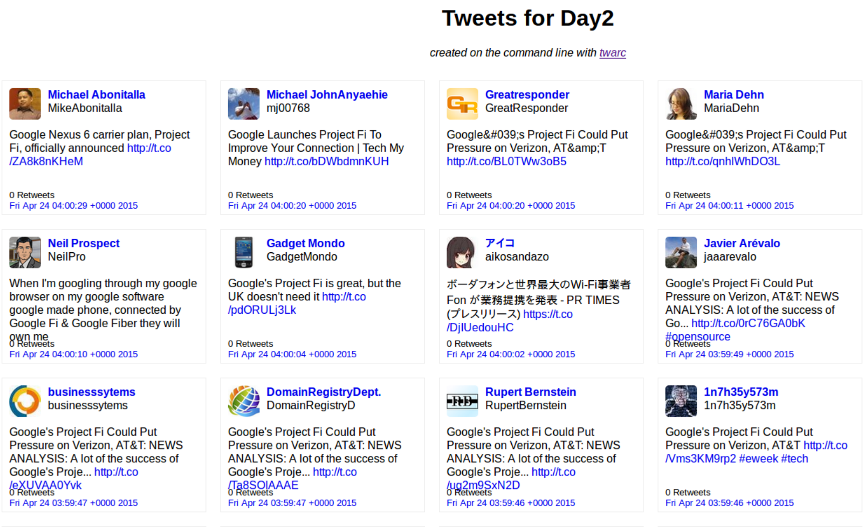
ホットなWeb記事
web_hot_news
Googleがついに独自のモバイル通信サービス「Project Fi」を開始、格安SIM新時代を切り開くサービス内容とは? - GIGAZINE http://t.co/77Uu1Spvy4
0 Retweets
Thu Apr 23 04:23:18 +0000 2015

Alice Paul
AlicePaul_CA
RT @MarshaCollier: Google introduces Project Fi, its very own phone service http://t.co/8xNMwQlqhT #Nexus6
3 Retweets
Thu Apr 23 04:23:16 +0000 2015

dWebServe
dwebserve
Project Fi: 5 fast answers about Google's new phone service http://t.co/wtuvXHPT4p @addthis @digg #taggoogle
0 Retweets
Thu Apr 23 04:23:15 +0000 2015

Cooldotz
cooldotz
Google Goes Wireless with Project Fi - The Mac Observer - http://t.co/E5YNrVrQS8 - #technews
0 Retweets
Thu Apr 23 04:23:13 +0000 2015

Fig. 4.1. Wall - Day 1



Tweets for Day2

created on the command line with twarc

Michael Abonitalla
MikeAbonitalla

Google Nexus 6 carrier plan, Project Fi, officially announced http://t.co/ZA8k8nKHeM

0 Retweets
Fri Apr 24 04:00:29 +0000 2015

Michael JohnAnyaehie
mj00768

Google Launches Project Fi To Improve Your Connection | Tech My Money http://t.co/bDWbdmnKUH

0 Retweets
Fri Apr 24 04:00:20 +0000 2015

Greatresponder
GreatResponder

Google&#039;s Project Fi Could Put Pressure on Verizon, AT&amp;T http://t.co/BL0TWw3oB5

0 Retweets
Fri Apr 24 04:00:20 +0000 2015

Maria Dehn
MariaDehn

Google&#039;s Project Fi Could Put Pressure on Verizon, AT&amp;T http://t.co/qnhlWhDO3L

0 Retweets
Fri Apr 24 04:00:11 +0000 2015

Neil Prospect
NeilPro

When I'm googling through my google browser on my google software google made phone, connected by Google Fi & Google Fiber they will own me

0 Retweets
Fri Apr 24 04:00:10 +0000 2015

Gadget Mondo
GadgetMondo

Google's Project Fi is great, but the UK doesn't need it http://t.co/pdORULj3Lk

0 Retweets
Fri Apr 24 04:00:04 +0000 2015

アイコ
aikosandazo

ボーダフォンと世界最大のWi-Fi事業者Fon が業務提携を発表 - PR TIMES (プレスリリース) https://t.co/DjlUedouHC

0 Retweets
Fri Apr 24 04:00:02 +0000 2015

Javier Arévalo
jaaarevalo

Google's Project Fi Could Put Pressure on Verizon, AT&T: NEWS ANALYSIS: A lot of the success of Go... http://t.co/0rC76GA0bK #opensource

0 Retweets
Fri Apr 24 03:59:49 +0000 2015

businesssytems
businesssytems

Google's Project Fi Could Put Pressure on Verizon, AT&T: NEWS ANALYSIS: A lot of the success of Google's Proje... http://t.co/eXUVAA0Yvk

0 Retweets
Fri Apr 24 03:59:47 +0000 2015

DomainRegistryDept.
DomainRegistryD

Google's Project Fi Could Put Pressure on Verizon, AT&T: NEWS ANALYSIS: A lot of the success of Google's Proje... http://t.co/Ta8SOIAAAE

0 Retweets
Fri Apr 24 03:59:47 +0000 2015

Rupert Bernstein
RupertBernstein

Google's Project Fi Could Put Pressure on Verizon, AT&T: NEWS ANALYSIS: A lot of the success of Google's Proje... http://t.co/ug2m9SxN2D

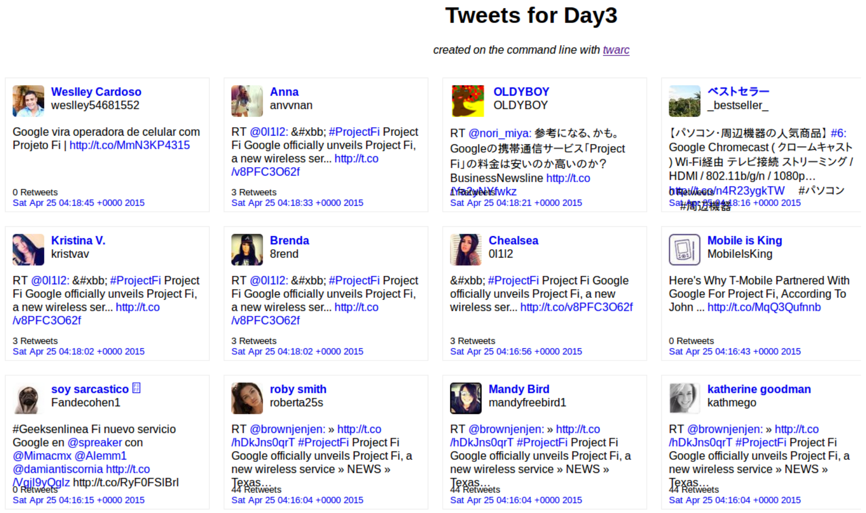
0 Retweets
Fri Apr 24 03:59:46 +0000 2015

1n7h35y573m
1n7h35y573m

Google's Project Fi Could Put Pressure on Verizon, AT&T http://t.co/Vms3KM9rp2 #eweek #tech

0 Retweets
Fri Apr 24 03:59:46 +0000 2015

Fig. 4.2. Wall - Day 2



Tweets for Day3

created on the command line with twarc

Weslley Cardoso
weslley54681552

Google vira operadora de celular com Projeto Fi | http://t.co/MmN3KP4315

0 Retweets
Sat Apr 25 04:18:45 +0000 2015

Anna
anvvnan

RT @0l1l2: &#xbb; #ProjectFi Project Fi Google officially unveils Project Fi, a new wireless ser... http://t.co/v8PFC3O62f

3 Retweets
Sat Apr 25 04:18:33 +0000 2015

OLDYBOY
OLDYBOY

RT @nori_miya: 参考になる、かも。Googleの携帯通信サービス「Project Fi」の料金は安いのか高いのか？ BusinessNewsline http://t.co/Ya2yNYfwkz

1 Retweets
Sat Apr 25 04:18:21 +0000 2015

ベストセラー
bestseller

【パソコン・周辺機器の人気商品】#6: Google Chromecast (クロームキャスト) Wi-Fi経由 テレビ接続 ストリーミング / HDMI / 802.11b/g/n / 1080p… http://t.co/n4R23ygkTW #パソコン #周辺機器

0 Retweets
Sat Apr 25 04:18:16 +0000 2015

Kristina V.
kristvav

RT @0l1l2: &#xbb; #ProjectFi Project Fi Google officially unveils Project Fi, a new wireless ser... http://t.co/v8PFC3O62f

3 Retweets
Sat Apr 25 04:18:02 +0000 2015

Brenda
8rend

RT @0l1l2: &#xbb; #ProjectFi Project Fi Google officially unveils Project Fi, a new wireless ser... http://t.co/v8PFC3O62f

3 Retweets
Sat Apr 25 04:18:02 +0000 2015

Chealsea
0l1l2

&#xbb; #ProjectFi Project Fi Google officially unveils Project Fi, a new wireless ser... http://t.co/v8PFC3O62f

3 Retweets
Sat Apr 25 04:16:56 +0000 2015

Mobile is King
MobileIsKing

Here's Why T-Mobile Partnered With Google For Project Fi, According To John ... http://t.co/MqQ3Qufnnb

0 Retweets
Sat Apr 25 04:16:43 +0000 2015

soy sarcastico
Fandecohen1

#Geeksenlinea Fi nuevo servicio Google en @spreaker con @Mimacmx @Alemm1 @damiantiscornia http://t.co/Vgjl9yQglz http://t.co/RyF0FSlBrl

0 Retweets
Sat Apr 25 04:16:15 +0000 2015

roby smith
roberta25s

RT @brownjenjen: » http://t.co/hDkJns0qrT #ProjectFi Project Fi Google officially unveils Project Fi, a new wireless service » NEWS » Texas...

44 Retweets
Sat Apr 25 04:16:04 +0000 2015

Mandy Bird
mandyfreebird1

RT @brownjenjen: » http://t.co/hDkJns0qrT #ProjectFi Project Fi Google officially unveils Project Fi, a new wireless service » NEWS » Texas...

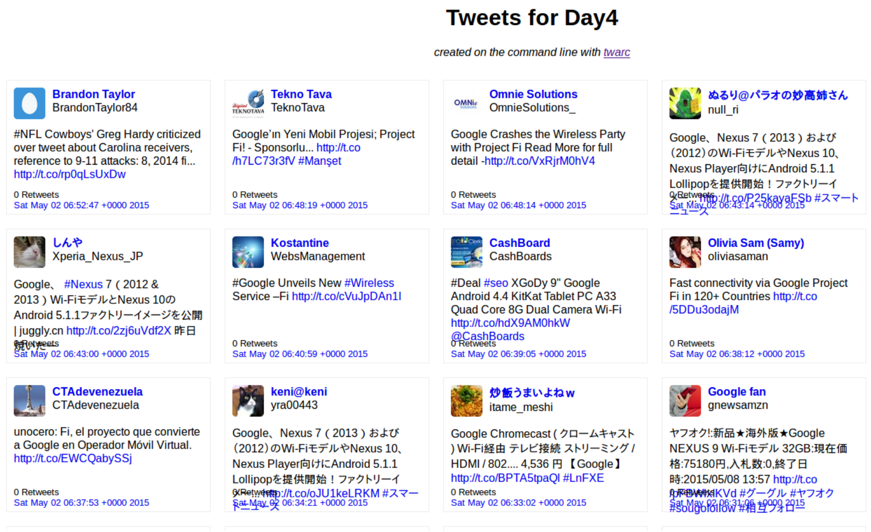
44 Retweets
Sat Apr 25 04:16:04 +0000 2015

katherine goodman
kathmego

RT @brownjenjen: » http://t.co/hDkJns0qrT #ProjectFi Project Fi Google officially unveils Project Fi, a new wireless service » NEWS » Texas...

44 Retweets
Sat Apr 25 04:16:04 +0000 2015

Fig. 4.3. Wall - Day 3



Tweets for Day4

created on the command line with twarc

Brandon Taylor
BrandonTaylor84

#NFL Cowboys' Greg Hardy criticized over tweet about Carolina receivers, reference to 9-11 attacks: 8, 2014 fi... http://t.co/rp0qLsUxDw

0 Retweets
Sat May 02 06:52:47 +0000 2015

Tekno Tava
TeknoTava

Google'ın Yeni Mobil Projesi; Project Fi! - Sponsorlu... http://t.co/h7LC73r3fV #Manşet

0 Retweets
Sat May 02 06:48:19 +0000 2015

Omnie Solutions
OmnieSolutions_

Google Crashes the Wireless Party with Project Fi Read More for full detail -http://t.co/VxRjrM0hV4

0 Retweets
Sat May 02 06:48:14 +0000 2015

ぬるり@パラオの妙高姉さん
null_ri

Google、Nexus 7 (2013) および（2012）のWi-FiモデルやNexus 10、Nexus Player向けにAndroid 5.1.1 Lollipopを提供開始！ファクトリーイメー… http://t.co/P25kayaFSb #スマートニュース

0 Retweets
Sat May 02 06:43:14 +0000 2015

しんや
Xperia_Nexus_JP

Google、 #Nexus 7 (2012 & 2013) Wi-FiモデルとNexus 10のAndroid 5.1.1ファクトリーイメージを公開 | juggly.cn http://t.co/2zj6uVdf2X 昨日焼いた

0 Retweets
Sat May 02 06:43:00 +0000 2015

Kostantine
WebsManagement

#Google Unveils New #Wireless Service –Fi http://t.co/cVuJpDAn1I

0 Retweets
Sat May 02 06:40:59 +0000 2015

CashBoard
CashBoards

#Deal #seo XGoDy 9" Google Android 4.4 KitKat Tablet PC A33 Quad Core 8G Dual Camera Wi-Fi http://t.co/hdX9AM0hkW @CashBoards

0 Retweets
Sat May 02 06:39:05 +0000 2015

Olivia Sam (Samy)
oliviasaman

Fast connectivity via Google Project Fi in 120+ Countries http://t.co/5DDu3odajM

0 Retweets
Sat May 02 06:38:12 +0000 2015

CTAdevenezuela
CTAdevenezuela

unocero: Fi, el proyecto que convierte a Google en Operador Móvil Virtual. http://t.co/EWCQabySSj

0 Retweets
Sat May 02 06:37:53 +0000 2015

keni@keni
yra00443

Google、Nexus 7 (2013) および（2012）のWi-FiモデルやNexus 10、Nexus Player向けにAndroid 5.1.1 Lollipopを提供開始！ファクトリーイメー… http://t.co/oJU1keLRKM #スマートニュース

0 Retweets
Sat May 02 06:34:21 +0000 2015

炒飯うまいよねｗ
itame_meshi

Google Chromecast (クロームキャスト) Wi-Fi経由 テレビ接続 ストリーミング / HDMI / 802.... 4,536 円 【Google】 http://t.co/BPTA5tpaQl #LnFXE

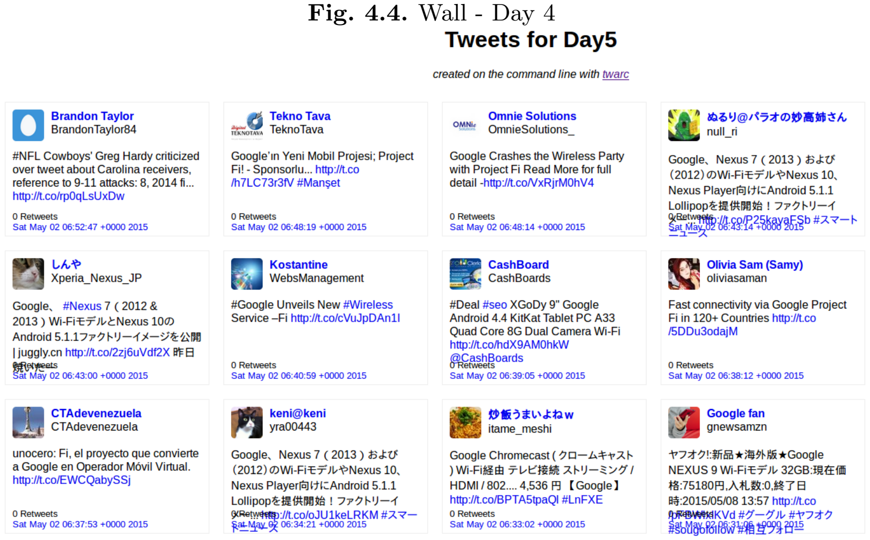
0 Retweets
Sat May 02 06:33:02 +0000 2015

Google fan
gnewsamzn

ヤフオク!:新品★海外版★Google NEXUS 9 Wi-Fiモデル 32GB:現在価格:75180円,入札数:0,終了日時:2015/05/08 13:57 http://t.co/pFBWxlKVd #グーグル #ヤフオク #sougofollow #相互フォロー

0 Retweets
Sat May 02 06:31:06 +0000 2015

Fig. 4.4. Wall - Day 4



Tweets for Day5

created on the command line with twarc

Brandon Taylor
BrandonTaylor84

#NFL Cowboys' Greg Hardy criticized over tweet about Carolina receivers, reference to 9-11 attacks: 8, 2014 fi... http://t.co/rp0qLsUxDw

0 Retweets
Sat May 02 06:52:47 +0000 2015

Tekno Tava
TeknoTava

Google'ın Yeni Mobil Projesi; Project Fi! - Sponsorlu... http://t.co/h7LC73r3fV #Manşet

0 Retweets
Sat May 02 06:48:19 +0000 2015

Omnie Solutions
OmnieSolutions_

Google Crashes the Wireless Party with Project Fi Read More for full detail -http://t.co/VxRjrM0hV4

0 Retweets
Sat May 02 06:48:14 +0000 2015

ぬるり@パラオの妙高姉さん
null_ri

Google、Nexus 7 (2013) および（2012）のWi-FiモデルやNexus 10、Nexus Player向けにAndroid 5.1.1 Lollipopを提供開始！ファクトリーイメー… http://t.co/P25kayaFSb #スマートニュース

0 Retweets
Sat May 02 06:43:14 +0000 2015

しんや
Xperia_Nexus_JP

Google、 #Nexus 7 (2012 & 2013) Wi-FiモデルとNexus 10のAndroid 5.1.1ファクトリーイメージを公開 | juggly.cn http://t.co/2zj6uVdf2X 昨日焼いた

0 Retweets
Sat May 02 06:43:00 +0000 2015

Kostantine
WebsManagement

#Google Unveils New #Wireless Service –Fi http://t.co/cVuJpDAn1I

0 Retweets
Sat May 02 06:40:59 +0000 2015

CashBoard
CashBoards

#Deal #seo XGoDy 9" Google Android 4.4 KitKat Tablet PC A33 Quad Core 8G Dual Camera Wi-Fi http://t.co/hdX9AM0hkW @CashBoards

0 Retweets
Sat May 02 06:39:05 +0000 2015

Olivia Sam (Samy)
oliviasaman

Fast connectivity via Google Project Fi in 120+ Countries http://t.co/5DDu3odajM

0 Retweets
Sat May 02 06:38:12 +0000 2015

CTAdevenezuela
CTAdevenezuela

unocero: Fi, el proyecto que convierte a Google en Operador Móvil Virtual. http://t.co/EWCQabySSj

0 Retweets
Sat May 02 06:37:53 +0000 2015

keni@keni
yra00443

Google、Nexus 7 (2013) および（2012）のWi-FiモデルやNexus 10、Nexus Player向けにAndroid 5.1.1 Lollipopを提供開始！ファクトリーイメー… http://t.co/oJU1keLRKM #スマートニュース

0 Retweets
Sat May 02 06:34:21 +0000 2015

炒飯うまいよねｗ
itame_meshi

Google Chromecast (クロームキャスト) Wi-Fi経由 テレビ接続 ストリーミング / HDMI / 802.... 4,536 円 【Google】 http://t.co/BPTA5tpaQl #LnFXE

0 Retweets
Sat May 02 06:33:02 +0000 2015

Google fan
gnewsamzn

ヤフオク!:新品★海外版★Google NEXUS 9 Wi-Fiモデル 32GB:現在価格:75180円,入札数:0,終了日時:2015/05/08 13:57 http://t.co/pFBWxlKVd #グーグル #ヤフオク #sougofollow #相互フォロー

0 Retweets
Sat May 02 06:31:06 +0000 2015

Fig. 4.5. Wall - Day 5

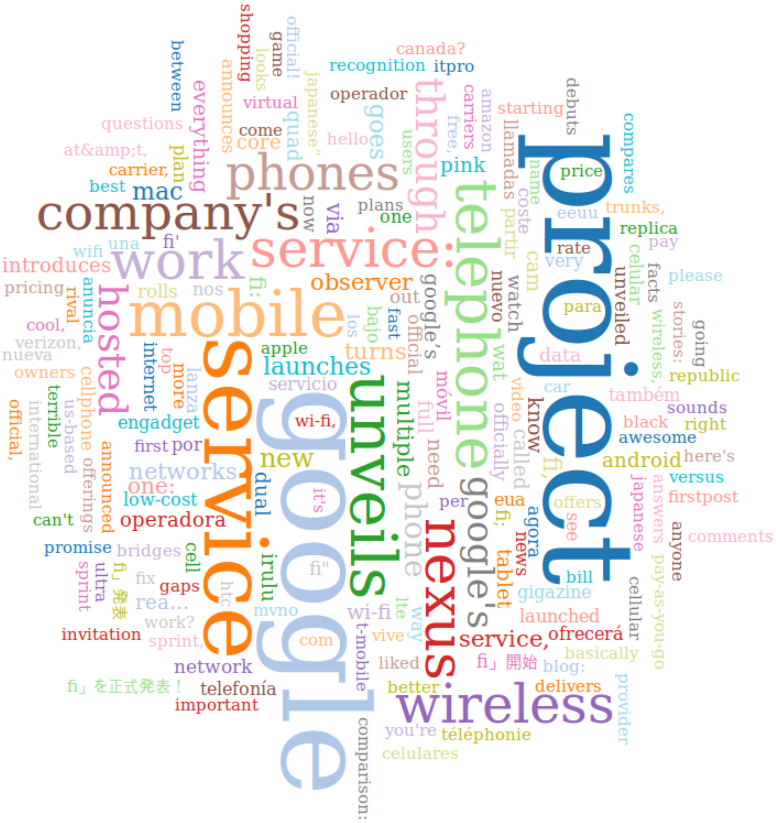


Fig. 4.6. Word Cloud - Day 1

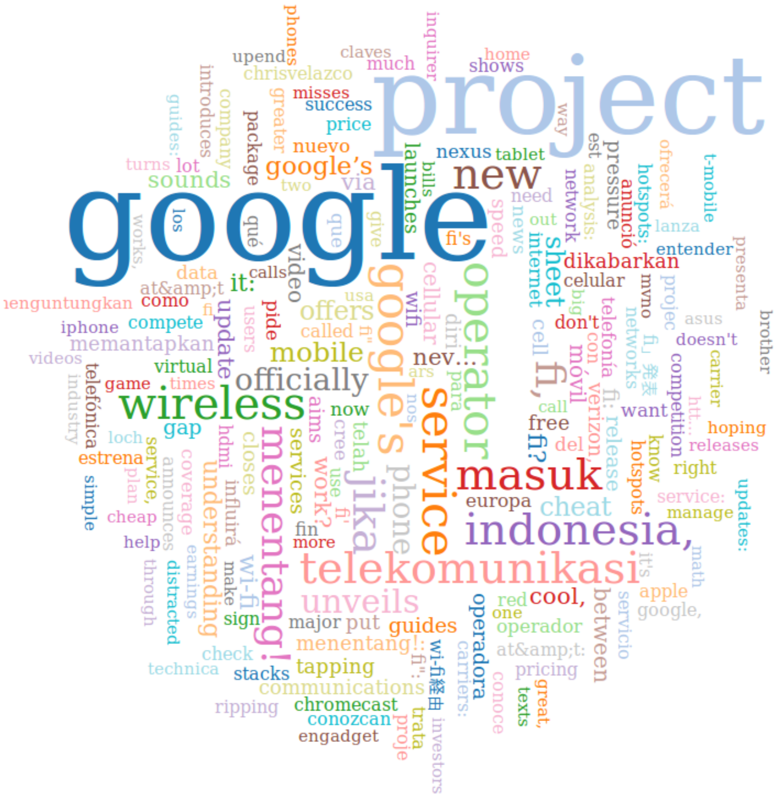


Fig. 4.7. Word Cloud - Day 2

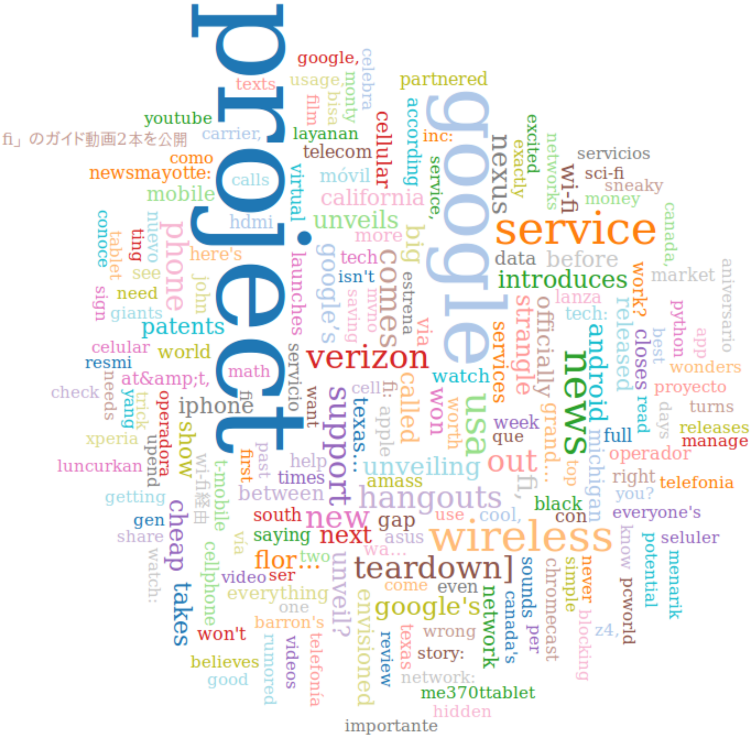


Fig. 4.8. Word Cloud - Day 3

Fig. 4.9. Word Cloud - Day 4

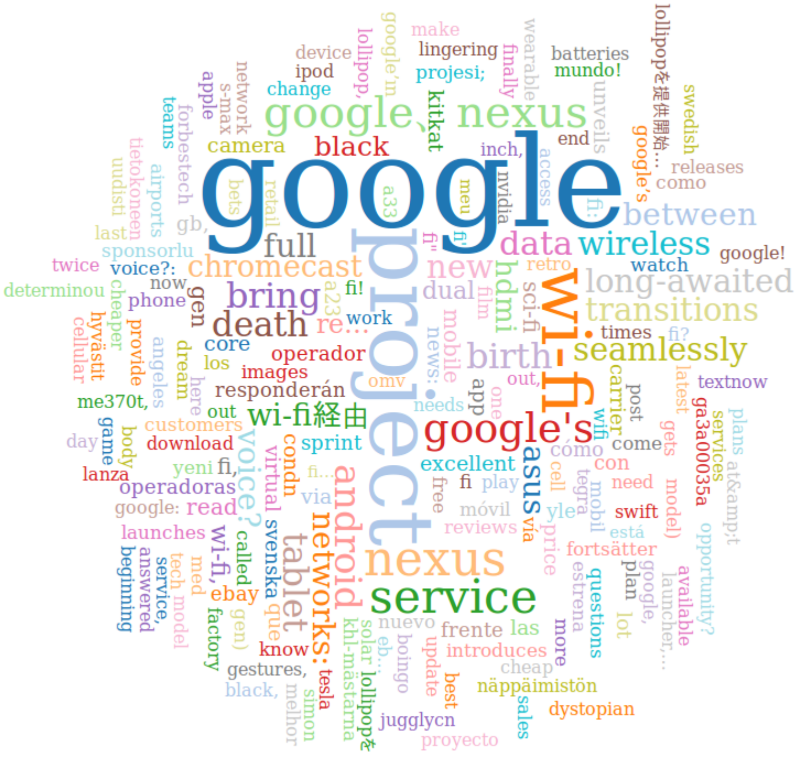


Fig. 4.10. Word Cloud - Day 5

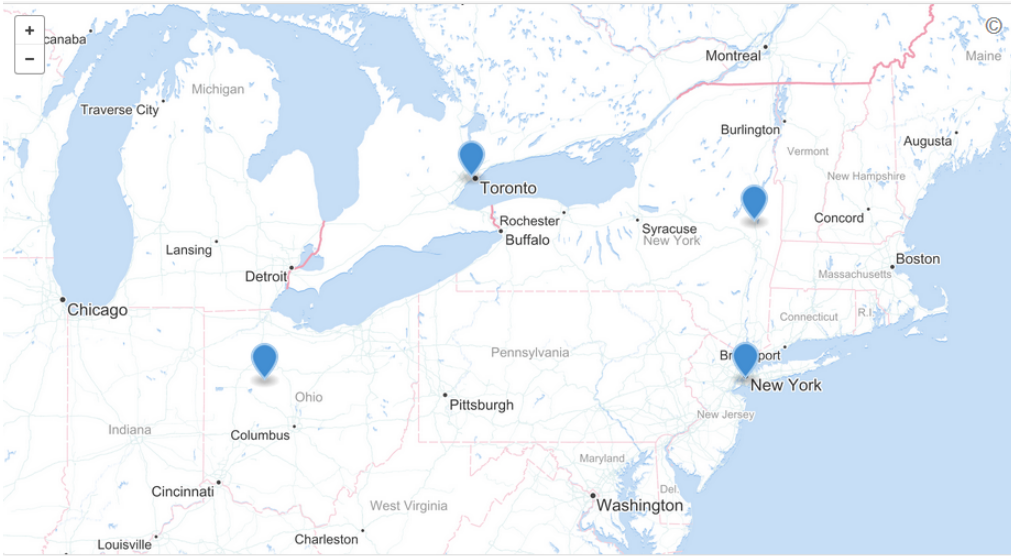


Fig. 4.11. Geographical Location - Day 1

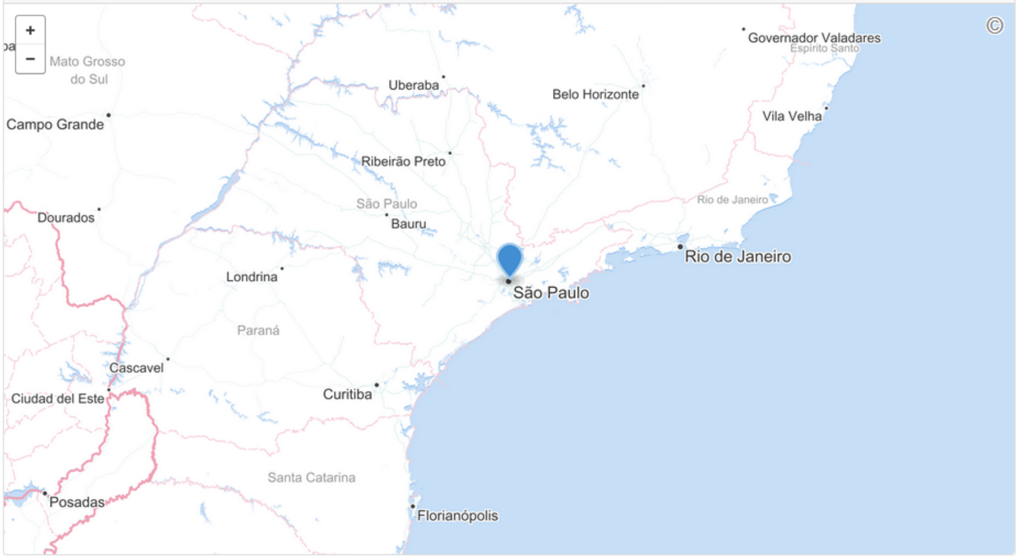


Fig. 4.12. Geographical Location - Day 2

4.3 Code Listing

```
from twarc import Twarc
import json

CONSUMER_KEY = 'CfHUyBhlMaLv5Mn8r2IziXpLs'
CONSUMER_SECRET = '
    PqqtbhbyNb5mcJ2dHkSIT2wupOMuEqfSINGYvV8KDIOPuqgDkN'
ACCESS_TOKEN = '29202483-
    qK6twPLeurVc8Ls8zBxdFtaFGyzm76LUBbtXOMMk1'
ACCESS_TOKEN_SECRET = '
    aOIFdI1TVJjsIPWNO1rAFx2IECzVSCPY4kOnEKBA0pCdA'

w = open('tweetDay5.json', 'w')

t = Twarc(CONSUMER_KEY, CONSUMER_SECRET, ACCESS_TOKEN,
    ACCESS_TOKEN_SECRET)
count = 1
for tweet in t.search("google fi"):
        w.write(json.dumps(tweet))
        w.write('\n')
        print count
        count += 1
        if count >1000:
                break
```

Listing 4.1. Python program for fetching tweets using twarc.

References

1. Cdf in r. http://stats.stackexchange.com/questions/30858/how-to-calculate-cumulative-distribution-in-r.
2. n-grams in python. http://stackoverflow.com/questions/13423919/computing-n-grams-using-python.
3. twarc. https://github.com/edsu/twarc.