

SOCIAL COMPUTING CSC 591 P2 (A) REPORT

The implementation approach I followed to solving the problem is as described below.

The project was implemented with the help of tweepy, a python library for accessing the twitter API. After obtaining the Access Token, Access Secret, Consumer Key and the Consumer Secret for the given application, the authorization for the access is acquired. The StreamListener class filters all tweets with the hashtag 'P2CSC555F15' for further processing.

CHECK – IN

The user's location is pre determined from a set of given locations. First, if the another person checks in at the same position as that of the user, then the user responds to the other person with a check-in of a particular format.

For example, the user's location was set as hunt
One of the check-in tweets that was the input –
I checked in at #hunt #id_cpancha_5 #P2CSC555F15

```
MY RESPONSE TWEET CHECKED
@chinpanz10
Name: sharaths1993
MY_MODE: Loud
EXPECTED_MODE: Loud
```

Fig 1 – Output response to check-in from neighbours

The word 'checked' was searched for in the tweet. The string between the first hashtag and the second hashtag was checked. If the string matched the user's location, the response tweet was constructed accordingly by getting the screen name from the json object of the tweet. The constructed tweet is then updated as a status for the user. The checked-in tweets are appended onto the file 'checkedtweets.txt'.

MAKE – CALLS

To check if the user can request an incoming call, the user responds with a tweet.

For example, one of the call requests is as follows –
CALL #id_yjmehta_15 #P2CSC555F15

```
MY RESPONSE TWEET IN CALL
@yjmehta
Call from: Leia
URGENCY: 0
#id_yjmehta_15 #P2CSC555F15
```

Fig 2 – Output response to call

The word CALL was searched for in the tweet. The response id is the substring between the two hashtags of the tweet. The Call from field is filled randomly from a pre defined list of callers. The urgency

is also chosen randomly. The call tweets are written into the 'calltweets.txt' file. The tweet response is then constructed and the status of the user is updated.

FEEDBACK FOR ACTION

For each call received, the action taken has a feedback from the neighbours. For this, the utility function has to be used and the action should be determined based on the inputs to the utility function. Parameters such as the noise, location should be taken into consideration. Currently, the program chooses a random value from the list and the status is updated accordingly. The tweets are written into the 'actiontweets.txt' file. A snapshot of the output as the feedback for the action is

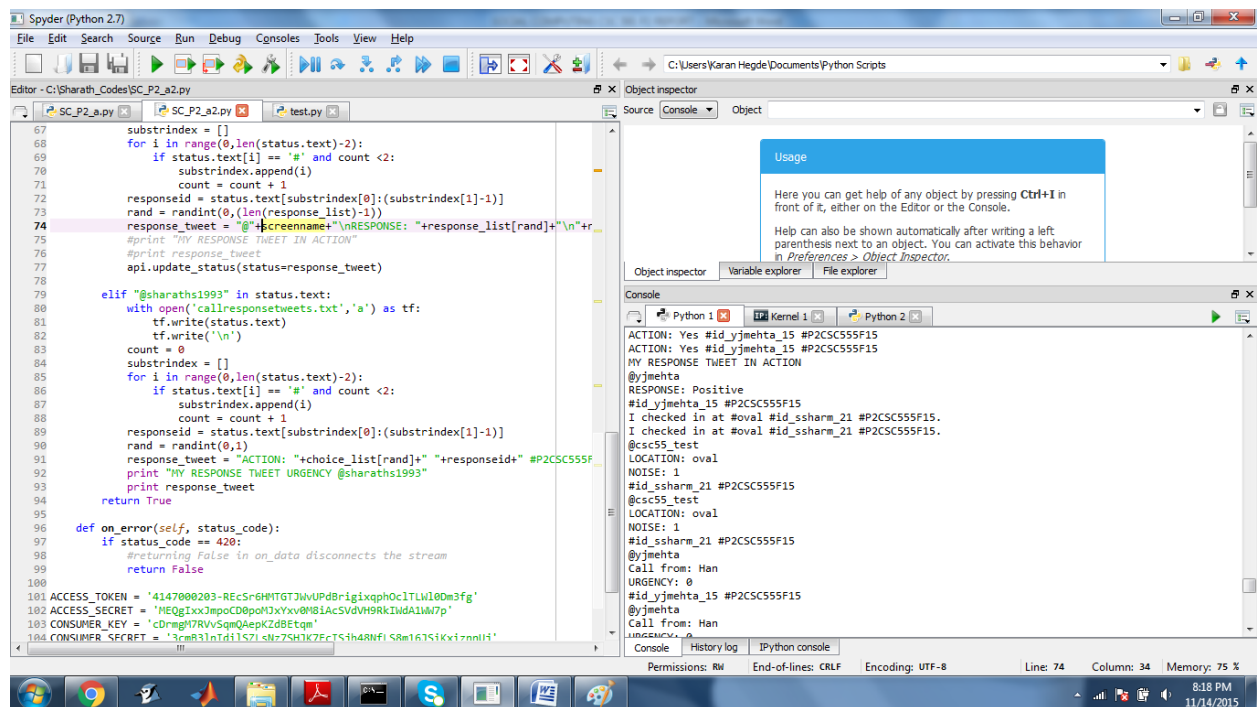
```
ACTION: Yes #id_yjmehta_15 #P2CSC555F15
MY RESPONSE TWEET IN ACTION
@yjmehta
RESPONSE: Positive
#id_yjmehta_15 #P2CSC555F15
```

Fig 3 –Neighbor feedback for action taken

CALL RESPONSE

For each call received, the user should indicate the action was taken. In this case, the utility function comes into use. Depending on the various parameters such as the strength of the relation, type of relation (family, friend, colleague, stranger) the corresponding action should be performed. In this case, the values are chosen at random. The filtered tweets were further selected as the call statements are directed at the screen name of the user. Accordingly, the response is built by obtaining the values from the call and the status is updated.

A few screenshots of the output are as shown below –



The screenshot displays the Spyder Python IDE interface. The editor window on the left contains a Python script with the following visible code:

```
67 substrindex = []
68 for i in range(0, len(status.text)-2):
69     if status.text[i] == '#' and count < 2:
70         substrindex.append(i)
71         count = count + 1
72 responseid = status.text[substrindex[0]:(substrindex[1]-1)]
73 rand = randint(0, (len(response_list)-1))
74 response_tweet = "@" + screenname + "\nRESPONSE: " + response_list[rand] + "\n" + r
75 #print "MY RESPONSE TWEET IN ACTION"
76 #print response_tweet
77 api.update_status(status=response_tweet)
78
79 elif "@sharath1993" in status.text:
80     with open('callresponsetweets.txt', 'a') as tf:
81         tf.write(status.text)
82         tf.write('\n')
83     count = 0
84     substrindex = []
85     for i in range(0, len(status.text)-2):
86         if status.text[i] == '#' and count < 2:
87             substrindex.append(i)
88             count = count + 1
89     responseid = status.text[substrindex[0]:(substrindex[1]-1)]
90     rand = randint(0, 1)
91     response_tweet = "ACTION: " + choice_list[rand] + " " + responseid + " #P2CSC555F15"
92     print "MY RESPONSE TWEET URGENCY @sharath1993"
93     print response_tweet
94     return True
95
96 def on_error(self, status_code):
97     if status_code == 420:
98         #returning False in on_data disconnects the stream
99         return False
100
101 ACCESS_TOKEN = "4147000203-RECSr6HMTGTJWUPdBrigixaphOc1TLWl0m3fg"
102 ACCESS_SECRET = "MEQgIxxJmpoCD0p0NjYxv0N8IAcSVdVH9RKIMdA1W7p"
103 CONSUMER_KEY = "cDrmgH7RVuVsqmQaepKZ0BETqm"
104 CONSUMER_SECRET = "3rm031n1d115Z1sN2ZSH1K7E2r7S1h4RNF1SRm167S1Xvi7nnH1"
```

The console window on the right shows the output of the script:

```
ACTION: Yes #id_yjmehta_15 #P2CSC555F15
ACTION: Yes #id_yjmehta_15 #P2CSC555F15
MY RESPONSE TWEET IN ACTION
@yjmehta
RESPONSE: Positive
#id_yjmehta_15 #P2CSC555F15
I checked in at #oval #id_ssharm_21 #P2CSC555F15.
@csc55_test
LOCATION: oval
NOISE: 1
#id_ssharm_21 #P2CSC555F15
@cs55_test
LOCATION: oval
NOISE: 1
#id_ssharm_21 #P2CSC555F15
@yjmehta
Call from: Han
URGENCY: 0
#id_yjmehta_15 #P2CSC555F15
@yjmehta
Call from: Han
```

The screenshot shows the Spyder Python IDE interface. The editor window displays a Python script for a Twitter bot. The script includes functions for handling status updates and sending tweets. The console window shows the output of the script, including the bot's name, location, and the tweets it has sent.

```
67 substrindex = []
68 for i in range(0, len(status.text)-2):
69     if status.text[i] == '#' and count < 2:
70         substrindex.append(i)
71         count = count + 1
72     responseid = status.text[substrindex[0]:(substrindex[1]-1)]
73     rand = randint(0, (len(response_list)-1))
74     response_tweet = '@'+screenname+'\nRESPONSE: '+response_list[rand]+' \n'+r
75     #print "MY RESPONSE TWEET IN ACTION"
76     #print response_tweet
77     api.update_status(status=response_tweet)
78
79 elif '@sharaths1993' in status.text:
80     with open('callresponsestweets.txt', 'a') as tf:
81         tf.write(status.text)
82         tf.write('\n')
83         count = 0
84         substrindex = []
85         for i in range(0, len(status.text)-2):
86             if status.text[i] == '#' and count < 2:
87                 substrindex.append(i)
88                 count = count + 1
89             responseid = status.text[substrindex[0]:(substrindex[1]-1)]
90             rand = randint(0, 1)
91             response_tweet = "ACTION: "+choice_list[rand]+" "+responseid+" #P2CSC555F15
92             #print "MY RESPONSE TWEET URGENCY @sharaths1993"
93             #print response_tweet
94             return True
95
96 def on_error(self, status_code):
97     if status_code == 420:
98         #returning False in on_data disconnects the stream
99         return False
100
101 ACCESS_TOKEN = "4147080203-REc5r6WMTGT7WUPdBrigixaph0c1TLWl0dM3fg"
102 ACCESS_SECRET = "MEG0xvJmpeC08p013Yxv0813Ac5VdVH9RKiWdA1W7p"
103 CONSUMER_KEY = "CDmg07RUVvSgnQ4epKZ0BEtqm"
104 CONSUMER_SECRET = "3cm83lntdlSL7sLnZ7SHU7ZEcT5h4RNF1S8m16T5Kxi7anH"
```

Console Output:

```
#id_sshekha3_1 #P2CSC555F15
ACTION: Yes #id_sshekha3_1 #P2CSC555F15.
ACTION: Yes #id_sshekha3_1 #P2CSC555F15.
MY RESPONSE TWEET IN ACTION
@srxvid
RESPONSE: Negative
#id_sshekha3_1 #P2CSC555F15
@srxvid
RESPONSE: Positive
#id_sshekha3_1 #P2CSC555F15
@srxvid
RESPONSE: Positive
#id_sshekha3_1 #P2CSC555F15
I checked in at #hunt #id_sshekha3_2 #P2CSC555F15.
I checked in at #hunt #id_sshekha3_2 #P2CSC555F15.
MY RESPONSE TWEET CHECKED
@srxvid
Name: sharaths1993
MY_MODE: Loud
EXPECTED_MODE: Loud
#id_sshekha3_2 #P2CSC555F15
I checked in at #hunt #id_cpancha_3 #P2CSC555F15
I checked in at #hunt #id_cpancha_3 #P2CSC555F15
```

The screenshot shows the Spyder Python IDE interface. The editor window displays a Python script for a Twitter bot. The script includes a class for handling status updates and sending tweets. The console window shows the output of the script, including the bot's name, location, and the tweets it has sent.

```
21 class MyStreamListener(tweepy.StreamListener):
22
23     def on_status(self, status):
24         screenname = status.user.screen_name
25
26         if "checked" in status.text:
27             with open('checkedtweets.txt', 'a') as tf:
28                 tf.write(status.text)
29                 tf.write('\n')
30                 count = 0
31                 substrindex = []
32                 for i in range(0, len(status.text)-2):
33                     if status.text[i] == '#' and count < 3:
34                         substrindex.append(i)
35                         count = count + 1
36                 location = status.text[substrindex[0]:(substrindex[1]-1)]
37                 if location == mylocation:
38                     responseid = status.text[substrindex[1]+1:(substrindex[2]-1)]
39                     response_tweet = '@'+screenname+'\nName: '+myname+'\nMY_MODE: '+my_mc
40                     #print "MY RESPONSE TWEET CHECKED"
41                     api.update_status(status=response_tweet)
42                     #print response_tweet
43
44         elif "CALL" in status.text:
45             with open('calltweets.txt', 'a') as tf:
46                 tf.write(status.text)
47                 tf.write('\n')
48                 count = 0
49                 substrindex = []
50                 for i in range(0, len(status.text)-2):
51                     if status.text[i] == '#' and count < 2:
52                         substrindex.append(i)
53                         count = count + 1
54                 responseid = status.text[substrindex[0]:(substrindex[1]-1)]
55                 rand = randint(0, (len(call_list)-1))
56                 rand2 = randint(0, (len(urgency_list)-1))
57                 response_tweet = '@'+screenname+'\ncall from: '+call_list[rand]+' \nURGENC
58                 #print "MY RESPONSE TWEET IN CALL"
```

Console Output:

```
EXPECTED_MODE: Loud
#id_sshekha3_2 #P2CSC555F15
@srxvid
Name: RSJogaikar
MY_MODE: silent
EXPECTED_MODE: silent
#id_sshekha3_2 #P2CSC555F15.
@srxvid
Name: RSJogaikar
MY_MODE: silent
EXPECTED_MODE: silent
#id_sshekha3_2 #P2CSC555F15.
@sandhyannayak
RESPONSE: Positive
#id_snnayak_98 #P2CSC555F15
@sandhyannayak
RESPONSE: Positive
#id_snnayak_98 #P2CSC555F15
I checked in at #hunt #id_sshekha3_1 #P2CSC555F15.
I checked in at #hunt #id_sshekha3_1 #P2CSC555F15.
MY RESPONSE TWEET CHECKED
@srxvid
Name: sharaths1993
```