Understanding the data

In [ ]:

```
import numpy as np
import pandas as pd
import matplotlib.pyplot as plt
import seaborn as sns
%pylab inline
import copy
from googletrans import Translator
import emoji
import re
```

In [3]:

```
#Names =
['u_id','m_id','time','forward_count','comment_count','like_count','content']
Names = ['u_id','m_id','time','content']

train_dataset= pd.read_fwf("G:\\weibo_train_data.txt",header=None,names=Names ,encodi
ng='utf-8',delimiter="\t")

# fixed width formatted lines.
```

Observing the dataset

In [3]:

```
train_dataset.head(10)
```

Out[3]:

	u_id	m_id	time
0	d38e9bed5d98110dc2489d0d1cac3c2a	7d45833d9865727a88b960b0603c19f6	2015-02-23 17:41:29
1	fa13974743d3fe6ff40d21b872325e9e	8169f1d45051e08ef213bf1106b1225d	2015-02-14 12:49:58
2	da534fe87e7a52777bee5c30573ed5fd	68cd0258c31c2c525f94febea2d9523b	2015-03-31 13:58:06

	u_id	m_id	time
3	e06a22b7e065e559a1f0bf7841a85c51	00b9f86b4915aedb7db943c54fd19d59	2015-06-11 20:39:57
4	f9828598f9664d4e347ef2048ce17734	c7f6f66044c0c5a3330e2c5371be6824	2015-03-10 18:02:38
5	d80f3d3c5c1d658e82b837a4dd1af849	bfc0819b83ec59ce767287077f2b3507	2015-02-13 01:09:41
6	f349a67d1cd7c8683c5bbc5f8486e193	83674a60e5310195fc35d97ea8f45c46	2015-07-15 01:16:24
7	24b621c98f2594b698c0b1d60c9ae6db	2cbd3d514ed5ad3dab81aa043c8b3d0a	2015-05-19 10:24:57
8	e44d81d630e4f382f657e72aa4b685da	8a88a25f9f26ed9f79080eaacc1a8668	2015-02-11 11:03:36
9	fbe6c953632e1b3dda66cf6118b6ab12	f359a74cb4ac6150a3af8325eda04ea0	2015-03-22 00:54:34

In [4]:

```
print("Predict_Dataset has "+str(train_dataset.shape[0])+" records")
```

```
Predict_Dataset has 1229618 records
```

In [6]:

```
print("Predict_Dataset has "+str(train_dataset.shape[1])+" attributes")
```

```
Predict_Dataset has 4 attributes
```

In [6]:

```
train_content_mix=train_dataset['content']
```

In [7]:

```
train_content_split = pd.DataFrame(train_content_mix.str.split('\t',expand=True))
print(train_content_split.head(5))
```

```
   0  1  2                                                  3
0  0  0  0  丽江旅游(sz002033)#股票##炒股##财经##理财##投资#推荐包赢股，盈利对半分成...
1  0  0  0  #丁辰灵的红包#挣钱是一种能力，抢红包拼的是技术。我抢到了丁辰灵 和@阚洪岩 一起发出的现金...
2  0  0  0                                淘宝网这些傻逼。。。气的劳资有火没地儿发~尼玛，你们都瞎了
3  0  4  3                                  看点不能说的，你们都懂[笑cry]
4  0  0  0                                             111多张
```

In [8]:

```
#frames = [train_data,train_content_split]
#esult = pd.concat(frames)
#df['uid']=train_data['u_id']

train_dataset2 = pd.concat([train_dataset,train_content_split], axis=1)
print(train_dataset2.head(5))
```

```
                              u_id                                m_id  \
0  d38e9bed5d98110dc2489d0d1cac3c2a  7d45833d9865727a88b960b0603c19f6
1  fa13974743d3fe6ff40d21b872325e9e  8169f1d45051e08ef213bf1106b1225d
2  da534fe87e7a52777bee5c30573ed5fd  68cd0258c31c2c525f94febea2d9523b
3  e06a22b7e065e559a1f0bf7841a85c51  00b9f86b4915aedb7db943c54fd19d59
4  f9828598f9664d4e347ef2048ce17734  c7f6f66044c0c5a3330e2c5371be6824

                  time                                            content  0  \
0  2015-02-23 17:41:29  0\t0\t0\t丽江旅游(sz002033)#股票##炒股##财经##理财##投资#推荐包...
0
1  2015-02-14 12:49:58  0\t0\t0\t#丁辰灵的红包#挣钱是一种能力，抢红包拼的是技术。我抢到了丁辰灵和@阚洪...  0
2  2015-03-31 13:58:06            0\t0\t0\t淘宝网这些傻逼。。。气的劳资有火没地儿发~尼玛，你们都瞎了  0
3  2015-06-11 20:39:57                          0\t4\t3\t看点不能说的，你们都懂[笑cry]  0

4  2015-03-10 18:02:38                                   0\t0\t0\t111多张  0

   1  2                                                  3
0  0  0  丽江旅游(sz002033)#股票##炒股##财经##理财##投资#推荐包赢股，盈利对半分成...
1  0  0  #丁辰灵的红包#挣钱是一种能力，抢红包拼的是技术。我抢到了丁辰灵 和@阚洪岩 一起发出的现金...

2  0  0                                淘宝网这些傻逼。。。气的劳资有火没地儿发~尼玛，你们都瞎了
3  4  3                                  看点不能说的，你们都懂[笑cry]
4  0  0                                             111多张
```

In [9]:

```
del train_dataset2['content']
train_dataset2.rename(columns={0:'forward_count'},inplace=True)
train_dataset2.rename(columns={1:'comment_count'},inplace=True)
```

```
train_dataset2.rename(columns={2:'like_count'},inplace=True)
train_dataset2.rename(columns={3:'content'},inplace=True)
train_dataset2.head(5)
```

Out[9]:

	u_id	m_id	time	f
0	d38e9bed5d98110dc2489d0d1cac3c2a	7d45833d9865727a88b960b0603c19f6	2015-02-23 17:41:29	0
1	fa13974743d3fe6ff40d21b872325e9e	8169f1d45051e08ef213bf1106b1225d	2015-02-14 12:49:58	0
2	da534fe87e7a52777bee5c30573ed5fd	68cd0258c31c2c525f94febea2d9523b	2015-03-31 13:58:06	0
3	e06a22b7e065e559a1f0bf7841a85c51	00b9f86b4915aedb7db943c54fd19d59	2015-06-11 20:39:57	0
4	f9828598f9664d4e347ef2048ce17734	c7f6f66044c0c5a3330e2c5371be6824	2015-03-10 18:02:38	0

In [17]:

```
translate_dataframe = pd.DataFrame(data=train_dataset2['content'].head(10))
translator = Translator()
translate_dataframe["English_content"] = translate_dataframe['content'].map(lambda x
: translator.translate(x, src="zh-CN", dest="en").text)
```

content

```
                                                content
0  丽江旅游(sz002033)#股票##炒股##财经##理财##投资#推荐包赢股，盈利对半分成...
1  #丁辰灵的红包#挣钱是一种能力，抢红包拼的是技术。我抢到了丁辰灵 和@阚洪岩 一起发出的现金...
2                          淘宝网这些傻逼。。。气的劳资有火没地儿发~尼玛，你们都瞎了
3                                  看点不能说的，你们都懂[笑cry]
4                                                111多张
```

In [18]:

```
print(translate_dataframe)
```

```
                                                content  \
0  丽江旅游(sz002033)#股票##炒股##财经##理财##投资#推荐包赢股，盈利对半分成...
1  #丁辰灵的红包#挣钱是一种能力，抢红包拼的是技术。我抢到了丁辰灵 和@阚洪岩 一起发出的现金...
2                          淘宝网这些傻逼。。。气的劳资有火没地儿发~尼玛，你们都瞎了
3                                  看点不能说的，你们都懂[笑cry]
4                                                111多张
5  有生之年！我最喜欢的up主跟我的三体勾搭到一起了！幸福感爆棚！ @黑桐谷歌  http://...
6          论优衣库试衣间隔音效果好坏？ http://t.cn/RL5aSzp（分享自 @知乎）
7  如此平凡的日常一幕，还能够再积累多少呢。 终有一天，当我们到了看着这张照片能感受到一阵怀念的...
8  #罗永浩的红包#二十三，糖瓜儿粘，抢个红包乐翻天！我抢到了罗永浩 和@_王先森就是我 一起发...
9  有好东西分享给你！闪记笔记记事，最好用的中文待办软件，还等什么？快去下载： http://t...

                                     English_content
0  Lijiang Tourism (sz002033)#Stock##炒股##财经##理财##...
1  #丁辰灵的红包# Earning money is a kind of ability. I...
2  Taobao.com is stupid. . . The labor of the gas...
3  You can't say anything, you all know [laughing...
4                                    More than 111
5  For a lifetime! My favorite up master is with ...
6  On the effect of UNIQLO's fitting interval sou...
7  How much more can you accumulate in such an or...
8  #罗永浩的红包# Twenty-three, sugar melons sticky, gr...
9  Have something to share with you! Flash note n...
```

In [19]:

```
train_dataset2.head(30)
```

Out[19]:

	u_id	m_id	time
0	d38e9bed5d98110dc2489d0d1cac3c2a	7d45833d9865727a88b960b0603c19f6	2015-02-23 17:41:29
1	fa13974743d3fe6ff40d21b872325e9e	8169f1d45051e08ef213bf1106b1225d	2015-02-14 12:49:58
			2015-

	u_id	m_id	time
2	da534fe87e7a52777bee5c30573ed5fd	68cd0258c31c2c525f94febea2d9523b	03-31 13:58:06
3	e06a22b7e065e559a1f0bf7841a85c51	00b9f86b4915aedb7db943c54fd19d59	2015-06-11 20:39:57
4	f9828598f9664d4e347ef2048ce17734	c7f6f66044c0c5a3330e2c5371be6824	2015-03-10 18:02:38
5	d80f3d3c5c1d658e82b837a4dd1af849	bfc0819b83ec59ce767287077f2b3507	2015-02-13 01:09:41
6	f349a67d1cd7c8683c5bbc5f8486e193	83674a60e5310195fc35d97ea8f45c46	2015-07-15 01:16:24
7	24b621c98f2594b698c0b1d60c9ae6db	2cbd3d514ed5ad3dab81aa043c8b3d0a	2015-05-19 10:24:57
8	e44d81d630e4f382f657e72aa4b685da	8a88a25f9f26ed9f79080eaacc1a8668	2015-02-11 11:03:36
9	fbe6c953632e1b3dda66cf6118b6ab12	f359a74cb4ac6150a3af8325eda04ea0	2015-03-22 00:54:34
10	f9a3ca6bc1e75d173cfc98ec4b108072	c7bc3445e8b90db8cc5e045f606dc1ee	2015-02-11 19:29:04
11	3c68bbb9da57fcc752c8a493d91bdd3a	77e14cf9d460715e84c51747c3641a9b	2015-04-28 00:14:05
12	104e8d55e98eb3cd834810088af039fe	ee0b2c9d35bfeb0fbc5b3a8677f4a18c	2015-02-14 23:42:23
13	0d15005d6397fb5ce1d45e7c834f7370	9c954d63fcfea19dca8d81a4f3b53861	2015-06-19 14:35:03
			2015-

	u_id	m_id	time
14	875a4a77b339d93f819e2c4de5bd0b57	f2cdcdbcec9ff47cbb3c6a636e4b92a3	2015-07-01 04:11:48
15	380a2219670f50dc87efce3380bea6e8	46f10244d02afa85d12346ce28e3cec5	2015-03-11 08:00:24
16	b9b88b0fc105fb08a552e782afa4342e	cb907eb1bdbc198ed0944cc3b7e24f91	2015-05-04 22:10:22
17	f18eb14365c0d7248fab1b9c464f4e70	096543bd8746869982d1a7557164dd0d	2015-02-18 21:37:17
18	0fc17bf5e2dc789dd48505df1f5b14fd	4c1e2418127811d212d0e3867a99db3e	2015-07-13 05:07:28
19	dd749a5af07c04ce7de451273a983671	419dd71d562883ef836e774bc3f4e163	2015-07-30 14:24:28
20	a984551b159fcdc0a48f9e38ecb1488f	baa0051d359555601ab61df684787f0f	2015-02-03 20:09:49
21	2e0467b73d0f6f9e5607a6174581fdd8	2fd200a7f670138c2026091c3b01532a	2015-04-15 15:49:11
22	819656f05994b00b7260daf7346586a7	95590e88cac5d8c9d1a496bc3bd42f07	2015-05-27 14:50:18
23	91ce7c63b272f2037a3e702c10163fa3	8b4e85a881afaff91f276eac7bfb6604	2015-02-13 18:48:37
24	4680e73f9e7a6b87dec62a86a7821c17	b2db095af290b3a36cf798a3e17528d8	2015-03-12 15:19:54
25	976e85e3ededdd9b2c2a3179eb7ae8ab	9540ee0cf7ccfae523020c8025e7095f	2015-03-21 22:04:55

	u_id	m_id	time
26	6623347e5f19f35f2d02ad515b96524c	9a2f48a870843d1964a03c6642b309d5	2015-07-21 01:06:53
27	cf727e70b6661387cf6aadf01d2eb32c	bff281350f035db0e84c25394865d86a	2015-02-19 06:02:06
28	de0836c1c5d40a5cae64a964a0b54894	c3345fd72cad53ca9bffd63634170ba0	2015-04-20 22:36:23
29	c8848f18da5911d0389c3ac70fe13204	fa352495e646a3f7ff979267c490fd89	2015-06-11 23:46:08

In [ ]:

```
emoji_pattern=re.compile("["
                         u"\U0001F600-\U0001F64F"
                         u"\U0001F300-\U0001F5FF"
                         u"\U0001F680-\U0001F6FF"
                         u"\U0001F1E0-\U0001F1FF"
                         "]+",flags=re.UNICODE)
train_dataset2['content']=emoji_pattern.sub(r'',str(train_dataset2['content']))
```

In [1]:

```
translate_dataframe = pd.DataFrame(data=train_dataset2['content'].head(30))
translator = Translator()
translate_dataframe["English_content"] = translate_dataframe['content'].map(lambda x
: translator.translate(x, src="zh-CN", dest="en").text)
```

In [19]:

```
train_dataset2.forward_count.describe()
```

Out[19]:

```
count     1229618
unique       1243
top             0
freq      1009457
Name: forward_count, dtype: object
```

In [14]:

```
train_dataset2.comment_count.describe()
```

Out[14]:

```
count     1229618
unique        527
top             0
freq       975602
Name: comment_count, dtype: object
```

In [15]:

```
train_dataset2.like_count.describe()
```

Out[15]:

```
count     1229618
unique       1020
top             0
freq       920818
Name: like_count, dtype: object
```

In [10]:

```
train_dataset3=pd.DataFrame(train_dataset2.time.str.split(' ',1).tolist(),columns=['date','new_time'])
```

In [9]:

```
train_dataset3.head()
```

Out[9]:

	date	new_time
0	2015-02-23	17:41:29
1	2015-02-14	12:49:58
2	2015-03-31	13:58:06
3	2015-06-11	20:39:57
4	2015-03-10	18:02:38

In [11]:

```
train_dataset4 = pd.concat([train_dataset2,train_dataset3], axis=1)
del train_dataset4['time']
del train_dataset4['content']
train_dataset4.rename(columns={0:'forward_count'},inplace=True)
train_dataset4.rename(columns={1:'comment_count'},inplace=True)
train_dataset4.rename(columns={2:'like_count'},inplace=True)
train_dataset4.rename(columns={3:'content'},inplace=True)
train_dataset4.rename(columns={'new_time':'time'},inplace=True)
train_dataset4.head(5)
```

Out[11]:

	u_id	m_id	forward_c
0	d38e9bed5d98110dc2489d0d1cac3c2a	7d45833d9865727a88b960b0603c19f6	0
1	fa13974743d3fe6ff40d21b872325e9e	8169f1d45051e08ef213bf1106b1225d	0
2	da534fe87e7a52777bee5c30573ed5fd	68cd0258c31c2c525f94febea2d9523b	0
3	e06a22b7e065e559a1f0bf7841a85c51	00b9f86b4915aedb7db943c54fd19d59	0
4	f9828598f9664d4e347ef2048ce17734	c7f6f66044c0c5a3330e2c5371be6824	0

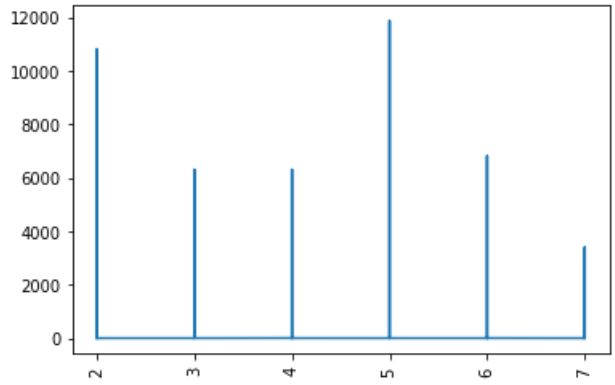
Observing trend of number of likes with respect to each month

In [15]:

```
#Month vs Like_Count
train_dataset5=train_dataset4.sort_values('date',ascending=True)
train_dataset5['like_count']=train_dataset5['like_count'].astype(float)
train_dataset5['month']=pd.DatetimeIndex(train_dataset5['date']).month
plt.plot(train_dataset5['month'],train_dataset5['like_count'])
plt.xticks(rotation='vertical')
```

Out[15]:

```
(array([1., 2., 3., 4., 5., 6., 7., 8.]),
 <a list of 8 Text xticklabel objects>)
```



Observing trend of number of forwards with respect to each month

-

In [ ]:

```
#Month vs Forward_Count
train_dataset5=train_dataset4.sort_values('date',ascending=True)
train_dataset5['forward_count']=train_dataset5['forward_count'].astype(float)
train_dataset5['month']=pd.DatetimeIndex(train_dataset5['date']).month
plt.bar(train_dataset5['month'],train_dataset5['forward_count'])
#plt.xticks(rotation='vertical')
```

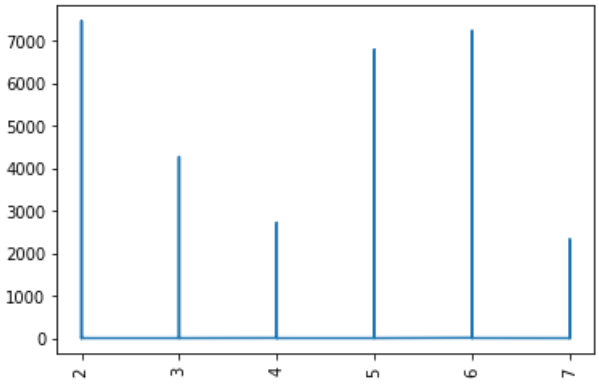
Observing trend of number of comments with respect to each month

In [52]:

```
#Month vs Comment_Count
train_dataset5=train_dataset4.sort_values('date',ascending=True)
train_dataset5['comment_count']=train_dataset5['comment_count'].astype(float)
train_dataset5['month']=pd.DatetimeIndex(train_dataset5['date']).month
plt.plot(train_dataset5['month'],train_dataset5['comment_count'])
plt.xticks(rotation='vertical')
```

Out[52]:

```
(array([1., 2., 3., 4., 5., 6., 7., 8.]),
 <a list of 8 Text xticklabel objects>)
```

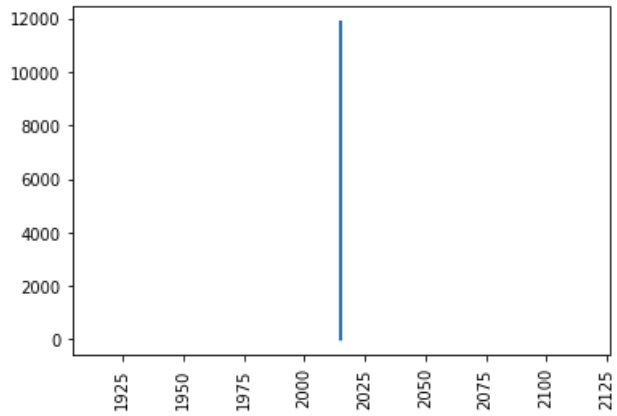


In [53]:

```
#Year vs Like_Count
train_dataset5=train_dataset4.sort_values('date',ascending=True)
train_dataset5['like_count']=train_dataset5['like_count'].astype(float)
train_dataset5['year']=pd.DatetimeIndex(train_dataset5['date']).year
plt.plot(train_dataset5['year'],train_dataset5['like_count'])
plt.xticks(rotation='vertical')
```

Out[53]:

```
(array([1900., 1925., 1950., 1975., 2000., 2025., 2050., 2075., 2100.,
        2125., 2150.]), <a list of 11 Text xticklabel objects>)
```



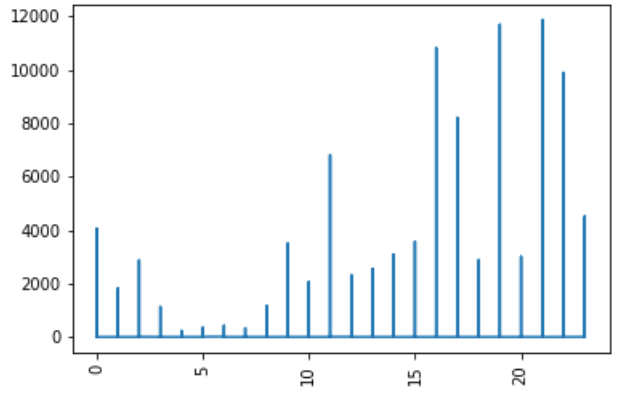
Observing trend of number of likes on hourly basis

In [16]:

```
#Hour vs Like_Count
train_dataset5=train_dataset4.sort_values('time',ascending=True)
train_dataset5['like_count']=train_dataset5['like_count'].astype(float)
train_dataset5['hour']=pd.DatetimeIndex(train_dataset5['time']).hour
plt.plot(train_dataset5['hour'],train_dataset5['like_count'])
plt.xticks(rotation='vertical')
```

Out[16]:

```
(array([-5.,  0.,  5., 10., 15., 20., 25.]),
 <a list of 7 Text xticklabel objects>)
```



In [ ]:

```
#Hour vs forward_Count
train_dataset5=train_dataset4.sort_values('time',ascending=True)
train_dataset5['forward_count']=train_dataset5['forward_count'].astype(float)
train_dataset5['hour']=pd.DatetimeIndex(train_dataset5['time']).hour
plt.bar(train_dataset5['hour'],train_dataset5['forward_count'])
plt.xticks(rotation='vertical')
```

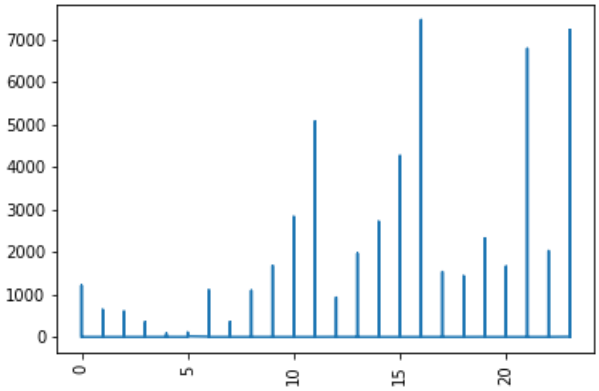
In [18]:

```
#Hour vs comment_Count
```

```
train_dataset5=train_dataset4.sort_values('time',ascending=True)
train_dataset5['comment_count']=train_dataset5['comment_count'].astype(float)
train_dataset5['hour']=pd.DatetimeIndex(train_dataset5['time']).hour
plt.plot(train_dataset5['hour'],train_dataset5['comment_count'])
plt.xticks(rotation='vertical')
```

Out[18]:

```
(array([-5.,  0.,  5., 10., 15., 20., 25.]),
 <a list of 7 Text xticklabel objects>)
```



DMA course project review 2

Preprocessing of data

1.
   - Team ID - 5A09
   - Sem - 5TH
   - Div - 'A'
   - School - KLE Technological university

1.
   - Topic ID - 5ADMACP14
   - Project Title - Sina Weibo Intereaction Prediction

1. Problem Statement - To predict the user behaviors such as forwarding, commenting and liking.

1.
   - Team Leader - Deepti Nadkarni - 01FE16BCS062 (Roll no-58)
   - Members
     - Apoorva Malemath - 01FE16BCS041 (Roll no-39)
     - Arundati Dixit - 01FE16BCS046 (Roll no-44)
     - Ashish Kar - 01FE16BCS047 (Roll no-45)

In [1]:

```
import numpy as np
import pandas as pd
import matplotlib.pyplot as plt
import seaborn as sns
%pylab inline
import copy
from googletrans import Translator
import pandas as pd
import numpy as np
import csv
import re
import jieba
import time
import json
from sklearn.feature_extraction.text import CountVectorizer
from sklearn import linear_model
from sklearn.externals import joblib
from nltk.corpus import stopwords as e_stopwords
from datetime import datetime, timedelta
import jieba
import sys

from nltk.corpus import stopwords
from nltk.tokenize import word_tokenize
```

```
Populating the interactive namespace from numpy and matplotlib
```

DATA PREPROCESSING

- Raw data is not directly adequate for analysis.
- Training data
  - uid
  - mid
  - time
  - forward_count
  - comment_count
  - like_count
  - content
- Predicting data
  - uid
  - mid
  - time
  - content
- Previous Observations made
  - Training data has 12,296,18 tuples.
  - Predicting data has 43,845 tuples.
  - Significant occurance of the value zero.
- Translation

In [3]:

```
train1= pd.read_csv("E:\\5th Sem\\DMA Project\\DMA Project Sina Weibo\\CSV\\weibo_train1.csv")
```

In [3]:

```
train2= pd.read_csv("E:\\5th Sem\\DMA Project\\DMA Project Sina Weibo\\CSV\\weibo_train2.csv")
```

In [4]:

```
frames=[train1,train2]
train=pd.concat(frames)
```

In [5]:

```
train.shape
```

Out[5]:

```
(1229618, 11)
```

In [6]:

```
train.head(5)
```

Out[6]:

	u_id	m_id	forward_c
0	d38e9bed5d98110dc2489d0d1cac3c2a	7d45833d9865727a88b960b0603c19f6	0
1	fa13974743d3fe6ff40d21b872325e9e	8169f1d45051e08ef213bf1106b1225d	0
2	da534fe87e7a52777bee5c30573ed5fd	68cd0258c31c2c525f94febea2d9523b	0
3	e06a22b7e065e559a1f0bf7841a85c51	00b9f86b4915aedb7db943c54fd19d59	0
4	f9828598f9664d4e347ef2048ce17734	c7f6f66044c0c5a3330e2c5371be6824	0

In [5]:

```
tc=np.array_split(train,400)
```

In [7]:

```
i=0
for i in range(400):
    ith=str(i)
    f="G:\\concatfiles\\f"+ith+".txt"
    tc[i].to_csv(f,sep=',',index=False,encoding='utf-8')
```

Translation

- u_id
- m_id
- forward_count
- comment_count
- like_count
- content
- date
- time
- content_media_count
- content_spchar
- non_emoji_content
- en_content

In [8]:

```
translated=pd.DataFrame(columns=list(['u_id', 'm_id', 'forward_count', 'comment_count
', 'like_count',
       'content', 'date', 'time', 'content_media_count', 'content_spchar',
       'non_emoji_content', 'en_content', 'Unnamed: 1']))
```

Translation has been performed on the content column seperately by considering it as a seperate file, thus the files are concatinaated

In [9]:

```
for j in range(0,218):
    filename="G:\\concatfiles\\f"+str(j)+".txt"
    transname="G:\\translated\\ts"+str(j)+".zh-CN.en.txt"
    print(filename)
    print(transname)
    f=pd.read_csv(filename)
    t= pd.read_csv(transname,sep="5A09")
    frames = [f,t]
#result = pd.concat(frames, ignore_index=False)
    #df="df"+str(j)
    df=(pd.concat(frames, join='outer', ignore_index=False,keys=None, levels=None, na
mes=None, verify_integrity=False, copy=True, axis=1))
    translated=translated.append(df)
    #translated.append(df,ignore_index=True)
```

```
G:\concatfiles\f0.txt
G:\translated\ts0.zh-CN.en.txt
G:\concatfiles\f1.txt
G:\translated\ts1.zh-CN.en.txt
```

```
C:\Users\DELL\Anaconda3\lib\site-packages\ipykernel_launcher.py:7: ParserWarning: Fal
```

```
ling back to the 'python' engine because the 'c' engine does not support regex separa
tors (separators > 1 char and different from '\s+' are interpreted as regex); you can
avoid this warning by specifying engine='python'.
  import sys
```

```
G:\concatfiles\f2.txt
G:\translated\ts2.zh-CN.en.txt
G:\concatfiles\f3.txt
G:\translated\ts3.zh-CN.en.txt
G:\concatfiles\f4.txt
G:\translated\ts4.zh-CN.en.txt
G:\concatfiles\f5.txt
G:\translated\ts5.zh-CN.en.txt
G:\concatfiles\f6.txt
G:\translated\ts6.zh-CN.en.txt
G:\concatfiles\f7.txt
G:\translated\ts7.zh-CN.en.txt
G:\concatfiles\f8.txt
G:\translated\ts8.zh-CN.en.txt
G:\concatfiles\f9.txt
G:\translated\ts9.zh-CN.en.txt
G:\concatfiles\f10.txt
G:\translated\ts10.zh-CN.en.txt
G:\concatfiles\f11.txt
G:\translated\ts11.zh-CN.en.txt
G:\concatfiles\f12.txt
G:\translated\ts12.zh-CN.en.txt
G:\concatfiles\f13.txt
G:\translated\ts13.zh-CN.en.txt
G:\concatfiles\f14.txt
G:\translated\ts14.zh-CN.en.txt
G:\concatfiles\f15.txt
G:\translated\ts15.zh-CN.en.txt
G:\concatfiles\f16.txt
G:\translated\ts16.zh-CN.en.txt
G:\concatfiles\f17.txt
G:\translated\ts17.zh-CN.en.txt
G:\concatfiles\f18.txt
G:\translated\ts18.zh-CN.en.txt
G:\concatfiles\f19.txt
G:\translated\ts19.zh-CN.en.txt
G:\concatfiles\f20.txt
G:\translated\ts20.zh-CN.en.txt
G:\concatfiles\f21.txt
G:\translated\ts21.zh-CN.en.txt
G:\concatfiles\f22.txt
G:\translated\ts22.zh-CN.en.txt
G:\concatfiles\f23.txt
G:\translated\ts23.zh-CN.en.txt
G:\concatfiles\f24.txt
G:\translated\ts24.zh-CN.en.txt
G:\concatfiles\f25.txt
G:\translated\ts25.zh-CN.en.txt
```

G:\concatfiles\f26.txt
G:\translated\ts26.zh-CN.en.txt
G:\concatfiles\f27.txt
G:\translated\ts27.zh-CN.en.txt
G:\concatfiles\f28.txt
G:\translated\ts28.zh-CN.en.txt
G:\concatfiles\f29.txt
G:\translated\ts29.zh-CN.en.txt
G:\concatfiles\f30.txt
G:\translated\ts30.zh-CN.en.txt
G:\concatfiles\f31.txt
G:\translated\ts31.zh-CN.en.txt
G:\concatfiles\f32.txt
G:\translated\ts32.zh-CN.en.txt
G:\concatfiles\f33.txt
G:\translated\ts33.zh-CN.en.txt
G:\concatfiles\f34.txt
G:\translated\ts34.zh-CN.en.txt
G:\concatfiles\f35.txt
G:\translated\ts35.zh-CN.en.txt
G:\concatfiles\f36.txt
G:\translated\ts36.zh-CN.en.txt
G:\concatfiles\f37.txt
G:\translated\ts37.zh-CN.en.txt
G:\concatfiles\f38.txt
G:\translated\ts38.zh-CN.en.txt
G:\concatfiles\f39.txt
G:\translated\ts39.zh-CN.en.txt
G:\concatfiles\f40.txt
G:\translated\ts40.zh-CN.en.txt
G:\concatfiles\f41.txt
G:\translated\ts41.zh-CN.en.txt
G:\concatfiles\f42.txt
G:\translated\ts42.zh-CN.en.txt
G:\concatfiles\f43.txt
G:\translated\ts43.zh-CN.en.txt
G:\concatfiles\f44.txt
G:\translated\ts44.zh-CN.en.txt
G:\concatfiles\f45.txt
G:\translated\ts45.zh-CN.en.txt
G:\concatfiles\f46.txt
G:\translated\ts46.zh-CN.en.txt
G:\concatfiles\f47.txt
G:\translated\ts47.zh-CN.en.txt
G:\concatfiles\f48.txt
G:\translated\ts48.zh-CN.en.txt
G:\concatfiles\f49.txt
G:\translated\ts49.zh-CN.en.txt
G:\concatfiles\f50.txt
G:\translated\ts50.zh-CN.en.txt
G:\concatfiles\f51.txt
G:\translated\ts51.zh-CN.en.txt

G:\concatfiles\f52.txt
G:\translated\ts52.zh-CN.en.txt
G:\concatfiles\f53.txt
G:\translated\ts53.zh-CN.en.txt
G:\concatfiles\f54.txt
G:\translated\ts54.zh-CN.en.txt
G:\concatfiles\f55.txt
G:\translated\ts55.zh-CN.en.txt
G:\concatfiles\f56.txt
G:\translated\ts56.zh-CN.en.txt
G:\concatfiles\f57.txt
G:\translated\ts57.zh-CN.en.txt
G:\concatfiles\f58.txt
G:\translated\ts58.zh-CN.en.txt
G:\concatfiles\f59.txt
G:\translated\ts59.zh-CN.en.txt
G:\concatfiles\f60.txt
G:\translated\ts60.zh-CN.en.txt
G:\concatfiles\f61.txt
G:\translated\ts61.zh-CN.en.txt
G:\concatfiles\f62.txt
G:\translated\ts62.zh-CN.en.txt
G:\concatfiles\f63.txt
G:\translated\ts63.zh-CN.en.txt
G:\concatfiles\f64.txt
G:\translated\ts64.zh-CN.en.txt
G:\concatfiles\f65.txt
G:\translated\ts65.zh-CN.en.txt
G:\concatfiles\f66.txt
G:\translated\ts66.zh-CN.en.txt
G:\concatfiles\f67.txt
G:\translated\ts67.zh-CN.en.txt
G:\concatfiles\f68.txt
G:\translated\ts68.zh-CN.en.txt
G:\concatfiles\f69.txt
G:\translated\ts69.zh-CN.en.txt
G:\concatfiles\f70.txt
G:\translated\ts70.zh-CN.en.txt
G:\concatfiles\f71.txt
G:\translated\ts71.zh-CN.en.txt
G:\concatfiles\f72.txt
G:\translated\ts72.zh-CN.en.txt
G:\concatfiles\f73.txt
G:\translated\ts73.zh-CN.en.txt
G:\concatfiles\f74.txt
G:\translated\ts74.zh-CN.en.txt
G:\concatfiles\f75.txt
G:\translated\ts75.zh-CN.en.txt
G:\concatfiles\f76.txt
G:\translated\ts76.zh-CN.en.txt
G:\concatfiles\f77.txt
G:\translated\ts77.zh-CN.en.txt

G:\concatfiles\f78.txt
G:\translated\ts78.zh-CN.en.txt
G:\concatfiles\f79.txt
G:\translated\ts79.zh-CN.en.txt
G:\concatfiles\f80.txt
G:\translated\ts80.zh-CN.en.txt
G:\concatfiles\f81.txt
G:\translated\ts81.zh-CN.en.txt
G:\concatfiles\f82.txt
G:\translated\ts82.zh-CN.en.txt
G:\concatfiles\f83.txt
G:\translated\ts83.zh-CN.en.txt
G:\concatfiles\f84.txt
G:\translated\ts84.zh-CN.en.txt
G:\concatfiles\f85.txt
G:\translated\ts85.zh-CN.en.txt
G:\concatfiles\f86.txt
G:\translated\ts86.zh-CN.en.txt
G:\concatfiles\f87.txt
G:\translated\ts87.zh-CN.en.txt
G:\concatfiles\f88.txt
G:\translated\ts88.zh-CN.en.txt
G:\concatfiles\f89.txt
G:\translated\ts89.zh-CN.en.txt
G:\concatfiles\f90.txt
G:\translated\ts90.zh-CN.en.txt
G:\concatfiles\f91.txt
G:\translated\ts91.zh-CN.en.txt
G:\concatfiles\f92.txt
G:\translated\ts92.zh-CN.en.txt
G:\concatfiles\f93.txt
G:\translated\ts93.zh-CN.en.txt
G:\concatfiles\f94.txt
G:\translated\ts94.zh-CN.en.txt
G:\concatfiles\f95.txt
G:\translated\ts95.zh-CN.en.txt
G:\concatfiles\f96.txt
G:\translated\ts96.zh-CN.en.txt
G:\concatfiles\f97.txt
G:\translated\ts97.zh-CN.en.txt
G:\concatfiles\f98.txt
G:\translated\ts98.zh-CN.en.txt
G:\concatfiles\f99.txt
G:\translated\ts99.zh-CN.en.txt
G:\concatfiles\f100.txt
G:\translated\ts100.zh-CN.en.txt
G:\concatfiles\f101.txt
G:\translated\ts101.zh-CN.en.txt
G:\concatfiles\f102.txt
G:\translated\ts102.zh-CN.en.txt
G:\concatfiles\f103.txt
G:\translated\ts103.zh-CN.en.txt
G:\concatfiles\f104.txt

G:\translated\ts104.zh-CN.en.txt
G:\concatfiles\f105.txt
G:\translated\ts105.zh-CN.en.txt
G:\concatfiles\f106.txt
G:\translated\ts106.zh-CN.en.txt
G:\concatfiles\f107.txt
G:\translated\ts107.zh-CN.en.txt
G:\concatfiles\f108.txt
G:\translated\ts108.zh-CN.en.txt
G:\concatfiles\f109.txt
G:\translated\ts109.zh-CN.en.txt
G:\concatfiles\f110.txt
G:\translated\ts110.zh-CN.en.txt
G:\concatfiles\f111.txt
G:\translated\ts111.zh-CN.en.txt
G:\concatfiles\f112.txt
G:\translated\ts112.zh-CN.en.txt
G:\concatfiles\f113.txt
G:\translated\ts113.zh-CN.en.txt
G:\concatfiles\f114.txt
G:\translated\ts114.zh-CN.en.txt
G:\concatfiles\f115.txt
G:\translated\ts115.zh-CN.en.txt
G:\concatfiles\f116.txt
G:\translated\ts116.zh-CN.en.txt
G:\concatfiles\f117.txt
G:\translated\ts117.zh-CN.en.txt
G:\concatfiles\f118.txt
G:\translated\ts118.zh-CN.en.txt
G:\concatfiles\f119.txt
G:\translated\ts119.zh-CN.en.txt
G:\concatfiles\f120.txt
G:\translated\ts120.zh-CN.en.txt
G:\concatfiles\f121.txt
G:\translated\ts121.zh-CN.en.txt
G:\concatfiles\f122.txt
G:\translated\ts122.zh-CN.en.txt
G:\concatfiles\f123.txt
G:\translated\ts123.zh-CN.en.txt
G:\concatfiles\f124.txt
G:\translated\ts124.zh-CN.en.txt
G:\concatfiles\f125.txt
G:\translated\ts125.zh-CN.en.txt
G:\concatfiles\f126.txt
G:\translated\ts126.zh-CN.en.txt
G:\concatfiles\f127.txt
G:\translated\ts127.zh-CN.en.txt
G:\concatfiles\f128.txt
G:\translated\ts128.zh-CN.en.txt
G:\concatfiles\f129.txt
G:\translated\ts129.zh-CN.en.txt
G:\concatfiles\f130.txt

G:\translated\ts130.zh-CN.en.txt
G:\concatfiles\f131.txt
G:\translated\ts131.zh-CN.en.txt
G:\concatfiles\f132.txt
G:\translated\ts132.zh-CN.en.txt
G:\concatfiles\f133.txt
G:\translated\ts133.zh-CN.en.txt
G:\concatfiles\f134.txt
G:\translated\ts134.zh-CN.en.txt
G:\concatfiles\f135.txt
G:\translated\ts135.zh-CN.en.txt
G:\concatfiles\f136.txt
G:\translated\ts136.zh-CN.en.txt
G:\concatfiles\f137.txt
G:\translated\ts137.zh-CN.en.txt
G:\concatfiles\f138.txt
G:\translated\ts138.zh-CN.en.txt
G:\concatfiles\f139.txt
G:\translated\ts139.zh-CN.en.txt
G:\concatfiles\f140.txt
G:\translated\ts140.zh-CN.en.txt
G:\concatfiles\f141.txt
G:\translated\ts141.zh-CN.en.txt
G:\concatfiles\f142.txt
G:\translated\ts142.zh-CN.en.txt
G:\concatfiles\f143.txt
G:\translated\ts143.zh-CN.en.txt
G:\concatfiles\f144.txt
G:\translated\ts144.zh-CN.en.txt
G:\concatfiles\f145.txt
G:\translated\ts145.zh-CN.en.txt
G:\concatfiles\f146.txt
G:\translated\ts146.zh-CN.en.txt
G:\concatfiles\f147.txt
G:\translated\ts147.zh-CN.en.txt
G:\concatfiles\f148.txt
G:\translated\ts148.zh-CN.en.txt
G:\concatfiles\f149.txt
G:\translated\ts149.zh-CN.en.txt
G:\concatfiles\f150.txt
G:\translated\ts150.zh-CN.en.txt
G:\concatfiles\f151.txt
G:\translated\ts151.zh-CN.en.txt
G:\concatfiles\f152.txt
G:\translated\ts152.zh-CN.en.txt
G:\concatfiles\f153.txt
G:\translated\ts153.zh-CN.en.txt
G:\concatfiles\f154.txt
G:\translated\ts154.zh-CN.en.txt
G:\concatfiles\f155.txt
G:\translated\ts155.zh-CN.en.txt
G:\concatfiles\f156.txt

G:\translated\ts156.zh-CN.en.txt
G:\concatfiles\f157.txt
G:\translated\ts157.zh-CN.en.txt
G:\concatfiles\f158.txt
G:\translated\ts158.zh-CN.en.txt
G:\concatfiles\f159.txt
G:\translated\ts159.zh-CN.en.txt
G:\concatfiles\f160.txt
G:\translated\ts160.zh-CN.en.txt
G:\concatfiles\f161.txt
G:\translated\ts161.zh-CN.en.txt
G:\concatfiles\f162.txt
G:\translated\ts162.zh-CN.en.txt
G:\concatfiles\f163.txt
G:\translated\ts163.zh-CN.en.txt
G:\concatfiles\f164.txt
G:\translated\ts164.zh-CN.en.txt
G:\concatfiles\f165.txt
G:\translated\ts165.zh-CN.en.txt
G:\concatfiles\f166.txt
G:\translated\ts166.zh-CN.en.txt
G:\concatfiles\f167.txt
G:\translated\ts167.zh-CN.en.txt
G:\concatfiles\f168.txt
G:\translated\ts168.zh-CN.en.txt
G:\concatfiles\f169.txt
G:\translated\ts169.zh-CN.en.txt
G:\concatfiles\f170.txt
G:\translated\ts170.zh-CN.en.txt
G:\concatfiles\f171.txt
G:\translated\ts171.zh-CN.en.txt
G:\concatfiles\f172.txt
G:\translated\ts172.zh-CN.en.txt
G:\concatfiles\f173.txt
G:\translated\ts173.zh-CN.en.txt
G:\concatfiles\f174.txt
G:\translated\ts174.zh-CN.en.txt
G:\concatfiles\f175.txt
G:\translated\ts175.zh-CN.en.txt
G:\concatfiles\f176.txt
G:\translated\ts176.zh-CN.en.txt
G:\concatfiles\f177.txt
G:\translated\ts177.zh-CN.en.txt
G:\concatfiles\f178.txt
G:\translated\ts178.zh-CN.en.txt
G:\concatfiles\f179.txt
G:\translated\ts179.zh-CN.en.txt
G:\concatfiles\f180.txt
G:\translated\ts180.zh-CN.en.txt
G:\concatfiles\f181.txt
G:\translated\ts181.zh-CN.en.txt
G:\concatfiles\f182.txt
G:\translated\ts182.zh-CN.en.txt

G:\translated\ts182.zh-CN.en.txt
G:\concatfiles\f183.txt
G:\translated\ts183.zh-CN.en.txt
G:\concatfiles\f184.txt
G:\translated\ts184.zh-CN.en.txt
G:\concatfiles\f185.txt
G:\translated\ts185.zh-CN.en.txt
G:\concatfiles\f186.txt
G:\translated\ts186.zh-CN.en.txt
G:\concatfiles\f187.txt
G:\translated\ts187.zh-CN.en.txt
G:\concatfiles\f188.txt
G:\translated\ts188.zh-CN.en.txt
G:\concatfiles\f189.txt
G:\translated\ts189.zh-CN.en.txt
G:\concatfiles\f190.txt
G:\translated\ts190.zh-CN.en.txt
G:\concatfiles\f191.txt
G:\translated\ts191.zh-CN.en.txt
G:\concatfiles\f192.txt
G:\translated\ts192.zh-CN.en.txt
G:\concatfiles\f193.txt
G:\translated\ts193.zh-CN.en.txt
G:\concatfiles\f194.txt
G:\translated\ts194.zh-CN.en.txt
G:\concatfiles\f195.txt
G:\translated\ts195.zh-CN.en.txt
G:\concatfiles\f196.txt
G:\translated\ts196.zh-CN.en.txt
G:\concatfiles\f197.txt
G:\translated\ts197.zh-CN.en.txt
G:\concatfiles\f198.txt
G:\translated\ts198.zh-CN.en.txt
G:\concatfiles\f199.txt
G:\translated\ts199.zh-CN.en.txt
G:\concatfiles\f200.txt
G:\translated\ts200.zh-CN.en.txt
G:\concatfiles\f201.txt
G:\translated\ts201.zh-CN.en.txt
G:\concatfiles\f202.txt
G:\translated\ts202.zh-CN.en.txt
G:\concatfiles\f203.txt
G:\translated\ts203.zh-CN.en.txt
G:\concatfiles\f204.txt
G:\translated\ts204.zh-CN.en.txt
G:\concatfiles\f205.txt
G:\translated\ts205.zh-CN.en.txt
G:\concatfiles\f206.txt
G:\translated\ts206.zh-CN.en.txt
G:\concatfiles\f207.txt
G:\translated\ts207.zh-CN.en.txt
G:\concatfiles\f208.txt
G:\translated\ts208.zh-CN.en.txt

```
G:\concatfiles\f209.txt
G:\translated\ts209.zh-CN.en.txt
G:\concatfiles\f210.txt
G:\translated\ts210.zh-CN.en.txt
G:\concatfiles\f211.txt
G:\translated\ts211.zh-CN.en.txt
G:\concatfiles\f212.txt
G:\translated\ts212.zh-CN.en.txt
G:\concatfiles\f213.txt
G:\translated\ts213.zh-CN.en.txt
G:\concatfiles\f214.txt
G:\translated\ts214.zh-CN.en.txt
G:\concatfiles\f215.txt
G:\translated\ts215.zh-CN.en.txt
G:\concatfiles\f216.txt
G:\translated\ts216.zh-CN.en.txt
G:\concatfiles\f217.txt
G:\translated\ts217.zh-CN.en.txt
```

In [11]:

```
translated1=pd.DataFrame()
for j in range(219,339):
    filename="G:\\concatfiles\\f"+str(j)+".txt"
    transname="G:\\translated\\ts"+str(j)+".zh-CN.en.txt"
    print(filename)
    print(transname)
    f=pd.read_csv(filename)
    t= pd.read_csv(transname,sep="5A09")
    frames = [f,t]
#result = pd.concat(frames, ignore_index=False)
    #df="df"+str(j)
    df=(pd.concat(frames, join='outer', ignore_index=False,keys=None, levels=None, na
mes=None, verify_integrity=False, copy=True, axis=1))
    translated1=translated1.append(df)
    #translated.append(df,ignore_index=True)
```

```
G:\concatfiles\f219.txt
G:\translated\ts219.zh-CN.en.txt
G:\concatfiles\f220.txt
G:\translated\ts220.zh-CN.en.txt
```

```
C:\Users\DELL\Anaconda3\lib\site-packages\ipykernel_launcher.py:8: ParserWarning: Fal
ling back to the 'python' engine because the 'c' engine does not support regex separa
tors (separators > 1 char and different from '\s+' are interpreted as regex); you can
avoid this warning by specifying engine='python'.
```

```
G:\concatfiles\f221.txt
G:\translated\ts221.zh-CN.en.txt
G:\concatfiles\f222.txt
G:\translated\ts222.zh-CN.en.txt
G:\concatfiles\f223.txt
```

G:\translated\ts223.zh-CN.en.txt
G:\concatfiles\f224.txt
G:\translated\ts224.zh-CN.en.txt
G:\concatfiles\f225.txt
G:\translated\ts225.zh-CN.en.txt
G:\concatfiles\f226.txt
G:\translated\ts226.zh-CN.en.txt
G:\concatfiles\f227.txt
G:\translated\ts227.zh-CN.en.txt
G:\concatfiles\f228.txt
G:\translated\ts228.zh-CN.en.txt
G:\concatfiles\f229.txt
G:\translated\ts229.zh-CN.en.txt
G:\concatfiles\f230.txt
G:\translated\ts230.zh-CN.en.txt
G:\concatfiles\f231.txt
G:\translated\ts231.zh-CN.en.txt
G:\concatfiles\f232.txt
G:\translated\ts232.zh-CN.en.txt
G:\concatfiles\f233.txt
G:\translated\ts233.zh-CN.en.txt
G:\concatfiles\f234.txt
G:\translated\ts234.zh-CN.en.txt
G:\concatfiles\f235.txt
G:\translated\ts235.zh-CN.en.txt
G:\concatfiles\f236.txt
G:\translated\ts236.zh-CN.en.txt
G:\concatfiles\f237.txt
G:\translated\ts237.zh-CN.en.txt
G:\concatfiles\f238.txt
G:\translated\ts238.zh-CN.en.txt
G:\concatfiles\f239.txt
G:\translated\ts239.zh-CN.en.txt
G:\concatfiles\f240.txt
G:\translated\ts240.zh-CN.en.txt
G:\concatfiles\f241.txt
G:\translated\ts241.zh-CN.en.txt
G:\concatfiles\f242.txt
G:\translated\ts242.zh-CN.en.txt
G:\concatfiles\f243.txt
G:\translated\ts243.zh-CN.en.txt
G:\concatfiles\f244.txt
G:\translated\ts244.zh-CN.en.txt
G:\concatfiles\f245.txt
G:\translated\ts245.zh-CN.en.txt
G:\concatfiles\f246.txt
G:\translated\ts246.zh-CN.en.txt
G:\concatfiles\f247.txt
G:\translated\ts247.zh-CN.en.txt
G:\concatfiles\f248.txt
G:\translated\ts248.zh-CN.en.txt
G:\concatfiles\f249.txt

G:\translated\ts249.zh-CN.en.txt
G:\concatfiles\f250.txt
G:\translated\ts250.zh-CN.en.txt
G:\concatfiles\f251.txt
G:\translated\ts251.zh-CN.en.txt
G:\concatfiles\f252.txt
G:\translated\ts252.zh-CN.en.txt
G:\concatfiles\f253.txt
G:\translated\ts253.zh-CN.en.txt
G:\concatfiles\f254.txt
G:\translated\ts254.zh-CN.en.txt
G:\concatfiles\f255.txt
G:\translated\ts255.zh-CN.en.txt
G:\concatfiles\f256.txt
G:\translated\ts256.zh-CN.en.txt
G:\concatfiles\f257.txt
G:\translated\ts257.zh-CN.en.txt
G:\concatfiles\f258.txt
G:\translated\ts258.zh-CN.en.txt
G:\concatfiles\f259.txt
G:\translated\ts259.zh-CN.en.txt
G:\concatfiles\f260.txt
G:\translated\ts260.zh-CN.en.txt
G:\concatfiles\f261.txt
G:\translated\ts261.zh-CN.en.txt
G:\concatfiles\f262.txt
G:\translated\ts262.zh-CN.en.txt
G:\concatfiles\f263.txt
G:\translated\ts263.zh-CN.en.txt
G:\concatfiles\f264.txt
G:\translated\ts264.zh-CN.en.txt
G:\concatfiles\f265.txt
G:\translated\ts265.zh-CN.en.txt
G:\concatfiles\f266.txt
G:\translated\ts266.zh-CN.en.txt
G:\concatfiles\f267.txt
G:\translated\ts267.zh-CN.en.txt
G:\concatfiles\f268.txt
G:\translated\ts268.zh-CN.en.txt
G:\concatfiles\f269.txt
G:\translated\ts269.zh-CN.en.txt
G:\concatfiles\f270.txt
G:\translated\ts270.zh-CN.en.txt
G:\concatfiles\f271.txt
G:\translated\ts271.zh-CN.en.txt
G:\concatfiles\f272.txt
G:\translated\ts272.zh-CN.en.txt
G:\concatfiles\f273.txt
G:\translated\ts273.zh-CN.en.txt
G:\concatfiles\f274.txt
G:\translated\ts274.zh-CN.en.txt
G:\concatfiles\f275.txt
G:\translated\ts275.zh-CN.en.txt

G:\translated\ts275.zh-CN.en.txt
G:\concatfiles\f276.txt
G:\translated\ts276.zh-CN.en.txt
G:\concatfiles\f277.txt
G:\translated\ts277.zh-CN.en.txt
G:\concatfiles\f278.txt
G:\translated\ts278.zh-CN.en.txt
G:\concatfiles\f279.txt
G:\translated\ts279.zh-CN.en.txt
G:\concatfiles\f280.txt
G:\translated\ts280.zh-CN.en.txt
G:\concatfiles\f281.txt
G:\translated\ts281.zh-CN.en.txt
G:\concatfiles\f282.txt
G:\translated\ts282.zh-CN.en.txt
G:\concatfiles\f283.txt
G:\translated\ts283.zh-CN.en.txt
G:\concatfiles\f284.txt
G:\translated\ts284.zh-CN.en.txt
G:\concatfiles\f285.txt
G:\translated\ts285.zh-CN.en.txt
G:\concatfiles\f286.txt
G:\translated\ts286.zh-CN.en.txt
G:\concatfiles\f287.txt
G:\translated\ts287.zh-CN.en.txt
G:\concatfiles\f288.txt
G:\translated\ts288.zh-CN.en.txt
G:\concatfiles\f289.txt
G:\translated\ts289.zh-CN.en.txt
G:\concatfiles\f290.txt
G:\translated\ts290.zh-CN.en.txt
G:\concatfiles\f291.txt
G:\translated\ts291.zh-CN.en.txt
G:\concatfiles\f292.txt
G:\translated\ts292.zh-CN.en.txt
G:\concatfiles\f293.txt
G:\translated\ts293.zh-CN.en.txt
G:\concatfiles\f294.txt
G:\translated\ts294.zh-CN.en.txt
G:\concatfiles\f295.txt
G:\translated\ts295.zh-CN.en.txt
G:\concatfiles\f296.txt
G:\translated\ts296.zh-CN.en.txt
G:\concatfiles\f297.txt
G:\translated\ts297.zh-CN.en.txt
G:\concatfiles\f298.txt
G:\translated\ts298.zh-CN.en.txt
G:\concatfiles\f299.txt
G:\translated\ts299.zh-CN.en.txt
G:\concatfiles\f300.txt
G:\translated\ts300.zh-CN.en.txt
G:\concatfiles\f301.txt
G:\translated\ts301.zh-CN.en.txt

G:\concatfiles\f302.txt
G:\translated\ts302.zh-CN.en.txt
G:\concatfiles\f303.txt
G:\translated\ts303.zh-CN.en.txt
G:\concatfiles\f304.txt
G:\translated\ts304.zh-CN.en.txt
G:\concatfiles\f305.txt
G:\translated\ts305.zh-CN.en.txt
G:\concatfiles\f306.txt
G:\translated\ts306.zh-CN.en.txt
G:\concatfiles\f307.txt
G:\translated\ts307.zh-CN.en.txt
G:\concatfiles\f308.txt
G:\translated\ts308.zh-CN.en.txt
G:\concatfiles\f309.txt
G:\translated\ts309.zh-CN.en.txt
G:\concatfiles\f310.txt
G:\translated\ts310.zh-CN.en.txt
G:\concatfiles\f311.txt
G:\translated\ts311.zh-CN.en.txt
G:\concatfiles\f312.txt
G:\translated\ts312.zh-CN.en.txt
G:\concatfiles\f313.txt
G:\translated\ts313.zh-CN.en.txt
G:\concatfiles\f314.txt
G:\translated\ts314.zh-CN.en.txt
G:\concatfiles\f315.txt
G:\translated\ts315.zh-CN.en.txt
G:\concatfiles\f316.txt
G:\translated\ts316.zh-CN.en.txt
G:\concatfiles\f317.txt
G:\translated\ts317.zh-CN.en.txt
G:\concatfiles\f318.txt
G:\translated\ts318.zh-CN.en.txt
G:\concatfiles\f319.txt
G:\translated\ts319.zh-CN.en.txt
G:\concatfiles\f320.txt
G:\translated\ts320.zh-CN.en.txt
G:\concatfiles\f321.txt
G:\translated\ts321.zh-CN.en.txt
G:\concatfiles\f322.txt
G:\translated\ts322.zh-CN.en.txt
G:\concatfiles\f323.txt
G:\translated\ts323.zh-CN.en.txt
G:\concatfiles\f324.txt
G:\translated\ts324.zh-CN.en.txt
G:\concatfiles\f325.txt
G:\translated\ts325.zh-CN.en.txt
G:\concatfiles\f326.txt
G:\translated\ts326.zh-CN.en.txt
G:\concatfiles\f327.txt
G:\translated\ts327.zh-CN.en.txt

```
G:\concatfiles\f328.txt
G:\translated\ts328.zh-CN.en.txt
G:\concatfiles\f329.txt
G:\translated\ts329.zh-CN.en.txt
G:\concatfiles\f330.txt
G:\translated\ts330.zh-CN.en.txt
G:\concatfiles\f331.txt
G:\translated\ts331.zh-CN.en.txt
G:\concatfiles\f332.txt
G:\translated\ts332.zh-CN.en.txt
G:\concatfiles\f333.txt
G:\translated\ts333.zh-CN.en.txt
G:\concatfiles\f334.txt
G:\translated\ts334.zh-CN.en.txt
G:\concatfiles\f335.txt
G:\translated\ts335.zh-CN.en.txt
G:\concatfiles\f336.txt
G:\translated\ts336.zh-CN.en.txt
G:\concatfiles\f337.txt
G:\translated\ts337.zh-CN.en.txt
G:\concatfiles\f338.txt
G:\translated\ts338.zh-CN.en.txt
```

In [12]:

```
translated2=pd.DataFrame()
for j in range(340,400):
    filename="G:\\concatfiles\\f"+str(j)+".txt"
    transname="G:\\translated\\ts"+str(j)+".zh-CN.en.txt"
    print(filename)
    print(transname)
    f=pd.read_csv(filename)
    t= pd.read_csv(transname,sep="5A09")
    frames = [f,t]
#result = pd.concat(frames, ignore_index=False)
    #df="df"+str(j)
    df=(pd.concat(frames, join='outer', ignore_index=False,keys=None, levels=None, na
mes=None, verify_integrity=False, copy=True, axis=1))
    translated2=translated2.append(df)
    #translated.append(df,ignore_index=True)
```

```
G:\concatfiles\f340.txt
G:\translated\ts340.zh-CN.en.txt
G:\concatfiles\f341.txt
G:\translated\ts341.zh-CN.en.txt
G:\concatfiles\f342.txt
G:\translated\ts342.zh-CN.en.txt
```

```
C:\Users\DELL\Anaconda3\lib\site-packages\ipykernel_launcher.py:8: ParserWarning: Fal
ling back to the 'python' engine because the 'c' engine does not support regex separa
tors (separators > 1 char and different from '\s+' are interpreted as regex); you can
avoid this warning by specifying engine='python'.
```

G:\concatfiles\f343.txt
G:\translated\ts343.zh-CN.en.txt
G:\concatfiles\f344.txt
G:\translated\ts344.zh-CN.en.txt
G:\concatfiles\f345.txt
G:\translated\ts345.zh-CN.en.txt
G:\concatfiles\f346.txt
G:\translated\ts346.zh-CN.en.txt
G:\concatfiles\f347.txt
G:\translated\ts347.zh-CN.en.txt
G:\concatfiles\f348.txt
G:\translated\ts348.zh-CN.en.txt
G:\concatfiles\f349.txt
G:\translated\ts349.zh-CN.en.txt
G:\concatfiles\f350.txt
G:\translated\ts350.zh-CN.en.txt
G:\concatfiles\f351.txt
G:\translated\ts351.zh-CN.en.txt
G:\concatfiles\f352.txt
G:\translated\ts352.zh-CN.en.txt
G:\concatfiles\f353.txt
G:\translated\ts353.zh-CN.en.txt
G:\concatfiles\f354.txt
G:\translated\ts354.zh-CN.en.txt
G:\concatfiles\f355.txt
G:\translated\ts355.zh-CN.en.txt
G:\concatfiles\f356.txt
G:\translated\ts356.zh-CN.en.txt
G:\concatfiles\f357.txt
G:\translated\ts357.zh-CN.en.txt
G:\concatfiles\f358.txt
G:\translated\ts358.zh-CN.en.txt
G:\concatfiles\f359.txt
G:\translated\ts359.zh-CN.en.txt
G:\concatfiles\f360.txt
G:\translated\ts360.zh-CN.en.txt
G:\concatfiles\f361.txt
G:\translated\ts361.zh-CN.en.txt
G:\concatfiles\f362.txt
G:\translated\ts362.zh-CN.en.txt
G:\concatfiles\f363.txt
G:\translated\ts363.zh-CN.en.txt
G:\concatfiles\f364.txt
G:\translated\ts364.zh-CN.en.txt
G:\concatfiles\f365.txt
G:\translated\ts365.zh-CN.en.txt
G:\concatfiles\f366.txt
G:\translated\ts366.zh-CN.en.txt
G:\concatfiles\f367.txt
G:\translated\ts367.zh-CN.en.txt
G:\concatfiles\f368.txt

G:\translated\ts368.zh-CN.en.txt
G:\concatfiles\f369.txt
G:\translated\ts369.zh-CN.en.txt
G:\concatfiles\f370.txt
G:\translated\ts370.zh-CN.en.txt
G:\concatfiles\f371.txt
G:\translated\ts371.zh-CN.en.txt
G:\concatfiles\f372.txt
G:\translated\ts372.zh-CN.en.txt
G:\concatfiles\f373.txt
G:\translated\ts373.zh-CN.en.txt
G:\concatfiles\f374.txt
G:\translated\ts374.zh-CN.en.txt
G:\concatfiles\f375.txt
G:\translated\ts375.zh-CN.en.txt
G:\concatfiles\f376.txt
G:\translated\ts376.zh-CN.en.txt
G:\concatfiles\f377.txt
G:\translated\ts377.zh-CN.en.txt
G:\concatfiles\f378.txt
G:\translated\ts378.zh-CN.en.txt
G:\concatfiles\f379.txt
G:\translated\ts379.zh-CN.en.txt
G:\concatfiles\f380.txt
G:\translated\ts380.zh-CN.en.txt
G:\concatfiles\f381.txt
G:\translated\ts381.zh-CN.en.txt
G:\concatfiles\f382.txt
G:\translated\ts382.zh-CN.en.txt
G:\concatfiles\f383.txt
G:\translated\ts383.zh-CN.en.txt
G:\concatfiles\f384.txt
G:\translated\ts384.zh-CN.en.txt
G:\concatfiles\f385.txt
G:\translated\ts385.zh-CN.en.txt
G:\concatfiles\f386.txt
G:\translated\ts386.zh-CN.en.txt
G:\concatfiles\f387.txt
G:\translated\ts387.zh-CN.en.txt
G:\concatfiles\f388.txt
G:\translated\ts388.zh-CN.en.txt
G:\concatfiles\f389.txt
G:\translated\ts389.zh-CN.en.txt
G:\concatfiles\f390.txt
G:\translated\ts390.zh-CN.en.txt
G:\concatfiles\f391.txt
G:\translated\ts391.zh-CN.en.txt
G:\concatfiles\f392.txt
G:\translated\ts392.zh-CN.en.txt
G:\concatfiles\f393.txt
G:\translated\ts393.zh-CN.en.txt
G:\concatfiles\f394.txt
G:\translated\ts394.zh-CN.en.txt

G:\translated\ts394.zh-CN.en.txt
G:\concatfiles\f395.txt
G:\translated\ts395.zh-CN.en.txt
G:\concatfiles\f396.txt
G:\translated\ts396.zh-CN.en.txt
G:\concatfiles\f397.txt
G:\translated\ts397.zh-CN.en.txt
G:\concatfiles\f398.txt
G:\translated\ts398.zh-CN.en.txt
G:\concatfiles\f399.txt
G:\translated\ts399.zh-CN.en.txt

In [15]:

```
frames=[translated,translated1,translated2]
```

In [17]:

```
translated=translated.append(translated1)
```

In [18]:

```
translated=translated.append(translated2)
```

In [20]:

```
translated.head(611759).to_csv("E://DMA_PRE//Translated1.csv", sep=',',index=False,
encoding= 'utf-8')
translated.tail(611758).to_csv("E://DMA_PRE//Translated2.csv", sep=',',index=False,
encoding= 'utf-8')
```

In [10]:

```
translated.shape
```

Out[10]:

(670177, 13)

TEXT PREPROCESSING

REMOVAL OF NOISE - URL

In [11]:

```
def remurl(content):
    try:
        URLless_string = re.sub(r'\w+:\/{2}[\d\w-]+(\.[\d\w-]+)*(?:(?:\/[^\s/]*))*',
'', content)
        return URLless_string
```

```
    except Exception as e:
        print(str(e))
```

In [13]:

```
df_urlrem = pd.DataFrame(columns=['en_contenturl','url_rem'])
for i in range(50000):
    non_emo=translated['en_content'].iloc[i]
    content=translated['en_content'].iloc[i]
    new_content=remurl(content)

    df_urlrem = df_urlrem.append({'en_contenturl': non_emo,'url_rem':new_content}, ig
nore_index=True)
```

```
expected string or bytes-like object
expected string or bytes-like object
expected string or bytes-like object
expected string or bytes-like object
expected string or bytes-like object
expected string or bytes-like object
expected string or bytes-like object
expected string or bytes-like object
expected string or bytes-like object
expected string or bytes-like object
expected string or bytes-like object
```

OUTPUT

In [22]:

```
df_urlrem.head(5)
```

Out[22]:

	en_contenturl	url_rem
0	Lijiang Tourism (sz002033) # ## stock stocks F...	Lijiang Tourism (sz002033) # ## stock stocks F...
1	Chen Ling Ding # # red envelopes to make money...	Chen Ling Ding # # red envelopes to make money...
2	Taobao these sucker. . . Industrial gas fire n...	Taobao these sucker. . . Industrial gas fire n...
3	Aspect can not say, you know everything [laugh...	Aspect can not say, you know everything [laugh...
4	Over 111 Zhang	Over 111 Zhang

REMOVAL OF STOPWORDS

In [23]:

```
remStopword=pd.DataFrame()
```

In [24]:

```
def removeStopwords(data):
    stop_words = set(stopwords.words('english'))
    words = word_tokenize(data)
    wordsFiltered = []
    try:
        for w in words:
            if (w not in stop_words) :
                wordsFiltered.append(w)
        return wordsFiltered
    except Exception as e:
        print(str(e))
```

In [1]:

```
df_new = pd.DataFrame(columns=['en_contentsw','Stopwrod_removed'])
for i in range(50000):
    non_emo=df_urlrem['url_rem'].iloc[i]
    remStopword=removeStopwords(df_urlrem['url_rem'].iloc[i])
    list1=[non_emo,remStopword]

    df_new = df_new.append({'en_contentsw': non_emo, 'Stopword_removed': remStopword}
, ignore_index=True)
```

OUTPUT

In [114]:

```
df_new.drop(['Stopwrod_removed'], axis=1).head(5)
```

Out[114]:

	en_contentsw	Stopword_removed
0	Lijiang Tourism (sz002033) # ## stock stocks F...	[Lijiang, Tourism, (, sz002033,), #, #, #, st...
1	Chen Ling Ding # # red envelopes to make money...	[Chen, Ling, Ding, #, #, red, envelopes, make,...
2	Taobao these sucker. . . Industrial gas fire n...	[Taobao, sucker, ., ., ., Industrial, gas, fir...
3	Aspect can not say, you know everything [laugh	[Aspect, say, ,, know, everything, [, laughs,

	en_contentsw	Stopword_removed
4	Over 111 Zhang	[Over, 111, Zhang]

STEMMING

In [96]:

```
import nltk
from nltk.stem.porter import PorterStemmer
porter_stemmer = PorterStemmer()
```

In [97]:

```
def stemming(tokens):
    # First Word tokenization
    nltk_tokens =tokens
    stem = []
    #Next find the roots of the word
    try:
        for w in nltk_tokens:
            s=porter_stemmer.stem(w)
            stem.append(s)
        return stem
    except Exception as e:
        print(str(e))
```

In [32]:

```
df_stem = pd.DataFrame(columns=['en_contentst','Stemming'])
for i in range(50000):
    content=df_new['Stopword_removed'].iloc[i]
    stem=stemming(df_new['Stopword_removed'].iloc[i])
    list1=[content,stem]
    df_stem = df_stem.append({'en_contentst': content, 'Stemming': stem}, ignore_inde
x=True)
```

OUTPUT

In [116]:

```
df_stem.head(5)
```

Out[116]:

	en_contentst	Stemming
0	[Lijiang, Tourism, (, sz002033,), #, #, #, st...	[lijiang, tourism, (, sz002033,), #, #, #, st...

	en_contentst	Stemming
1	[Chen, Ling, Ding, #, #, red, envelopes, make,...	[chen, ling, ding, #, #, red, envelop, make, m...
2	[Taobao, sucker, ., ., ., Industrial, gas, fir...	[taobao, sucker, ., ., ., industri, ga, fire, ...
3	[Aspect, say, ,, know, everything, [, laughs, ...	[aspect, say, ,, know, everyth, [, laugh, cri,]]
4	[Over, 111, Zhang]	[over, 111, zhang]

LEMMATIZATION

Returns the base or dictionary form of the word (lemma) Ex:

- feet --> foot
- wolves --> wolf

In [34]:

```
###LEMMATIZATION
import nltk
from nltk.stem import WordNetLemmatizer
```

In [35]:

```
def lemmatization(tokens):
    wordnet_lemmatizer = WordNetLemmatizer()
    nltk_tokens =tokens
    lem = []
    #Next find the roots of the word
    try:
        for w in nltk_tokens:
            l=wordnet_lemmatizer.lemmatize(w)
            lem.append(l)
        return lem
    except Exception as e:
        print(str(e))
```

In [36]:

```
df_lem = pd.DataFrame(columns=['Stemmingle','lemmatization'])
for i in range(50000):
    content=df_stem['Stemming'].iloc[i]
    lem=stemming(df_stem['Stemming'].iloc[i])
    list1=[content,lem]
    df_lem = df_lem.append({'Stemmingle': content, 'lemmatization': lem}, ignore_inde
x=True)
```

In [117]:

```
df_lem.head(5)
```

Out[117]:

	Stemmingle	lemmatization
0	[lijiang, tourism, (, sz002033,), #, #, #, st...	[lijiang, tourism, (, sz002033,), #, #, #, st...
1	[chen, ling, ding, #, #, red, envelop, make, m...	[chen, ling, ding, #, #, red, envelop, make, m...
2	[taobao, sucker, ., ., ., industri, ga, fire, ...	[taobao, sucker, ., ., ., industri, ga, fire, ...
3	[aspect, say, ,, know, everyth, [, laugh, cri,]]	[aspect, say, ,, know, everyth, [, laugh, cri,]]
4	[over, 111, zhang]	[over, 111, zhang]

In [38]:

```
#nltk.download('wordnet')
```

Converting to lower case

In [39]:

```
def tolower(tokens):
    nltk_tokens=tokens
    x = [element.lower() for element in nltk_tokens]
    return x
```

In [40]:

```
df_lower = pd.DataFrame(columns=['lemmatizationtl','lower'])
for i in range(50000):
    content=df_lem['lemmatization'].iloc[i]
    low=tolower(df_lem['lemmatization'].iloc[i])
    list1=[content,low]
    df_lower = df_lower.append({'lemmatizationtl': content, 'lower': low}, ignore_ind
ex=True)
```

OUTPUT

In [118]:

```
df_lower.head(5)
```

Out[118]:

	lemmatizationtl	lower
0	[lijiang, tourism, (, sz002033,), #, #, #, st...	[lijiang, tourism, (, sz002033,), #, #, #, st...
1	[chen, ling, ding, #, #, red, envelop, make,	[chen, ling, ding, #, #, red, envelop, make,

	m... lemmatizationtl	m... lower
2	[taobao, sucker, ., ., ., industri, ga, fire, ...	[taobao, sucker, ., ., ., industri, ga, fire, ...
3	[aspect, say, ,, know, everyth, [, laugh, cri,]]	[aspect, say, ,, know, everyth, [, laugh, cri,]]
4	[over, 111, zhang]	[over, zhang]

Removing numbers

In [42]:

```
def rem_num(tokens):
        for item in tokens:
            if item.isdigit():
                tokens.remove(item)
        return tokens
```

In [104]:

```
df_remnum = pd.DataFrame(columns=['lowerrnum','no_num'])
for i in range(50000):
    content=df_lower['lower'].iloc[i]
    nonum=rem_num(df_lower['lower'].iloc[i])
    df_remnum = df_remnum.append({'lowerrnum': content, 'no_num': nonum}, ignore_inde
x=True)
```

REMOVE PUNTUATION

In [121]:

```
def rem_punctuation(tokens):
        input_text = ' '.join(tokens).lower()
        s = re.sub(r"[-()\"#/@;:<>{}\[\]`+=~|.!?,]", "", input_text)
        #print(input_text)
        words = word_tokenize(s)
        return words
```

In [122]:

```
df_rempunc = pd.DataFrame(columns=['no_numrp','no_punc'])
for i in range(50000):
    content=df_remnum['no_num'].iloc[i]
    nopun=rem_punctuation(df_remnum['no_num'].iloc[i])
    list1=[content,nopun]
    df_rempunc = df_rempunc.append({'no_numrp': content, 'no_punc': nopun}, ignore_in
dex=True)
```

OUTPUT

In [186]:

```
T=translated.head(50000)
```

In [192]:

```
frames=[T,df_urlrem, df_new, df_stem, df_lem, df_lower, df_remnum, df_rempunc]
```

In [190]:

```
T = T.reset_index(drop=True)
```

In [193]:

```
Train=(pd.concat(frames, join='outer', ignore_index=False,keys=None, levels=None, nam
es=None, verify_integrity=False, copy=True, axis=1))
```

In [196]:

```
Train=Train.drop(['content_spchar',
'non_emoji_content','content','en_content','Unnamed: 1', 'en_contenturl',
       'url_rem', 'en_contentsw', 'Stopwrod_removed', 'Stopword_removed',
       'en_contentst', 'Stemming', 'Stemmingle', 'lemmatization',
       'lemmatizationtl', 'lower', 'lowerrnum', 'no_num', 'no_numrp'], axis=1)
```

In [197]:

```
Train = Train.rename(columns={'no_punc': 'content'})
```

In [198]:

```
Train.shape
```

Out[198]:

```
(10000, 10)
```

In [210]:

```
Train=Train.drop(['index'],axis=1)
```

In [211]:

```
Train.to_csv("E:\\DMA_PRE\\PREPROCESSED.csv", sep=',',index=False, encoding= 'utf-8')
```

In [214]:

```
Train.columns
```

Out[214]:

```
Index(['u_id', 'm_id', 'forward_count', 'comment_count', 'like_count', 'date',
       'time', 'content_media_count', 'content'],
```

```
      dtype='object')
```

In [212]:

```
Train
```

Out[212]:

	u_id	m_id	forwa
0	d38e9bed5d98110dc2489d0d1cac3c2a	7d45833d9865727a88b960b0603c19f6	0
1	fa13974743d3fe6ff40d21b872325e9e	8169f1d45051e08ef213bf1106b1225d	0
2	da534fe87e7a52777bee5c30573ed5fd	68cd0258c31c2c525f94febea2d9523b	0
3	e06a22b7e065e559a1f0bf7841a85c51	00b9f86b4915aedb7db943c54fd19d59	0
4	f9828598f9664d4e347ef2048ce17734	c7f6f66044c0c5a3330e2c5371be6824	0
5	d80f3d3c5c1d658e82b837a4dd1af849	bfc0819b83ec59ce767287077f2b3507	0
6	f349a67d1cd7c8683c5bbc5f8486e193	83674a60e5310195fc35d97ea8f45c46	0
7	24b621c98f2594b698c0b1d60c9ae6db	2cbd3d514ed5ad3dab81aa043c8b3d0a	0
8	e44d81d630e4f382f657e72aa4b685da	8a88a25f9f26ed9f79080eaacc1a8668	0
9	fbe6c953632e1b3dda66cf6118b6ab12	f359a74cb4ac6150a3af8325eda04ea0	0
10	f9a3ca6bc1e75d173cfc98ec4b108072	c7bc3445e8b90db8cc5e045f606dc1ee	21
11	3c68bbb9da57fcc752c8a493d91bdd3a	77e14cf9d460715e84c51747c3641a9b	0
12	104e8d55e98eb3cd834810088af039fe	ee0b2c9d35bfeb0fbc5b3a8677f4a18c	9
13	0d15005d6397fb5ce1d45e7c834f7370	9c954d63fcfea19dca8d81a4f3b53861	0
14	875a4a77b339d93f819e2c4de5bd0b57	f2cdcdbcec9ff47cbb3c6a636e4b92a3	0
15	380a2219670f50dc87efce3380bea6e8	46f10244d02afa85d12346ce28e3cec5	0

	u_id	m_id	forwa
16	b9b88b0fc105fb08a552e782afa4342e	cb907eb1bdbc198ed0944cc3b7e24f91	0
17	f18eb14365c0d7248fab1b9c464f4e70	096543bd8746869982d1a7557164dd0d	0
18	0fc17bf5e2dc789dd48505df1f5b14fd	4c1e2418127811d212d0e3867a99db3e	0
19	dd749a5af07c04ce7de451273a983671	419dd71d562883ef836e774bc3f4e163	0
20	a984551b159fcdc0a48f9e38ecb1488f	baa0051d359555601ab61df684787f0f	0
21	2e0467b73d0f6f9e5607a6174581fdd8	2fd200a7f670138c2026091c3b01532a	0
22	819656f05994b00b7260daf7346586a7	95590e88cac5d8c9d1a496bc3bd42f07	6
23	91ce7c63b272f2037a3e702c10163fa3	8b4e85a881afaff91f276eac7bfb6604	0
24	4680e73f9e7a6b87dec62a86a7821c17	b2db095af290b3a36cf798a3e17528d8	0
25	976e85e3ededdd9b2c2a3179eb7ae8ab	9540ee0cf7ccfae523020c8025e7095f	0
26	6623347e5f19f35f2d02ad515b96524c	9a2f48a870843d1964a03c6642b309d5	0
27	cf727e70b6661387cf6aadf01d2eb32c	bff281350f035db0e84c25394865d86a	0
28	de0836c1c5d40a5cae64a964a0b54894	c3345fd72cad53ca9bffd63634170ba0	0
29	c8848f18da5911d0389c3ac70fe13204	fa352495e646a3f7ff979267c490fd89	0
...	...	...	...
9970	5ae3749e4e089c3c76843debbff80283	ee9da01dfefe7e7f0bf571e8136aad20	2
9971	5587f41cfaf471df0f37b74a298295a7	8547fbb068298fbe75eeb18afec247cb	0
9972	93cc443d5df3c53a1fd0d8e12286eb1b	123bf93f7de888ee43f57e70b39ed72b	0
9973	c4b747dca344890718884e10805be401	247ae6b3c354a33cdd80472b95012d59	6

	u_id	m_id	forwa
9974	b4e7bc5d347de90c0629fe2227d96484	6603b30d73b5e23f9fe7c8ac1f39b6d8	34
9975	7564bab83ea84e4c0985b023aac58c7d	0924b77fad659942de9955042d9795fe	0
9976	6c62c5eee1ad56b97e00e34c6eeddf83	87c3f916a2c129a3443daaf564452b31	0
9977	875a4a77b339d93f819e2c4de5bd0b57	30e19c5b0642f61d89d29bf18bb0987e	0
9978	d592754c20ee397a6932e9dcc5323a49	ee85cb92cc924761a4546bfeed373cb5	0
9979	717fcfb02acdd129954856331cbeb70f	cc8d59d3006f70566119f124bf3be0bc	0
9980	c4b747dca344890718884e10805be401	82344020ee957d6b66544191ef914c51	3
9981	f41f1ffcafafd818c3cdaab632f51c0f	2889c9f8c306edee7e46124142fb692b	0
9982	c60533fdb5278412b14379f693f77dd5	c18182c9b3a2c0786f6e61fd601eebcd	1
9983	dd171c22e4560775c6b474d2e76ff6df	03658e9964e036191ccd1ccb3c4a3030	0
9984	3b0e32348721c39b3a8ad0259f6d6671	62e854ca6b734c6128dabc288b3837e3	0
9985	ba8fa60737fd5ccde031861384b7c70b	487808510c03c57547a1d869002f9b6b	0
9986	0f24ca3980734abb64119b0c47f63872	e7f75a763842a8a7a80da9a7dcf259fc	1
9987	6e7d0f59a83e1639fe5dbe90d58924bd	a8aac705174dbfa882169a8b91f0a555	0
9988	9e44aadf543ea2cfa063add522f61791	d1a57fde410538993c2c34d23d0f1c7b	61
9989	eacb9ef7d9f8b48fe0e815231cd7e9a7	6e656816c63ed6193f7c11abc7c08636	0
9990	b7a40e686113044148b88872b5cc3a3c	f57414659c4ae247b653445d08f8002a	1
9991	fb5acde8c0d7bb225b3dae767d6cdff3	4d81c630fafd742250af37213fed2cb6	1
9992	25ca1cea6595c04a5f009a11dd0e676b	cd42a5f05c3cd93ae1733ad1f07a7b18	0

	u_id	m_id	forwa
9993	1155b6265c1f6c648a622ef87a5d40a2	0002403c7426903f9e9983761aae5d06	0
9994	2fb80ade87859a21c10b98391df7d23c	efd2e25f97bca46f356b41d07f21f556	0
9995	f965a0cebbb0ecdd9ab3e1df7029d679	febab0099a7fdd541fbc541a85f00578	0
9996	ec0f81cdfb9895775f2071853bf75e75	7dff0cc3150ae91a7f8df06ccdccbf51	0
9997	586d67e2d15faeb2cb6db6f7a44312f1	4a1fd3c45f0514021acacc830ce17f44	0
9998	7634e89faee952e49bfb983de1a6518c	323cac43f508302afee6875402338ffd	0
9999	ba58044ec2e74ab69e4a5e2fe1b732e8	396772cb616a0ed336dc9bcc0e73acfb	1

10000 rows × 9 columns

TEXT

In [1]:

```
import numpy as np
import pandas as pd
import matplotlib.pyplot as plt
import seaborn as sns
%pylab inline
import copy
from googletrans import Translator
import pandas as pd
import numpy as np
import csv
import re
import jieba
import time
import json
from sklearn.feature_extraction.text import CountVectorizer
from sklearn import linear_model
from sklearn.externals import joblib
from nltk.corpus import stopwords as e_stopwords
from datetime import datetime, timedelta
import jieba
import sys

from nltk.corpus import stopwords
from nltk.tokenize import word_tokenize
```

```
Populating the interactive namespace from numpy and matplotlib
```

In [2]:

```
train1= pd.read_csv("G:\\preprocessed_1.csv")
```

In [4]:

```
train2= pd.read_csv("G:\\preprocessed_2.csv")
```

In [7]:

```
frames=[train1,train2]
train=pd.concat(frames)
```

In [8]:

```
train.shape[0]
```

Out[8]:

```
1223517
```

In [69]:

```
translated=train
```

TEXT PREPROCESSING

REMOVAL OF NOISE - URL

In [70]:

```
def remurl(content):
    try:
        URLless_string = re.sub(r'\w+:\/{2}[\d\w-]+(\.[\d\w-]+)*(?:(?:\/[^\s/]*))*', '', content)
        return URLless_string
    except Exception as e:
        print(str(e))
        return content
```

Removal of numbers

In [71]:

```
def rem_num(tokens):
        try:
            for item in tokens:
                if item.isdigit():
                    tokens.remove(item)
            return tokens
        except Exception as e:
            print(str(e))
            return tokens
```

REMOVAL OF STOPWORDS

In [72]:

```
remStopword=pd.DataFrame()
```

In [73]:

```
def removeStopwords(data):
    stopw="STOPWORDS.txt"
    stop_words = set(stopwords.words('english'))
    words = word_tokenize(data)
    wordsFiltered = []
    try:
        for w in words:
            if (w not in stopw or w not in stop_words) :
                wordsFiltered.append(w)
        return wordsFiltered
    except Exception as e:
        print(str(e))
        return data
```

STEMMING

In [74]:

```
import nltk
from nltk.stem.porter import PorterStemmer
porter_stemmer = PorterStemmer()
```

In [75]:

```
def stemming(tokens):
    # First Word tokenization
    nltk_tokens =tokens
    stem = []
    #Next find the roots of the word
    try:
        for w in nltk_tokens:
            s=porter_stemmer.stem(w)
            stem.append(s)
        return stem
    except Exception as e:
```

```
        print(str(e))
        return tokens
```

LEMMATIZATION

In [76]:

```
###LEMMATIZATION
import nltk
from nltk.stem import WordNetLemmatizer
```

In [77]:

```
def lemmatization(tokens):
    wordnet_lemmatizer = WordNetLemmatizer()
    nltk_tokens =tokens
    lem = []
    #Next find the roots of the word
    try:
        for w in nltk_tokens:
            l=wordnet_lemmatizer.lemmatize(w)
            lem.append(l)
        return lem
    except Exception as e:
        print(str(e))
        return tokens
```

In [78]:

```
#nltk.download('wordnet')
```

Converting to lower case

In [79]:

```
def tolower(tokens):
    try:
        nltk_tokens=tokens
        x = [element.lower() for element in nltk_tokens]
        return x
    except Exception as e:
        print(str(e))
        return tokens
```

In [80]:

```
 def rem_punctuation(tokens):
        try:
            input_text = ' '.join(tokens).lower()
            s = re.sub(r"[-()\"#/@;:<>{}`+=~|.!?,]", "", input_text)
            #print(input_text)
            words = word_tokenize(s)
            return words
        except Exception as e:
            print(str(e))
            return tokens
```

In [81]:

```
## REMOVE PUNTUATION
```

In [105]:

```
df_urlrem = pd.DataFrame(columns=['en_contenturl','url_rem'])
for i in range(10):
    non_emo=translated['en_content'].iloc[i]
```

```
    content=translated['en_content'].iloc[i]
    new_content=remurl(content)

    df_urlrem = df_urlrem.append({'en_contenturl': non_emo,'url_rem':new_content},
ignore_index=True)
print("done1")
df_remnum = pd.DataFrame(columns=['url_rem',])
for i in range(10):
    content=df_urlrem['url_rem'].iloc[i]
    nonum=rem_num(df_urlrem['url_rem'].iloc[i])
    list1=[content,nonum]
    df_remnum = df_remnum.append({'url_rem': content, 'no_num': nonum}, ignore_index=True)
print("done2")
df_new = pd.DataFrame(columns=['no_num','Stopwrod_removed'])
for i in range(10):
    non_emo=df_remnum['no_num'].iloc[i]
    letters_only = re.sub("[^a-zA-Z]"," ",str(df_remnum['no_num'].iloc[i]))
    remStopword=removeStopwords(letters_only)
    list1=[non_emo,remStopword]
    df_new = df_new.append({'no_num': non_emo, 'Stopword_removed': remStopword}, ignore_index=True)

print("done3")
df_stem = pd.DataFrame(columns=['en_contentst','Stemming'])
for i in range(10):
    content=df_new['Stopword_removed'].iloc[i]
    stem=stemming(df_new['Stopword_removed'].iloc[i])
    list1=[content,stem]
    df_stem = df_stem.append({'en_contentst': content, 'Stemming': stem}, ignore_index=True)
print("done4")
df_lem = pd.DataFrame(columns=['Stemmingle','lemmatization'])
for i in range(10):
    content=df_stem['Stemming'].iloc[i]
    lem=stemming(df_stem['Stemming'].iloc[i])
    list1=[content,lem]
    df_lem = df_lem.append({'Stemmingle': content, 'lemmatization': lem}, ignore_index=True)
print("done5")
df_lower = pd.DataFrame(columns=['lemmatizationtl','lower'])
for i in range(10):
    content=df_lem['lemmatization'].iloc[i]
    low=tolower(df_lem['lemmatization'].iloc[i])
    list1=[content,low]
    df_lower = df_lower.append({'lemmatizationtl': content, 'lower': low}, ignore_index=True)
print("done6")
df_rempunc = pd.DataFrame(columns=['lemmatizationtlp','no_punc'])
for i in range(10):
    content=df_lower['lemmatizationtl'].iloc[i]
    nopun=rem_punctuation(df_lower['lemmatizationtl'].iloc[i])
    list1=[content,nopun]
    df_rempunc = df_rempunc.append({'lemmatizationtlp': content, 'no_punc': nopun}, ignore_index=Tr
ue)
print("done7")
df_rempunc.to_csv("G://preprocessed_FULL.csv", sep=',',index=False, encoding= 'utf-8')
```

```
done1
'str' object has no attribute 'remove'
'str' object has no attribute 'remove'
'str' object has no attribute 'remove'
'str' object has no attribute 'remove'
done2
done3
done4
done5
done6
done7
```

In [89]:

```
frames=[translated,df_urlrem, df_new, df_stem, df_lem, df_lower, df_remnum, df_rempunc]
```

In [93]:

```
Train=Train.reset_index(inplace=True, drop=True)
```

In [104]:

In [104]:

```
df_rempunc
```

Out[104]:

	lemmatizationtlp	no_punc
0	[lijiang, tourism, sz, stock, stock, financ, i...	[lijiang, tourism, sz, stock, stock, financ, i...
1	[chen, ling, ding, red, envelop, to, make, mon...	[chen, ling, ding, red, envelop, to, make, mon...
2	[taobao, these, sucker, industri, ga, fire, no...	[taobao, these, sucker, industri, ga, fire, no...
3	[aspect, can, not, say, you, know, everyth, la...	[aspect, can, not, say, you, know, everyth, la...
4	[over, zhang]	[over, zhang]
5	[lifetim, My, favorit, up, with, the, main, bo...	[lifetim, my, favorit, up, with, the, main, bo...
6	[On, uniqlo, dress, room, sound, insul, is, go...	[on, uniqlo, dress, room, sound, insul, is, go...
7	[So, ordinari, everyday, scene, but, also, how...	[so, ordinari, everyday, scene, but, also, how...
8	[overh, of, red, xxiii, tanggua, children, sti...	[overh, of, red, xxiii, tanggua, children, sti...
9	[there, are, good, thing, to, share, with, you...	[there, are, good, thing, to, share, with, you...

In [102]:

```
Train
```

In [30]:

```
Train.to_csv("G://preprocessed2L.csv", sep=',',index=False, encoding= 'utf-8')
```

In [106]:

```
df=pd.read_csv("G://preprocessed_FULL.csv")
```

In [107]:

```
df
```

Out[107]:

	lemmatizationtlp	no_punc
0	['lijiang', 'tourism', 'sz', 'stock', 'stock',...	['lijiang', 'tourism', 'sz', 'stock', 'stock',...
1	['chen', 'ling', 'ding', 'red', 'envelop', 'to...	['chen', 'ling', 'ding', 'red', 'envelop', 'to...
2	['taobao', 'these', 'sucker', 'industri', 'ga'...	['taobao', 'these', 'sucker', 'industri', 'ga'...
3	['aspect', 'can', 'not', 'say', 'you', 'know',...	['aspect', 'can', 'not', 'say', 'you', 'know',...
4	['over', 'zhang']	['over', 'zhang']
5	['lifetim', 'My', 'favorit', 'up', 'with', 'th...	['lifetim', 'my', 'favorit', 'up', 'with', 'th...
6	['On', 'uniqlo', 'dress', 'room', 'sound', 'in...	['on', 'uniqlo', 'dress', 'room', 'sound', 'in...
7	['So', 'ordinari', 'everyday', 'scene', 'but',...	['so', 'ordinari', 'everyday', 'scene', 'but',...
8	['overh', 'of', 'red', 'xxiii', 'tanggua', 'ch...	['overh', 'of', 'red', 'xxiii', 'tanggua', 'ch...
9	['there', 'are', 'good', 'thing', 'to', 'share...	['there', 'are', 'good', 'thing', 'to', 'share...

Team 5A09 DMA Course: Project Sina Weibo Interaction Prediction Challenge

Determining Statistical Factors

Authors: Apoorva Malemath, Arundati Dixit, Ashish Kar, Deepti Nadkarni

In [1]:

```
import import_ipynb
import pandas as pd
from genUidStat import loadData,genUidStat
from evaluation import precision
from runTime import runTime
```

```
importing Jupyter notebook from genUidStat.ipynb
importing Jupyter notebook from evaluation.ipynb
importing Jupyter notebook from runTime.ipynb
```

Information on Loaded Modules

genUidStat.ipynb

Loads train and predict dataset as well as generated UID stats with statistical measures for further analysis

evaluation.ipynb

evaluation function accoding to official rule:
http://tianchi.aliyun.com/competition/information.htm?spm=5176.100067.5678.2.Grh4pl&raceId=5

runTime.ipynb

A basic run time function for run time calculation

Prerequisites

1. Generate UID Stats with statistical measures for FCL

We will find Mean, Median, Max and Min of Forward, Comment and Likes for every unique UID in train dataset for our further statistical analysis

In [1]:

```
df=pd.read_csv("train_uid_stat.csv")
```

```
---------------------------------------------------------------------------
NameError                                 Traceback (most recent call last)
<ipython-input-1-e05e2296a0b5> in <module>()
----> 1 df=pd.read_csv("train_uid_stat.csv")

NameError: name 'pd' is not defined
```

Example For UID stats

Say in train dataset, For UID x there are two MID(ie two posts):

Train Dataset:

UID Stats:

Now Consider that same user has 4 mids in predict dataset, so prediction of FCL by factor "mean" will be as follows:

Predict Dataset

Similary by factor "max" :

Predict Dataset

In [10]:

```
df.head(50)
```

Out[10]:

	u_id	forward_min	forward_max	forward_median	fo
0	000127c6126e2b0019f255ed21ac1cb7	0	1	0	0
1	0001565a5edece1669577e2ace9a6a3d	0	0	0	0

	uc_id	forward_min	forward_max	forward_median	fo
2	00033a6513b86b2705de9ffa9d37ffb6	0	0	0	0
3	0004fe2742507420eaa73e119dc83ac5	0	6	0	0
4	000c663a24a2f91f4ba156fcd4f8b9f2	0	1	0	0
5	000ce19d2fccb1f22421bec50bf25b08	0	0	0	0
6	000d7bf7406392b2212dfb4fe907d946	0	0	0	0
7	0012edb614365800e901c7f2b47e9129	0	0	0	0
8	001349a053bdecf1a71960f29288ced1	0	0	0	0
9	0015c42ec93854687a258a7f170c6acf	0	0	0	0
10	0018b27ecc1370e4208b6b2f175e6651	0	0	0	0
11	001d259734bccab73fdc373803c1fcd3	0	8	0	1
12	001d458a43e7fd1d9f8e2eba54d5d2fe	0	0	0	0
13	001e00fddab72bf7e6be3455e199904a	0	1	0	0
14	001fe802d7b8a3f5782b25acf0410440	0	0	0	0
15	00203b1aa005f8e374c1e681f5c2ba20	0	0	0	0
16	00212e7163d4aa64f2d7956a35027aa3	0	0	0	0
17	00218f81fb7713a915d74a1d44f95b0b	0	0	0	0
18	00235ea45eb587598e730a01e0c95435	0	0	0	0
19	0024afeb386597432b7fe0d0a4bd9520	0	0	0	0
20	0025fdc5741eb5afeeda3c90b8b35450	0	0	0	0
21	0026416dd4943c8b119896c1e824227b	0	0	0	0
22	0026bf3bb797997289aa4bd80d2965f6	0	0	0	0
23	002c120b0b15d1f749c8d07d54ea6420	0	0	0	0
24	002d3fc1ec528dcc11e7bab8ddc12ffd	0	0	0	0
25	002fffb6806ef6a0d9507d6038e11fce	0	0	0	0
26	0030024edee05cfcfd490fecb30ce8f8	0	0	0	0
27	0030649a18ba85357aa55953cd22c366	0	4	0	1
28	003242bfec263f03e8c7e8df606d961b	0	0	0	0
29	00366f1ae39b881bdb2ba8687a4c912a	0	0	0	0
30	0036be364a3cc98d55ebf494f675b719	0	0	0	0
31	0037104157ee5dd0987be03d750f0fd3	0	5	0	0
32	003ad9ea4f54e3e0a026ebdc5e62a8a6	0	0	0	0
33	003c25fdfbca4e966b10ce51f9ba03f7	0	0	0	0
34	003dcd8f6f3e00e39ee62e9736d1f5b2	0	0	0	0
35	003e42ef93574546b6e1726e1e0217f8	0	0	0	0

	u_id	forward_min	forward_max	forward_median	fo
35	005e42cf955745460b6c1726e1e021f8	0	0	0	0
36	00406a0f0da6c5c6129fcb9a34c25fb9	0	1	0	0
37	0042372f4af23d7f58847774c2b890ef	0	0	0	0
38	00424c292ede415019a992a18b95d0c1	0	0	0	0
39	00442eeb3443714e759190887f121b3a	0	0	0	0
40	00453c524df35f85fe30088fab42211c	0	1	0	0
41	0048c95badfc4e78939ee6dbaf846e83	0	0	0	0
42	0049ace5cf556923a4d2fae28df69412	0	2	0	0
43	004b28c230de2e046213fb7d66357240	0	0	0	0
44	004ded7ec093ef68e3b8c5de725e5963	0	1	0	0
45	004fd2e20d20a88a9ea09bafd8ea1365	0	0	0	0
46	005401d2c80b0df8622baf3863f6ebd1	0	5	0	0
47	00572eb39291a2c26c6fdb3efae9c595	0	2	0	0
48	005956e5440af20d49160cc6b4c3f7c8	0	2	0	0
49	005989ae18cab8d7896bc1ab84dcfe88	0	0	0	0

2. Use Offical Formula to dertermine accuracy for statistical factors

Predict with fixed Value

1. Default Values

About 80% of the training data are: 0 0 0 (forward_count,comment_count,like_count) and also, 96% of uid in predict dataset is present in train dataset, for remaining 4% which are new, we need some default values. inspired by this, we try some fixed value for all uid:

Function to take Fixed FCL Values, Give Accuracy and Generate Predicted FCL

In [2]:

```
@runTime
def predict_with_fixed_value(forward,comment,like,submission=True):
 # type check
```

```
 if isinstance(forward,int) and isinstance(forward,int) and isinstance(forward,int):
  pass
 else:
  raise TypeError("forward,comment,like should be type 'int' ")

 traindata,testdata = loadData()

 #score on the training set
 train_real_pred = traindata[['forward_count','comment_count','like_count']]
 train_real_pred['fp'],train_real_pred['cp'],train_real_pred['lp'] = forward,comment,
like
 print ("Score on the training set:{0:.2f}%".format(precision(train_real_pred.values
)*100))

 #predict on the test data with fixed value, generate submission file
 if submission:
  test_pred = testdata[['u_id','m_id']]
  test_pred['fp'],test_pred['cp'],test_pred['lp'] = forward,comment,like

  result = []
  filename = "weibo_predict_{}_{}_{}.txt".format(forward,comment,like)
  for _,row in test_pred.iterrows():
   result.append("{0}\t{1}\t{2},{3},{4}\n".format(row[0],row[1],row[2],row[3],row[4])
)
  f = open(filename,'w')
  f.writelines(result)
  f.close()
  print ('generate submission file "{}"'.format(filename))
```

2. UID Statistics (Mean, Max, Min, Median)

Another wise solution is to predict respectively with uid's statistics(E.g mean,median) , their score on the training data:

Function to take Statistical Factor, Give Accuracy and Generate Predicted FCL

In [3]:

```
@runTime
def predict_with_stat(stat="median",submission=True):
 """
 stat:
  string
  min,max,mean,median
 """
 stat_dic = genUidStat()
 traindata,testdata = loadData()
```

```
 #get stat for each uid
 forward,comment,like = [],[],[]
 for uid in traindata['u_id']:
  if uid in stat_dic:
   forward.append(int(stat_dic[uid]["forward_"+stat]))
   comment.append(int(stat_dic[uid]["comment_"+stat]))
   like.append(int(stat_dic[uid]["like_"+stat]))
  else:
   forward.append(0)
   comment.append(0)
   like.append(0)
 #score on the training set
 train_real_pred = traindata[['forward_count','comment_count','like_count']]
 train_real_pred['fp'],train_real_pred['cp'],train_real_pred['lp'] = forward,comment,
like
 print ("Score on the training set:{0:.2f}%".format(precision(train_real_pred.values
)*100))

 #predict on the test data with fixed value, generate submission file
 if submission:
  test_pred = testdata[['u_id','m_id']]
  forward,comment,like = [],[],[]
  for uid in testdata['u_id']:
   if uid in stat_dic:
    forward.append(int(stat_dic[uid]["forward_"+stat]))
    comment.append(int(stat_dic[uid]["comment_"+stat]))
    like.append(int(stat_dic[uid]["like_"+stat]))
   else:
    forward.append(0)
    comment.append(0)
    like.append(0)


  test_pred['fp'],test_pred['cp'],test_pred['lp'] = forward,comment,like

  result = []
  filename = "weibo_predict_{}.txt".format(stat)
  for _,row in test_pred.iterrows():
   result.append("{0}\t{1}\t{2},{3},{4}\n".format(row[0],row[1],row[2],row[3],row[4])
)
  f = open(filename,'w')
  f.writelines(result)
  f.close()
  print ('generate submission file "{}"'.format(filename))
```

Ready to check accuracy of various statistical factors.....

In [27]:

```
if __name__ == "__main__":
```

```
if __name__ == "__main__":
  predict_with_stat(stat="median",submission=True)
```

```
G:\Anaconda\lib\site-packages\ipykernel_launcher.py:24: SettingWithCopyWarning:
A value is trying to be set on a copy of a slice from a DataFrame.
Try using .loc[row_indexer,col_indexer] = value instead

See the caveats in the documentation: http://pandas.pydata.org/pandas-docs/stable/ind
exing.html#indexing-view-versus-copy
```

```
Score on the training set:32.73%
```

```
G:\Anaconda\lib\site-packages\ipykernel_launcher.py:42: SettingWithCopyWarning:
A value is trying to be set on a copy of a slice from a DataFrame.
Try using .loc[row_indexer,col_indexer] = value instead

See the caveats in the documentation: http://pandas.pydata.org/pandas-docs/stable/ind
exing.html#indexing-view-versus-copy
```

```
generate submission file "weibo_predict_median.txt"
predict_with_stat run time: 135.31s
```

In [29]:

```
if __name__ == "__main__":
  predict_with_fixed_value(0,1,1,submission=True)
```

```
G:\Anaconda\lib\site-packages\ipykernel_launcher.py:13: SettingWithCopyWarning:
A value is trying to be set on a copy of a slice from a DataFrame.
Try using .loc[row_indexer,col_indexer] = value instead

See the caveats in the documentation: http://pandas.pydata.org/pandas-docs/stable/ind
exing.html#indexing-view-versus-copy
  del sys.path[0]
```

```
Score on the training set:26.43%
```

```
G:\Anaconda\lib\site-packages\ipykernel_launcher.py:19: SettingWithCopyWarning:
A value is trying to be set on a copy of a slice from a DataFrame.
Try using .loc[row_indexer,col_indexer] = value instead

See the caveats in the documentation: http://pandas.pydata.org/pandas-docs/stable/ind
exing.html#indexing-view-versus-copy
```

```
generate submission file "weibo_predict_0_1_1.txt"
predict_with_fixed_value run time: 68.95s
```

In [4]:

```
if __name__ == "__main__":
  predict_with_stat(stat="mean",submission=True)
```

```
G:\Anaconda\lib\site-packages\ipykernel_launcher.py:24: SettingWithCopyWarning:
A value is trying to be set on a copy of a slice from a DataFrame.
Try using .loc[row_indexer,col_indexer] = value instead

See the caveats in the documentation: http://pandas.pydata.org/pandas-docs/stable/ind
exing.html#indexing-view-versus-copy
```

```
Score on the training set:30.17%
```

```
G:\Anaconda\lib\site-packages\ipykernel_launcher.py:42: SettingWithCopyWarning:
A value is trying to be set on a copy of a slice from a DataFrame.
Try using .loc[row_indexer,col_indexer] = value instead

See the caveats in the documentation: http://pandas.pydata.org/pandas-docs/stable/ind
exing.html#indexing-view-versus-copy
```

```
generate submission file "weibo_predict_mean.txt"
predict_with_stat run time: 132.35s
```

In [5]:

```
if __name__ == "__main__":
  predict_with_stat(stat="max",submission=True)
```

```
G:\Anaconda\lib\site-packages\ipykernel_launcher.py:24: SettingWithCopyWarning:
A value is trying to be set on a copy of a slice from a DataFrame.
Try using .loc[row_indexer,col_indexer] = value instead

See the caveats in the documentation: http://pandas.pydata.org/pandas-docs/stable/ind
exing.html#indexing-view-versus-copy
```

```
Score on the training set:7.13%
```

```
G:\Anaconda\lib\site-packages\ipykernel_launcher.py:42: SettingWithCopyWarning:
A value is trying to be set on a copy of a slice from a DataFrame.
Try using .loc[row_indexer,col_indexer] = value instead

See the caveats in the documentation: http://pandas.pydata.org/pandas-docs/stable/ind
exing.html#indexing-view-versus-copy
```

```
generate submission file "weibo_predict_max.txt"
predict_with_stat run time: 132.56s
```

In [6]:

```
if __name__ == "__main__":
  predict_with_stat(stat="min",submission=True)
```

```
G:\Anaconda\lib\site-packages\ipykernel_launcher.py:24: SettingWithCopyWarning:
A value is trying to be set on a copy of a slice from a DataFrame.
```

```
Try using .loc[row_indexer,col_indexer] = value instead

See the caveats in the documentation: http://pandas.pydata.org/pandas-docs/stable/ind
exing.html#indexing-view-versus-copy
```

```
Score on the training set:26.07%
```

```
G:\Anaconda\lib\site-packages\ipykernel_launcher.py:42: SettingWithCopyWarning:
A value is trying to be set on a copy of a slice from a DataFrame.
Try using .loc[row_indexer,col_indexer] = value instead

See the caveats in the documentation: http://pandas.pydata.org/pandas-docs/stable/ind
exing.html#indexing-view-versus-copy
```

```
generate submission file "weibo_predict_min.txt"
predict_with_stat run time: 131.45s
```

In [7]:

```
if __name__ == "__main__":
  predict_with_fixed_value(0,0,0,submission=True)
```

```
G:\Anaconda\lib\site-packages\ipykernel_launcher.py:13: SettingWithCopyWarning:
A value is trying to be set on a copy of a slice from a DataFrame.
Try using .loc[row_indexer,col_indexer] = value instead

See the caveats in the documentation: http://pandas.pydata.org/pandas-docs/stable/ind
exing.html#indexing-view-versus-copy
  del sys.path[0]
```

```
Score on the training set:25.98%
```

```
G:\Anaconda\lib\site-packages\ipykernel_launcher.py:19: SettingWithCopyWarning:
A value is trying to be set on a copy of a slice from a DataFrame.
Try using .loc[row_indexer,col_indexer] = value instead

See the caveats in the documentation: http://pandas.pydata.org/pandas-docs/stable/ind
exing.html#indexing-view-versus-copy
```

```
generate submission file "weibo_predict_0_0_0.txt"
predict_with_fixed_value run time: 72.05s
```

In [4]:

```
if __name__ == "__main__":
  predict_with_fixed_value(0,0,1,submission=True)
```

```
G:\Anaconda\lib\site-packages\ipykernel_launcher.py:13: SettingWithCopyWarning:
A value is trying to be set on a copy of a slice from a DataFrame.
Try using .loc[row_indexer,col_indexer] = value instead

See the caveats in the documentation: http://pandas.pydata.org/pandas-docs/stable/ind
```

```
exing.html#indexing-view-versus-copy
  del sys.path[0]
```

```
Score on the training set:26.11%
```

```
G:\Anaconda\lib\site-packages\ipykernel_launcher.py:19: SettingWithCopyWarning:
A value is trying to be set on a copy of a slice from a DataFrame.
Try using .loc[row_indexer,col_indexer] = value instead

See the caveats in the documentation: http://pandas.pydata.org/pandas-docs/stable/ind
exing.html#indexing-view-versus-copy
```

```
generate submission file "weibo_predict_0_0_1.txt"
predict_with_fixed_value run time: 66.76s
```

In [5]:

```
if __name__ == "__main__":
  predict_with_fixed_value(0,1,0,submission=True)
```

```
G:\Anaconda\lib\site-packages\ipykernel_launcher.py:13: SettingWithCopyWarning:
A value is trying to be set on a copy of a slice from a DataFrame.
Try using .loc[row_indexer,col_indexer] = value instead

See the caveats in the documentation: http://pandas.pydata.org/pandas-docs/stable/ind
exing.html#indexing-view-versus-copy
  del sys.path[0]
```

```
Score on the training set:25.95%
```

```
G:\Anaconda\lib\site-packages\ipykernel_launcher.py:19: SettingWithCopyWarning:
A value is trying to be set on a copy of a slice from a DataFrame.
Try using .loc[row_indexer,col_indexer] = value instead

See the caveats in the documentation: http://pandas.pydata.org/pandas-docs/stable/ind
exing.html#indexing-view-versus-copy
```

```
generate submission file "weibo_predict_0_1_0.txt"
predict_with_fixed_value run time: 68.40s
```

In [6]:

```
if __name__ == "__main__":
  predict_with_fixed_value(1,0,0,submission=True)
```

```
G:\Anaconda\lib\site-packages\ipykernel_launcher.py:13: SettingWithCopyWarning:
A value is trying to be set on a copy of a slice from a DataFrame.
Try using .loc[row_indexer,col_indexer] = value instead

See the caveats in the documentation: http://pandas.pydata.org/pandas-docs/stable/ind
exing.html#indexing-view-versus-copy
  del sys.path[0]
```

```
Score on the training set:22.22%
```

```
G:\Anaconda\lib\site-packages\ipykernel_launcher.py:19: SettingWithCopyWarning:
A value is trying to be set on a copy of a slice from a DataFrame.
Try using .loc[row_indexer,col_indexer] = value instead

See the caveats in the documentation: http://pandas.pydata.org/pandas-docs/stable/ind
exing.html#indexing-view-versus-copy
```

```
generate submission file "weibo_predict_1_0_0.txt"
predict_with_fixed_value run time: 67.65s
```

In [7]:

```
if __name__ == "__main__":
  predict_with_fixed_value(1,0,1,submission=True)
```

```
G:\Anaconda\lib\site-packages\ipykernel_launcher.py:13: SettingWithCopyWarning:
A value is trying to be set on a copy of a slice from a DataFrame.
Try using .loc[row_indexer,col_indexer] = value instead

See the caveats in the documentation: http://pandas.pydata.org/pandas-docs/stable/ind
exing.html#indexing-view-versus-copy
  del sys.path[0]
```

```
Score on the training set:23.44%
```

```
G:\Anaconda\lib\site-packages\ipykernel_launcher.py:19: SettingWithCopyWarning:
A value is trying to be set on a copy of a slice from a DataFrame.
Try using .loc[row_indexer,col_indexer] = value instead

See the caveats in the documentation: http://pandas.pydata.org/pandas-docs/stable/ind
exing.html#indexing-view-versus-copy
```

```
generate submission file "weibo_predict_1_0_1.txt"
predict_with_fixed_value run time: 69.00s
```

In [8]:

```
if __name__ == "__main__":
  predict_with_fixed_value(1,1,0,submission=True)
```

```
G:\Anaconda\lib\site-packages\ipykernel_launcher.py:13: SettingWithCopyWarning:
A value is trying to be set on a copy of a slice from a DataFrame.
Try using .loc[row_indexer,col_indexer] = value instead

See the caveats in the documentation: http://pandas.pydata.org/pandas-docs/stable/ind
exing.html#indexing-view-versus-copy
  del sys.path[0]
```

```
Score on the training set:21.28%
```

```
G:\Anaconda\lib\site-packages\ipykernel_launcher.py:19: SettingWithCopyWarning:
A value is trying to be set on a copy of a slice from a DataFrame.
Try using .loc[row_indexer,col_indexer] = value instead
```

```
Try using .loc[row_indexer,col_indexer] = value instead

See the caveats in the documentation: http://pandas.pydata.org/pandas-docs/stable/indexing.html#indexing-view-versus-copy
```

```
generate submission file "weibo_predict_1_1_0.txt"
predict_with_fixed_value run time: 71.24s
```

In [9]:

```
if __name__ == "__main__":
  predict_with_fixed_value(1,1,1,submission=True)
```

```
G:\Anaconda\lib\site-packages\ipykernel_launcher.py:13: SettingWithCopyWarning:
A value is trying to be set on a copy of a slice from a DataFrame.
Try using .loc[row_indexer,col_indexer] = value instead

See the caveats in the documentation: http://pandas.pydata.org/pandas-docs/stable/indexing.html#indexing-view-versus-copy
  del sys.path[0]
```

```
Score on the training set:10.18%
```

```
G:\Anaconda\lib\site-packages\ipykernel_launcher.py:19: SettingWithCopyWarning:
A value is trying to be set on a copy of a slice from a DataFrame.
Try using .loc[row_indexer,col_indexer] = value instead

See the caveats in the documentation: http://pandas.pydata.org/pandas-docs/stable/indexing.html#indexing-view-versus-copy
```

```
generate submission file "weibo_predict_1_1_1.txt"
predict_with_fixed_value run time: 69.53s
```

Overall Results

Current Weibo Sina Interation Prediction Leaderboard by Aliyun.com

References

https://github.com/wepe/AliTianChi/tree/master (A Statistical Analysis on Weibo Sina Interaction Prediction 2014 Challenge)

3.BAG OF WORDS

We convert text to a numerical representation called a feature vector. A feature vector can be as simple as a list of numbers.

The bag-of-words model is one of the feature extraction algorithms for text.

1.The first step in this model is defining the vocabulary

2.The second step is to convert sentences into a frequency vector based on the vocabulary.

In [11]:

```
#Reading data from document
import pandas as pd
df_pre=pd.read_csv("E:\\DMA_PRE\\PREPROCESSED.csv")
df=pd.read_csv("E:\\DMA_PRE\\PREPROCESSED.csv")
```

In [12]:

```
from sklearn import linear_model
```

In [13]:

```
#Adjustments to be done for the data
df['content']=df['content'].str.replace(",", "")
```

In [14]:

```
df['content']=df['content'].str.replace("'","")
```

In [15]:

```
#creating a list for all content
l=[]
for i in range(0,10000):
    l.append(df['content'].iloc[i])
l
```

Out[15]:

```
['[lijiang tourism sz002033 stock stock financ invest bank recommend baoy share divid
half earn princip group]',
 '[chen ling ding red envelop make money abil grab red envelop fight technolog i grab
red envelop cash issu chen ding ling kan hongyan burst happi valentin "s" day fan
togeth around red envelop â•® â¯ 3â¯ â•\\xadhttp tcnrzdivjf]',
 '[taobao sucker industri ga fire send children nima blind]',
 '[aspect say know everyth laugh cri]',
 '[over zhang]',
 '[lifetim my favorit main bodi hook three togeth burst happi black tong googl]',
 '[on uniqlo dress room sound insul good bad share know almost]',
 '[so ordinari everyday scene also much longer accumul one day get look pictur i feel
```

[so ordinari everyday scene also much longer accumul one day got look pictur i feel burst nostalgia age rememb pain accompani]',
 '[overh red xxiii tanggua children stick grab red envelop win i grab littl _ big red overh wang xiansen issu year goat yet come good luck come red envelop tri luck http t cnrzggwg6]',
 '[there good thing share flash note note todo best chine softwar go download]',
 '[microsoft azur machin learn exampl machin learn analysi network intru log practic hand big data `` secur field began hot public report data storag data transfer data r etriev data visual map attack even compon also limit similar data analysi topn descri pt statist]',
 '[life string life mani trial let us pain grow pain like heart like world music also pain like soul sail morn open water lili monday "s" bless hope bring posit energi wee k good night rose]',
 '[readi test first cluster data]',
 '[cream chee cake butter milk mix beaten egg yolk milk sugar flour mix 31 mix lemon juic protein pass 5 preheat rice cooker ladl half protein rest protein mix mix turn 6 help spread butter rice cooker inner wall semiwet towel cover outlet began cook stew 10min 5min stew 20min]',
 '[it wuhan drug drive car driver detain secur xinhua wuhan june report feng guodong report learn wuhan public secur bureau inspect law enforc recent driver check car tak e driver ma fruit]',
 '[thi year matter line look articl share new]',
 '[i share inspir small articl]',
 '[anni "s" red envelop make money abil grab red envelop fight technolog i grab red e nvelop cash issu anni alipay wallet good luck year goat open you tri luck â•® â¯ 3â¯ â•\\xadhttp tcnrzkxypd]',
 '[haneda airport]',
 '[share singl kutluk song `` piano play soul netea cloud music]',
 '[share album netea cloud music]',
 '[read issu clear recent realli protect mode tri help block silli trivial process `` state ah i wonder i open custom pattern compani respond outstand colleg student rebel li if employ compani regress point outrag mean one thing share know almost]',
 '[sogou laboratori text classif corpu]',
 '[dilut cool cdrom access entri master `` athero cultur iso ckook cd book cool dilut book primer learn access the book divid chapter introduc basic knowledg offic softwar databa access learn content behind lay foundat reader introduc basic creat access cla ss object rw5vgdt]',
 '[publish blog p2p network financ manag platform rooki need know net loan industri d oorway `` stock invest like p2p network financ manag platform investor also larg reta il point definit larg name hand rel larg amount money platform heavi invest investor retail vote]',
 '[hc m & a onlin zhongguancun industri logic share tech2ipo]',
 '[ruin alcohol boss after dish recommend tapa home desper amway finest wine everi ti me i drank two can soda compromi]',
 '[veri backward countri i use someon el chemic sensor i doctor know peopl behind ind ustri engin engin doctor doctor domest chine academi scienc abil develop complex medi c devic shadow develop pet cheap dr zhang qiang founder siemen doctor electron engin lot peopl work hospit i even write code internship understand]',
 '[xiaohui at1989 us share beat king realli lost lot last year qq music but shuashuai microphon almost lost purpo face_with_tears_of_joy face_with_tears_of_joy fortun catc h film keep tempo feel littl embarrass still handsom jump thumbs_up thumbs_up varieti scene worthi superstar unit state beat shoot]',
 '[direct song share grinning face with sweat share accord zhao "s" home dinner `` ne

tea cloud music]',
 '[exhaust life emot intellig]',
 '[end world tree read best way life worthi white hair back desper everi student]',
 '[it microbusi busi innov new channel want concern kelan diamond engag new capit coo
per capit side requir found team must continu go back new entrepreneuri team investor
account]',
 '[quick taxi shake happi do think next taxi subsid quickli upgrad latest version fas
t taxi app new year "s" eve day seventh day 1200 1900 kidney million voucher imp
upon dip festiv deft receiv coupon http tcnrwp4k4v]',
 '[after reform process leadership]',
 '[ah let wall collap]',
 '[i alway felt free `` word practic signif someon ask is empti `` answer free `` rea
lli mean free rather gain greater done thing incom i will thing time question well pr
epar give emot materi reward i think lot thing lot easier]',
 '[share wuhan often stay hot toxic heavi peopl often stay late wang plu weather hot
hot motiv heat outsid bodi accumul lot heat toxin mouth pain eye often red yellow uri
n symptom healthcar wax gourd loofah cool role clear away heat melon gourd ad amount
water fri addit small amount salt add season]',
 '[hua teach sleep four command difficult get older share headlin today]',
 '[tomcatã‰ã¨ã-ãƒhttp ã— ã ” â¶â¯ãœã¸ ã— âªãžâªhttp â· ãƒãžãš ãžã ’ â± â¾â² â » ãšã‡â
²ã‹ã „ ã± 51ctoâ¼â¼ãšãµâ² copyright â¿ â€]',
 '[big data architectur internet common scenario resolv]',
 '[today live support educ south china agricultur univ erp research societi countrysi
d come end after ten day train student "" interest reduc game technolog becom increas
ingli matur deepli move passion everi one us we countrysid team harmoni energet look
student will mind inevit sad huanong youth]',
 '[share pictur]',
 '[publish articl reproduc bowen reserv xu yili the sharp drop quasitruth ``]',
 '[dou wei abject `` behind entertain channel _ _ phoenix]',
 '[penang submicro newspap]',
 '[good voic record sing listen address]',
 '[ke zhendong kai red envelop money abil grab red envelop fight technolog i grab red
envelop cash ke zhendong kai microsurgeri pro issu togeth burst happi valentin "s" da
y fan togeth around red envelop â•® â¯ 3â¯ â•\\xadhttp tcnrzgojmj]',
 '[comment pinellia song springmvc l bowen work `` spring mvc demo tutori sourc code
download http tcnrp9nahl `` view origin http tcnztyprna]',
 '[plum _ big break fire door manufactur lalalolol linshao fan club isshi_ want life
piec qiu huijun congratul dear user congratul becom sina weibo `` identifi seventh an
niversari `` offsit activ second prize lucki user plea visit http tcnrabjizn check gi
ft]',
 '[due forc majeur develop spare time aosc os3 beta bounc month new relea time plea w
ait notif aosc linux chine geek]',
 '[the world small matter dodg accid intersect sucker dig booger dig booger dig booge
r]',
 '[i "m" asian song list rainjihoon highest chart song good music need action support
come favorit song]',
 '[child mother pick mountain back sever flower white flower insert instal water plas
tic bottl fill summer now i know good name call gardenia]',
 '[* get start function program python *
downloadfunctionalprogramminginpythonitishardtogetaconsistentopinionon justwhatfunctio
alprogrammingi ev http tcnrlthloc]',
 '[thi twoday mobil phone market happi ah ha ha]',

'[thi piec i "ve" seen technic content]',
'[hadoop ever dream recoveri hbase ecosystem glori hand want dream think right thing right way]',
'[postgi wall i wish offspr py]',
'[share track netea cloud music]',
'[it commentari appl "s" wise give tv review recent media report appl given effort d evelop brand tv product senior technolog analyst dawson jandawson wrote marketwatch p oint develop appl tv pay econom account much less continu develop appl]',
'[the system run line two month instabl method usabl tri effect still ask old man co de review found cau timeout paramet therefor best approach ad bunch printf debug gdb hard look code]',
'[angularmateri shake giant crater]',
'[winter fell love partner get rid summer hover winter spring worship worship]',
'[network applic propaganda public awar network secur survey `` relea last week show launch ceremoni network secur awar peopl optimist and similarli ordinari peopl famili ar industri inform secur situat also poor even urgent person properti loss face may i nvolv bodili injuri signif social econom impact]',
'[after listen whole person good it seem i realli understand art]',
'[valentin "s" day microblog king confess make money abil grab red envelop fight tec hnolog i grab red envelop cash kotex offici microblog issu open year goat luck you tr i luck â•® â¯ 3â¯ â•\\xadhttp tcnrzfogwt]',
'[origin longer use microblog sever day see microbo yesterday morn 500 sever heart s uddenli uncomfort know probabl woman like someth look microblog said found lone night fight delet qq i put microblog delet letter said similar word]',
'[share pictur read download lofter client]',
'[quick taxi voucher see happi camp iphon taxi voucher shake nonstop april 800 pm see happi camp tfboy million shake phoenix legend quick taxi voucher first collar cou pon zhang fun click receiv]',
'[guan yi teacher ponder mr marber yong mention best grasp charm rigidli adher i gue ss know mysteri]',
'[video record share microoff thin littl princess night car lovemak session share vl ook microrecord fast chip slow hand]',
'[samsung like appl that raspberri pie cell phone diy good]',
'[20150225 http tcnrwppml0 fame unicorn lowkey hor raffi krikorian interview twitter backend technolog advic first time entrepreneur what better place startup 10th annive rsari youtub youtub oral histori star daili]',
'[walmart intern exposur iphon 6s price increa $ some foreign media report intern wa lmart memo appear file nextgen iphon retail price compar iphon increa least $ therefo r 16gb contract price soar $ $ 64gb capac twoyear verizon contract unit price also in crea us $]',
'[with raid5 faulttol control use three 500gb sata hard drive store audio section ht tp tcnra5j9q]',
'[scallop punch the first day learn word]',
'[hiroko absolut pitch take play audio tour up main circuit hiroko beep beep mile mi le anim]',
'[â· scope manag manual pm principl teach becom master questionnair & interview it d eni experienc design experi life strong empathi better tri figur user thought percept desir user even user less research also identifi need make good design but problem fu zzi design inaccuraci produc % % margin error recommend]',
'[becom full stack engin often voluntari often forc project manpow shortag laugh]',
'[alibaba cloud comput big data bet tri catch amazon three year number chilean net]'
,
'[resourc share love valentin "s" day vector graphic download today share gift love

'[resourc share love valentin "s" day vector graphic download today share gift love valentin "s" day vector graphic download adob use vector graphic illustr vector tool open use materi greet card promot activ adverti themat map design materi whether pers on commerci use free]',
'[thi fish small i want steal]',
'[tian liang red envelop realli true i drew fan red envelop tian liang fast taxi pro vid fast taxi yuan red envelop `` year goat yet come firstcom good luck come tri luck http tcnrzgpsbi]',
'[share track netea cloud music]',
'[vmware popular within data center outstand second quarter perform vmware inc annou nc secondquart report earn per share financ analyst "" expect revenu growth meet sche dul]',
'[want open happi take thing hard pain http tcnrlqzya1]',
'[three great compass life hatr peopl live togeth get rid love one leav want get som eth unabl obtain get afford fit forc live wonder]',
'[a bit mean]',
'[applic three case potenti threat larg data mine big data mani big citi world metro poli algorithm use data analysi establish smart citi tokyo japan put everi car becom accur mobil data `` ea traffic congest even reduc number road accid death curb crime futur may occur even tragedi today let us look three applic case http tcnraxlyin]',
'[errata new classic new classic cultur guava fragranc `` accord context p47 first r ow third word `` read `` plea reprint correct]',
'[openstack demo kubernet openstack http tcnrw0puih]',
'[yesterday i finish lunch time express order sever cough nap colleagu feel sorri to day lunch break i decid work quietli left station hid corridor like simpl thing the n ext thing ye dri first dri which thing drag long shashi hou finish how arrang next ti me good anxieti crazi]',
'[i think i say much even feel clear iq doge fact i rough man anim happi birthday le e jun "ll" alway littl princ lee chang xuan happi birthday red_heart noth better litt l princ crown]',
'[student ticket work fortun weather good today tear]',
'[nima offic hot]',
'[deepin held april particip fifth de jure contend `` law major debatecum2015 copyri ght themat debat race debat union law colleg hubei provinc law debat leagu `` wuhan m unicip bureau cultur copyright agenc cosponsor interest junior partner readi encount to oh]',
'[dangdang bestsel bestsel second grief groceri store white night ``]',
'[also intend play outsid saw sea â€‹â€‹peopl queu secur quickli line go i expect bu si morn kunm]',
'[comput audio term audio endpoint translat some place translat audio endpoint]',
'[cover mat summer]',
'[to play pop goddess]',
'[app plan good way promot program divid app version appl io googl android two type version app mani market channel i rememb year ago see blog articl android loo channel articl light domest distribut channel hundr]',
'[0ctf final blue lotu team hot pursuit flappypig score closer closer twentythr qual ifi suspen big cheer player final victori front treasur safe blue lotu]',
'[handwrit hand letter inspir quot design mark van leeuwen choo one love phone wallp ap motiv prai come to forc]',
'[whim look first row compani "s" web design i found also someth instant there play] ',
'[eric brewer contain micro servic futur comput geek headlin csdnnet ``]',
'[come call wind call rain]',

'[i like play rain even larg question `` svg onmouseov alert]',
'[share netea news last year henan fire abstract time hornet "s" nest ``]',
'[]',
'[wed music shop care collat classif super set wed music encyclopedia thi set data t
otal divid eight seri take 200 yuan get eight seri larg classif mani small class invo
lv aspect wed activ use readi use well worth refer collect weddingrel person click vi
ew]',
'[welcom everyon materi commun wwwrrsccom everyon commit creativ commun look forward
cg enthusiast nation profess cg tutori materi share platform everyon communityba mate
ri enthusiast member focu exchang cg materi resourc tutori cover cg tutori ae templat
video clip industri softwar websit templat 3d model graphic materi book magazin enjoy
]',
'[bowen issu network inform secur 429 day `` recent capit 429 network secur day `` c
ampaign offici kick publicfac network inform secur expo held beij exhibit center the
expo support ministri public secur ministri industri inform technolog beij]',
'[realli comfort home shi shi sleep wake]',
'[android show consumpt data]',
'[i rub weibo also vim j k oper]',
'[start work dog mode keyboard realli itchi feel good look forward new challeng begi
n selfless code smirk]',
'[read trump card agent agent colleg `` machin still firework music period bloodi sc
ene handl instant delight â˜…â˜…â˜…â˜…â˜…]',
'[kobe problem configyaml accord manual written exampl error help faci]',
'[mani peopl brave hsiao go highest histori cinema tachykinin `` first day box offic
nearli million hypervari]',
'[thi twoday breath gone girl `` theori everyth `` former destin `` whiplash `` read
littl depress got rhythm sick]',
'[i watch 01 tsinghua dream innov entrepreneurship educ experi share check]',
'[depthlearn algorithm make recommend base spotifi music content paper phd candid re
servoir lab laboratori univ ghent belgium ghent univ sander dieleman written blog art
icl research interest music audio signal classif recommend level charact studi specia
l depth learn learn characterist the follow translat http tcnrw8kqb]',
'[in scenario write mmap memori file sometim produc hundr ms delay would result exce
ed servic sla kernel file write back throttl mechan hang write process result rel lar
g delay time updat old understand kernel ha ha i share red sill articl]',
'[famili guy `` broken fought million make qin king]',
'[green rice grassroot hospit cut stick busi valu primari hospit radiolog imag sever
shortag doctor doctor diagno limit level tradit pac product extend internet doctor an
ywh via internet anytim anywh access primari hospit diagnosi patient imag primari car
e doctor internet platform effect improv level practit reflect valu mhealth]',
'[]',
'[sugar funni toot sister son heart]',
'[freemark list length traver subscript nest sort lauy itey art site]',
'[network repr introduct interact onlin analysi introduct network analysi repres ``
elijah meek maya krishnan github http tcnrlqrw4h]',
'[spring festiv musts tonight]',
'[levay drink 1080p adverti shot from chi hoon funni babi eleph forc control]',
'[i like simpl effici i use weico microblog client smart night mode charact custom r
ead environ well offlin mode save traffic want experi better jab link download weico
pro http tcnrapgr4n]',
'[i share 2gua articl]',
'[thank support concern beij shanghai boss door]',

```
 '[laugh cri graduat work quickli marri children ah age small quickli settl laugh
cri alway someon say peopl upset laugh cri i listen smile face fact realli want cur p
ig]',
 '[googl technolog three major us telecommun compani reach agreement promot googl wal
let]',
 '[i answer phone twice ali mobil secur abandon sad sad]',
 '[big new year "s" fail overtak shoulder all water drown circl friend app]',
 '[qq particl credit invit user experi forc open qualif reli credit line mention open
loan qq particl way i invit particl credit code invit user experi strong technolog ne
ed tell open quota loan submit person inform use credit particul offici websit offici
mayli trend net]',
 '[]',
 '[linux use eclip cpp code brow handi featur basic static analysi vs]',
 '[* duq u20 technic analysi * http former tcnr2rpnkg http tcnrlquf0g]',
 '[quantum comput deal big data challeng china hkust first quantum machin learn machi
n learn algorithm core artifici intellig machin simul human learn behavior intellig g
ain experi past experi order improv over perform restructur intern knowledg structur
unknown event accur infer machin learn scienc engin mani field]',
 '[handian red envelop i give red envelop stuf handian $ ta send lucki red envelop
togeth hope meet luck good togeth heart]',
 '[exhibit site tilt photographi techniqu union nation tour dalian station begin staf
f readi wait excit tilt photographi feast in dalian littl friend come exhibitor]',
 '[micro philosophi six weak life angri punish peopl "s" mistak troubl tortur fault r
egret frustrat destruct past worri frighten virtual risk lone imprison selfmad prison
low selfesteem discredit use strength other]',
 '[everi pass meet landscap i water hyacinth appl android phone download http tcn8sou
hyh]',
 '[good search wikipedia syba "s" powerdesign case tool set easili use manag inform s
ystem analysi design includ almost entir process databa model design data flow made u
se powerdesign conceptu data model physic data model client may gener varieti applic
develop tool data wareh may creat structur]',
 '[configur control product node deploy]',
 '[millet strip unit state spent i abl put simpl patch panel fine realli admir i want
buy usb interfac patch panel bedroom conveni charg phone yuan realli super cheap apri
l 8th day rice millet phone entir depart well stock sale accessori over fold free sen
d million yuan red envelop millet phone realli thumb give mi five]',
 '[to make defici network secur manag system http tcnzj93ij0]',
 '[a dollar red envelop cash messi heart sinamail dare 1000 yuan red envelop sent act
iv link http tcnrzrfbfp]',
 '[line line ah `` i "m" live pepper small partner rush crowd forward help thumb]',
 '[report work week line despi mood ppt open mail care open ms mail interview nima in
tang load forc time good wast ah]',
 '[silicon valley `` final season spoiler thoma mead richard armitag talk]',
 '[it state council implement flexibl educ membership allow student drop innov entrep
reneurship xinhua beij may xinhua state council recent issu opinion deepen reform hig
her educ innov entrepreneurship `` hereinaft refer opinion `` full deploy deepen refo
rm higher educ innov entrepreneurship]',
 '[chrome version window start googl drive bundl]',
 '[anhui heli leverag sap plm enhanc research develop core compet crowd]',
 '[send payment prostitut popular alipay even got rememb password function]',
 '[share pictur]',
 '[technolog state council to encourag electr supplier channel sink stimul entreprene
```

url]',
 '[morn librari hou loo sack dad oracl 12c inmemori option analyt applic `` imo open arow sga buffer zone oracl databa oracl line data column convert storag format suppli frontend applic use analysi for olap system]',
 '[quick taxi voucher popular river lake hongyun fortuna appear way escort beef cattl your arrog taxi voucher http tcnrlzohdo]',
 '[protect king field comput 138 billion year histori univ birth less year thing tomo rrow afternoon pineappl scienc prize nobel physic prize winner georg smoot togeth exp lor topic comput univ invit dr wang jian ali baba the univ comput `` welcom crowd sig n link http tcnra61izq ali ali cloud cloud secur]',
 '[kaleidoscop netea founder ding lei eight element success diamond bachelor ding habit radio larg extent influenc father believ "s" proud profess futur becom electron electr engin recommend http t cn ra4ogz1]',
 '[partli cloudi today with high â° c low â° c via]',
 '[fit machin learn folk]',
 '[i forgot give gentl]',
 '[complet `` old oath wed process plan case light effect accompani majest music warm audienc light wed citi fulli close groom stage bride happi flower door angel dancer c hase light follow relea bubbl master ceremoni narrat child mom dad "s" babi day wande r around warmth affect grow learn understand learn toler detail zqgfb02]',
 '[choo variabl introduc basi drop basi `` translat chine origin call `` group group jihai know boundari repent save donor care ah]',
 '[in mysql check pttablechecksum pttablesync sync tabl ensur data consist postgresql similar tool pg_compar sever pgdiff]',
 '[apach mrql mrql mapreduc queri languag pronounc miracl queri process optim system largesc distribut data analysi queri support four mode mr type run hado]',
 '[readili take return sunri]',
 '[to help friend move frontend 3040w y]',
 '[noth el howl two maroon funk _ ^ show "s" voic alway heard greasi relieved_fac rel ieved_fac relieved_fac listen address http tcnrwbdspp sing record]',
 '[it huawei watch start play the screen quit good mwc2015 huawei except talkbandb2 n 1 also brought first androidwear watch huawei]',
 '[it googl led fund agricultur data compani $ million sina financ york may even news us depart agricultur data startup farmersbusinessnetwork fbn announc tuesday googl ve ntur latest round financ googleventur]',
 '[your uncl eat show affect face bowl oper]',
 '[zizhuyuan walk dinner readili beat]',
 '[skip back exhaust abdomin train right time suddenli feel veget night someth becom thing past]',
 '[special wed plan packag price quotat price list inform leaflet color page total we d set line leaflet sourc file psd format pro modifi suit practic inform plea see http tcnzqs39v4]',
 '[harass other commiss judgment judgment view detail http tcnrwhnpf5]',
 '[get technic document group daniel wrote nice describ current mood]',
 '[wrote first point open sourc compon hope like doge defici exhibit love trinea code home googdev codekk mu class open sourc group android studi group net hu kai softli t inkl languag tcahead]',
 '[]',
 '[scallop punch the first day learn word]',
 '[share feng zhiwei cultur blog bowen pictur transdisciplinari research comput lingu ist]',
 '[what word back tm kind never compromi attitud toward life]',
 '[the concept microserv larger microserv concept bit bigger optim organiz level fact

or optim concept technic factor combin made question type produc correspond solut may abl fit anoth problem phil will qcon london http tcnrabcwmg]',
 '[fast disk green offic space fast disk lucki tree water everi day catch worm harves t fruit fertil win ipad mini moonston mobil power fast disk sport water bottl award t ree fall lap small partner quickli call friend quick accept award]',
 '[i microblog reach level lv17 also three exclu opportun collect gift tyrant turn ta three _3306 ljlzzgood tong yan group come help win lotteri extra chanc for detail ple a poke â† ’ http tcnra8m1nt]',
 '[day ban bribe trench ceremoni day ban `` limit number open test 515 almost cut han d fortun i escap 0252 second success also grab 029 yuan red envelop one hundr million gamer love game masterpiec realli pride click collar red envelop]',
 '[get earli then engag]',
 '[]',
 '[io regular express]',
 '[do know printer perform print oper also save text print have heard printer penetr use penetr test skill network printer typic deploy offic network enterpri access netw ork attack use advantag take look magic attack translat 91ri team http tcnrlgwfhu]',
 '[what retreat away past futur maintain present dzongsar khyent rinpoch]',
 '[now lot domest secur vendor sell safe product sell penetr test servic i sell sell mao shield i know deploy websit secur product waf penetr test happen laugh laugh laug h laugh]',
 '[attitud chang height]',
 '[red red envelop someth internet "s" heartwarm heart i drew quick taxi provid fast taxi yuan red envelop `` red envelop someth internet come tri luck red envelop distanc happi]',
 '[share courteou test phone pharmaci realli easi click view http tcnr263zsv]',
 '[futur neusoft unieap & saca platform product smart manufactur becom import start p oint from china secur network]',
 '[wu larg cherri late mani open]',
 '[hello everyon i sign android taiwan chine network play android phone i come exchan g url http tcnrwxuwpg]',
 '[xxxxxxxxxxx]',
 '[the biggest lie histori unsubscrib repli td `` angri angri angri]',
 '[command word pass string stdin command]',
 '[we tri variou method attent game play run five kilomet everi day listen music nima eye close eye open mind code although statu correct bodi much ah thi point actual wan t wake open comput i sleep mom egg ah it "s" forc run km everi day empti energi storm]',
 '[want eat hot pot ah ah ah]',
 '[color drunk]',
 '[sina note congratul your weibo account system recogn sina seventh anniversari `` s econd prize lucki user activ plea visit http tcnr2ngsq check gift erinberrylabellevi complianc sui highsp traffic polic brigad chan fgjewkap4j]',
 '[technolog share resili tree sway draggabl html5 canva anim base tree swing html5 c anva anim html5 anim featur drag branch whole tree swing realist simul swing tree res t whole process quit realist great effect worth learn]',
 '[today "s" awar day `` "gongan" secur desk calendar]',
 '[no i give dogma i give formula let follow formula fal formula dead formula creat f ramework around framework gradual becom shackl osho i heart solut ``]',
 '[univ taobao shop sell `` univ graduat yin zhengyi kunm colleg presid ho taobao ann ounc kunm institut talent shop `` set teacher student graduat ceremoni outstand graduat becam first batch shelv babi ``]',

'[today first day 24yearold year old i feel realli differ]',
'[python penetr test sourc code open sourc project worth inten sql inject tool sqlmap dn secur monitor dnsrecon brute forc test tool patat xss exploit xsser http tcnrpc93lj web server stress test tool hulk ssl secur scanner sslyze]',
'[googl team vmware googl cloud servic station vcloud air googl vmware announc alliance cloud comput googl "s" cloud servic station vmware "s" vcloud air servic googl cloud storag bigqueri analyz big data servic data storag cloud cloud dn tightli integr `` vcloud air]',
'[* for small tabl front page applic * product sometim tabl use label want add js event content tabl tag may obtain correspond tabl tag http tcnrljdo2g]',
'[i rainjihoon present flower two valu â€‹â€‹of love feel meng meng da rice circl pro come send flower aid]',
'[denmark becom world "s" first cashless countri _ sina phone]',
'[china "s" px project list _ netea new]',
'[vm success stori with develop inform system taxat qingdao municip local taxat bureau face low util server resourc deploy flexibl enough termin equip data secur manag challeng by use vmware virtual desktop cloud solut achiev rapid deploy improv manag oper mainten level format desktop mobil cloud tax offic]',
'[thi appl "s" qualiti control pill comput program actual py kernel panic trigger directli four languag â€‹â€‹restart]',
'[share pictur]',
'[a help websit http tcnh3g98 pm project team colleagu differ countri prior knowledg public holiday countri well everyon "s" holiday arrang import holiday observ india]',
'[how youth wast `` tang xu song sunflow]',
'[anhui dialect dialect dialect map anhui provinc `` singl system complex system multipl dialect it mandarin dialect nonmandarin dialect]',
'[not pursu fussi indiff live live spectacular secur more happi less worri no matter littl long happi like everi day hungri eat tire sleep i woke smile what "s" like life put season tast subway life three thousand human thing learn laugh]',
'[wuhan busi owner ask teacher teach employ love `` church employ fall love make enterpri develop better wuhan compani hire special teacher teacher speak neither relat profess technic cour speak content relat busi manag twohour program special teach young compani learn fall love it turn compani employ love `` welfar]',
'[befor make invest thoroughli research compani text heart last lifetim ceil enterpri industri product servic becom satur near state oversuppli must clear follow compani belong correspond invest differ circumst]',
'[i share movi tenth articl]',
'[on journey life road alway comprehend honestli make ordinari person there lot talent peopl blade grass sway afraid step afraid wildfir afraid graze tenaci vital exist truth there noth aliv also mean]',
'[he said hot come man tucao microcom apostol sub tucao comic set `` see]',
'[* googl phish protect tool bit code bypass * 2015072523 1430 sourc secur broadcast read point chan favorit share xiao bian scienc googl password warn googl introduc new c april year http tcnrls6nf]',
'[scallop punch the first day learn word]',
'[i realli quit bad everi day peopl want cut hair wig]',
'[what special group guy test code black farmer summer charg agricultur higg code fighter said crazi reveng angri angri]',
'[north american search activ way]',
'[repli sherlock mongolia see time lower cost photo like wast sherlock neg matter love take pictur ah templ tower fact would like see street marketplac custom differ year i want see templ tower thing hundr year comment omelett network]',

'[run citi let "s" play togeth play citi modern citi indiff anonym human natur feedb
ack]',
'[docker still far matur docker use real system compani littl right share segmentfau
lt problem portal]',
'[such low input high return big wheel activ how littl thi unscientif time fight cha
ract turn dial award come bowl i particip activ decor day hao li day nonstop yuan pri
ze draw iphone6 `` tri luck]',
'[it first half busi sector accept onlin shop complaint from content point view cons
um complaint complaint commod consumpt among top four merchandi commun equip vehicl h
ousehold applianc consum complaint servic front rank four remot shop residenti servic
telecommun servic internet servic]',
'[i "ve" miss chanc save world never miss premium prize i particip activ idl mobil t
raffic control send red envelop `` tri luck]',
'[love love love love someon gave gift]',
'[share netea news `` bottl walk stair `` air raid `` five stitch head smash]',
'[read afternoon viewport complex we need machin understand]',
'[movi rio `` let think heart feel lone met group similar lucki thing a year ago joi
n group i feel like bird blue macaw blu suddenli met group parrot jungl rio de janeir
o do give look easili defeat understand ridicul other alon look similar long]',
'[turn light go bed day2 befor twelv good embarrass]',
'[the landlord "s" blog realli good learn have idea abstract layer increa factori fa
ctori client make full use code logic abstract layer code last subcategori entiti rep
lac specif implement would contribut expan understood subclass entiti argument commen
t jelli think c design pattern factori model ``]',
'[as long benefit enjoy liter level turn tyrant depend think littl excit happi child
ren small partner speed crowd collect rank microblog exclu courtesi http tcnrzhzgrk]'
,
'[logo wed theme psd templat artist font font design wed logo diy materi set data fo
nt templat instead font psd file format photoshop softwar open total model detail]',
'[get start vagrant document]',
'[lee jun ki hair red envelop abil make money grab red envelop fight technolog i gra
b red envelop cash lee jun ki alipay wallet issu togeth good luck year goat open you
tri luck â•® â¯ 3â¯ â•\\xadhttp tcnrzgwmiw]',
'[someon guess index cute]',
'[paragraph web crawler open sourc softwar cy163 blog]',
'[i like theoktopu "s" instagram `` unknownartist unknownsourc random nativ awesom a
rt pic imag photo 4cha http tcnrahyaqq]',
'[shijiazhuang achiev full coverag public safeti video surveil]',
'[dahe red envelop make money abil grab red envelop fight technolog i grab dahe let
fli red envelop red envelop cash sent togeth good luck year goat open you tri luck
â•® â¯ 3â¯ â•\\xadhttp tcnrzdldrt]',
'[histori amaz child watch serv see sad]',
'[i share gather chine calligraphi articl]',
'[internet agricultur explor internet local oper agricultur modern tradit recent yea
r penetr internet technolog agricultur two gradual close togeth internet becom anoth
agricultur commun focal point real estat hot]',
'[quick taxi voucher fifteenth day guess riddl beam lantern festiv go play fast car
microblog microlett repli answer guess get yuan taxi voucher three riddl chanc the an
swer publish march more lantern pack fast chip]',
'[after phone lose unlock phone phone "s" dial mode enter * `` 15digit number phone
"s" screen if user "s" phone lost immedi provid number servic provid help let phone l
ock even peopl get phone replac sim card phone use excerpt your person inform safe ``
bowen viewpoint broadview]',

'[grand canyon]',
'[run lap wyatt yue lap i ran 1002 km]',
'[thi prai unravel english tokyo ghoul op http tcnr2at009 echo echo app]',
'[move mouth red envelop wow realli rich i abl get move mouth red envelop cash i bel iev i draw]',
'[i like simpl effici i use weico microblog client smart night mode charact custom r ead environ well offlin mode save traffic want experi better jab link download weico pro http tcnrapgr4n]',
'[i bought amazon zcn "" the godfath `` trilog collector "s" edit packag read world "s" topsel novel librari kindl ebook "" us mario puzo]',
'[spring young star i believ strike onli real creat]',
'[run 150000 watch program pay rmb afford buy buy buy]',
'[i want chang buy ipad super want chang want chang nexus5 i want money grass]',
'[inventori strang android devic android system founder begin develop android system want appli digit camera acquir googl android appli smart phone http cn r28wbff]',
'[illu mood share hunt read]',
'[today met taxi driver open peopl "s" uber like industri fast assembl model]',
'[i share articl]',
'[a day ago compani architect told great god p5 % time write code p8 p9 % time met i mport identifi main issu persuad other import urgent first the career go backward hea vier commun `` live realli interest code home day open varieti write code night _]',
'[lynx year promot middl i lynx red envelop cash i believ i draw]',
'[zhengzhou univ colleg water resourc environ youth leagu school environ water resou rc zhengzhou univ student union i "m" sorri view microblog user could find press f5 r efresh tri see microblog]',
'[strang place]',
'[php use cooki access number achiev statist code _php basi _ script home]',
'[fantasi westward journey net elit enthusiast would like ask one thing wz seal dark pile % today seal chine new year event three ring cheap netea give explan ah origin w z weak enough cut properti dark give way fantasi westward journey i realli angri sad] ',
'[ice cream live chen jiali rainbow facial paralysi tough battl sho san user jhdzeu4 2co mo warm babi microblog send gift congratul your sina microblog system identifi si de event prize lucki user plea visit http tcnrweb3h check gift]',
'[]',
'[it made money tang yan red envelop total 254052 yuan cash fast chip slow handfr]',
'[daili random wallpap wor line set point too lazi care lazi care]',
'[i updat sina disc android client v333 time optim onlin experi read document â ‘ â ‘ ¡ offlin see list file come experi microdisk v333 download http tcnratl1ij]',
'[a microblog red envelop distribut good tri luck]',
'[rm file 110 delet multipl file]',
'[]',
'[what "s" social movement pk interest]',
'[fan bingb red envelop money abil grab red envelop fight technolog i grab red envel op cash issu fan bingb yuchen kiki good luck year goat open you tri luck â•® â¯ 3â¯ â •\\xadhttp tcnrztwqvc]',
'[rainjihoon i like work damn love `` refuel forc stand star life persist oh bang ba ng da]',
'[good morn uf knowledg spread one person anoth knowledg diminish impair the valuabl popular idea applic knowledg lead increa return grow number innov imagin ``]',
'[drop formal launch onebutton home `` featur allow smooth way home]',
'[my show cinderella cover `` upload fm845514 rock poetri come listen download lych

fm listen offlin]',
 '[befor packet produc album i tucao said thirti men also babi toss quietli alreadi r
ecord half sheet amaz dish wu said group band open studio i accompani practic kick un
wittingli three year he stumbl thirti band also establish unfortun late dream effort
terribl to peopl around effort in order wu "s" everyon togeth]',
 '[ann "s" great red grab red envelop map auspici red envelop i drew micro color ball
lotteri new year red envelop `` micro lotteri provid great fan ann "s" red come tri l
uck]',
 '[relogin remind yuan heart wyse ny sent crawler]',
 '[qihoo recruit natur languag process engin]',
 '[expert we "re" real program walk share headlin today]',
 '[subson framework use graphic "xian" _ wang lei blog]',
 '[how could hundr percent reason peopl world]',
 '[fangshimin frozen meat expir decad wonder "s" surpri expir year thi report confu c
onfu two as xinhua new agenc report quot interview southern weekend cctv basi intervi
ew some offici love lip servic base consid xinhua new agenc report hui yingjiang shi
roujia news http tcnrlgokua]',
 '[littl brother fed "s" annual red envelop grab good luck i drew fed "s" brother
red envelop quick taxi provid fast taxi yuan red envelop `` year goat yet come firstc
om good luck come tri luck http tcnrzdjktm]',
 '[ma di sesam leaf red envelop abil make money grab red envelop fight technolog i gr
ab ma di sesam leaf sea red envelop cash sent togeth tide tide lvi good luck year goa
t open you tri luck â•® â¯ 3â¯ â•\\xadhttp tcnrzkz3gj]',
 '[funni do want tri thi unman aerial vehicl `` drive cool ha ha]',
 '[i intern sina financ planner plan invit code want friend came oh liyong yao zhang
jia qi senior financ planner qi shot march march everi morn 1000 limit grab stamp ins
tal plan buy guid click invit code dim]',
 '[inform amoy us logic vulner beauti qq phone number free look logic flaw w0ai1uo "s
" blog amoy reset us open bypass pay straight]',
 '[aptitud search ^ tin find deb packag]',
 '[share link]',
 '[oh god i final manag get 2car point]',
 '[golang a go librari disqu new distribut queue author redi go languag chine network
]',
 '[share pictur]',
 '[exclu decrypt secret capit airbnb uber behind social share fruit]',
 '[]',
 '[rainjihoon grab red envelop red envelop map auspici red envelop i drew quick taxi
provid fast taxi yuan red envelop `` red envelop rainjihoon fan come tri luck]',
 '[i feel ah child natur nurtur along fine worri blind correct especi age two evolv b
iolog properti best take advantag favor growth]',
 '[nasa space daili chart big run spacex launch nasa cargo research to intern space s
tation a spacex falcon rocket lift space launch]',
 '[openstack technolog confer hot ing do think summit guest speaker do care arrang gu
est product showca live oh with need minut abl easili understand hp helion openstack
develop platform hybrid cloud manag platform new skill getâˆš wood not faith scene cl
oud lose heart quick see follow figur tast hp helion cloud charm]',
 '[git git common oper command summari open sourc distribut version control system ef
fici highsp process small larg project version manag git linu torvald to help
manag]',
 '[note grow pm â· practic choo good product manag most peopl "s" mind soul product m
anag product manag howev practic product manag play big role and choo good product

manag after read articl abl get answer want recommend]',
 '[albatron qingcloud beij second district meet today limelight quickli attack]',
 '[thi thing could get prize talk otherwi also play togeth happi cool i receiv microb log level exclu ceremoni prize forc go tri charact]',
 '[insight life spring spring style eleg winter winter life beauti smart laugh world laugh you cri one person cri when forget the pain thing love someon much forget also need met it import help other â€¢ wrong time help]',
 '[on microblog some peopl love say see comment everyon yell i rest assur `` i guess go jingdong taobao buy thing see comment decid buy brush comment chine characterist]',
 '[announc winner linkedin econom graph challeng http tcnras90el challeng http tcnras 90ew]',
 '[airdroid let comput mobil devic send pictur document music etc becom simpl fivesta r recommend download http tcnrzkvpje]',
 '[everi citi belong uniqu tempera]',
 '[shape constraint larg data analysi in articl author describ method shape constrain t system data flow analysi he spoke factor use system analysi work set averag transac t size request updat rate consist he also pass two use case stream video face recogni t detail big data architectur discuss]',
 '[success life must sever success refu accept debat]',
 '[admir attitud toward life although far away appear wander wander essenti life love unruli voic china]',
 '[want learn it skill junior partner big welfar free highdefinit video download free onlin technic q & a cour fee free limit edit limit opportun oh firstserv basi simpli regist activ f code abl enjoy preferenti readi go]',
 '[it electr provid frequent price oolong remedi need fill legisl gap in fact emerg e lectron busi platform oolong price common occurr regul fill legal void give consum re ason healthi environ consum]',
 '[ministri industri the intellig manufactur start point vigor promot internet `` jul i ministri spokesman chief engin zhang feng introduc first half industri commun indus tri develop `` press confer the next step ministri intellig manufactur start point vi gor promot internet develop intellig equip intellig manufactur organ implement major project speed intellig plant]',
 '[it love fantast art etc su beep beep mile mile infr video ventur enthusiast suffic i fund b station fill huge fund gap futur purcha copyright b station tri launch new b arbarian contract program `` call user contract new fan favorit fan favorit cartoon freedom reward `` unequ amount]',
 '[who goodheart children afflict realiti lunaticridden effort begin world noth world may sidelin]',
 '[]',
 '[share rain song carat lover `` shrimp share music]',
 '[test nanj nation husband oneniceapp nice]',
 '[site react transact model â· issu â· miniflycn qvd]',
 '[recommend book digit market market subvert imagin gold mou â· digit market competi t highlight classic case full color `` how subvert inher market idea make good use di git market how give full play imagin creat richer solid effect brand commun space cla ssic case win open histori mirror enlighten futur door]',
 '[i answer what recommend linux termin oper mainten know almost tmux featur screen a nyway littl host i use simpl power]',
 '[sogou star rank fieri award rain 164w vote rank no chart popular male singer small partner quickli vote]',
 '[ganchang ah]',
 '[i share mrs nora wang xin articl much money small gift]',

'[keyword search tool upgrad announc enhanc function time dear webmast i glad inform perfect keyword search tool upgrad after upgrad enhanc function hundr time the number search keyword demonstr improv time 50000 ad keyword rank data new popular page rank data add realtim traffic keyword data seo research associ network]',
'[get start python reptil overview python bole onlin ``]',
'[laugh cri xia yi gold multi dali]',
'[zhao wei dear `` perform definit boxoff poison act play well need accumul precipit]',
'[just like photo malign lake jasper nation park alberta canada nakedplanet photogra phi chrisburkard http tcnrari4pn]',
'[summari jia bingxi liu shan zhang kaixiang chen jian survey robot visual servo con trol vision system control strategi acta automatica sinica 861873]',
'[i grab afford app welfar zodiac natal buddha red gold lucki small partner share co me tri afford app]',
'[oper mainten linux softwar instal begin onlin youth cheer]',
'[loui koo small red envelop rob good luck i drew red envelop loui koo quick taxi pr ovid fast taxi yuan red envelop `` year goat yet come firstcom good luck come tri luck http tcnrzeutbc]',
'[handsom man awak good morn]',
'[becom php expert difficult beginn intermedi profess elit obviou use php everyon wa nt good php expert cour rome built day beginn expert kind process go featur code anal ysi project creat php environ tri differ framework http tcnraamku0]',
'[sun]',
'[origin plate jun want talk everyon use rightclick middl finger scientist found hab it use index finger right mou button iq higher % ordinari peopl with ring finger also understand peopl realli use common index finger separ index finger qinghua univ]',
'[i hang cold http tcnszjykp]',
'[internet rumor recent imagenet race event process result research still caution ge t smart ah]',
'[thou thou thou thou thou thou thou thou thou thou thou thou thou thou thou th ou thou thou thou thou thou thou thou thou thou thou thou thou thou thou th ou thou thou thou thou thou thou thou thou thou thou thou thou thou thou th ou thou thou thou thou thou thou thou thou thou thou thou thou thou thou th ou thou thou thou thou thou thou thou thou thou thou thou thou thou thou th ou thou thou thou thou thou seest ah ah ah ah ah ah ah doge doge doge doge doge doge doge]',
'[guo dafeng look forward question leaf share well]',
'[one _ baidu encyclopedia one â„ â¡ â " â„ â¡ â^ japan tea ceremoni term is rikyu set pearl murata flow moral principl shao gull militari field stream propo majo r tea road tea ceremoni sevenchul three thousand one thousand imperi clan urasenk hay ami zongda pass repair develop one `` word first propo two rikyu "s" discipl mountain mountain]',
'[prai intellig hardwar get back ration invest appl watch list deton intellig hardwa r field wave panic buy full thunder natur touch sensit nerv investor time intellig ha rdwar project everywh despit heat market steadili improv alway time calm wave intelli g hardwarerel innov project still continu ferment recent investor gradual return rati on intellig hardwar invest]',
'[the northwest "s" largest data center start build china mobil xinjiang data center cover area â€‹â€‹169 acr total construct area â€‹â€‹119000 squar meter built idc inte rnet data center rack host capac a spend million yuan build construct scale 39000 squ ar meter provid chassi expect put use end for inform plea click]',
'[it seem way sleep]',

'[lanshu could stand]',
'[exampl similar fuzzi optim studi product configur]',
'[internet believ wang kuanhsiung red xxiii tanggua children stick grab red envelop win i grab littl red envelop internet believ wang kuanhsiung alipay wallet issu toget h year goat yet come good luck come red envelop tri luck http tcnrwvta00]',
'[x86 develop board make smart home system]',
'[liaon new year revenu drop 179 % half provinc incom expenditur _ netea]',
'[flo rida low feat tpain `` day flexed_bicep origin dinner work home air condit blo w watch afc game pleasant but i like and accid sleep night]',
'[technolog share zoom easi use jqueri plugin jqueri photo album one impress applic process pictur help make project add amaz imag transit effect zoom fullscreen display imag transit effect jqueri plugin support key switch keyboard support mobil devic]',
'[super marin `` my rate â˜…â˜…â˜…â˜… stori younger age white meng watercress app]',
'[how eleg high jump share blog nan page ``]',
'[* mozilla develop network celebr 10th anniversari * mdn "mozillasmultilingualresou rceforwebdocu" hasbeenavailabletodevelopersforanentiredecadeto http tcnrladifk]',
'[]',
'[]',
'[share track netea cloud music]',
'[i share articl]',
'[oh built github blog need know basi qing liang yu "s" answer know almost]',
'[everi night heart stopper sad]',
'[quguan media made comment sinc i also call harass let delet comment]',
'[ani time complet red envelop money abil grab red envelop fight technolog i grab re d envelop cash time finish _zea alipay wallet issu togeth good luck year goat open yo u tri luck â•® â¯ 3â¯ â•\\xadhttp tcnrwgp413]',
'[i look reason explain alreadi behind]',
'[uncomfort tear tear tear want go home want go tangshan also could walk get earli t omorrow also bunch thing also pick luggag crazi crazi crazi http tcnr2w6wew]',
'[i "m" asian song list rainjihoon highest chart song good music need action support come favorit song]',
'[left `` relea fire passag we want pull hand abl marri love live bed result year fo r woman regrett best age contractu wanmo good job men sad remind `` time substanc tim e belov woman feel love bring gospel grin]',
'[the sun well]',
'[dialogu kevin kelli noth new style spark origin titl shenzhen maker cultur week di alogu kevin kelli noth new style idea spark fli white beard round glass plain shirt k evin kelli kevin kelli maintain classic style noth new if http tcnr23zwvf]',
'[share share red red red share share share red red red]',
'[shower wind today with high â° c low â° c via]',
'[share articl good program introduct]',
'[i share li xiaopeng articl]',
'[king sprayer]',
'[zhengzhou univ colleg water resourc environ youth leagu colleg water resourc envir on zhengzhou univ student volunt help group as today "s" cet examin room occup small partner found librari full surpri notic find thing small partner accid lost rectangul ar brown wallet id bi hu yong id no http tcnr2jbmm5]',
'[happi lesson whi hard say share know almost net say fear reject other]',
'[if want add peopl process user v fan come push meter microblog search found point direct push rice network]',
'[hamster also sell meng meng born]',

'[phone baidu cloud end technolog practic `` gener manag sharon baidu mobil phone pr
oduct phd like feng deputi director baidu attend make open remark team 2nd floor hall
client baidu "s" mobil phone platform perform optim scheme `` keynot share layer clou
d venu architectur design person mobil search recommend combat `` keynot share dri sh
are technolog number live full warm atmosph phone baidu]',
'[distress money lunch break line minut cup coff]',
'[day scallop punch i finish hear today]',
'[i want go will hand hand]',
'[you "re" connect wifi xie dan dabo free shop provid]',
'[angel love `` theme wed show love sea angel love `` backstori larg wed show origin
touch legend in beauti love sea descend kind innoc littl angel that day encount like
sea boy that day love love sea beauti love heart yearn infinit use commit dedic etern
detail]',
'[a group safeti test busi group busi safeti test senior project type open public in
form the accept test lowrisk vulner high school receiv high medium low secur threat v
ulner lucr vulner box project particip immedi http tcnr2faayr]',
'[mountain measur time photo]',
'[to see public project i help want support one help `` give perfect home volunt ma
elect surgeri cost 15000 ramnsat `` never give effort]',
'[befor earthquak use new technolog relief]',
'[overh red envelop year goat yet come good luck come grab red envelop win i grab
overh red envelop cash good luck year goat open let red envelop reviol http tcnrzggwg
6]',
'[insist day turn around cell shi]',
'[walk 616km one hour minut 4415 kcal plump ``]',
'[ha ha ha ha ha ha ha ha]',
'[how ubuntu system screen shot _ share baidu experi]',
'[jane chen jie ride sunshin two small children main start pair program under scienc
cuisong yan v5ria]',
'[jihua group issu new antidrug unit seiz equip yunnan stateown asset manag committ
one minut quick overview basic inform secur concept]',
'[ì†ëª© ë°œëª© í•œë²ˆì ” © ë¬¼ë¦¬ê³ ë,˜ì „ œ ë° ” ì§€ ê¸´ìž¥ ìš¤íƒ ë‹¤ë ˜ œ í ˙ ¸ë ˜ ¤ ì€ ìž… ë²œë¦¬ ë‹ˆ ì•…ì–´ ì^˜ì¤€ poodl ê·¸ëž˜ë „ ì€´ì€œ ê±¸ ì–´ë–¡í•´ _ ìš¤íƒ ë‹ ¤ë ˜ œ í ˙ ¸ë ˜ ¤ poodl ë§ˆë‹¹ ìž^ëš ” ì§ ˙ ì—€ í,¤ìš°ê³ ì‹¶ë‹¤ cr greenteepark insta
gram]',
'[it lenovo cto whi want pre superfish lenovo group cto bide huo teng hugh peterhort
ensiu superfish respond associ last week relea automat remov tool help user complet r
emov superfish product lenovo user]',
'[shower today with high â° c low â° c via]',
'[look good heart cecilia love love love]',
'[i heard tomorrow test fortysix howev matter i cock doge]',
'[introduc success git branch model technic translat in articl i propo develop model
i "ve" introduc develop model project whether work privat year prove success i "m" go
write long time ago]',
'[cs student peter beshai creat bucket popular basketb shot visual tool comput scien
c ubc]',
'[spring festiv year also thought see show loot red]',
'[break record time car bao tong templ came back took two hour previou record halfho
ur khan]',
'[lei jun millet phone i publish articl]',
'[buffer founder nontechn advic founder give develop task outsourc compani even
program you find technic cofound first product requir program mani time mou click mak
e mvp origin chain http tcnr2ff6xj http tcnr2ff6xj dailil',

'[upload photo album 720pim ``]',
'[way work i publish articl product manag enterpri mode `` ad especi popular recent share concept economi concept true share product equip labor share resourc share in practic two often mix togeth therefor use share over econom summar variou combin specif form plea add discuss group http tcnrzukbq5]',
'[publish articl reproduc bowen reproduc ã€-candl chart move averageã€— usag move averag line ``]',
'[guo tao red envelop money abil grab red envelop fight technolog i grab red envelop cash issu guo tao alipay wallet burst happi valentin "s" day fan togeth around red envelop â•® â¯ 3â¯ â•\\xadhttp tcnrzgpmyl]',
'[write stori pictur wall i custom decor wall paint frog network small partner come onlook share wall frog]',
'[easier said done speak louder word descript user "s" behavior compar introduc user "s" charact give impress former profound think]',
'[tucao emperor rare home wash bowl]',
'[power team]',
'[air qualiti index chongq new ui good comfort]',
'[turn ms yu mathemat model algorithm quanshou lu conjunct matlab exampl absolut dri peopl i tell download address http tcnzozibjj]',
'[swift objectori program orient program protocol orient program compil]',
'[zhmm]',
'[wow last season play mani % achiev pvp]',
'[quick taxi voucher first hit fast car look miss queen `` campaign offici open come queen adult upload video we "re" look click map view long event detail see beauti well taxi voucher click collect http tcnraxihm6]',
'[vest red envelop wow realli rich i abl get cash big red vest privat memori alipay wallet issu togeth everi day draw red envelop one day becom tyrant you tri luck http tcnrzg7pmk]',
'[new year red envelop everyon send i wish fortun red envelop]',
'[one user said like televi shop sell magnet therapi pant sure plea foreign mirror seem tall video kan hung industri vehicl advantag new sagitar bodi share youku]',
'[jung ji hoon rain625 happi birthday heart]',
'[scallop punch the first day learn word]',
'[head first design pattern chine version share amazon]',
'[final understand worship worship effect reliev stress increa selfconfid costsav effect last resort]',
'[whi want take pictur i shit also happi let pull ah ``]',
'[adob launch prize catch insect `` hackeron adob compani lengthi applic product line led flash softwar product long face lot secur risk in order avoid deterior situat company chosen open attitud enthusiast abl help even http tcnrwnrqfp]',
'[video jiangsu haimen send high school entranc exam ceremoni shock tmd realli spectacular]',
'[ibm rank enterpriseclass hybrid cloud largest supplier synergi research group survey ibm two consecut quarter enterpriseclass hybrid cloud privat cloud provid championship report also includ fourth quarter ibm cloud comput market top supplier rank]',
'[free stuff feel good secret the origin zero `` special price notat evok warm emot becom sourc irrat excit if item discount cent cent would buy possibl if promot free cent scrambl reach definit share kindl]',
'[it china increa militari spend japan see content clearli read chine side also accel transform defen strategi inland pure militari power ensur right ocean navi air forc countri transform transform would requir substanti r & d fund]',
'[readili member festiv fan sauc red envelop hand i abl get red envelop cash microbl

og member thi begin said yuan red envelop do say i continu draw red envelop go]',
 '[it technolog share font awesom great icon font librari css framework font awesom w
eb font contain almost commonli use icon twitter facebook address http tcnrwdnrrx use
r custom icon font includ size color shadow effect properti control css the front er
come experi crowd]',
 '[linkedin depth analysi larg data platform]',
 '[os x 10104 trim provid way open thirdparti ssd although jump risk `` prompt]',
 '[take night train night watch point nervou strip though see whatev outcom psycholog
comfort qaq]',
 '[docker contain â€¢ file system isol process contain run complet separ root file
system resourc isol system resourc like cpu memori alloc]',
 '[programm note after asynchron program happili often use synchron program solv prob
lem order execut parallel execut asynchron solv problem howev common program model so
meon launch applic patent see googl patent search site patent public number us a1 ``
patent]',
 '[fangshimin mayb jour common denomin pervert hatr islam luckyangyang conan xu hao s
ometh common defend china "s" internet blockad xu hao said would direct great superio
r wall conan reason i find strang like parti http tcnra1mqcu]',
 '[jung ji hoon rain625 happi birthday rain carat lover rainjihoon rainjihoon photo r
oll beaming_face_with_smiling_ey credit tag]',
 '[steve job art imperi person realiti distort field origin chain http tcnrafqb67 htt
p tcnra2svlq daili]',
 '[lemon butter]',
 '[most small compani still die must first solv problem exist the second step consid
secur issu]',
 '[red envelop children pull group less point]',
 '[cool ai english speak group day new start week lunch word mania learn english fact
say myth chine peopl learn english say english pull common languag chine peopl especi
emphasi direct translit almost insan hyster hysteria friend young a dream smile smile
]',
 '[the number red envelop chen abil make money grab red envelop fight technolog i gra
b red envelop cash chen alipay wallet issu togeth good luck year goat open you tri lu
ck â•® â¯ 3â¯ â•\\xadhttp tcnrzdoijr]',
 '[rubi ast fun profit experi]',
 '[healthi pig fruit crazi]',
 '[hire fullstack engin expert engin earli startup everi engin must full stack everyo
n everyth the compani develop certain stage necessari introduc expert field talent so
introduct expert personnel your compani expert interview expert daili]',
 '[cadisplaylink magic combin magic uibezierpath share kittenyang]',
 '[i rainjihoon present flower two valu â€‹â€‹of love feel meng meng da rice circl pr
o come send flower aid]',
 '[artifici intellig topic dmlc deep depth leagu distribut machin learn opensourc pla
tform resolv problem dmlc paper describ exist xgboost cxxnet minerva paramet server r
abit compon main solut implement perform brief descript recent project plan the lates
t avail programm magazin 6a]',
 '[cctvhttp tcnhoweu]',
 '[search mean â™ « i listen song adam levinelost star â™ « http tcnrlmaozh qq music
phone listen qq music qq grade accel liter]',
 '[nude supermodel recent tinghuo bodi art nude model touch conscienc tell fall love
supermodel bare biliti magic power cc women bodi]',
 '[it figur galaxi s6 s6 edg new plan samsung determin exposur offici announc
galaxys6 mwc2015 mobil world congress held barcelona â€‹â€‹spain march year deriv mod

el galaxys6edg eve confer main foreign video un]',
 '[]',
 '[or good]',
 '[today i one hundr word cut back six vocabulari word my english bad at least i hear t possess great person you come back word right origin word cut school bulli certif]'
,
 '[upload photo album 720pim ``]',
 '[rain jung ji hoon 13th anniversari debut rainjihoon rain carat lover heart heart h eart rainjihoon hi name jung ji hoon i love peopl call jung ji hoon]',
 '[kim power manag softwar crm help manag presal sale sale follow one one well known help sale manag keep abreast progress local offic sale problem reduc churn employ pri vat chanc avoid personnel chang impact custom churn lost busi person maxim potenti cu stom improv effici singl success free trial]',
 '[i "m" asian song list rainjihoon new song â‰¥ï¹â‰¤ â‰¥ï¹â‰¤]',
 '[just log dragonfli fm fm final found artifact explo content news fiction music com ic talk show want hear consequ well key 3000 radio station broadcast hour day fast pu t away right â† ’ http tcns6fted share dragonfli fm]',
 '[today i micropl sign gain 369m free space good luck index star tri luck micro data disk sina "s" brand cloud storag larg storag space download massiv resourc mobil comp ut synchron hand micro unlimit come experi]',
 '[taohuawu blossom templ peach anli peac tao huaxian race peach also peach chang dri nk i publish articl]',
 '[video summari forecast model an overview predict model `` max kuhn cloud http tcnr2ntg1x see three aspect predict model `` http weibo com clxqhkc2e]',
 '[]',
 '[uber use discount code 3ncx8 save â¥ first uber ride servic convert]',
 '[margin call]',
 '[i share]',
 '[xen emerg new vulner linod amazon rackspac emerg repair new vulner xen open sourc hypervisor nerv major public cloud compani rackspac amazon "s" tight hasten launch ma ssiv attack attack repair restart system]',
 '[there jiuyangzhenj human resourc configur ä¹éˆ´ç™½é¹¤ç^ª beat dragon palm fast br eak slow break the core rhythm rhythm fast slow loss rhythm fast run without direct s tand still slow with pace fast spring wildfir slow epe front fast fast slow slower]',
 '[shenzhen first meal breakfast chaozhou shop children teenag rice roll handmad nood l meat children wolfberri leaf soup delici afford shi]',
 '[the evolut concurr web servic save system memori cpu author xu hanbin first number concurr connect grow number concurr connect web system face exponenti growth recent y ear becom high concurr norm web system small challeng in simpl crude way solv increa web system machin http tcnransdgw]',
 '[last week buy food afraid go home eat think]',
 '[health swine mad]',
 '[love train `` theme wed romant encount sound conductor femal dear passeng train st ation dock wed the train met number departur toward etern happi train one pair specia l train usher passeng mari bride groom small swiss let us warm applau welcom aboard c olumn happi sweet journey began detail view]',
 '[dongfeng car season joy peugeot new car sale ha ha cheaper buy car also put prioriti latest model do new traffic flow saliva show friend laugh open car let other envi jealou hate sinist act quickli]',
 '[* zorin os rc steamo 20 preview * zorino steamo high degr concern linux distribut zorino base deriv ubuntu relea suitabl window chang linux]',
 '[plea call packag small meat rememb shout]',
 '[publish blog network data secur crisi `` & lai shan network secur data secur natio

[publish blog network data secur crisi ... & fai shan network secur data secur natio
n internet emerg respon center cncert relea china "s" internet network secur situat r
eport `` data show china "s" network secur situat optimist cnchttp tcnrarpko4]',
'[sunset sky cloud vscocam via instagram]',
'[i updat ipad microblog client v371 â ' microblog page text optim easier turn comme
ntari prai â ' ¡ hot new comment featur â ' ¢ bug fix come experi app store download
http tcnh98rbi]',
'[â· percept â· sae recommend unknowingli use sae almost two year]',
'[tsinghua univ faculti game held east playground may afternoon tsinghua univ one fa
mou slogan motherland work healthili year `` impress stand]',
'[how smart citi boom public enterpri layout beachhead sinc state local twelfth five
year plan `` introduc mani citi build smart citi develop prioriti % subprovinci citi
nationwid percent prefecturelevel citi total citi construct propo smart citi total pl
an invest nearli thousand billion the full text address http tcnr2lerux]',
'[publish bowen `` send bless net send bless sign send bless net offici websit regis
tr center send bless network registrar network center send bless net result pool tech
nolog "s" rebat share shop mode platform first set discount rebat share one pure buye
r shop guid platform shop rebat offer exclu discount merchandi latest]',
'[today success receiv point microblog control red guevara life network]',
'[empti `` bool true walday]',
'[* node can not join cluster how debug issu sst * galeraclusterhastheabilitytoaddne
wnodestotheclusterbyhandlinginternallythetransferoftheentir http tcnrlauam6]',
'[sister funni photo shoot colleg entranc examin student refuel]',
'[langfang bgp multilin telecommun room server host price beij idc compani langfang
run telecommun equip room]',
'[i "ve" manag make appoint charm blue note2 youth yield new qualiti thousand yuan m
achin choic note2 opportun particip interact topic charm win world "s" first blue cha
rm blue charm blue note2 new preempt m code]',
'[f x amber solo album contain song beauti `` lyric version public imag imag amber c
hildhood photo variou stage model debut composit combin beauti soft voic warm acoust
guitar amber express unfold wing dream face difficulti give beauti appear overcom]',
'[februari afternoon news alibaba group today announc establish hk $ billion hong ko
ng young entrepreneur fund nonprofit make support hong kong open cau young peopl fost
er entrepreneurship]',
'[it sharp solar public charg station built free charg smart machin global technolog
roundup accord japan iphonemania websit report juli even away home appl "s" smartphon
free charg possibl]',
'[ali literatur q2 market share leader fiction book flag top activ user januari seco
nd quarter activ user poni fiction 188175 million first place palm read iread second
qq read three read app accumul month move second quarter bat total number activ user
rank point view ali rank first literatur tencent read text second place rel weak baid
u literatur http t cn rlonfdk]',
'[zhang xin yi red envelop wow realli rich i abl get cash big red zhang xin yi
alipay wallet issu togeth everi day draw red envelop one day becom tyrant you tri luc
k http tcnrzjgnvi]',
'[i updat flymebo 124 applic download address http tcnrzn3x8d charm feng network dow
nload http tcnzwsy14f]',
'[in hospit registr led display he show age look past front man age year similar
age fill face suddenli longer feel younger big pentium also run around live dream lif
estyl still pure pursuit three major free person freedom freedom thought financ freed
om achiev]',
'[tang yan grab red envelop red envelop map auspici red envelop i drew quick taxi
provid fast taxi yuan red envelop `` red envelop tang yan fan come tri luck]',

```
 '[note orphan heard catalyst microsoft china msdn]',
 '[graduat remun i cri watch internet compani pay began restless whole local tyrant h
ttp tcnrlafijp]',
 '[tung cheng ann]',
 '[intellig medic sector rose 345 % share daili limit pass centuri etc nobel prize wi
nner solv problem hightech thiev prevent inform secur awar educ help http tcnrvopuci]
',
 '[mysql learn master six skill]',
 '[shut portug cheer happi year snake magic read mind sketch imag test inner world qu
asi terribl ä ’ jesu do]',
 '[stand back network outlet navinfo brought fun drive 20 3snew first time eight mont
h drive interest wedriv usher age cell phone carplan interconnect scheme welink car r
ulesclass network oper system wedriv os]',
 '[year old get year old love doll year old final money buy year old love dress point
what start youth so mani thing tie togeth youth better leav youth foolish liu yu give
bullet ``]',
 '[night return liber]',
 '[seen `` the man earth scienc fiction fuck kid â˜…]',
 '[how quickli locat accid expo global variabl barret lee blog park etherdream thx]',
 '[it googl stock price high stimul nasdaq composit record san francisco juli even ne
ws relea strong fiscal second quarter earn googl stock price friday hit record high p
ush nasdaq composit index also hit record high]',
 '[microblog android client android client feedback version 528 model nx511j system 5
02 wlan rotten chang chang]',
 '[post pic]',
 '[no return key traffic manag breath light brush android rom]',
 '[spoil children low energi cradl cultur alpin cold soil make upright pine cypress b
y sima qian castrat articl everi word gem bead li yu imprison word environ one new
emperor indulg sensual result loss life cheng corrupt final selfdef hardship worst ``
word wisdom realli ah]',
 '[talk evolut php5 garbag collect algorithm garbag collect introduct host php langua
g program php programm need manual handl memori resourc alloc relea written c php zen
d exten except thi mean php realiz refu see http tcnzyhzjk7]',
 '[rain carat carat lover lover rainjihoon nice tv larg rain shuai shuai meng meng]',
 '[so mani electr supplier help countri amazon sell thing feel quit strang think]',
 '[all suffer world drive forc gospel brother sister bring peopl suffer promi abil lo
rd jesu christ come touch bibl ver]',
 '[i concern topic owhat sing sensat boyfriend take express view]',
 '[zhang huichun origin "ve"]',
 '[i rainjihoon present flower two valu â€‹â€‹of love feel meng meng da rice circl pr
o come send flower aid]',
 '[read rever `` â˜… bad ``]',
 '[masochist new tactic point watercress longer play like music listen]',
 '[depart delici can yellow peach sweet sour afford essenti food good âš™oâš™ oh flig
ht inform plea contact]',
 '[what wood confidenti agreement said american consequ tube real estat loan useless
american cent chine bank govern guarant american right interf branch offic oper plu r
eagent compani boss huang keep low profil american fear control independ branch]',
 '[cooper student fear algorithm understand web server json jsonp explain long time]'
,
 '[anoth day it "s" april 0416 pm i "m" aliv nlp tree roadnorth]',
 '[thi dog actual murder owner]',
```

'[chrome android chrome io fundament two differ browser blink render web page webvie
w io "s" mobil web]',
'[think mani time headphon lie jay full expect occas muffl lone afternoon summer aft
ernoon heat restless night quietli greet sink decad ago would second person know girl
heart]',
'[so mani peopl migrat x86 wait ge uncl there say 21st centuri expen talent for it s
ystem construct talent primari factor guarant system stabil today user migrat legaci
risc architectur x86 platform whether migrat realiz kind effect user feedback here ta
ke look see http tcnrlv2o1a]',
'[scallop punch the littl princ day read chapter]',
'[when young one thing alon pretti cool grow alon bleak one thing now i feel lone on
e thing sometim peopl need real despair the real despair pain sad matter realli despe
r peopl calm]',
'[nlp job agenc us corpor recruit recruit machin learn data mine senior research eng
in]',
'[sand fan day i come net book curtain clean servic longer worri clean good look oh
hee hee]',
'[big new year "s" street nobodi ah]',
'[the first session month baidu takeaway food good subsidi presumptu eat lotteri win
appl watch offer nonstop gift constantli quickli spread news lotteri click]',
'[rain gurgl stream would rather like live rain thought come rain everi day]',
'[where microblog search bright star rain fan click fast rain enhanc bright index he
lp ta becom beauti star sink hundr million microblog user see bright ta]',
'[php need know five thing year "s" schedul php schedul vote rfc php relea octob des
pit delay plea publish year php thu view detail schedul http tcnrayvygx]',
'[time let forget person want forget start new life so romanc feel alon long get eve
ryth leisur thought harm love miss want get rid person heart plea fill heart heart fu
ll one hurt]',
'[today went hospit check "s" panic]',
'[i share child liu articl]',
'[thesi receiv free test card deal charli picki mentor charli content heavi word tri
pl pressur charli essenc million chine foreign academ journal dissert billion interne
t page english compar ensur origin other like thesi]',
'[code school good ha know learn mode realli ignor]',
'[silicon valley second quarter good time updat]',
'[i share]',
'[css preprocessor postprocessor css `` when come css preprocessor familiar focu art
icl drawn introduct css processor nearli year new frontend commun trend zhao lei "s"
blog]',
'[what hell would like squall kick spirit]',
'[best comparison chart javascript librari fusionchart vanchart highchart js larg ic
on librari compar refer select turn need]',
'[smart encod h265 4k done high definit rich color low bit rate roi svc multistream
low latenc intellig detect achiev face detect behavior detect scene investig sen dete
ct abnorm audio microphon disconnect scream burst]',
'[school sto moral integr comminut yet]',
'[safe use common sen vain mobil devic mongolia mine half loss 78214 million yuan di
vidend]',
'[the major rural market mani peopl market key grab shu electr supplier genesi becam
new year spring festiv flavor]',
'[june wednesday morn quot updat small stuffi forum http tcnrvsn2m4 littl stuffi fru
it shop]',
'[know almost salt club time beij way cougerenao littl friend plan]'

'[know almost safe club time beij way cougerehao little friend plan]',
'[domest school univ sixtyon holiday middl high school class plu mad also delight]',
'[googl decid mark unsaf site remov adverti plan site longer abl profit adverti reve nu]',
'[unconsci april ubw back tomorrow gintama day back power game well eight day look i ng tear tear tear]',
'[never give]',
'[interview alibaba research zhao haip from facebook ali baba]',
'[fan bingb red envelop money abil grab red envelop fight technolog i grab red envel op cash issu fan bingb yuchen kiki good luck year goat open you tri luck â•® â¯ 3â¯ â •\\xadhttp tcnrztwqvc]',
'[i rainjihoon present flower two valu â€‹â€‹of love feel meng meng da rice circl pr o come send flower aid]',
'[share content xxxxx]',
'[i particip vote smartisan mobil phone new slogan collect activ vote smartisan init i i vote you "ll" find i "m" one `` thi option come stand http tcnrlmxqg5]',
'[ref tag android510_r3]',
'[it alibaba promi confed flag icon withdrawn product link american black church cha rleston last week occur shoot crime killer ago confed flag confed armi demonstr use c ivil war civil right organ identifi confed flag behalf racism lead flaglow movement ` `]',
'[xu qing red envelop money abil grab red envelop fight technolog i grab red envelop cash xu qing accompani note alipay wallet issu togeth good luck year goat open you tr i luck â•® â¯ 3â¯ â•\\xadhttp tcnrzy5elf]',
'[old love halo `` theme wed plan wed scene banquet hall entranc arch place moon flo wer door play first glanc visual impact banquet hall tabl wall decor color balloon ad give warm romant put small candl tabl entranc ceremoni place four corner pavilion fak e vine dot main channel roll red carpet detail]',
'[what "s" duang xiaob]',
'[china unicom month sb stop hair cancel day interest]',
'[i last year biggest mistak comput term mac os x io devic use appl id and bought lo t appl "s" applic know shashi hou consolid account function tear although famili shar e but still good merger]',
'[today new part css3 line way see css4 â•¯ â-¡ â•°]',
'[advent internet urban middl voic continu enlarg govern gradual kidnap least issu b eij haze internet user voic show govern conspiraci environ issu evolv polit issu part improv air qualiti also manufactur consid damag these injuri contrari rule law justic i share articl]',
'[year goat fortun togeth red envelop fingertip togeth coffeestar distribut chine ne w year red envelop red envelop beam speed]',
'[respect past glass wine wish go back year altman]',
'[modern blue link launch applic appl watch user user download applic directli app s tore free]',
'[mark]',
'[how get psycholog shadow divorc divorc chen he qq space also strip i read content want cri think miss layla realli good woman think realli love sort thing chen he firs t review first concern realli love ah divorc realli piti detail http tcnrzkmqir]',
'[video world warcraft older grow tenth anniversari edit share youku look almost cri]',
'[pm kipl law practicemanag advic success failur treat equal understand failur also onesid understand success scientist ask `` you alway test new type batteri failur con tinu scientist repli `` i never fail fail i alreadi know 50000 kind method produc bat teri there failur useless find peopl use failur]',

'[it mobil phone foundri rush open huawei millet phone order return lead appl "s" do
min iphone6 â€‹â€‹time suppli chain vendor also adjust proport custom busi new `` rep
ort comb found huawei millet repr domest manufactur subject variou suppli chain manuf
actur compet]',
'[a period consequ run run half hour two blister feet khan i tent attribut shoe prob
lem]',
'[note pm practicegrow linux improv effici ten bash skill in mani case use bash prog
ram problem encount repeat everi time i need rethink solut problem until one day i st
and sit write gener function bashrc file deploy comput recommend]',
'[year life improv emot intellig work rel smooth result arrog thing degr live world
prepar blindli intox temporari victori as everyon know victori small road sign order
achiev final victori must improv emot intellig]',
'[web frontend popular articl last month recal js code jslint refineri upgrad `` js
memori analysi tool develop contrast popular frontend framework complet guid consol m
aster]',
'[adverti quarter]',
'[statist two line time one develop model paramet nonparametr model provid robust st
atist model solut increa model select larg model choo small model model effect littl
reason variancebia tradeoff thi reason two complet contradictori two line the scholar
seem tangl]',
'[take colleg entranc examin score fine you right nine colleg entranc zhenti 9998 %
candid if particip colleg entranc examin year like admit beij univ tsinghua univ]',
'[you know australia]',
'[jqueri slide navig preview thumbnail plug _ excel record set ui3g share tucao exce
l design mind]',
'[lu zhenwang red envelop readili pump red envelop hand i "ve" drawn luzhen wang qia
n ink yuan red envelop cash come issu oh thi begin said billion red envelop do say i
continu draw red envelop go]',
'[in end make aunt stop control]',
'[i realli niubi javascript debug even without onetim write line take account extent
garbag javascript languag simpli call miracl]',
'[good morn xinyu jealou guard quiet hustl bustl guard beauti life ever sun wind rai
n journey life intertwin yearn heart forward heart tire rest]',
'[it lott era onlin shop sale maintain doubledigit growth circul most chine tourist
may know lott group korea rank fifth asset addit circul tentacl also extend food tour
ism servic chemistri construct manufactur financ industri]',
'[rt itgovern free guid write audit report inform system via twitter]',
'[it start eve nation bank ps4 psn at present fast speed media confirm nation bank p
lay properli intern version ps4 psvita host use version number dlc download affect]',
'[spa spm st sta saa fml perform tune much look concept dizzi starv]',
'[from littl tiger tfboy i drunk gap andi brother said sing song ancient time `` sur
pri]',
'[let search across divid languag on cross languag inform retriev technolog crosslan
guag inform retriev research subject field inform retriev the past year due rapid dev
elop internet research area widespread academ attent appli technolog search help us f
ind use inform exampl see http tcnzj8wnoo]',
'[shijiazhuang oil workshop pancak crisp repeatedli]',
'[experi share highfrequ frontend interview question remov duplic array element vari
eti method recent want chang job sever question list compani zhongguancun interview a
sk mention delet array element method said third encount problem mani onlin solut sha
re author write i hope help beginn play]',
'[how attract maker cultur group mtk two killer creativ lab mace share ee intel edsi

on comparison advantag]',
 '[white baymax like titan tower osaka japan expo perhap robot materi scienc make glo
w]',
 '[run 500km use minut second speed minut second 41778 kcal plump http tcnrwdr8mm]',
 '[i know almost answer shanghai two student googl internship difficult onli noth lik
e case educ other roll the technolog fun would ask question]',
 '[it china nation tourism administr survey travel agenc stop suppli product way catt
l beij april accord nation tourism administr websit news nation tourism administr for
m work group travel agent stop investig media reflect way cattl tourism tourism suppl
i good problem]',
 '[book moonlight fell left hand `` douban]',
 '[i run ran in principl run blank perhap order obtain blank run even among gap also
moment "s" thought sneak film thi matter cour human mind exist real blank the human s
pirit strong enough sit vacuum even consist have said share kindl]',
 '[uh never execut pr hr career develop push camera hackathon right develop total two
half plu pm also pr asid said ah go day cut head]',
 '[amax china new monthli]',
 '[with film longj tea]',
 '[baidu takeaway posit blindli call bibimbap worri i brought eat punch sent look sto
re help marin human good]',
 '[]',
 '[today i micropl sign gain 213m free space good luck index star tri luck micro data
disk sina "s" brand cloud storag larg storag space download massiv resourc mobil comp
ut synchron hand micro unlimit come experi]',
 '[i publish articl]',
 '[light shameless ah]',
 '[rumor new trailer lanxiang special shovel room internet compani year new york
alipay ctrip suffer via testerhom]',
 '[for long time come librari meow in fact i come librari siesta sleep]',
 '[recent difficult control temper may forc variou deadlin probabl need vacat adjust
psychiatr go look i say peopl inadvert harm sorri condemn without name]',
 '[alipay set pictur would allow alipay you want uninst alipay taobao draw line]',
 '[there vision know know one day would love feel never part one day understand heart
ach feel halfdrunk half awak found actual want think fate i met moment destin etern]'
,
 '[real estat o2o sector lynx ji hou `` saif lead investor million yuan a round finan
c focu new home financ internet agenc love love kat hou `` expo $ million seri d fina
nc real estat busi platform kat hou announc complet million yuan a round financ led s
aif invest]',
 '[liu xuan red envelop money abil grab red envelop fight technolog i grab red envelo
p cash issu liu xuan alipay wallet good luck year goat open you tri luck â•® â¯ 3â¯ â
•\\xadhttp tcnrzddfbg]',
 '[puff my origin index difficult recov ah seem say quit accur come cece hard recov e
ye opposit sex click test address]',
 '[there engag crowdfund oper friend zhiyi sheng talk]',
 '[luffi admir i set one piec _ _ baidu post bar]',
 '[sun teacher still see passion speech leav univ poor]',
 '[as ghost symphoni shoe introduc skill use gns3 build univ `` topolog home feel pra
i king everyon share chain quickli point http tcnrareqf6 cool cisco lectur]',
 '[i % fill make knife payment prostitut tencent applic treasur]',
 '[scallop punch the first day learn word learn hear read two articl]',
 '[i rainjihoon present flower two valu â€‹â€‹of love feel meng meng da rice circl pr
o come send flower aid]'

o come send flower aid]',
 '[ifttt via weibo]',
 '[mile app exclu singl review cuisin buy back time point]',
 '[the disea men commit buddi hospit doctor told rash i got indirect amnesia the doct
or ask what specif symptom when gener incid buddi said i see beauti woman forget i "m
" marri doctor roll away point disea i cure]',
 '[do group peopl around zhongyi chang job frequent six month year jump thirti contin
u maintain highfrequ jump bring better frequenc hop basic mark time]',
 '[ssl vpn without want move offic worri data insecur take free trial sinfor ssl vpn
as ssl vpn market six consecut year rank no1 nation it brand deepli convinc ssl vpn l
aunch applic virtual app secur reinforc emm enterpri mobil manag program help user so
lv secur problem mobil termin access product detail click link]',
 '[bang bang]',
 '[start altman]',
 '[rockefel "s" first massproduc bicycl intellig gener smart bike realiz navig map de
vic bind statist rout share call remind mobil social network function millet millet p
ick hype market]',
 '[share chuan "s" courag `` netea cloud music]',
 '[care line agent switchboard oper posit descript job descript posit titl posit care
lin oper care hotlin oper depart depart corpor locat]',
 '[those pretend mysteri long critic contempt merchant beggar robber beautifi mind cl
ear parasit laugh ulterior motiv make businessman frighten emphasi fair exchang peopl
]',
 '[zhengzhou univ colleg water resourc environ youth leagu school environ water resou
rc zhengzhou univ student union night talk park month actual mind set key success lif
e mind main line life shen fulok selfinduct wide live land scatter pearl put back one
one put even valu string http tcnr2vxz7b]',
 '[walk work last week hee hee in fact i afraid scare intent turn microchannel motion
smile]',
 '[simpl yet effici smooth i use weico microblog client android awesom end intellig n
ight mode custom font well offlin assist help save traffic want tri better experi mic
roblog jab link download weico]',
 '[sam nest prison five basic fit train program see http tcnr2wy8kj]',
 '[paint realli tucao]',
 '[sometim i also think go shop buy nice cloth dress also want sleep frankli rest i f
orgotten i need work]',
 '[i particip vote initi ali secur which think team win championship year alictf i vo
te `` wargam `` option come stand http t cn raxygm1]',
 '[one read `` cea public anim video short video whi lovabl]',
 '[openstack rt digitalfilmtr `` the effici gain cloudba work flow far superior anyth
in http tcnr2le40n]',
 '[punch scallop day raft snare awe loin misgiv prolif vener]',
 '[too lazi get tonight school stop secur bad fact i would say teacher swiftli fli pa
st dig booger]',
 '[xiao bian recent ask question vmware esx deploy xiao bian know answer howev small
would like share small seri know littl knowledg deploy esx]',
 '[befor serv ross gardler particip activ guest come linux user group wednesday even
welcom attend]',
 '[docker play cool lot chang inher think contain make littl wast]',
 '[ah come back look better time given statement comment want cri group peopl cute]',
 '[share pictur]',
 '[i think teacher resign professor "s" friend pay discret friend accord statement te
acher injust wu chou nonengag teacher look he call teacher north outsid north outsid

proud also resign outsid north distinguish alumni mass chang titl and professor oh oh oh]',
 '[share nakajima miyuki song wa ta ka u made cri `` shrimp share music]',
 '[recommend highqual photograph imag websit eputcom i open settl i hope review share work i rose score photograph thank]',
 '[one advantag cloud comput join compani choic oracl integr `` integr work play well integr hardwar softwar badminton basketb footbal paint varieti interest group also regularli organ fun game outing chariti birthday parti day spare time employ meet int erest compani choic cloud comput charm]',
 '[you black peopl black java java]',
 '[togeth wife see beauti edg talk see i halfset could stand]',
 '[today wallpap doge]',
 '[baidu translat realli love give children "s" day special gift]',
 '[we play guess sina love i bet bean guess shengp fu juventu vs real madrid pressur game there varieti color cool prize wide rang sport quiz well strong partner pk group fan wait guess sina love play yo http tcnrarq3bj]',
 '[the end overtim rest continu tomorrow]',
 '[live smile spring]',
 '[i saw helpless parent teacher episod mark share potato grow]',
 '[cattl devil dare complain wukong back goku ask zala cattl devil said watch wife pr incess iron tomb note fell love peak peopl say peak handsom nobl ideolog consciou som eth goku thi go back cattl devil i afraid put head dedic take countri]',
 '[c worth note sever major chang detail sourc articl danni kalev front c standard co mmitt the biggest chang c whi you should care lai yonghao made see http tcnzjwjkyl]',
 '[magic beat khan angri angri baidu magic beat]',
 '[as memori peopl whether feel unsung hero unsung hero prove to particip it save `` long ibm storag robot provid knowledg base challeng sen challeng game given storag ch eat step step help enhanc store knowledg make exist sen overflow h5 link http tcnrlks taq]',
 '[down mean give]',
 '[rainjihoon star forc list gaon weibo chart highest chart rank no move finger champ ion]',
 '[read news abl get instal app read salari 7hy1 invit code read news read news abl g et paid read app download http tcnrasea6]',
 '[walk road black cat lie drop chai chai deform grass cape w_q2001 hao congratul dea r user congratul becom sina weibo `` identifi seventh anniversari `` secondclass offs it activ award lucki user plea visit http tcnraxoewg check gift]',
 '[can anyon tell dizzi 3d word english say doge]',
 '[rt sodevi protip get rid work email slack flowdock phone work induc stress anxieti sure decrea]',
 '[good morn]',
 '[live network goddess zhang xin yuan ps shot recent famou friend luo beibei `` expo group photograph zhang xin yuan the former network goddess recent photograph upon exp osur might lead countless friend tucao thi cosmet sequela `` can live ps creatur `` z hang xin yuan thi poison yet `` p anoth detail http tcnr2lniam]',
 '[tribut chai jing haze context i give excerpt % smog coalfir fuel presenc larg numb er lowqual coal clean sever grade gasolin behind abroad larg car environ friendli dev ic british pollut seriou good govern govern haze must complet upgrad energi oil ga en ter era must break monopoli two barrel oil environ protect depart respon need fundame nt chang]',
 '[top domain name registr top15 western digit china top share drop % chine offici we bsit data ali million net cloud renam china internet us orang cloud offic offici resi

st si nike the first commerci network linkag interconnect world offici china busi net work new network microblog network]',
 '[technolog samsung smarter temporarili relea smart watch wise]',
 '[oracl oracl "s" vision china "s" comprehen profess chine video site site set numbe r page contain blend short technic vision web conferenc technolog zone download cente r time oracl expert onlin answer question member excel onlin resourc miss where plea hit hee hee]',
 '[xin zhang red envelop wow realli rich i abl get cash big red xin zhang alipay wall et issu togeth everi day draw red envelop one day becom tyrant you tri luck http tcnr ztzye8]',
 '[wed wed industri employ train materi essenti train materi compani compani "s" new employ old employ team build train market materi easi understand great valu click det ail]',
 '[cloud comput great potenti nonstandard transact hinder rapid adopt march report le arn shanghai univ financ respect school announc establish central asian cloud comput center center aim upgrad technolog level domest cloud comput transact enhanc intellec tu properti right speak core competit research close collabor open market channel]',
 '[data nginx rewrit comment lostian]',
 '[secur short take microsoft get vagu window updat enterpri inform secur staff know] ',
 '[when midday leisur good accord consumpt consum durabl good durabl good tradit calc ul accord normal use time enough use certain commod analyz exampl use normal purcha r epeat categori good such mobil phone or replac such car or idl it seen altern replac idl also differ consum demand]',
 '[mario save]',
 '[xuan bao mummi "s" littl world genuin discount purcha buyer show face_blowing_a_ki ss face_blowing_a_kiss]',
 '[share rain song carat lover `` shrimp share music]',
 '[rule law inform secur manag system close smith barney rose 566 % report 1345 yuan chang hand 315 % http tcnrwmqo22]',
 '[i simpli tell us bit probabl close beta problem after sever round test smoothli re pair happen yesterday outbreak data contribut problem so repair strain i "m" sorri pa rent loyal test]',
 '[interdomain multicast hcnphcie 717 spar mode multicast topolog role sm dr dr regist first jump posit joinsm rpt perform posit last hop dr among recipi share tree form send report messag trigger dr rp hop hop join share tree form rp root receiv lea v share tree and sm rpt]',
 '[yuan median incom tone com look product phase there direct sell price yuan easi pu sh name id love name id qq 860369]',
 '[to move `` big revi 30 upgrad experi new ui interact design more profess sport tut ori scienc custom fit program know sportsman beauti togeth similar sport coach profes s sport movement peopl accompani around upgrad benefit big run come to move allaround sport partner]',
 '[esriuc esri speed fast end first day confer put video present offici websit watch onlin download but problem listen much food whim set gi subtitl group special traderel foreign video english subtitl poor english listen domest audienc greatli facilit]',
 '[read rail time travel via push cool network]',
 '[python c `` help "modul" "" `` queri document]',
 '[2014117 dew root `` beij confer super near full video share youku]',
 '[jiax citi administr reli govern cloud seek new inform technolog develop chine egov ern network]',
 '[t cyber âG¢âG¢secur busi issu chi confer told t denvarvduifatoliacyberseguritvabus

'[^ cyber â€‹â€‹secur busi issu cbi confer told ^ denysfudyifotoffacybersecurityabus
inessissu cbiconferencetoldwarwickashfordsecurityeditorlikeanyoth http tcnrlwgsa5]',
'[read the big bang fourth quarter `` season "s" climax sheldon armi drunken kiss â˜
…â˜…â˜…â˜…]',
'[from netea cloud music xu also said similar word my fortun lost life in fact progr
ess idea world deserv like expand univ theori everi atom away peopl will approach in
addit gratitud i think import thing cherish for belong reason manner go flow]',
'[recent microblog rare miss lot great god share bad]',
'[read come end `` good grasp rhythm rel tight alway worri some nation director lear
n south korean director give like â˜…â˜…â˜…â˜…]',
'[doge]',
'[illustr it rain small holiday qingm spring "s" may day holiday beckon us such good
weather take trip say stay away if fed crowd domest scenic spot travel track taken co
untri xiao bian today recommend valu money tourism destin may day eastern europ rose
kingdom `` bulgaria]',
'[use proper way cloud note easili synchron manag note termin tripl backup storag da
ta secur guarant free larg storag space unlimit growth after activ get free 2gb cloud
storag space hurri experi]',
'[]',
'[thi compani recruit programm profess titl in consid give]',
'[vank tianyu "to" solut third center shenzhen first hopsca `` share premi intern re
fer]',
'[prezi_ baidu encyclopedia prezi major oper quick action scale idea interest presen
t softwar it broke tradit singlelin time powerpoint use systemat structur approach in
tegr present present rout suddenli pull one object anoth object rotat movement visual
impact through multiend]',
'[nuclear begin knight blog]',
'[idri a languag depend type offici websit ah]',
'[tian we soon enter feedback economi `` broadband capit tian â€‹â€‹chairman china g
reen compani annual meet said with futur ubiquit intellig termin comput power data be
com import asset manufactur circul consum record form close loop feedback soon enter
economy]',
'[elk stack best practic weibo share record oper mainten by pulpit effici oper maint
en group via instapap]',
'[suddenli flash thought i daughter i want marri person care older singl young men c
ontinu codeword â€‹â€‹]',
'[in middl night boss sent piec architectur diagram said let "s" look feel mean laug
h cri]',
'[* featuretoggl 32 relea *
aminorreleaseofmyopensourcefeaturetogglelibraryhasjustbeenreleasedtonugetversion32was
resultofuserrequeststobeabletoco http tcnrlgaahi]',
'[â ' garlic fri bacon slice â€‹â€‹bacon shred onion garlic cut â ' ¡ wok cool oil m
eat scoop transpar lamp volum â ' ¢ cook littl soy sauc onion ginger pepper saut last
â ' £ littl garlic salt stir chicken]',
'[eh101 helicopt _ baidu encyclopedia eh101 merlin `` multipurpo helicopt develop eu
ropean agusta westland success maiden flight june allweath capabl it use antisubmarin
escort search rescu airborn earli warn electron countermeasur share baidu encyclopedi
a]',
'[until day compani colleagu understand alway like sing first two sentenc blue lotu
xu wei obviou upset code written noth stop yearn freedom `` the road long mani unknow
n]',
'[share pictur]',
'[univ roommat gave someth at time i scare tear tear tear]',

'[web crawler analysi pyspid exampl python bole onlin ``]',
'[mother know chang drink yo fa i]',
'[i sent book php core technolog best practic `` click receiv]',
'[dubbo learn process share experi implement principl brief dubbo share experi prefa c depart last year began varieti transform first step servic modul primari]',
'[liu shi red envelop wow realli rich i abl get cash issu red envelop alipay shi shi wallet everi day draw red envelop one day becom tyrant you tri luck http tcnrzjsp5w]'
,
'[one hour build person websit blog bole onlin ``]',
'[]',
'[pan shiyi red realli true i drew red envelop pan shiyi quick taxi provid fast taxi yuan red envelop `` said haunt red year quickli seiz berserk ah http tcnrzg4tgo]',
'[saw 10g interstellar us drama alway touch]',
'[today i micropl sign gain 183m free space good luck index star tri luck microdisk sina "s" brand cloud storag larg storag space download massiv resourc mobil phone dat a comput synchron micro unlimit come experi]',
'[overtim realli tire love life less overtim]',
'[as long benefit enjoy liter level turn tyrant depend think littl excit happi child ren small partner speed crowd collect rank microblog exclu courtesi http tcnrzpg1ck]'
,
'[hou vote to end spi agenc "" bulk collect phone data an anonym]',
'[charact ah sicstu great peopl recogn i use bug report well money back gave free pe rson licen zhu han peng]',
'[bilibili luo day accord origin song antiqu right royal world origin up main pv pay shell turtl]',
'[stock market crash today stuck sweet potato oschina said osc "s" theme respon stoc k market to see subject true love opensourc chine new topic]',
'[there crook fraudul messag sent]',
'[i radar probe ice microbo leida receiv piec mcdonald "s" mccafe fruit snow ice lar g halfpric 95 yuan coupon quit forc ah come radar probe explor varieti benefit took t urn attack surpri side]',
'[howev bloom reward friend life]',
'[tang three littl red envelop fact true i drew red envelop tang three littl faster taxi provid fast taxi yuan red envelop `` year goat yet come firstcom good luck come tri luck http tcnrzdrncx]',
'[firefox track protect technolog reduc page load time % mozilla develop best paper award web 20 secur privaci confer paper pdf introduc firefox track protect technolog track]',
'[quick taxi voucher see happi camp voucher prize shake nonstop march 0022 see happi camp wu yifan william chan yang yang shake million quick taxi voucher now must take t axi voucher i poke receiv]',
'[data driven secur read chapter read twice r transfer code book python increasingli feel book realli good read book combin combat effect]',
'[there boss call xue bao xue bao interest success programm cock wire cock wire counterattack boss lowkey phone seem buy tv send the compani list want pull wind vote guess mani sale go employ drive bmw cadillac i know pack rat recycl use lifan nima gl ass broke know middl laugh]',
'[mark]',
'[rubi red envelop wow realli rich i abl get cash red envelop sent rubi alipay walle t everi day draw red envelop one day becom tyrant you tri luck http tcnrzyeoy3]',
'[rain rain lover carat anhui satellit tv spring festiv even 110508_the best pattaya station crazi crazi crazi crazi crazi crazi finish sleep bye bye bye bye]',

```
 '[can live cuisin]',
 '[nanj mani xctf final open class geekpwn safeti activ past two year begin engag joi
nt togeth]',
 '[read method histori complet studi method]',
 '[imit peopl stupid game patch play enigma broken irregular oper extent]',
 '[yolk yuan year kevin___zi time pass easi brief moment mind help across year epitom
hard forc fun time i suddenli felt littl sad time yeah retain fengyun vibrant heart l
ush time cheer cheer cheer friendship forev]',
 '[snow defend knife row nine new replac old charact `` like have come listen share h
imalaya good voic]',
 '[men chen bo reduc predict share hunt read]',
 '[state council layout cloud comput cloud servic revenu increa % report state counci
l recent issu promot cloud comput innov develop inform industri cultiv new form opini
on `` hereinaft refer opinion `` accel deploy cloud comput creat inform industri new
format cloud comput becom import support china "s" inform technolog import form const
ruct network power]',
 '[guid prototyp ml nlp code with us a tutori seri `` focu natur languag process mach
in tutori seri black box order avoid much possibl sourc lesson work with text `` extr
act inform content document collect prefrequ statist portion]',
 '[share video]',
 '[joseph reluctantli tinkl front life teambit frontend engin leav question noncomput
profess reli commun continu grow csdnnet]',
 '[one zombi powder yesterday today fell back]',
 '[british tang intellig control annual report disclosur notic cauthor]',
 '[i rainjihoon present flower two valu â€‹â€‹of love feel meng meng da rice circl pr
o come send flower aid]',
 '[i receiv microblog grade exclu ceremoni red envelop everyon upgrad welfar macbook
iphone6 big name sign ceremoni art ticket coupon million cash rose level enjoy privil
eg put life pinnacl come pick]',
 '[crestron site supervi oper]',
 '[share pictur]',
 '[no money delay day hell yet receiv food fortun brother leav mom lose social secur]
',
 '[under lightyear parti line develop independ b2c websit discuss address registr
method updat]',
 '[knew chong yu headquart reloc wangj snow jiejia 2015022811 new land wit new hope n
ew journey wit new dream new pace wit new splendor for past know man speak thousand m
ile weather wave farewel old futur creat peac mind tower build make vulner welcom att
ent know chong yu yang jilong zhao cosin hi_heig]',
 '[whi sometim better http http as secur compani often site stormpath develop ask que
stion best practic secur frequent ask question i whether]',
 '[raspberri pi attend]',
 '[recommend singl one by one]',
 '[who rememb thing would realli exist ``]',
 '[articl deci tree machin learn seri machin learn packag user r deci tree `` base r]
',
 '[gitcaf guess next laugh cri doge forgiv bohemian laugh cri]',
 '[share track netea cloud music]',
 '[i use oneclick checkin `` applic sign microdisk android add 182m space applic may
automat sign post letter like talk network disk bb site see detail http cn zjiuanq]',
 '[distribut cloudba machin learn collect save big data `` becom common requir mani a
pplic when size explo growth data distribut storag becom necessari collect data cente
```

r nonprocess lead natur distribut storag use distribut comput distribut data construc
t machin learn ml solut becom particularli essenti]',
 '[just regist litchi fm cute]',
 '[listen word chairman play phone earli child sleep]',
 '[real madrid champion leagu lore hero abandon buyout exposur purcha million transfe
r aguero `` i say real madrid realli one buy unexpect still find someon premier leagu
stand shooter time shooter wang unstabl state obviou peak aguero real madrid clutch l
ot money world footbal dignitari world footbal "s" actual one give two side realli sa
d]',
 '[a fun app best shall appli sister partner children time look recommend fan bingb m
assu sicong three friend bo grin]',
 '[peke univ seven year first time literatur read room]',
 '[if everyon like half hour ahead consciou railway station railway station could
block]',
 '[]',
 '[in addit one understand stori happi sad mani person feel]',
 '[nvidia open up cpuba physx code nvidia decid open cpuba implement physx 33 http tc
nrwevlia]',
 '[long time updat today came back toss long empti harvest lengnuanzizhi human natur
happen reason]',
 '[scallop punch the first day learn word]',
 '[for peopl ideal give like make long time let go so either endur pain glori aliv fu
ll regret quietli grow old]',
 '[cloud secur multimedia]',
 '[thi still see weather choo road travel walk]',
 '[it figur theeyetrib show smart watch eye tracker ceosunealstrup give explan help u
ser better manag applic small screen save smart watch electr consumpt mix accel eye t
racker start user watch look screen]',
 '[basic knowledg content area â€‹â€‹experti eh staff repr deep understand]',
 '[hao feng technolog stock warren stock stock financ invest bank stock dealer requir
minimum capit 100000 yuan monthli profit play % qq group]',
 '[good morn greet life kind rhythm must light shadow left right sunni rain tast chan
g without fierc twist turn fan submiss submiss visit good morn xinyu topic]',
 '[i i forget hurt heart realli hate ah tear]',
 '[phone shell ugli even fade thrown]',
 '[wordpress default crosssit vulner subject thi vulner allow attack use theme built
ico icon gener file genericon carri attack research institut said applic theme plugin
featur affect known affect default theme twentyfourteen plug jetpack instal one milli
on]',
 '[microsoft "s" visual studio exten github]',
 '[today i one hundr word cut back ielt word back day i "ll" abl see world left eye s
troke cheek left hand gentli right hand count money bank card `` translat english i "
m" actual poet back word origin word cut school bulli certif]',
 '[beij recruit oracl oracl univ sale recruit five year nation grid insur industri cu
stom experi interest plea contact marymeng oraclecom love]',
 '[share great singer li jian song]',
 '[]',
 '[are big topic amount ban said ban realli serv final time look chai jing haze surve
y dome fate subtitl english chai jing under dome english subtitl youtub]',
 '[share pictur helloxli]',
 '[how evalu left ear rat ali p9 leav ah]',
 '[web attitud print layout]',
 '[drug administr employ collect secur charg build strong firew]',

'[not memori wonder worth rememb long river year peopl thing ruthlessli swept away h
owev everyth relat youth alway settl bottom river becom indel memori we obsess mayb t
hing peopl lost dream passion tong hua go back one "s" youth ``]',
'[had walk word back 15year maximum target start]',
'[in fact peopl belong arachnida]',
'[just suddenli i rememb i want keep xiaoya]',
'[if depend peopl love left go back good care plan i realli want go back courag leav
]',
'[limit bind microblog movement win bracelet gift limit bind microblog movement win
bracelet gift]',
'[chi chi ãƒ¥ ãƒ¥ rousseau rousseau japan subtitl moriya onlin tegoshi yuya hd mv
yin yue taiwan share yin yue taiwan]',
'[pm deal peopl fill station point 8 point effect mental if hold everyth mental poin
t point result other demand slight mistak make unaccept conver hold eight point mind
even parti commit achiev point also hit 8fold ask eight point encourag time manner fi
nd better imagin minut mental reach result]',
'[day earli morn avalanch almost helpless stop ah]',
'[keenteam three consecut win lu student adjourn keen team record team defeat unit q
qpcmgr pdf reader within second obtain system privileg thi keen team pwn2own fifth ch
ampionship game congratul promised_lu qqpcmgr chine hacker three consecut rock509 big
bullfrog keen]',
'[under china "s" world order huge spend oversea help china replac unit state europ
number one gold master develop countri from china "s" foreign invest increa nearli 10
fold help win new alli enhanc trade access oil natur resourc i think term data inform
present interact stori scroll map way prai easi understand sourc http tcnrlan4k4]',
'[collect shandong dazhong guy rensan jun honor guard captain exclu interview lee ta
o famili share headlin today]',
'[openstack contributor littl troubl ah]',
'[it panason develop take firstfocu `` antiamerican lytro camera compani introduc in
dustri "s" first consum light field camera relea last year power secondgen product il
lum first camera refocu concept field imag brought lot inspir]',
'[veri long patient see deep understand python metaclass metaclass python bole onlin
``]',
'[ye you can blame your pointyhair boss on peter principl nerval "s" l]',
'[togeth see sea met beauti world togeth i issu travel travel road knowledg insight
if interest travel experi see forward microblog draw red win cash travel rout prize y
o]',
'[do see nobuo account risk ann chong er almost outsid home]',
'[huang gang logist suppli chain red envelop readili drawn red envelop hand i "ve" d
rawn huang gang logist suppli chain red envelop cash proud mei xue wu issu togeth oh
thi begin said billion red envelop do say i continu draw red envelop go]',
'[i micro disk find file awesom `` innodb transact lock mvccpdf `` i download see]',
'[share recent done h5 anim plea scan follow qr code]',
'[i wrote new articl websocket protocol analysi `` share simpl book]',
'[do dare challeng anim farm colonel number micro wipe three station "s" rank rose n
o hardwon victori keep]',
'[new year red envelop everyon send i wish fortun red envelop]',
'[new year "s" day fifteen laughter cri laughter cri laughter cri]',
'[]',
'[appear dew owo thi take time laugh cri laugh cri laugh cri]',
'[road java chine garbl solut four java code conver process front three blog focu in
troduc charact encod issu three blog blogger preliminari understand variou charact en

cod understand question java chine must understand but begin understand follow see ht
tp tcnrwz15ui]',
 '[recent garbag imessag began rage]',
 '[xiao liang rain carat carat lover male god jung ji hoon presid tea ten million tho
ught solidifi air rai wind bless lone i care i "m" happi meet i think happi]',
 '[boo]',
 '[sap support collect ase hot topic syba daniel dave putz teach debug ase seri witho
ut land]',
 '[too cool us imperi master core technolog ah share netea news the new selfi camera
motion artifact lili sale ``]',
 '[it turn end i defin]',
 '[other day i bought alfr launchbar]',
 '[* openstack devstack chang ip address instal * question ihavedevstackinstalledonau
buntu1204andicouldgetloggedintodashboard nowichangedtheip http tcnr2nhoij]',
 '[gold silver gain gold silver invest rose stumbl fall rise market chang heart throa
t are face market shout slogan pit father thi time think technolog last word wait wit
h strength return trust winwin largest seek gold silver exchang foreign currenc instr
uctor 1604040889 real cartridg singl group call group experi group verif xc valep wtf
]',
 '[nanj bureau statist recent relea transcript develop nanj last year undoubtedli big
gest surpri servic sector valu ad servic achiev nanj last year 4925 billion yuan 115
percent previou year thi commend doubledigit growth among nation "s" subprovinci citi
growth rate first citi "s" gdp grew 101 % lead econom growth sunan push hand]',
 '[word elimin vocabulari sound smarter honestli absolut realli alway]',
 '[learn transform workplac know abl transform transform huge project peopl fail comp
let live progress project transform success life directli proport possibl other
transform make project owner â€¢ with legal help other transform project complet poss
ibl do take transform tofu]',
 '[feng tang `` thing grow car wash first episod latt talk radio `` like have come li
sten share himalaya good voic]',
 '[so mani year alway thought capabl deal littl guilti plummet mayb reason follow hea
rt]',
 '[peopl state face doom `` watercress moment app]',
 '[i updat ipad microblog client v371 â ' microblog page text optim easier turn comme
ntari prai â ' ¡ hot new comment featur â ' ¢ bug fix come experi app store download
http tcnh98rbi]',
 '[kaifa renam shenzhen scienc technolog]',
 '[microblog red envelop feel cheap chicken test cheaper black market larg quantiti]'
,
 '[develop roadmap outlin zz]',
 '[it front teeth show oldest method dental treatment ultra pain recent research grou
p univ bologna itali found borg oldest dental treatment year ago teeth relev paper pu
blish scientif report scientificreport ``]',
 '[zy desperado zhu haiq nest aitc de segment small qi di recent coast coast nat unit
ari unitari long dc congratul dear user congratul becom sina weibo `` recogn secondcl
ass offsit activ award lucki user plea visit http tcnra6rmiv check gift]',
 '[quick taxi voucher riddl lantern festiv answer announc today bird go fast taxi wei
bo microchannel see answer three riddl guess lantern happi send pack fast chip http t
cnrwmk4ym]',
 '[junip srx branch seri firew configur manag manual junip "s" srx seri firew base ju
no oper system secur product juno integr rout switch secur rich set network servic at
present compani "s" full line junip router http tcnrwkuxyd]',

'[easi nope find good job happi]',
'[today i micropl sign gain 109m free space good luck index star tri luck microdisk sina "s" cloud storag network drive]',
'[winter almost upon us kill boy let man born]',
'[ha ha ha ha barrag common vocabulari mark]',
'[publish articl reproduc bowen reproduc ni wan gong practic law outlin ``]',
'[microsoft imagin cup msp_ zhen heart cherish msp seaw broth twentytwo]',
'[academician expert offer advic map educ recent professor pla inform engin univ academician survey initi wisdom henan zhongyuan geograph inform technolog innov center cosponsor first map geograph inform henan provinc educ summit forum held henan univ]',
'[030 * * * * php q addresstoscriptphp use crontab run everi minut php script]',
'[share singl `` spirit machin cut knife without headphon look lyric come point cool dog music iphon version]',
'[grab grab the hottest music super artifact i final grab one thousandth lucki i wow did grab friend catch anxiou rememb buy next set alarm clock come share music mall buy chanc get white color valu â€‹â€‹of highest global music super phone yo]',
'[alyssa * fart * hacker friend find valu â· â· "s" stori]',
'[new year "s" slightli bmc softwar happi new year dear friend thank hor regular companion year ram excit bmc softwar sincer wish happi new year good luck famili happi forev wellb dure spring festiv bmc "s" technic support hotlin work usual 4001206164 protect guard reunion]',
'[magazin test perform test web servic in chapter show around problem solv perform test perform test web servic detail explan thing stage perform test http tcn rl6ztq7]',
'[enjoy blog era backbon ultim internet tv usher era today opportun home user demand factor aspect entertain result experi tv product gradual develop past singlefunct type televi today "s" internet tv color display aspect control method audiovisu experi internet etc great chang]',
'[the ctfstega today new cloth `` sb sb sb `` steganographi two problem first card get http tcnr2s8mlf i tell honor futur good doge]',
'[not complet movement peopl regular exerci usual larg physic gap calm __ lee five peopl climb baishishan three men two women togeth i alway fit physic fit better cheer run front they slow look back man woman walk side side anoth man woman even hand go fml share know almost]',
'[simpl yet effici smooth i use weico microblog client android awesom end intellig night mode custom font well offlin assist help save traffic want tri better experi microblog jab link download weico]',
'[favorit datetimepick]',
'[microsoft "s" origin plan global nokia retail store renam accord econom time india report microsoft plan global scale 16000 origin nokia retail store renam nokia brand replac microsoft brand in way microsoft global fast expan `` retail coverag differ countri differ plan india brazil first detail http tcnrarfoj]',
'[2015esri develop confer six highlight earli exposur sourc china network day broke microsoft develop confer black & hololen holograph glass pocket everyon "s" eye if tactil feel hardwar product bring excit immedi innov softwar product let see possibl realiz well possibl brought http tcnrarvequ]',
'[most earli ventur compani "s" employ poor nobodi believ compani top player join startup compani gener follow one two peopl good startup compani rubbish]',
'[run 2006km time hour minut minut second pace 156532 kcal behind basic three kilomet go tire p]',
'[snail i fli shoe yo jump small seedl shoe auiset ss sisi user plea visit http tcnr adtmtk check gift]',
'[is art extrem control]',

 '[and calm think]',
 '[grab grab vote vote browser i use grab vote wang finger move bit ticket hand like ben & poor "s" larg nationwid celebr wang grab vote browser power everyth take home]',
 '[new carat lover video site rank tudou & letv no1 mighti mighti jesu jesu qq no you ku no zhejiang tv lover carat carat anhui tv lover rainjihoon rain rain carat lover]',
 '[it expert tip the phone grab red envelop identifi phish site xinhua beij februari report sun bo liu yan guo xiao tong spring festiv xinhua interview `` invit wellknown secur expert china "s" public secur univ professor wang dawei phd nation inform technolog center secur studi senior pank feng guest column togeth major user]',
 '[hua hua dish go campu card]',
 '[game control evolut sword `` feel histori evolut rpg littl bit surpri move star]',
 '[deep learn statist incomplet theorem interact markov random process nononsen lot good content mathemat know may ``]',
 '[come experi innov softwar wifi univ free access tool wifi artifact `` watch movi brush microblog play game traffic]',
 '[as long benefit enjoy liter level turn tyrant depend think littl excit happi children small partner speed crowd collect rank microblog exclu courtesi http tcnzoljzti]',
 '[ddd]',
 '[new year child draw atmosph spring festiv chine new year greet card pda know small paint partner bless draw ju matrix grew â€¢ ã…, â€¢ ùˆâœ§ draw need turn]',
 '[hand]',
 '[for 20yearold man learn balanc import no longer depend escap personnel face balanc relationship aspect life particularli import the socal balanc take care aspect must first find suitabl pivot point order heart weigh compon sort thing a matur person live sen balanc peopl]',
 '[there object move ing]',
 '[é ’ é ’ æ²³è¾¹è‰ winking_face_with_tongu via twitter]',
 '[html5 applic classic cool html5 jqueri anim applic exampl sourc code ad html5 jqueri make famili color use html5 creat brilliant dynam anim special effect thi articl share classic cool html5 jqueri anim applic like friend share collect]',
 '[ration born feynman stop mr feynman ``]',
 '[predict interv random forest predict interv random forest `` pdf http tcnr2xn4wa]',
 '[wish]',
 '[share track netea cloud music]',
 '[like person i happi togeth love person happi i think togeth]',
 '[go ioe mysql victori postgresql open sourc chine commun amazon "s" custom aurora mysql basi alleg perform mysql time 57 aw "s" wareh use postgresql good relat databa see scenario]',
 '[befor complain beat headphon mainten event twist turn final appl new machin end the result quit satisfactori seal affix]',
 '[german industri 40 huawei "s" full join medit `` whether "s" full join huawei propo german industri 40 even ibm made wisdom earth desir build connect i believ collabor across ecosystem innov ict system becom intellig world read download lofter client]',
 '[one bian fun tai css face question 20150302 aninterestingcssinterview pocket]',
 '[a cover ruin mood day also direct respon person joke no day alon ask class moren take cover return mood good simpli ignor smile came said group come collect cover nobodi said noth oh i "m" sorri ah night see contact ignor just person charg wool pull way smile]',

```
 '[broken line kite]',
 '[parti male taro sago good eat up point now ah ah ah ah ah \\\\ â‰§ â-½ â‰¦ burst]
',
 '[but van gogh iron open new station enabl old station shutter and pay high ticket m
oney need run longer distanc save littl journey bring thrill highsp rail]',
 '[phone taobao evolut architectur practic]',
 '[hungri what good user experi o2o era realli want]',
 '[innov safeti evalu system improv network secur tax risk manag]',
 '[* onlin boot camp modern php legaci applic php class * previou 20yearsofphpand1 au
thor paulmjonespostedon 20150701categori ev http tcnrlhl7cc]',
 '[your first graphql server]',
 '[new xiaogan council "s" work strengthen five step `` network inform secur manag]',
 '[laugh sell invoic fought south yeah]',
 '[archci implement support github gitlab webhook configur hook long code submit gtih
ub intern gitlab trigger continu integr docker mean refer implement code commit]',
 '[odp simpli lump fece modifi field type data type chang rebuild tabl abandon field
name key tmd author go along tabl id rebuild tabl tabl id chang origin user right los
t requir user reappli permiss tabl]',
 '[illog art student gener emot doge]',
 '[consequ feel origin thing constant retreat forget must go back two month]',
 '[good bad way]',
 '[farewel campu life pinnacl shanghai campu graduat arm video]',
 '[starbuck china realli drunk organ star label card waiter said send packet via i as
k she said latest activ heart chuckl look care home pit father soon expir]',
 '[spring time run correct run postur share headlin today]',
 '[do eat spici red pepper kindli plug]',
 '[yunnan societi sister beaten strip nude photo femal student expo yunnan societi si
ster beat student student strip nude photo hair qq space `` today morn net post publi
sh number forum yunnan wenshan post bar net post said wenshan prefectur fune counti h
igh school girl led student school suspect place]',
 '[translat literatur larg screen cool doge]',
 '[yesterday 25th state council execut meet focu develop nextgen inform technolog hig
hend cnc machin tool robot aerospac equip marin engin equip hightech ship rail transp
ort equip advanc energysav new energi vehicl power equip new materi biomedicin highpe
rform medic equip agricultur machineri equip sector]',
 '[zhengzhou univ colleg water resourc environ youth leagu school environ water resou
rc zhengzhou univ student union i "m" sorri view microblog user could find press f5 r
efresh tri see microblog]',
 '[live temperatur â „ ƒ humid % % north publish 0315 comfort recommend longsleev tsh
irt pant cloth singl shirt frail elderli advi forward knit longsleev shirt vest pant]
',
 '[seen jurass world `` my rate â˜…â˜…â˜…â˜… chong year chuangzei rebellion spread th
roughout countri fall empir sangui liaodong command led troop alreadi surrend lead
qingbingruguan put bring disast great qing dynasti douban app]',
 '[guangdong b 5f1m7 catch illeg juli 171758 shenzhen nanshan district nanhai road fo
rc chang lane compet drunk chang lane affect normal run vehicl video http tcnrlmfblt
traffic polic guard shenzhen shenzhen traffic polic mobil train battalion shenzhen na
nshan polic]',
 '[it seem take boo ok timemachin]',
 '[new institut remot sen digit earth global open recruit director]',
 '[cinis_ci red envelop finger gentli move cinis_ci red envelop togeth what pure love
hot]',
```

```
 '[i attend chine new year red envelop send car raffl chanc win million worth bride p
rice you also particip togeth link http tcnrwgaxoi]',
 '[zookeep work system architectur it technolog blog big learn learn total progress]'
,
 '[write good]',
 '[i share articl xun yu night]',
 '[to sinafood like one just littl feedback veget phone provid solut]',
 '[map point thing land resourc communiqu issu accord land resourc issu china land re
sourc communiqu `` number map geograph inform industri enterpri increa 34 % gdp growt
h 121 % still varieti lawless total time carri law enforc inspect]',
 '[microblog recruit peopl play thing confirm venu road recruit new colleagu think st
ori share microblog recruit peopl play see http tcnzjgy8nq]',
 '[i "m" asian song list rainjihoon highest chart song good music need action support
come favorit song]',
 '[realli fail]',
 '[in hot cat owner see articl girlfriend nude man pass internet let world know i gir
lfriend `` order prove girlfriend man still pretti fight laugh cri slag man girlfrien
d see articl singl doge face where "s" detail http tcnrwgytbz]',
 '[red envelop fertil mario mario fertil i gave red envelop stuf $ ta send lucki red
envelop togeth hope meet luck good togeth heart]',
 '[i find microdisk fabul file `` python for data analysi pdf `` i download see]',
 '[tang yan red envelop wow realli rich i abl get cash issu red envelop alipay tang y
an wallet everi day draw red envelop one day becom tyrant you tri luck http tcnrze0sx
r]',
 '[i rainjihoon present flower two valu â€‹â€‹of love feel meng meng da rice circl pr
o come send flower aid]',
 '[rain carat lover februari year cctv movi channel program guid februari 2015 2157 p
remier annual emot suspen drama dew root `` rain man show charm enjoy visual feast go
od like good like good like terrif terrif terrif electr electr electr]',
 '[quick taxi voucher carpool happi green counterpart whether spring wind rain small
quickli stuck intim taxi voucher readi i receiv stamp http tcnrwfcldj]',
 '[l_hudson14 hudson let "s" get championship next round]',
 '[quick taxi shake happi do think next taxi subsid quickli upgrad latest version fas
t taxi app new year "s" eve day seventh day 1200 1900 kidney million voucher imp
upon dip festiv deft receiv coupon http tcnrwi45el]',
 '[costa rica open water aquarium biolog male]',
 '[trump card agent agent colleg `` my rate â˜…â˜…â˜…â˜…â˜… excel watercress app]',
 ...]
```

In [16]:

```
#code for bag of words model
import numpy as np
import re

#for building vocabulary
def tokenize_sentences(sentences):
    words = []
    for sentence in sentences:
        w = extract_words(sentence)
        words.extend(w)

    words = sorted(list(set(words)))
```

```
    return words

def extract_words(sentence):
    ignore_words = ['a','b','c','d','e','f','g','h','i','j','k','l','m','n','o','p',
'q','r','s','t','u','v','w','x','y','z','A','B','C','D','E','F','G','H','I','J','K',
'L','M','N','O','P','Q','R','S','T','U','V','W','X','Y','Z']
    words = re.sub("[^\w]", " ",  sentence).split() #nltk.word_tokenize(sentence)
    words_cleaned = [w.lower() for w in words if w not in ignore_words]
    return words_cleaned

#function which returns feature vector
def bagofwords(sentence, words):
    sentence_words = extract_words(sentence)
    # frequency word count
    bag = np.zeros(len(words),dtype=int)
    for sw in sentence_words:
        for i,word in enumerate(words):
            if word == sw:
                bag[i] += 1

    return np.array(bag)
```

In [17]:

```
#building the vocabulary for the list created
vocabulary1 = tokenize_sentences(l)
```

In [18]:

```
l1 = [x for x in vocabulary1 if not (x.isdigit() or x[1:].isdigit())]
```

In [19]:

```
b=pd.DataFrame()
```

In [20]:

```
#constructing bag of words
a=[]

for i in range(0,10000):
    #b.append(bagofwords(df['content'].iloc[i], vocabulary1),ignore_index=True)
    a.append(bagofwords(df['content'].iloc[i], vocabulary1))
```

In [21]:

```
type(a)
```

Out[21]:

```
list
```

In [22]:

```
bow=np.asarray(a)
```

In [23]:

```
df_pol=pd.read_csv("E:\\DMA_PRE\\weibo_polarity.csv")
```

In [24]:

```
type(bow)
print(bow.shape)
```

```
(10000, 16792)
```

In [25]:

```
df_pol.columns
```

Out[25]:

```
Index(['Unnamed: 0', 'u_id', 'm_id', 'forward_count', 'comment_count',
       'like_count', 'content', 'date', 'time', 'content_media_count',
       'content_#_count', 'content_@_count', 'content_?_count',
       'content_!_count', 'content_length', 'content_emoji_count', 'hour',
       'min', 'sec', 'forward_min', 'forward_max', 'forward_median',
       'forward_mean', 'comment_min', 'comment_max', 'comment_median',
       'comment_mean', 'like_min', 'like_max', 'like_median', 'like_mean',
       'Unnamed: 0.1', 'content_spchar', 'non_emoji_content', 'en_content',
       'Unnamed: 1', 'url_rem', 'contentwurl', 'polarity'],
      dtype='object')
```

In [26]:

```
bow1=np.insert(bow,16791,df_pol["content_media_count"],axis=1)
bow2=np.insert(bow1,16792,df_pol["forward_median"],axis=1)
bow3=np.insert(bow2,16793,df_pol["comment_median"],axis=1)
bow4=np.insert(bow3,16794,df_pol["like_median"],axis=1)
bow5=np.insert(bow4,16795,df_pol["polarity"],axis=1)
```

In [29]:

```
train_bow=bow5[0:8000]
pred_bow=bow5[8001:10000]
```

Linear Regression Model using BOW and additional factors

In [31]:

```
X_train1=train_bow
X_test1=pred_bow
Y_train1=train_df[["forward_count","like_count","comment_count"]]
Y_test1=predict_df[["forward_count"]]
```

```
lm=linear_model.LinearRegression()
model=lm.fit(X_train1,Y_train1)
pred1=lm.predict(X_test1)
pred1=pred1.round()
pred1=(np.maximum(pred1,0.))
```

In [32]:

```
tbow=bow5[0:8000]
```

In [33]:

```
cvbow=bow5[8001:10000]
```

In [34]:

```
type(tbow)
```

Out[34]:

```
numpy.ndarray
```

In [35]:

```
df_pre.shape
dftrain=df_pre[0:8000]
dfcv=df_pre[8001:10000]
```

In [36]:

```
Y_test1=dfcv[["forward_count","like_count","comment_count"]]
```

In [37]:

```
np.savetxt("E://DMA_PRE//weibo_predict_resultbow.csv",pred1,delimiter=',',header="for
ward_count,comment_count,like_count",comments="")
result=pd.read_csv("E://DMA_PRE//weibo_predict_resultbow.csv")
```

In [38]:

```
train_real_pred = Y_test1
train_real_pred['fp']=result['forward_count'].values
train_real_pred['cp']=result['comment_count'].values
train_real_pred['lp']=result['like_count'].values
train_real_pred=train_real_pred.round()
print ("Score on the training set:{0:.2f}%".format(precision(train_real_pred.values)
*100))
```

```
Score on the training set:9.13%
```

```
C:\Users\DELL\Anaconda3\lib\site-packages\ipykernel_launcher.py:2: SettingWithCopyWar
ning:
A value is trying to be set on a copy of a slice from a DataFrame.
```

```
Try using .loc[row_indexer,col_indexer] = value instead

See the caveats in the documentation: http://pandas.pydata.org/pandas-docs/stable/ind
exing.html#indexing-view-versus-copy

C:\Users\DELL\Anaconda3\lib\site-packages\ipykernel_launcher.py:3: SettingWithCopyWar
ning:
A value is trying to be set on a copy of a slice from a DataFrame.
Try using .loc[row_indexer,col_indexer] = value instead

See the caveats in the documentation: http://pandas.pydata.org/pandas-docs/stable/ind
exing.html#indexing-view-versus-copy
  This is separate from the ipykernel package so we can avoid doing imports until
C:\Users\DELL\Anaconda3\lib\site-packages\ipykernel_launcher.py:4: SettingWithCopyWar
ning:
A value is trying to be set on a copy of a slice from a DataFrame.
Try using .loc[row_indexer,col_indexer] = value instead

See the caveats in the documentation: http://pandas.pydata.org/pandas-docs/stable/ind
exing.html#indexing-view-versus-copy
  after removing the cwd from sys.path.
```

In [39]:

```
result
```

Out[39]:

	forward_count	comment_count	like_count
0	5.0	0.0	0.0
1	157.0	0.0	34.0
2	0.0	0.0	0.0
3	0.0	0.0	-0.0
4	0.0	0.0	-0.0
5	0.0	0.0	0.0
6	12.0	432.0	184.0
7	68.0	43.0	27.0
8	71.0	40.0	37.0
9	19.0	7.0	10.0
10	0.0	0.0	0.0
11	36.0	4.0	0.0
12	128.0	86.0	27.0
13	0.0	2.0	0.0
14	0.0	7.0	14.0

	forward_count	comment_count	like_count
15	97.0	11.0	3.0
16	177.0	0.0	0.0
17	0.0	0.0	0.0
18	2.0	1.0	0.0
19	68.0	99.0	31.0
20	6.0	0.0	0.0
21	87.0	111.0	12.0
22	38.0	0.0	0.0
23	4.0	53.0	8.0
24	0.0	90.0	0.0
25	0.0	0.0	0.0
26	14.0	40.0	23.0
27	499.0	160.0	138.0
28	55.0	86.0	18.0
29	125.0	0.0	60.0
...	...	...	...
1969	50.0	0.0	10.0
1970	0.0	10.0	2.0
1971	56.0	54.0	35.0
1972	62.0	86.0	36.0
1973	161.0	164.0	40.0
1974	0.0	0.0	0.0
1975	0.0	106.0	0.0
1976	0.0	0.0	0.0
1977	80.0	102.0	21.0
1978	137.0	92.0	10.0
1979	0.0	0.0	2.0
1980	0.0	0.0	0.0
1981	232.0	147.0	117.0
1982	2.0	1.0	0.0
1983	106.0	0.0	66.0
1984	14.0	2.0	0.0
1985	56.0	92.0	0.0

	forward_count	comment_count	like_count
1986	0.0	77.0	0.0
1987	0.0	0.0	0.0
1988	0.0	0.0	0.0
1989	0.0	170.0	15.0
1990	0.0	72.0	100.0
1991	21.0	0.0	-0.0
1992	131.0	37.0	26.0
1993	0.0	0.0	0.0
1994	56.0	0.0	0.0
1995	0.0	0.0	0.0
1996	0.0	0.0	0.0
1997	228.0	0.0	29.0
1998	0.0	21.0	13.0

1999 rows × 3 columns

In [40]:

```
import pandas as pd
import numpy as np
from sklearn import linear_model
from sklearn.model_selection import train_test_split
from sklearn.metrics import accuracy_score
from matplotlib import pyplot as plt
import statsmodels.api as sm
```

In [41]:

```
#from evaluation import precision
```

In [42]:

```
df1=pd.read_csv("E:\\DMA_PRE\\PREPROCESSED.csv")
df_bow=pd.read_csv("E:\\DMA_PRE\\bow_df.csv")
```

In [43]:

```
#result = pd.concat([df1, df], axis=1)
```

In [44]:

```
import import_ipynb
from evaluation import precision
from runTime import runTime
```

BAG OF WORDS USING COUNTER VECTORIZER

In [45]:

```
df1=pd.read_csv('E:\\DMA_PRE\\pre_bow.csv')
train_df=df1[0:8000]
train_df.shape
```

Out[45]:

```
(8000, 10)
```

In [46]:

```
predict_df=df1[8001:10000]
```

In [47]:

```
train_l=[]
for i in range(0,8000):
    train_l.append(df_pre['content'].iloc[i])
len(train_l)
```

Out[47]:

```
8000
```

In [48]:

```
pred_l=[]
for i in range(8001,10000):
    pred_l.append(df_pre['content'].iloc[i])
len(pred_l)
```

Out[48]:

```
1999
```

In [49]:

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

In [50]:

```
vect=CountVectorizer()
```

In [51]:

```
vect.fit(train_l)
```

Out[51]:

```
CountVectorizer(analyzer='word', binary=False, decode_error='strict',
        dtype=<class 'numpy.int64'>, encoding='utf-8', input='content',
```

```
        lowercase=True, max_df=1.0, max_features=None, min_df=1,
        ngram_range=(1, 1), preprocessor=None, stop_words=None,
        strip_accents=None, token_pattern='(?u)\\b\\w\\w+\\b',
        tokenizer=None, vocabulary=None)
```

In [52]:

```
x=vect.transform(train_l)
```

In [53]:

```
#vocabulary=vect.get_feature_names()
#print(vocabulary)
```

In [54]:

```
print(x[0,:])
```

```
  (0, 1593) 1
  (0, 1608) 1
  (0, 3688) 1
  (0, 3923) 1
  (0, 4535) 1
  (0, 5181) 1
  (0, 5286) 1
  (0, 6171) 1
  (0, 6991) 1
  (0, 9310) 1
  (0, 9783) 1
  (0, 10657) 1
  (0, 11334) 2
  (0, 11648) 1
  (0, 13110) 1
```

In [55]:

```
type(x)
```

Out[55]:

```
scipy.sparse.csr.csr_matrix
```

In [56]:

```
arrbow=x.toarray()
```

In [57]:

```
vect = CountVectorizer(analyzer = "word",   \
                       tokenizer = None,    \
                       preprocessor = None, \
                       stop_words = None,   \
                       max_features = 100)
```

In [58]:

```
print(vect)
```

```
CountVectorizer(analyzer='word', binary=False, decode_error='strict',
        dtype=<class 'numpy.int64'>, encoding='utf-8', input='content',
        lowercase=True, max_df=1.0, max_features=100, min_df=1,
        ngram_range=(1, 1), preprocessor=None, stop_words=None,
        strip_accents=None, token_pattern='(?u)\\b\\w\\w+\\b',
        tokenizer=None, vocabulary=None)
```

In [59]:

```
from sklearn.preprocessing import PolynomialFeatures
from sklearn import linear_model
from sklearn.metrics import precision_score
```

Model 1

In [60]:

```
off_train_data_features = vect.fit_transform(train_1)
off_train_data_features = off_train_data_features.toarray()
off_train_data_forward = train_df.forward_count

off_test_data_features = vect.fit_transform(pred_1)
off_test_data_features = off_test_data_features.toarray()
off_test_data_forward = predict_df.forward_count

X_train1=off_train_data_features
X_test1= off_test_data_features
Y_train1=dftrain[["forward_count","like_count","comment_count"]]
Y_test1=dfcv[["forward_count","like_count","comment_count"]]

lm=linear_model.LinearRegression()
model=lm.fit(X_train1,Y_train1)
pred1=lm.predict(X_test1)
pred1=pred1.round()
pred1=(np.maximum(pred1,0.))
np.savetxt("E://DMA_PRE//weibo_predict_resultbow1.csv",pred1,delimiter=',',header="fo
rward_count,comment_count,like_count",comments="")
result1=pd.read_csv("E://DMA_PRE//weibo_predict_resultbow1.csv")
result1=result1.abs()
result1=result1.astype(int)
train_real_pred = Y_test1
train_real_pred['fp']=result1['forward_count'].values
train_real_pred['cp']=result1['comment_count'].values
train_real_pred['lp']=result1['like_count'].values
train_real_pred=train_real_pred.round()
print ("Score on the training set:{0:.2f}%".format(precision(train_real_pred.values)
*100))
```

```
Score on the training set:15.43%
```

```
C:\Users\DELL\Anaconda3\lib\site-packages\ipykernel_launcher.py:24: SettingWithCopyWa
rning:
A value is trying to be set on a copy of a slice from a DataFrame.
Try using .loc[row_indexer,col_indexer] = value instead

See the caveats in the documentation: http://pandas.pydata.org/pandas-docs/stable/ind
exing.html#indexing-view-versus-copy
C:\Users\DELL\Anaconda3\lib\site-packages\ipykernel_launcher.py:25: SettingWithCopyWa
rning:
A value is trying to be set on a copy of a slice from a DataFrame.
Try using .loc[row_indexer,col_indexer] = value instead

See the caveats in the documentation: http://pandas.pydata.org/pandas-docs/stable/ind
exing.html#indexing-view-versus-copy
C:\Users\DELL\Anaconda3\lib\site-packages\ipykernel_launcher.py:26: SettingWithCopyWa
rning:
A value is trying to be set on a copy of a slice from a DataFrame.
Try using .loc[row_indexer,col_indexer] = value instead

See the caveats in the documentation: http://pandas.pydata.org/pandas-docs/stable/ind
exing.html#indexing-view-versus-copy
```

Model 2

In [61]:

```
print(x.shape)
```

```
(8000, 14753)
```

In [62]:

```
x[1]
```

Out[62]:

```
<1x14753 sparse matrix of type '<class 'numpy.int64'>'
 with 25 stored elements in Compressed Sparse Row format>
```

In [63]:

```
x[2]
```

Out[63]:

```
<1x14753 sparse matrix of type '<class 'numpy.int64'>'
 with 9 stored elements in Compressed Sparse Row format>
```

In [64]:

```
train=df_pol[0:8000]
cv=df_pol[8001:10000]
```

In [65]:

```
off_train_data_features = vect.fit_transform(train_l)
off_train_data_features = off_train_data_features.toarray()

off_train_data_features1=np.insert(off_train_data_features,100,train["content_media_count"],axis=1)
off_train_data_features2=np.insert(off_train_data_features1,101,train["forward_median"],axis=1)
off_train_data_features3=np.insert(off_train_data_features2,102,train["comment_median"],axis=1)
off_train_data_features4=np.insert(off_train_data_features3,103,train["like_median"],axis=1)
#off_train_data_features5=np.insert(off_train_data_features4,100,train["polarity"],axis=1)
off_train_data_features6=np.insert(off_train_data_features4,104,train["content_emoji_count"],axis=1)
#off_train_data_forward = train_df.forward_count

off_test_data_features = vect.fit_transform(pred_l)
off_test_data_features = off_test_data_features.toarray()
off_test_data_features1=np.insert(off_test_data_features,100,cv["content_media_count"],axis=1)
off_test_data_features2=np.insert(off_test_data_features1,101,cv["forward_median"],axis=1)
off_test_data_features3=np.insert(off_test_data_features2,102,cv["comment_median"],axis=1)
off_test_data_features4=np.insert(off_test_data_features3,103,cv["like_median"],axis=1)
#off_test_data_features5=np.insert(off_test_data_features4,100,cv["polarity"],axis=1)

off_test_data_features6=np.insert(off_test_data_features4,104,cv["content_emoji_count"],axis=1)
#off_test_data_forward = predict_df.forward_count

X_train1=off_train_data_features6
X_test1= off_test_data_features6
Y_train1=dftrain[["forward_count","like_count","comment_count"]]
Y_test1=dfcv[["forward_count","like_count","comment_count"]]

lm=linear_model.LinearRegression()
model=lm.fit(X_train1,Y_train1)
pred1=lm.predict(X_test1)
pred1=pred1.round()
pred1=(np.maximum(pred1,0.))

np.savetxt("E://DMA_PRE//weibo_predict_resultbow3.csv",pred1,delimiter=',',header="forward_count,comment_count,like_count",comments="")
result3=pd.read_csv("E://DMA_PRE//weibo_predict_resultbow3.csv")
result3=result3.abs()
```

```
result3=result3.astype(int)
train_real_pred = Y_test1
train_real_pred['fp']=result3['forward_count'].values
train_real_pred['cp']=result3['comment_count'].values
train_real_pred['lp']=result3['like_count'].values
train_real_pred=train_real_pred.round()
print ("Score on the training set:{0:.2f}%".format(precision(train_real_pred.values)
*100))
```

```
Score on the training set:15.89%
```

```
C:\Users\DELL\Anaconda3\lib\site-packages\ipykernel_launcher.py:38: SettingWithCopyWa
rning:
A value is trying to be set on a copy of a slice from a DataFrame.
Try using .loc[row_indexer,col_indexer] = value instead

See the caveats in the documentation: http://pandas.pydata.org/pandas-docs/stable/ind
exing.html#indexing-view-versus-copy
C:\Users\DELL\Anaconda3\lib\site-packages\ipykernel_launcher.py:39: SettingWithCopyWa
rning:
A value is trying to be set on a copy of a slice from a DataFrame.
Try using .loc[row_indexer,col_indexer] = value instead

See the caveats in the documentation: http://pandas.pydata.org/pandas-docs/stable/ind
exing.html#indexing-view-versus-copy
C:\Users\DELL\Anaconda3\lib\site-packages\ipykernel_launcher.py:40: SettingWithCopyWa
rning:
A value is trying to be set on a copy of a slice from a DataFrame.
Try using .loc[row_indexer,col_indexer] = value instead

See the caveats in the documentation: http://pandas.pydata.org/pandas-docs/stable/ind
exing.html#indexing-view-versus-copy
```

Model 3

In [92]:

```
off_train_data_features = vect.fit_transform(train_1)
off_train_data_features = off_train_data_features.toarray()

off_train_data_features1=np.insert(off_train_data_features,100,train["content_media_c
ount"],axis=1)
off_train_data_features2=np.insert(off_train_data_features1,101,train["forward_median
"],axis=1)
off_train_data_features3=np.insert(off_train_data_features2,102,train["comment_median
"],axis=1)
off_train_data_features4=np.insert(off_train_data_features3,103,train["like_median"]
,axis=1)
off_train_data_features5=np.insert(off_train_data_features4,104,train["polarity"],ax
is=1)
```

```
#off_train_data_forward = train_df.forward_count

off_test_data_features = vect.fit_transform(pred_l)
off_test_data_features = off_test_data_features.toarray()
off_test_data_features1=np.insert(off_test_data_features,100,cv["content_media_count"
],axis=1)
off_test_data_features2=np.insert(off_test_data_features1,101,cv["forward_median"],ax
is=1)
off_test_data_features3=np.insert(off_test_data_features2,102,cv["comment_median"],ax
is=1)
off_test_data_features4=np.insert(off_test_data_features3,103,cv["like_median"],axis=
1)
off_test_data_features5=np.insert(off_test_data_features4,104,cv["polarity"],axis=1)

#off_test_data_forward = predict_df.forward_count

X_train1=off_train_data_features4
X_test1= off_test_data_features4
Y_train1=dftrain["forward_count"]
Y_test1=dfcv["forward_count"]

lm=linear_model.LinearRegression()
model=lm.fit(X_train1,Y_train1)
pred1=lm.predict(X_test1)
pred1=pred1.round()
pred1=(np.maximum(pred1,0.))

np.savetxt("E://DMA_PRE//weibo_predict_resultbow4.csv",pred1,delimiter=',',header="fo
rward_count",comments="")
result4=pd.read_csv("E://DMA_PRE//weibo_predict_resultbow4.csv")
result4=result4.abs()
result4=result4.astype(int)

train_real_pred = Y_test1
```

In [93]:

```
off_train_data_features = vect.fit_transform(train_l)
off_train_data_features = off_train_data_features.toarray()

off_train_data_features1=np.insert(off_train_data_features,100,train["content_media_c
ount"],axis=1)
off_train_data_features2=np.insert(off_train_data_features1,101,train["forward_median
"],axis=1)
off_train_data_features3=np.insert(off_train_data_features2,102,train["comment_median
"],axis=1)
off_train_data_features4=np.insert(off_train_data_features3,103,train["like_median"]
,axis=1)
off_train_data_features5=np.insert(off_train_data_features4,100,train["polarity"],ax
is=1)
#off_train_data_forward = train_df.forward_count
```

```
off_test_data_features = vect.fit_transform(pred_1)
off_test_data_features = off_test_data_features.toarray()
off_test_data_features1=np.insert(off_test_data_features,100,cv["content_media_count"
],axis=1)
off_test_data_features2=np.insert(off_test_data_features1,101,cv["forward_median"],ax
is=1)
off_test_data_features3=np.insert(off_test_data_features2,102,cv["comment_median"],ax
is=1)
off_test_data_features4=np.insert(off_test_data_features3,103,cv["like_median"],axis=
1)
off_test_data_features5=np.insert(off_test_data_features4,100,cv["polarity"],axis=1)
#off_test_data_forward = predict_df.forward_count

X_train2=off_train_data_features4
X_test2= off_test_data_features4
Y_train2=dftrain["like_count"]
Y_test2=dfcv["like_count"]

lm=linear_model.LinearRegression()
model=lm.fit(X_train1,Y_train1)
pred1=lm.predict(X_test1)
pred1=pred1.round()
pred1=(np.maximum(pred1,0.))

np.savetxt("E://DMA_PRE//weibo_predict_resultbow5.csv",pred1,delimiter=',',header="co
mment_count",comments="")
result5=pd.read_csv("E://DMA_PRE//weibo_predict_resultbow5.csv")
result5=result5.abs()
result5=result5.astype(int)
```

In [94]:

```
off_train_data_features = vect.fit_transform(train_1)
off_train_data_features = off_train_data_features.toarray()

off_train_data_features1=np.insert(off_train_data_features,100,train["content_media_c
ount"],axis=1)
off_train_data_features2=np.insert(off_train_data_features1,101,train["forward_median
"],axis=1)
off_train_data_features3=np.insert(off_train_data_features2,102,train["comment_median
"],axis=1)
off_train_data_features4=np.insert(off_train_data_features3,103,train["like_median"]
,axis=1)
off_train_data_features5=np.insert(off_train_data_features4,100,train["polarity"],ax
is=1)
#off_train_data_forward = train_df.forward_count

off_test_data_features = vect.fit_transform(pred_1)
off_test_data_features = off_test_data_features.toarray()
off_test_data_features1=np.insert(off_test_data_features,100,cv["content_media_count"
],axis=1)
off_test_data_features2=np.insert(off_test_data_features1,101,cv["forward_median"],ax
is=1)
```

```
off_test_data_features3=np.insert(off_test_data_features2,102,cv["comment_median"],axis=1)
off_test_data_features4=np.insert(off_test_data_features3,103,cv["like_median"],axis=1)
off_test_data_features5=np.insert(off_test_data_features4,100,cv["polarity"],axis=1)
#off_test_data_forward = predict_df.forward_count

X_train3=off_train_data_features4
X_test3= off_test_data_features4
Y_train3=dftrain["comment_count"]
Y_test3=dfcv["comment_count"]

lm=linear_model.LinearRegression()
model=lm.fit(X_train1,Y_train1)
pred1=lm.predict(X_test1)
pred1=pred1.round()
pred1=(np.maximum(pred1,0.))

np.savetxt("E://DMA_PRE//weibo_predict_resultbow6.csv",pred1,delimiter=',',header="like_count",comments="")
result6=pd.read_csv("E://DMA_PRE//weibo_predict_resultbow6.csv")
result6=result6.abs()
result6=result6.astype(int)
```

In [95]:

```
train_real_pred = pd.concat([Y_test1,Y_test2,Y_test3],axis=1)
train_real_pred['fp']=result4['forward_count'].values
train_real_pred['cp']=result5['comment_count'].values
train_real_pred['lp']=result6['like_count'].values
train_real_pred=train_real_pred.round()
print ("Score on the training set:{0:.2f}%".format(precision(train_real_pred.values)*100))
```

```
Score on the training set:16.20%
```

In [96]:

```
train_real_pred
```

Out[96]:

	forward_count	like_count	comment_count	fp	cp	lp
8001	0	0	0	0	0	0
8002	3	0	0	0	0	0
8003	0	1	0	12	12	12
8004	0	0	0	4	4	4
8005	0	0	0	4	4	4
8006	0	0	0	117	117	117

	forward_count	like_count	comment_count	fp	cp	lp
8007	2	0	0	0	0	0
8008	0	0	0	3	3	3
8009	3	1	2	0	0	0
8010	0	0	0	0	0	0
8011	0	0	0	8	8	8
8012	1	0	0	5	5	5
8013	0	0	0	0	0	0
8014	0	0	0	0	0	0
8015	0	0	18	0	0	0
8016	0	0	0	1	1	1
8017	0	0	0	1	1	1
8018	0	1	0	10	10	10
8019	0	0	0	0	0	0
8020	0	0	1	0	0	0
8021	0	0	0	0	0	0
8022	0	4	1	0	0	0
8023	3	0	0	4	4	4
8024	0	1	0	0	0	0
8025	0	0	0	31	31	31
8026	1	1	0	0	0	0
8027	0	2	2	9	9	9
8028	0	0	0	1	1	1
8029	0	0	0	3	3	3
8030	0	0	0	6	6	6
...	...	...	...	...	...	...
9970	2	1	17	0	0	0
9971	0	0	1	0	0	0
9972	0	0	0	14	14	14
9973	6	1	0	23	23	23
9974	34	0	10	0	0	0
9975	0	0	0	3	3	3
9976	0	0	0	3	3	3
9977	0	0	0	1	1	1

	forward_count	like_count	comment_count	fp	cp	lp
9978	0	0	0	9	9	9
9979	0	0	0	0	0	0
9980	3	1	0	34	34	34
9981	0	0	0	1	1	1
9982	1	1	0	0	0	0
9983	0	0	0	0	0	0
9984	0	1	4	0	0	0
9985	0	0	0	0	0	0
9986	1	0	0	8	8	8
9987	0	0	0	0	0	0
9988	61	2	4	0	0	0
9989	0	0	0	1	1	1
9990	1	1	1	2	2	2
9991	1	0	0	38	38	38
9992	0	0	0	0	0	0
9993	0	0	0	0	0	0
9994	0	2	0	0	0	0
9995	0	1	0	0	0	0
9996	0	0	0	0	0	0
9997	0	0	0	0	0	0
9998	0	0	0	0	0	0
9999	1	0	0	0	0	0

1999 rows × 6 columns

In [147]:

```
from sklearn.ensemble import GradientBoostingRegressor
```

In [148]:

```
train=df_pol[0:8000]
cv=df_pol[8001:10000]
```

In [149]:

```
X_train6=train[["content_media_count","forward_median","comment_median","like_median"
]]
X_test6= cv[["content_media_count","forward_median","comment_median","like_median"]]
Y_train6=train[["forward_count"]]
Y_test6=cv[["forward_count"]]
```

In [150]:

```
gbrt=GradientBoostingRegressor(n_estimators=100)
gbrt.fit(X_train6, Y_train6)
y_pred1=gbrt.predict(X_test6)
y_pred1=y_pred1.round()
y_pred1=(np.maximum(y_pred1,0.))

np.savetxt("E://DMA_PRE//weibo_predict_resultbow6.csv",y_pred1,delimiter=',',header="
forward_count",comments="")
result6=pd.read_csv("E://DMA_PRE//weibo_predict_resultbow6.csv")
result6=result6.abs()
result6=result6.astype(int)
```

```
C:\Users\DELL\Anaconda3\lib\site-packages\sklearn\utils\validation.py:578: DataConver
sionWarning: A column-vector y was passed when a 1d array was expected. Please change
the shape of y to (n_samples, ), for example using ravel().
  y = column_or_1d(y, warn=True)
```

In [151]:

```
X_train7=train[["content_media_count","forward_median","comment_median","like_median"
]]
X_test7= cv[["content_media_count","forward_median","comment_median","like_median"]]
Y_train7=train[["like_count"]]
Y_test7=cv[["like_count"]]

gbrt=GradientBoostingRegressor(n_estimators=100)
gbrt.fit(X_train7, Y_train7)
y_pred2=gbrt.predict(X_test7)
y_pred2=y_pred2.round()
y_pred2=(np.maximum(y_pred2,0.))

np.savetxt("E://DMA_PRE//weibo_predict_resultbow7.csv",y_pred2,delimiter=',',header="
like_count",comments="")
result7=pd.read_csv("E://DMA_PRE//weibo_predict_resultbow7.csv")
result7=result7.abs()
result7=result7.astype(int)
```

```
C:\Users\DELL\Anaconda3\lib\site-packages\sklearn\utils\validation.py:578: DataConver
sionWarning: A column-vector y was passed when a 1d array was expected. Please change
the shape of y to (n_samples, ), for example using ravel().
  y = column_or_1d(y, warn=True)
```

In [152]:

```
X_train8=train[["content_media_count","forward_median","comment_median","like_median"
]]
X_test8= cv[["content_media_count","forward_median","comment_median","like_median"]]
Y_train8=train[["comment_count"]]
Y_test8=cv[["comment_count"]]
```

```
gbrt=GradientBoostingRegressor(n_estimators=100)
gbrt.fit(X_train8, Y_train8)
y_pred3=gbrt.predict(X_test8)
y_pred3=y_pred3.round()
y_pred3=(np.maximum(y_pred3,0.))

np.savetxt("E://DMA_PRE//weibo_predict_resultbow8.csv",y_pred3,delimiter=',',header="
comment_count",comments="")
result8=pd.read_csv("E://DMA_PRE//weibo_predict_resultbow8.csv")
result8=result8.abs()
result8=result8.astype(int)
```

```
C:\Users\DELL\Anaconda3\lib\site-packages\sklearn\utils\validation.py:578: DataConver
sionWarning: A column-vector y was passed when a 1d array was expected. Please change
the shape of y to (n_samples, ), for example using ravel().
  y = column_or_1d(y, warn=True)
```

In [157]:

```
train_real_pred = pd.concat([Y_test6,Y_test8,Y_test7],axis=1)
train_real_pred['fp']=result6['forward_count'].values
train_real_pred['cp']=result8['comment_count'].values
train_real_pred['lp']=result7['like_count'].values
train_real_pred=train_real_pred.round()
print ("Score on the training set:{0:.2f}%".format(precision(train_real_pred.values)
*100))
```

```
Score on the training set:99.51%
```

In [156]:

```
train_real_pred
```

Out[156]:

	forward_count	comment_count	like_count	fp	cp	lp
8001	0	0	0	0	0	0
8002	0	0	0	0	0	0
8003	0	0	0	0	0	0
8004	0	0	0	0	0	0
8005	0	0	0	0	0	0
8006	0	0	0	0	0	0
8007	0	0	0	0	0	0
8008	0	0	0	0	0	0
8009	0	0	0	0	0	0
8010	0	1	0	0	0	0

	forward_count	comment_count	like_count	fp	cp	lp
8011	0	0	0	0	0	0
8012	0	0	0	0	0	0
8013	1	1	0	0	0	0
8014	0	0	0	0	0	0
8015	0	0	0	0	0	0
8016	0	0	0	0	0	0
8017	0	0	0	0	0	0
8018	0	0	0	0	0	0
8019	0	0	0	0	0	0
8020	0	0	0	0	0	0
8021	0	0	0	0	0	0
8022	0	0	0	0	0	0
8023	0	0	0	0	0	0
8024	0	0	0	0	0	0
8025	0	0	0	0	0	0
8026	0	0	0	0	0	0
8027	0	0	0	0	0	0
8028	0	0	0	0	0	0
8029	0	0	0	0	0	0
8030	0	0	0	0	0	0
...	...	...	...	...	...	...
9970	0	0	0	0	0	0
9971	0	0	0	0	0	0
9972	0	0	0	0	0	0
9973	0	0	1	0	0	0
9974	0	0	0	0	0	0
9975	0	0	0	0	0	0
9976	0	0	0	0	0	0
9977	0	0	0	0	0	0
9978	0	0	0	0	0	0
9979	0	0	0	0	0	0
9980	0	0	0	0	0	0
9981	1	0	0	0	0	0
9982	0	0	0	0	0	0

	forward_count	comment_count	like_count	fp	cp	lp
9983	0	0	0	0	0	0
9984	0	0	0	0	0	0
9985	0	0	0	0	0	0
9986	0	0	0	0	0	0
9987	0	0	0	0	0	0
9988	0	0	0	0	0	0
9989	0	0	0	0	0	0
9990	0	0	0	0	0	0
9991	0	0	0	0	0	0
9992	0	0	0	0	0	0
9993	0	0	0	0	0	0
9994	0	0	0	0	0	0
9995	0	0	0	0	0	0
9996	0	0	0	0	0	0
9997	0	0	0	0	0	0
9998	0	0	0	0	0	0
9999	0	0	0	0	0	0

1999 rows × 6 columns

In [1]:

```
import pandas as pd
import numpy as np
import re
from sklearn import linear_model
from sklearn.linear_model import Lasso
from sklearn.model_selection import train_test_split
from sklearn.metrics import mean_squared_error
from matplotlib import pyplot as plt
from textblob import TextBlob
import statsmodels.api as sm
```

In [2]:

```
import import_ipynb
from evaluation import precision
from runTime import runTime
```

```
importing Jupyter notebook from evaluation.ipynb
importing Jupyter notebook from runTime.ipynb
```

In [3]:

```
df=pd.read_csv("G://preprocessed1L.csv")
```

In [4]:

```
df
```

Out[4]:

	u_id	m_id	forw
0	d38e9bed5d98110dc2489d0d1cac3c2a	7d45833d9865727a88b960b0603c19f6	0.0
1	fa13974743d3fe6ff40d21b872325e9e	8169f1d45051e08ef213bf1106b1225d	0.0
2	da534fe87e7a52777bee5c30573ed5fd	68cd0258c31c2c525f94febea2d9523b	0.0
3	e06a22b7e065e559a1f0bf7841a85c51	00b9f86b4915aedb7db943c54fd19d59	0.0

	u_id	m_id	forw
4	f9828598f9664d4e347ef2048ce17734	c7f6f66044c0c5a3330e2c5371be6824	0.0
5	d80f3d3c5c1d658e82b837a4dd1af849	bfc0819b83ec59ce767287077f2b3507	0.0
6	f349a67d1cd7c8683c5bbc5f8486e193	83674a60e5310195fc35d97ea8f45c46	0.0
7	24b621c98f2594b698c0b1d60c9ae6db	2cbd3d514ed5ad3dab81aa043c8b3d0a	0.0
8	e44d81d630e4f382f657e72aa4b685da	8a88a25f9f26ed9f79080eaacc1a8668	0.0
9	fbe6c953632e1b3dda66cf6118b6ab12	f359a74cb4ac6150a3af8325eda04ea0	0.0
10	f9a3ca6bc1e75d173cfc98ec4b108072	c7bc3445e8b90db8cc5e045f606dc1ee	21.0
11	3c68bbb9da57fcc752c8a493d91bdd3a	77e14cf9d460715e84c51747c3641a9b	0.0
12	104e8d55e98eb3cd834810088af039fe	ee0b2c9d35bfeb0fbc5b3a8677f4a18c	9.0
13	0d15005d6397fb5ce1d45e7c834f7370	9c954d63fcfea19dca8d81a4f3b53861	0.0
14	875a4a77b339d93f819e2c4de5bd0b57	f2cdcdbcec9ff47cbb3c6a636e4b92a3	0.0

	u_id	m_id	forw
15	380a2219670f50dc87efce3380bea6e8	46f10244d02afa85d12346ce28e3cec5	0.0
16	b9b88b0fc105fb08a552e782afa4342e	cb907eb1bdbc198ed0944cc3b7e24f91	0.0
17	f18eb14365c0d7248fab1b9c464f4e70	096543bd8746869982d1a7557164dd0d	0.0
18	0fc17bf5e2dc789dd48505df1f5b14fd	4c1e2418127811d212d0e3867a99db3e	0.0
19	dd749a5af07c04ce7de451273a983671	419dd71d562883ef836e774bc3f4e163	0.0
20	a984551b159fcdc0a48f9e38ecb1488f	baa0051d359555601ab61df684787f0f	0.0
21	2e0467b73d0f6f9e5607a6174581fdd8	2fd200a7f670138c2026091c3b01532a	0.0
22	819656f05994b00b7260daf7346586a7	95590e88cac5d8c9d1a496bc3bd42f07	6.0
23	91ce7c63b272f2037a3e702c10163fa3	8b4e85a881afaff91f276eac7bfb6604	0.0
24	4680e73f9e7a6b87dec62a86a7821c17	b2db095af290b3a36cf798a3e17528d8	0.0
25	976e85e3ededdd9b2c2a3179eb7ae8ab	9540ee0cf7ccfae523020c8025e7095f	0.0

	u_id	m_id	forw
26	6623347e5f19f35f2d02ad515b96524c	9a2f48a870843d1964a03c6642b309d5	0.0
27	cf727e70b6661387cf6aadf01d2eb32c	bff281350f035db0e84c25394865d86a	0.0
28	de0836c1c5d40a5cae64a964a0b54894	c3345fd72cad53ca9bffd63634170ba0	0.0
29	c8848f18da5911d0389c3ac70fe13204	fa352495e646a3f7ff979267c490fd89	0.0
...	...	...	...
99970	b7261d402db4a731e8ea832699333ab7	3c64d713aa3f533f2b696bb4e9a26f4f	0.0
99971	aee12f3eb0cae884ae6c470968357f0f	5a3c9abf2f272895af331b54c78b0a14	0.0
99972	8b3250e43d33021dd848c04f963a96ca	8954335080c5ee96d07c7fc9894425db	1.0
99973	875a4a77b339d93f819e2c4de5bd0b57	613062a089f25ad265c6545e6141f942	0.0
99974	fb0971d7bc981be9878e44f185b4ff70	a733f59ae091283276c40eb8dfddecd4	54.0
99975	70c198acee07ce8fb7bcad0d19761abe	197362440af5951ba4d8db88649d4ea4	0.0
99976	69a108e7167bddb6fd33d720d8ba5b0e	35f40952b56ceeffa9ea9eee953358b1	0.0

	u_id	m_id	forw
99977	b139236e024e15611377da5001f1addd	04853ce3abcd155419648b3bbf042331	29.0
99978	eeabfd5e894c191158402264553f5bb7	7449368fd0a194b43e4073edf20da3c6	0.0
99979	2aa971a0a69411b2a276eb4723eef2ed	42fc51422c74f9e0c30f5d62a63f67d6	0.0
99980	879d037b78bd5f54a062afcc22f170ad	cf5b15bca7963d8779b50572db17873e	0.0
99981	9946867fb7e729d3d7b5693ebe4274cd	afc99fac3c2dcb17fdf3753a873b53cf	8.0
99982	d11cd9eca4d042914c1dd7f682262e6a	e432f569d00115a071124db15c61a6c2	0.0
99983	e950cb6513973337917ac0bfe6546171	2080b964b7b425b4ffba4b5d4c81e121	0.0
99984	a2f4bf65ba121a22923ed6269167614d	102e1c8562303edcbf0a65ec4267cd77	0.0
99985	a91d955eb55921171386353d97f26e2c	26e48696568c7c9171691420d4274fa6	0.0
99986	aaa34d33b2cdbc230356bb944d797355	c6b10b0441814ed852eaf856b3097bf8	0.0
99987	e88330514585dc40b7cb8f48c0e0ea2a	1cdcf7dff5c60b5bf340090b2a1dd4b8	156

	u_id	m_id	forw
99988	ff6df56a5c138710f41896102ff3335c	6ef4eaeda467202214babd3a8e1b7959	0.0
99989	a74f8ea4ec2cd491e00e7112574e28d2	ef45216fb95a553755a56e96619f3f84	0.0
99990	5fdbebd81a32b63f5a9bd20e40302cc7	2c7140afac51d51ffd3ac0770cf215fe	0.0
99991	392b9ad1019a6143a55d46d6d694b39c	376a571c1977908241c861510e3da05f	0.0
99992	1f6fa8bd67f384066c3815f384641909	6b1a8189242187adaafd6d7c720c69d9	0.0
99993	0faddeeabf8b2cfd75afc6ad9c1ba2da	4ce1362d01c3c68b3f6b37ecee3e33cb	0.0
99994	2c29c907d3a5111e58f60b7997877a0e	29f21c87a34727423d41c44ab36970fc	0.0
99995	be4bb6460816182375c75128776e03a3	67ff839d6546e98db53fd730ae248209	2.0
99996	fd3acb53b6e5992b8e96d08a8e27f00d	e85246e3cb09c12e955e6586de4372f7	0.0
99997	c58f60297a4a46ea9e80c171f0c6a804	e46bb297ab952cb6ad354ce31c41922b	0.0
99998	6acc1900479dcceea56375d97916a40e	7d4b9cfe0362db61395790e2538d696e	0.0

	u_id	m_id	forw
99999	c8026e7713b9cffd6c21935ac407dfcc	b877e60aef68601c52ff55445281192c	2.0

100000 rows × 28 columns

In [5]:

```
df.shape[0]
```

Out[5]:

```
100000
```

In [13]:

```
df_new = pd.DataFrame(columns=['pol'])
for i in range(0,100000):
    try:
        a=TextBlob(df['no_punc'].iloc[i]).sentiment
        df_new=df_new.append({'pol':a[0]}, ignore_index=True)
    except Exception as e:
        print(str(e))
        df_new=df_new.append({'pol':999999}, ignore_index=True)
```

In [14]:

```
df['polarity']=df_new['pol']
```

In [15]:

```
df_new
```

Out[15]:

	pol
0	-0.166667
1	0.000000
2	-0.400000
3	0.300000
4	0.000000
5	0.000000
6	0.133333
7	0.000000
8	0.214286

	pol
9	0.850000
10	0.125000
11	0.260000
12	0.250000
13	-0.105556
14	0.000000
15	0.136364
16	-0.250000
17	0.140000
18	0.000000
19	0.000000
20	0.000000
21	-0.053125
22	0.000000
23	0.060000
24	0.000000
25	0.000000
26	0.000000
27	-0.100000
28	0.533333
29	0.100000
...	...
99970	0.500000
99971	0.000000
99972	0.000000
99973	0.227778
99974	0.300000
99975	0.000000
99976	0.500000
99977	0.214394
99978	0.550000
99979	0.000000
99980	0.100000
99981	0.200000

	pol
99982	0.285714
99983	0.000000
99984	-0.015909
99985	0.800000
99986	0.500000
99987	-0.150000
99988	-0.600000
99989	0.700000
99990	0.000000
99991	0.341667
99992	0.250000
99993	-0.800000
99994	0.187500
99995	0.285714
99996	0.000000
99997	0.000000
99998	0.407143
99999	0.301786

100000 rows × 1 columns

In [16]:

```
df.to_csv('E://DMA_PRED//polarityL1.csv')
```

In [3]:

```
import pandas as pd
import numpy as np
import re
from sklearn import linear_model
from sklearn.model_selection import train_test_split
from sklearn.metrics import mean_squared_error
from matplotlib import pyplot as plt
from textblob import TextBlob
```

In [4]:

```
import import_ipynb
from evaluation import precision
from runTime import runTime
```

```
importing Jupyter notebook from evaluation.ipynb
importing Jupyter notebook from runTime.ipynb
```

----------Polarity as a factor----------

In [5]:

```
dfpol=pd.read_csv("E:\DMA_PRE\polarity\weibo_polarity.csv")
```

In [6]:

```
dfpol.head(10)
```

Out[6]:

	Unnamed: 0	u_id	m_ic
0	0	d38e9bed5d98110dc2489d0d1cac3c2a	7d45833d9865727a88b960b0603c19f6
1	1	d38e9bed5d98110dc2489d0d1cac3c2a	00755196c77936bf44656ada98291c59

	Unnamed: 0	u_id	m_id
2	2	d38e9bed5d98110dc2489d0d1cac3c2a	4fedf3888b1e16592f0e0bdc8b393845
3	3	d38e9bed5d98110dc2489d0d1cac3c2a	91be0b8612265aae32725cd4fa80b222
4	4	d38e9bed5d98110dc2489d0d1cac3c2a	bd2af99ecf1298f5539f0ddfcdd3ed64
5	5	d38e9bed5d98110dc2489d0d1cac3c2a	182078c5a409834f2128b3c9c2c289c3
6	6	d38e9bed5d98110dc2489d0d1cac3c2a	2c9697e5d6f1d9d479540173c4c374cb
7	7	d38e9bed5d98110dc2489d0d1cac3c2a	0ce5d103d7712b398ee2e81f83f49751
8	8	d38e9bed5d98110dc2489d0d1cac3c2a	a651facd0523d2a85a0717b83928c6c8
9	9	d38e9bed5d98110dc2489d0d1cac3c2a	3e1895f6017e0214f7392013552ac96a

10 rows × 39 columns

In [7]:

```
dfpol.columns
```

Out[7]:

```
Index(['Unnamed: 0', 'u_id', 'm_id', 'forward_count', 'comment_count',
       'like_count', 'content', 'date', 'time', 'content_media_count',
       'content_#_count', 'content_@_count', 'content_?_count',
       'content_!_count', 'content_length', 'content_emoji_count', 'hour',
       'min', 'sec', 'forward_min', 'forward_max', 'forward_median',
       'forward_mean', 'comment_min', 'comment_max', 'comment_median',
       'comment_mean', 'like_min', 'like_max', 'like_median', 'like_mean',
       'Unnamed: 0.1', 'content_spchar', 'non_emoji_content', 'en_content',
       'Unnamed: 1', 'url_rem', 'contentwurl', 'polarity'],
      dtype='object')
```

In [8]:

```
dfpol['date']=pd.to_datetime(dfpol['date'],errors='coerce')
train_month=[g for n, g in dfpol.groupby(pd.Grouper(key='date',freq='M'))]
```

In [9]:

```
train_month[0]=pd.read_csv("E:\DMA_PRE\polarity\weibo_train_feb_cpts10000.csv")
train_month[1]=pd.read_csv("E:\DMA_PRE\polarity\weibo_train_march_cpts10000.csv")
train_month[2]=pd.read_csv("E:\DMA_PRE\polarity\weibo_train_april_cpts10000.csv")
train_month[3]=pd.read_csv("E:\DMA_PRE\polarity\weibo_train_may_cpts10000.csv")
train_month[4]=pd.read_csv("E:\DMA_PRE\polarity\weibo_train_june_cpts10000.csv")
train_month[5]=pd.read_csv("E:\DMA_PRE\polarity\weibo_train_july_cpts10000.csv")
```

In [10]:

```
frames1=[train_month[0],train_month[1],train_month[2],train_month[3],train_month[4]]
train=pd.concat(frames1)
predict=train_month[5]
```

Model 7: (Factors: Media, Length, Emoji, Median,Polarity)

In [11]:

```
X_train1=train[["content_media_count","content_length","forward_median","comment_medi
an","like_median","polarity"]]
Y_train1=train[["forward_count","comment_count","like_count"]]
X_test1=predict[["content_media_count","content_length","forward_median","comment_med
ian","like_median","polarity"]]
Y_test1=predict[["forward_count","comment_count","like_count"]]

pd.options.mode.use_inf_as_na = True
X_train1.fillna(X_train1.max(),inplace=True)
X_test1.fillna(X_test1.max(),inplace=True)
```

```
C:\Users\DELL\Anaconda3\lib\site-packages\pandas\core\generic.py:5430: SettingWithCopyWarning:
A value is trying to be set on a copy of a slice from a DataFrame

See the caveats in the documentation: http://pandas.pydata.org/pandas-docs/stable/indexing.html#indexing-view-versus-copy
  self._update_inplace(new_data)
```

In [12]:

```
lm1=linear_model.LinearRegression()
model1=lm1.fit(X_train1,Y_train1)
pred1=lm1.predict(X_test1)
pred1=pred1.round()
pred1=(np.maximum(pred1,0.))
```

In [13]:

```
print(model1.coef_)
print(model1.intercept_)
```

```
[[ -3.60345081e-01   6.74510609e-03  -9.03871388e+00  -5.69877934e+00
    1.46618263e+01  -3.00702607e-01]
 [ -4.52559256e-01   2.88854313e-04  -2.58935392e+00   3.20973326e-01
    2.56664104e+00   1.78063868e-01]
 [ -1.70607807e-01  -1.73336140e-03  -2.49319057e+00  -7.66383744e-01
    3.84877692e+00   1.34944972e-01]]
[ 0.09961692  0.27754213  0.23556883]
```

In [14]:

```
np.savetxt("E:\DMA_PRE\polarity\weibo_predict_result51.csv",pred1,delimiter=',',header="forward_count,comment_count,like_count",comments="")
result1=pd.read_csv("E:\DMA_PRE\polarity\weibo_predict_result51.csv")
```

In [15]:

```
print(mean_squared_error(Y_test1,result1))
```

```
21.2745912995
```

In [16]:

```
train_real_pred=Y_test1
train_real_pred['fp']=result1['forward_count']
train_real_pred['cp']=result1['comment_count']
train_real_pred['lp']=result1['like_count']
print("Score:{0:.2f}%".format(precision(train_real_pred.values)*100))
```

```
Score:35.39%
```

```
C:\Users\DELL\Anaconda3\lib\site-packages\ipykernel_launcher.py:2: SettingWithCopyWarning:
```

```
A value is trying to be set on a copy of a slice from a DataFrame.
Try using .loc[row_indexer,col_indexer] = value instead

See the caveats in the documentation: http://pandas.pydata.org/pandas-docs/stable/ind
exing.html#indexing-view-versus-copy

C:\Users\DELL\Anaconda3\lib\site-packages\ipykernel_launcher.py:3: SettingWithCopyWar
ning:
A value is trying to be set on a copy of a slice from a DataFrame.
Try using .loc[row_indexer,col_indexer] = value instead

See the caveats in the documentation: http://pandas.pydata.org/pandas-docs/stable/ind
exing.html#indexing-view-versus-copy
  This is separate from the ipykernel package so we can avoid doing imports until
C:\Users\DELL\Anaconda3\lib\site-packages\ipykernel_launcher.py:4: SettingWithCopyWar
ning:
A value is trying to be set on a copy of a slice from a DataFrame.
Try using .loc[row_indexer,col_indexer] = value instead

See the caveats in the documentation: http://pandas.pydata.org/pandas-docs/stable/ind
exing.html#indexing-view-versus-copy
  after removing the cwd from sys.path.
```

Analysis: Result with Polarity as factor are satisfactory considering the data used for train. This might prove to be a good factor for whole dataset prediction.

In [1]:

```
import pandas as pd
import numpy as np
from sklearn import linear_model
from sklearn.model_selection import train_test_split
from sklearn.metrics import accuracy_score
from matplotlib import pyplot as plt
import statsmodels.api as sm
```

In [2]:

```
import import_ipynb
from evaluation import precision
from runTime import runTime
```

```
importing Jupyter notebook from evaluation.ipynb
importing Jupyter notebook from runTime.ipynb
```

In [3]:

```
from evaluation2 import precision2
```

```
importing Jupyter notebook from evaluation2.ipynb
```

In [4]:

```
df1=pd.read_csv("E:\\5th Sem\\DMA Project\\Model Evaluation\\weibo_train1_cp.csv")
df2=pd.read_csv("E:\\5th Sem\\DMA Project\\Model Evaluation\\weibo_train2_cp.csv")
frames=[df1,df2]
train_dataset=pd.concat(frames)
predict_dataset=pd.read_csv("E:\\5th Sem\\DMA Project\\Model
Evaluation\\weibo_predict_cp.csv")
```

In [5]:

```
train_dataset['date']=pd.to_datetime(train_dataset['date'],errors='coerce')
```

In [6]:

```
train_month=[g for n, g in train_dataset.groupby(pd.Grouper(key='date',freq='M'))]
```

In [7]:

```
train_dataset['time']=pd.to_datetime(train_dataset['time'],errors='coerce')
```

In [8]:

```
train_hour=[g for n, g in train_dataset.groupby(pd.Grouper(key='time',freq='H'))]
```

In [9]:

```
train_month[0].to_csv("E:\\5th Sem\\DMA Project\\Model
Evaluation\\weibo_train_feb_cp.csv",sep=',',index=False,encoding='utf-8')
```

```
train_month[1].to_csv("E:\\5th Sem\\DMA Project\\Model
Evaluation\\weibo_train_march_cp.csv",sep=',',index=False,encoding='utf-8')
train_month[2].to_csv("E:\\5th Sem\\DMA Project\\Model
Evaluation\\weibo_train_april_cp.csv",sep=',',index=False,encoding='utf-8')
train_month[3].to_csv("E:\\5th Sem\\DMA Project\\Model
Evaluation\\weibo_train_may_cp.csv",sep=',',index=False,encoding='utf-8')
train_month[4].to_csv("E:\\5th Sem\\DMA Project\\Model
Evaluation\\weibo_train_june_cp.csv",sep=',',index=False,encoding='utf-8')
train_month[5].to_csv("E:\\5th Sem\\DMA Project\\Model
Evaluation\\weibo_train_july_cp.csv",sep=',',index=False,encoding='utf-8')
```

In [13]:

```
i=0
for i in range(0,24):
    path="E:\\5th Sem\\DMA Project\\Model
Evaluation\\weibo_train_hour_"+str(i)+"_cp.csv"
    train_hour[i].to_csv(path,sep=',',index=False,encoding='utf-8')
```

In [14]:

```
frames1=[train_month[0],train_month[1],train_month[2],train_month[3],train_month[4]]
train=pd.concat(frames1)
predict=train_month[5]
```

Model 1 (Factors: Media, #, @, ?, !, Length, Emoji)

In [15]:

```
X_train=train[["content_media_count","content_#_count","content_@_count","content_?_c
ount","content_!_count","content_length","content_emoji_count"]]
Y_train=train[["forward_count","comment_count","like_count"]]
X_test=predict[["content_media_count","content_#_count","content_@_count","content_?_
count","content_!_count","content_length","content_emoji_count"]]
Y_test=predict[["forward_count","comment_count","like_count"]]
```

In [16]:

```
print(X_train.shape,Y_train.shape)
print(X_test.shape,Y_test.shape)
```

```
(1044681, 7) (1044681, 3)
(184937, 7) (184937, 3)
```

In [17]:

```
pd.options.mode.use_inf_as_na = True
X_train.fillna(X_train.max(),inplace=True)
X_test.fillna(X_test.max(),inplace=True)
```

```
C:\Users\DELL\Anaconda3\lib\site-packages\pandas\core\generic.py:5430: SettingWithCopyWarning:
A value is trying to be set on a copy of a slice from a DataFrame

See the caveats in the documentation: http://pandas.pydata.org/pandas-docs/stable/indexing.html#indexing-view-versus-copy
  self._update_inplace(new_data)
```

In [18]:

```
lm=linear_model.LinearRegression()
model=lm.fit(X_train,Y_train)
pred=lm.predict(X_test)
pred=pred.round()
pred=(np.maximum(pred,0.))
```

In [19]:

```
print(pred[0:5])
```

```
[[ 4.  1.  2.]
 [ 3.  2.  3.]
 [ 3.  1.  1.]
 [ 7.  1.  2.]
 [ 9.  2.  3.]]
```

In [20]:

```
print(model.coef_)
```

```
[[-2.13937044 -0.36595303 -0.29844433 -0.04270781 -0.20884535  0.05323583
  -0.11373958]
 [-1.46356558 -0.15223682 -0.02946097 -0.00326344 -0.04880305  0.00769467
   0.1747673 ]
 [-2.6330917  -0.13749083 -0.32811723 -0.01394959 -0.071054    0.01544652
   0.26583545]]
```

In [21]:

```
print(model.intercept_)
```

```
[ 1.31437859  1.76293348  2.98461306]
```

In [22]:

```
np.savetxt("E:\\5th Sem\\DMA Project\\Model Evaluation\\weibo_predict_result2.csv",pred,delimiter=',',header="forward_count,comment_count,like_count",comments="")
result=pd.read_csv("E:\\5th Sem\\DMA Project\\Model Evaluation\\weibo_predict_result2.csv")
```

In [23]:

```
train_real_pred = Y_test
forward=result['forward_count'].values
```

```
comment=result['forward_count'].values
like=result['forward_count'].values
train_real_pred['fp'],train_real_pred['cp'],train_real_pred['lp'] = forward,comment,like
print ("Score on the training set:{0:.2f}%".format(precision(train_real_pred.values)*100))
```

```
C:\Users\DELL\Anaconda3\lib\site-packages\ipykernel_launcher.py:5: SettingWithCopyWarning:
A value is trying to be set on a copy of a slice from a DataFrame.
Try using .loc[row_indexer,col_indexer] = value instead

See the caveats in the documentation: http://pandas.pydata.org/pandas-docs/stable/indexing.html#indexing-view-versus-copy
  """
```

```
Score on the training set:3.69%
```

Model 2 (Media, Length, Emoji)

In [27]:

```
X_train=train[["content_media_count","content_length","content_emoji_count"]]
Y_train=train[["forward_count","comment_count","like_count"]]
X_test=predict[["content_media_count","content_length","content_emoji_count"]]
Y_test=predict[["forward_count","comment_count","like_count"]]
```

In [28]:

```
print(X_train.shape,Y_train.shape)
print(X_test.shape,Y_test.shape)
```

```
(1044681, 3) (1044681, 3)
(184937, 3) (184937, 3)
```

In [29]:

```
pd.options.mode.use_inf_as_na = True
X_train.fillna(X_train.max(),inplace=True)
X_test.fillna(X_test.max(),inplace=True)
```

```
C:\Users\DELL\Anaconda3\lib\site-packages\pandas\core\generic.py:5430: SettingWithCopyWarning:
A value is trying to be set on a copy of a slice from a DataFrame

See the caveats in the documentation: http://pandas.pydata.org/pandas-docs/stable/indexing.html#indexing-view-versus-copy
  self._update_inplace(new_data)
```

In [30]:

```
lm=linear_model.LinearRegression()
```

```
lm=linear_model.LinearRegression()
model=lm.fit(X_train,Y_train)
pred=lm.predict(X_test)
pred=pred.round()
pred=(np.maximum(pred,0.))
```

In [31]:

```
print(pred[0:5])
```

```
[[ 4.  1.  2.]
 [ 3.  2.  3.]
 [ 3.  1.  1.]
 [ 7.  1.  2.]
 [ 8.  1.  3.]]
```

In [32]:

```
print(model.coef_)
```

```
[[-2.19780299  0.05049652 -0.14777505]
 [-1.47377848  0.00679751  0.16018448]
 [-2.68756793  0.01390734  0.25398113]]
```

In [33]:

```
print(model.intercept_)
```

```
[ 1.23425639  1.73473692  2.94322868]
```

In [35]:

```
np.savetxt("E:\\5th Sem\\DMA Project\\Model Evaluation\\weibo_predict_result3.csv",p
red,delimiter=',',header="forward_count,comment_count,like_count",comments="")
result=pd.read_csv("E:\\5th Sem\\DMA Project\\Model
Evaluation\\weibo_predict_result3.csv")
```

In [36]:

```
train_real_pred = Y_test
forward=result['forward_count'].values
comment=result['forward_count'].values
like=result['forward_count'].values
train_real_pred['fp'],train_real_pred['cp'],train_real_pred['lp'] = forward,comment,l
ike
print ("Score on the training set:{0:.2f}%".format(precision(train_real_pred.values)
*100))
```

```
C:\Users\DELL\Anaconda3\lib\site-packages\ipykernel_launcher.py:5: SettingWithCopyWar
ning:
A value is trying to be set on a copy of a slice from a DataFrame.
Try using .loc[row_indexer,col_indexer] = value instead

See the caveats in the documentation: http://pandas.pydata.org/pandas-docs/stable/ind
```

```
exing.html#indexing-view-versus-copy
  """
```

```
Score on the training set:3.79%
```

Model 3(Media)

In [37]:

```
X_train=train[["content_media_count"]]
Y_train=train[["forward_count","comment_count","like_count"]]
X_test=predict[["content_media_count"]]
Y_test=predict[["forward_count","comment_count","like_count"]]
```

In [38]:

```
print(X_train.shape,Y_train.shape)
print(X_test.shape,Y_test.shape)
```

```
(1044681, 1) (1044681, 3)
(184937, 1) (184937, 3)
```

In [39]:

```
pd.options.mode.use_inf_as_na = True
X_train.fillna(X_train.max(),inplace=True)
X_test.fillna(X_test.max(),inplace=True)
```

```
C:\Users\DELL\Anaconda3\lib\site-packages\pandas\core\generic.py:5430: SettingWithCop
yWarning:
A value is trying to be set on a copy of a slice from a DataFrame

See the caveats in the documentation: http://pandas.pydata.org/pandas-docs/stable/ind
exing.html#indexing-view-versus-copy
  self._update_inplace(new_data)
```

In [40]:

```
lm=linear_model.LinearRegression()
model=lm.fit(X_train,Y_train)
pred=lm.predict(X_test)
pred=pred.round()
pred=(np.maximum(pred,0.))
```

In [41]:

```
print(pred[0:5])
```

```
[[ 3.  1.  1.]
 [ 4.  2.  4.]
 [ 3.  1.  1.]
 [ 3.  1.  1.]
```

```
 [ 3.  1.  1.]]
```

In [42]:

```
print(model.coef_)
```

```
[[-0.62388914]
 [-1.26343548]
 [-2.25659216]]
```

In [43]:

```
print(model.intercept_)
```

```
[ 3.86255173  2.09158813  3.67207903]
```

In [45]:

```
np.savetxt("E:\\5th Sem\\DMA Project\\Model Evaluation\\weibo_predict_result4.csv",p
red,delimiter=',',header="forward_count,comment_count,like_count",comments="")
result=pd.read_csv("E:\\5th Sem\\DMA Project\\Model
Evaluation\\weibo_predict_result4.csv")
```

In [33]:

```
train_real_pred = Y_test
train_real_pred['fp']=result['forward_count'].values
train_real_pred['cp']=result['comment_count'].values
train_real_pred['lp']=result['like_count'].values
train_real_pred=train_real_pred.round()
print ("Score on the training set:{0:.2f}%".format(precision(train_real_pred.values)
*100))
```

```
G:\Anaconda\lib\site-packages\ipykernel_launcher.py:2: SettingWithCopyWarning:
A value is trying to be set on a copy of a slice from a DataFrame.
Try using .loc[row_indexer,col_indexer] = value instead

See the caveats in the documentation: http://pandas.pydata.org/pandas-docs/stable/ind
exing.html#indexing-view-versus-copy

G:\Anaconda\lib\site-packages\ipykernel_launcher.py:3: SettingWithCopyWarning:
A value is trying to be set on a copy of a slice from a DataFrame.
Try using .loc[row_indexer,col_indexer] = value instead

See the caveats in the documentation: http://pandas.pydata.org/pandas-docs/stable/ind
exing.html#indexing-view-versus-copy
  This is separate from the ipykernel package so we can avoid doing imports until
G:\Anaconda\lib\site-packages\ipykernel_launcher.py:4: SettingWithCopyWarning:
A value is trying to be set on a copy of a slice from a DataFrame.
Try using .loc[row_indexer,col_indexer] = value instead

See the caveats in the documentation: http://pandas.pydata.org/pandas-docs/stable/ind
exing.html#indexing-view-versus-copy
```

```
  after removing the cwd from sys.path.
```

```
Score on the training set:3.59%
```

Model 4 (Time) Pre-requisite

In [34]:

```
train_dataset['hour']=pd.DatetimeIndex(train_dataset['time']).hour
print("done")
train_dataset['min']=pd.DatetimeIndex(train_dataset['time']).minute
print("done")
train_dataset['sec']=pd.DatetimeIndex(train_dataset['time']).second
print("done")
```

```
done
done
done
```

In [95]:

```
predict_dataset['hour']=pd.DatetimeIndex(predict_dataset['time']).hour
print("done")
predict_dataset['min']=pd.DatetimeIndex(predict_dataset['time']).minute
print("done")
predict_dataset['sec']=pd.DatetimeIndex(predict_dataset['time']).second
print("done")
```

```
done
done
done
```

In [98]:

```
train_dataset.head(614809).to_csv("G://DMA_PROJECT//weibo_train1_cpt.csv",sep=',',ind
ex=False,encoding='utf-8')
train_dataset.tail(614809).to_csv("G://DMA_PROJECT//weibo_train2_cpt.csv",sep=',',ind
ex=False,encoding='utf-8')
predict_dataset.to_csv("G://DMA_PROJECT//weibo_predict_cpt.csv",sep=',',index=False,
encoding='utf-8')
```

In [4]:

```
df1=pd.read_csv("G://DMA_PROJECT//weibo_train1_cpt.csv")
df2=pd.read_csv("G://DMA_PROJECT//weibo_train2_cpt.csv")
frames=[df1,df2]
train_dataset=pd.concat(frames)
predict_dataset=pd.read_csv("G://DMA_PROJECT//weibo_predict_cpt.csv")
```

In [100]:

```
train_dataset['date']=pd.to_datetime(train_dataset['date'],errors='coerce')
```

In [101]:

```
train_month=[g for n, g in train_dataset.groupby(pd.Grouper(key='date',freq='M'))]
```

In [102]:

```
train_month[0].to_csv("G://DMA_PROJECT//weibo_train_feb_cpt.csv",sep=',',index=False
,encoding='utf-8')
train_month[1].to_csv("G://DMA_PROJECT//weibo_train_march_cpt.csv",sep=',',index=Fals
e,encoding='utf-8')
train_month[2].to_csv("G://DMA_PROJECT//weibo_train_april_cpt.csv",sep=',',index=Fals
e,encoding='utf-8')
train_month[3].to_csv("G://DMA_PROJECT//weibo_train_may_cpt.csv",sep=',',index=False
,encoding='utf-8')
train_month[4].to_csv("G://DMA_PROJECT//weibo_train_june_cpt.csv",sep=',',index=False
,encoding='utf-8')
train_month[5].to_csv("G://DMA_PROJECT//weibo_train_july_cpt.csv",sep=',',index=False
,encoding='utf-8')
```

In [35]:

```
train_month[0]=pd.read_csv("G://DMA_PROJECT//weibo_train_feb_cpt.csv")
train_month[1]=pd.read_csv("G://DMA_PROJECT//weibo_train_march_cpt.csv")
train_month[2]=pd.read_csv("G://DMA_PROJECT//weibo_train_april_cpt.csv")
train_month[3]=pd.read_csv("G://DMA_PROJECT//weibo_train_may_cpt.csv")
train_month[4]=pd.read_csv("G://DMA_PROJECT//weibo_train_june_cpt.csv")
train_month[5]=pd.read_csv("G://DMA_PROJECT//weibo_train_july_cpt.csv")
```

In [36]:

```
frames1=[train_month[0],train_month[1],train_month[2],train_month[3],train_month[4]]
train=pd.concat(frames1)
predict=train_month[5]
```

Model 4 (Time: Hour, Min, Sec)

In [37]:

```
X_train=train[["hour","min","sec"]]
Y_train=train[["forward_count","comment_count","like_count"]]
X_test=predict[["hour","min","sec"]]
Y_test=predict[["forward_count","comment_count","like_count"]]
```

In [38]:

```
print(X_train.shape,Y_train.shape)
print(X_test.shape,Y_test.shape)
```

```
(1044681, 3) (1044681, 3)
(184937, 3) (184937, 3)
```

In [39]:

```
pd.options.mode.use_inf_as_na = True
X_train.fillna(X_train.max(),inplace=True)
X_test.fillna(X_test.max(),inplace=True)
```

```
G:\Anaconda\lib\site-packages\pandas\core\generic.py:5430: SettingWithCopyWarning:
A value is trying to be set on a copy of a slice from a DataFrame

See the caveats in the documentation: http://pandas.pydata.org/pandas-docs/stable/ind
exing.html#indexing-view-versus-copy
  self._update_inplace(new_data)
```

In [40]:

```
lm=linear_model.LinearRegression()
model=lm.fit(X_train,Y_train)
pred=lm.predict(X_test)
pred=pred.round()
pred=(np.maximum(pred,0.))
```

In [41]:

```
print(pred[0:5])
```

```
[[ 3.  1.  2.]
 [ 2.  1.  2.]
 [ 3.  1.  2.]
 [ 1.  1.  1.]
 [ 6.  1.  3.]]
```

In [42]:

```
print(model.coef_)
```

```
[[ 0.08066778 -0.06019321 -0.0382752 ]
 [ 0.02746039 -0.00495496 -0.00478928]
 [ 0.07295777 -0.03270772 -0.01755154]]
```

In [43]:

```
print(model.intercept_)
```

```
[ 5.15832365  1.15751239  2.61598361]
```

In [44]:

```
np.savetxt("G://DMA_PROJECT//weibo_predict_result5.csv",pred,delimiter=',',header="fo
rward_count,comment_count,like_count",comments="")
result=pd.read_csv("G://DMA_PROJECT//weibo_predict_result5.csv")
```

In [45]:

```
train_real_pred = Y_test
```

```
train_real_pred['fp']=result['forward_count'].values
train_real_pred['cp']=result['comment_count'].values
train_real_pred['lp']=result['like_count'].values
train_real_pred=train_real_pred.round()
print ("Score on the training set:{0:.2f}%".format(precision(train_real_pred.values)
*100))
```

```
G:\Anaconda\lib\site-packages\ipykernel_launcher.py:2: SettingWithCopyWarning:
A value is trying to be set on a copy of a slice from a DataFrame.
Try using .loc[row_indexer,col_indexer] = value instead

See the caveats in the documentation: http://pandas.pydata.org/pandas-docs/stable/ind
exing.html#indexing-view-versus-copy

G:\Anaconda\lib\site-packages\ipykernel_launcher.py:3: SettingWithCopyWarning:
A value is trying to be set on a copy of a slice from a DataFrame.
Try using .loc[row_indexer,col_indexer] = value instead

See the caveats in the documentation: http://pandas.pydata.org/pandas-docs/stable/ind
exing.html#indexing-view-versus-copy
  This is separate from the ipykernel package so we can avoid doing imports until
G:\Anaconda\lib\site-packages\ipykernel_launcher.py:4: SettingWithCopyWarning:
A value is trying to be set on a copy of a slice from a DataFrame.
Try using .loc[row_indexer,col_indexer] = value instead

See the caveats in the documentation: http://pandas.pydata.org/pandas-docs/stable/ind
exing.html#indexing-view-versus-copy
  after removing the cwd from sys.path.
```

```
Score on the training set:4.19%
```

Model 5 (Time: Hour)

In [46]:

```
X_train=train[["hour"]]
Y_train=train[["forward_count","comment_count","like_count"]]
X_test=predict[["hour"]]
Y_test=predict[["forward_count","comment_count","like_count"]]
```

In [47]:

```
print(X_train.shape,Y_train.shape)
print(X_test.shape,Y_test.shape)
```

```
(1044681, 1) (1044681, 3)
(184937, 1) (184937, 3)
```

In [48]:

```
pd.options.mode.use_inf_as_na = True
```

```
X_train.fillna(X_train.max(),inplace=True)
X_test.fillna(X_test.max(),inplace=True)
```

```
G:\Anaconda\lib\site-packages\pandas\core\generic.py:5430: SettingWithCopyWarning:
A value is trying to be set on a copy of a slice from a DataFrame

See the caveats in the documentation: http://pandas.pydata.org/pandas-docs/stable/ind
exing.html#indexing-view-versus-copy
  self._update_inplace(new_data)
```

In [49]:

```
lm=linear_model.LinearRegression()
model=lm.fit(X_train,Y_train)
pred=lm.predict(X_test)
pred=pred.round()
pred=(np.maximum(pred,0.))
```

In [50]:

```
print(pred[0:5])
```

```
[[ 3.  1.  1.]
 [ 4.  1.  2.]
 [ 3.  1.  2.]
 [ 3.  1.  2.]
 [ 3.  1.  2.]]
```

In [51]:

```
print(model.coef_)
```

```
[[ 0.07590631]
 [ 0.02704577]
 [ 0.0704154 ]]
```

In [52]:

```
print(model.intercept_)
```

```
[ 2.40776056  0.88480607  1.21309106]
```

In [53]:

```
np.savetxt("G://DMA_PROJECT//weibo_predict_result6.csv",pred,delimiter=',',header="fo
rward_count,comment_count,like_count",comments="")
result=pd.read_csv("G://DMA_PROJECT//weibo_predict_result6.csv")
```

In [54]:

```
train_real_pred = Y_test
train_real_pred['fp']=result['forward_count'].values
train_real_pred['cp']=result['comment_count'].values
train_real_pred['lp']=result['like_count'].values
```

```
train_real_pred=train_real_pred.round()
print ("Score on the training set:{0:.2f}%".format(precision(train_real_pred.values)
*100))
```

```
G:\Anaconda\lib\site-packages\ipykernel_launcher.py:2: SettingWithCopyWarning:
A value is trying to be set on a copy of a slice from a DataFrame.
Try using .loc[row_indexer,col_indexer] = value instead

See the caveats in the documentation: http://pandas.pydata.org/pandas-docs/stable/ind
exing.html#indexing-view-versus-copy

G:\Anaconda\lib\site-packages\ipykernel_launcher.py:3: SettingWithCopyWarning:
A value is trying to be set on a copy of a slice from a DataFrame.
Try using .loc[row_indexer,col_indexer] = value instead

See the caveats in the documentation: http://pandas.pydata.org/pandas-docs/stable/ind
exing.html#indexing-view-versus-copy
  This is separate from the ipykernel package so we can avoid doing imports until
G:\Anaconda\lib\site-packages\ipykernel_launcher.py:4: SettingWithCopyWarning:
A value is trying to be set on a copy of a slice from a DataFrame.
Try using .loc[row_indexer,col_indexer] = value instead

See the caveats in the documentation: http://pandas.pydata.org/pandas-docs/stable/ind
exing.html#indexing-view-versus-copy
  after removing the cwd from sys.path.
```

```
Score on the training set:3.66%
```

Model 6 Time: (Hour, Min, Sec), Media, Length, Emoji

In [55]:

```
X_train=train[["content_media_count","content_length","content_emoji_count","hour","m
in","sec"]]
Y_train=train[["forward_count","comment_count","like_count"]]
X_test=predict[["content_media_count","content_length","content_emoji_count","hour","
min","sec"]]
Y_test=predict[["forward_count","comment_count","like_count"]]
```

In [56]:

```
print(X_train.shape,Y_train.shape)
print(X_test.shape,Y_test.shape)
```

```
(1044681, 6) (1044681, 3)
(184937, 6) (184937, 3)
```

In [57]:

```
pd.options.mode.use_inf_as_na = True
```

```
X_train.fillna(X_train.max(),inplace=True)
X_test.fillna(X_test.max(),inplace=True)
```

```
G:\Anaconda\lib\site-packages\pandas\core\generic.py:5430: SettingWithCopyWarning:
A value is trying to be set on a copy of a slice from a DataFrame

See the caveats in the documentation: http://pandas.pydata.org/pandas-docs/stable/ind
exing.html#indexing-view-versus-copy
  self._update_inplace(new_data)
```

In [58]:

```
lm=linear_model.LinearRegression()
model=lm.fit(X_train,Y_train)
pred=lm.predict(X_test)
pred=pred.round()
pred=(np.maximum(pred,0.))
```

In [59]:

```
print(pred[0:5])
```

```
[[  4.   1.   1.]
 [  1.   2.   3.]
 [  2.   1.   1.]
 [  4.   1.   1.]
 [ 10.   2.   4.]]
```

In [60]:

```
print(model.coef_)
```

```
[[-2.16126691  0.04995623 -0.14079719  0.09097968 -0.05802816 -0.03386975]
 [-1.46714342  0.0068272   0.15966872  0.02589544 -0.00483482 -0.00418282]
 [-2.6652836   0.0137497   0.25624217  0.07054347 -0.03242168 -0.01633076]]
```

In [61]:

```
print(model.intercept_)
```

```
[ 2.62859063  1.63033587  3.36618167]
```

In [62]:

```
np.savetxt("G://DMA_PROJECT//weibo_predict_result7.csv",pred,delimiter=',',header="fo
rward_count,comment_count,like_count",comments="")
result=pd.read_csv("G://DMA_PROJECT//weibo_predict_result7.csv")
```

In [63]:

```
train_real_pred = Y_test
train_real_pred['fp']=result['forward_count'].values
train_real_pred['cp']=result['comment_count'].values
train_real_pred['lp']=result['like_count'].values
```

```
train_real_pred=train_real_pred.round()
print ("Score on the training set:{0:.2f}%".format(precision(train_real_pred.values)
*100))
```

```
G:\Anaconda\lib\site-packages\ipykernel_launcher.py:2: SettingWithCopyWarning:
A value is trying to be set on a copy of a slice from a DataFrame.
Try using .loc[row_indexer,col_indexer] = value instead

See the caveats in the documentation: http://pandas.pydata.org/pandas-docs/stable/ind
exing.html#indexing-view-versus-copy

G:\Anaconda\lib\site-packages\ipykernel_launcher.py:3: SettingWithCopyWarning:
A value is trying to be set on a copy of a slice from a DataFrame.
Try using .loc[row_indexer,col_indexer] = value instead

See the caveats in the documentation: http://pandas.pydata.org/pandas-docs/stable/ind
exing.html#indexing-view-versus-copy
  This is separate from the ipykernel package so we can avoid doing imports until
G:\Anaconda\lib\site-packages\ipykernel_launcher.py:4: SettingWithCopyWarning:
A value is trying to be set on a copy of a slice from a DataFrame.
Try using .loc[row_indexer,col_indexer] = value instead

See the caveats in the documentation: http://pandas.pydata.org/pandas-docs/stable/ind
exing.html#indexing-view-versus-copy
  after removing the cwd from sys.path.
```

```
Score on the training set:6.89%
```

Model 7: Stats Pre-requisite

In [5]:

```
df1=pd.read_csv("G://DMA_PROJECT//weibo_train1_cpt.csv")
df2=pd.read_csv("G://DMA_PROJECT//weibo_train2_cpt.csv")
frames=[df1,df2]
train_dataset=pd.concat(frames)
predict_dataset=pd.read_csv("G://DMA_PROJECT//weibo_predict_cpt.csv")
```

In [6]:

```
stat=pd.read_csv("G://DMA_PROJECT//train_uid_stat.csv")
```

In [7]:

```
trainstat=pd.merge(train_dataset,stat,on=['u_id'])
predictstat=pd.merge(train_dataset,stat,on=['u_id'])
```

In [9]:

```
trainstat.head(5)
```

Out[9]:

Out[9]:

	u_id	m_id	forward_c
0	d38e9bed5d98110dc2489d0d1cac3c2a	7d45833d9865727a88b960b0603c19f6	0
1	d38e9bed5d98110dc2489d0d1cac3c2a	00755196c77936bf44656ada98291c59	0
2	d38e9bed5d98110dc2489d0d1cac3c2a	4fedf3888b1e16592f0e0bdc8b393845	0
3	d38e9bed5d98110dc2489d0d1cac3c2a	91be0b8612265aae32725cd4fa80b222	0
4	d38e9bed5d98110dc2489d0d1cac3c2a	bd2af99ecf1298f5539f0ddfcdd3ed64	0

5 rows × 30 columns

In [10]:

```
trainstat.head(614809).to_csv("G://DMA_PROJECT//weibo_train1_cpts.csv",sep=',',index
=False,encoding='utf-8')
trainstat.tail(614809).to_csv("G://DMA_PROJECT//weibo_train2_cpts.csv",sep=',',index
=False,encoding='utf-8')
predictstat.to_csv("G://DMA_PROJECT//weibo_predict_cpts.csv",sep=',',index=False,enc
oding='utf-8')
```

In [4]:

```
df1=pd.read_csv("G://DMA_PROJECT//weibo_train1_cpts.csv")
df2=pd.read_csv("G://DMA_PROJECT//weibo_train2_cpts.csv")
frames=[df1,df2]
train_dataset=pd.concat(frames)
predict_dataset=pd.read_csv("G://DMA_PROJECT//weibo_predict_cpts.csv")
```

In [5]:

```
train_dataset['date']=pd.to_datetime(train_dataset['date'],errors='coerce')
```

In [6]:

```
train_month=[g for n, g in train_dataset.groupby(pd.Grouper(key='date',freq='M'))]
```

In [14]:

```
train_month[0].to_csv("G://DMA_PROJECT//weibo_train_feb_cpts.csv",sep=',',index=False
,encoding='utf-8')
train_month[1].to_csv("G://DMA_PROJECT//weibo_train_march_cpts.csv",sep=',',index=Fal
se,encoding='utf-8')
train_month[2].to_csv("G://DMA_PROJECT//weibo_train_april_cpts.csv",sep=',',index=Fal
se,encoding='utf-8')
train_month[3].to_csv("G://DMA_PROJECT//weibo_train_may_cpts.csv",sep=',',index=False
,encoding='utf-8')
train_month[4].to_csv("G://DMA_PROJECT//weibo_train_june_cpts.csv",sep=',',index=Fals
e,encoding='utf-8')
train_month[5].to_csv("G://DMA_PROJECT//weibo_train_july_cpts.csv",sep=',',index=Fals
e,encoding='utf-8')
```

In [7]:

```
train_month[0]=pd.read_csv("G://DMA_PROJECT//weibo_train_feb_cpts.csv")
train_month[1]=pd.read_csv("G://DMA_PROJECT//weibo_train_march_cpts.csv")
train_month[2]=pd.read_csv("G://DMA_PROJECT//weibo_train_april_cpts.csv")
train_month[3]=pd.read_csv("G://DMA_PROJECT//weibo_train_may_cpts.csv")
train_month[4]=pd.read_csv("G://DMA_PROJECT//weibo_train_june_cpts.csv")
train_month[5]=pd.read_csv("G://DMA_PROJECT//weibo_train_july_cpts.csv")
```

In [8]:

```
frames1=[train_month[0],train_month[1],train_month[2],train_month[3],train_month[4]]
train=pd.concat(frames1)
predict=train_month[5]
```

Model 7 Median,Time: (Hour, Min, Sec), Media, Length, Emoji

Only for Forward Count

In [9]:

```
X_train=train[["content_media_count","content_length","content_emoji_count","hour","m
in","sec","forward_median"]]
Y_train=train[["forward_count"]]
X_test=predict[["content_media_count","content_length","content_emoji_count","hour","
min","sec","forward_median"]]
Y_test=predict[["forward_count"]]
```

In [10]:

```
print(X_train.shape,Y_train.shape)
print(X_test.shape,Y_test.shape)
```

```
(1044681, 7) (1044681, 1)
(184937, 7) (184937, 1)
```

In [11]:

```
pd.options.mode.use_inf_as_na = True
X_train.fillna(X_train.max(),inplace=True)
X_test.fillna(X_test.max(),inplace=True)
```

```
G:\Anaconda\lib\site-packages\pandas\core\generic.py:5430: SettingWithCopyWarning:
A value is trying to be set on a copy of a slice from a DataFrame

See the caveats in the documentation: http://pandas.pydata.org/pandas-docs/stable/ind
exing.html#indexing-view-versus-copy
  self._update_inplace(new_data)
```

In [12]:

```
lm=linear_model.LinearRegression()
model=lm.fit(X_train,Y_train)
pred=lm.predict(X_test)
pred=pred.round()
pred=(np.maximum(pred,0.))
```

In [13]:

```
print(pred[0:5])
```

```
[[ 1.]
 [ 1.]
 [ 1.]
 [ 8.]
 [ 1.]]
```

In [14]:

```
print(model.coef_)
```

```
[[ -7.05031752e-01   2.07778896e-02   6.62467259e-02   4.04463406e-02
    6.80481359e-04   9.98897013e-05   1.52380773e+00]]
```

In [15]:

```
print(model.intercept_)
```

```
[-0.37795983]
```

In [16]:

```
np.savetxt("G://DMA_PROJECT//weibo_predict_result8.csv",pred,delimiter=',',header="fo
rward_count,comment_count,like_count",comments="")
result=pd.read_csv("G://DMA_PROJECT//weibo_predict_result8.csv")
```

In [17]:

```
train_real_pred=Y_test
train_real_pred['fp']=result['forward_count'].values
train_real_pred=train_real_pred.round()
print ("Score on the training set:{0:.2f}%".format(precision2(train_real_pred.values
)*100))
```

```
G:\Anaconda\lib\site-packages\ipykernel_launcher.py:2: SettingWithCopyWarning:
A value is trying to be set on a copy of a slice from a DataFrame.
Try using .loc[row_indexer,col_indexer] = value instead

See the caveats in the documentation: http://pandas.pydata.org/pandas-docs/stable/ind
exing.html#indexing-view-versus-copy
```

```
Score on the training set:45.81%
```

Model 7 Median,Time: (Hour, Min, Sec), Media, Length, Emoji

In [39]:

```
X_train=train[["content_media_count","content_length","content_emoji_count","hour","m
in","sec","forward_median","comment_median","like_median"]]
Y_train=train[["forward_count","comment_count","like_count"]]
X_test=predict[["content_media_count","content_length","content_emoji_count","hour","
min","sec","forward_median","comment_median","like_median"]]
Y_test=predict[["forward_count","comment_count","like_count"]]
```

In [40]:

```
print(X_train.shape,Y_train.shape)
print(X_test.shape,Y_test.shape)
```

```
(1044681, 9) (1044681, 3)
(184937, 9) (184937, 3)
```

In [41]:

```
pd.options.mode.use_inf_as_na = True
X_train.fillna(X_train.max(),inplace=True)
X_test.fillna(X_test.max(),inplace=True)
```

```
G:\Anaconda\lib\site-packages\pandas\core\generic.py:5430: SettingWithCopyWarning:
A value is trying to be set on a copy of a slice from a DataFrame

See the caveats in the documentation: http://pandas.pydata.org/pandas-docs/stable/ind
exing.html#indexing-view-versus-copy
  self._update_inplace(new_data)
```

In [42]:

```
lm=linear_model.LinearRegression()
model=lm.fit(X_train,Y_train)
pred=lm.predict(X_test)
pred=pred.round()
pred=(np.maximum(pred,0.))
```

In [43]:

```
print(pred[0:5])
```

```
[[  3.   4.   3.]
 [  3.   4.   3.]
 [  3.   4.   3.]
 [ 13.   7.   8.]
 [  2.   2.   2.]]
```

In [44]:

```
print(model.coef_)
```

```
[[ -2.63364425e-01   1.91174376e-02   5.54642973e-02   3.76150971e-02
    5.27858190e-04   1.68362209e-03   1.58279691e+00   1.67729136e+00
   -2.93017304e-01]
 [ -6.60961017e-01   2.55079540e-03   1.00504045e-01   1.24809468e-02
    2.35146358e-03   1.20243289e-03   7.17940397e-02   1.55348700e+00
    4.45266999e-03]
 [ -8.70474303e-01   9.23346652e-04   5.98607519e-02   2.34643687e-02
    4.45920014e-05   1.52298289e-03   1.45839466e-01   3.91354761e-01
    1.06591334e+00]]
```

In [45]:

```
print(model.intercept_)
```

```
[-1.04897282  0.36245024  0.43019058]
```

In [46]:

```
np.savetxt("G://DMA_PROJECT//weibo_predict_result9.csv",pred,delimiter=',',header="fo
rward_count,comment_count,like_count",comments="")
result=pd.read_csv("G://DMA_PROJECT//weibo_predict_result9.csv")
```

In [47]:

```
train_real_pred = Y_test
train_real_pred['fp']=result['forward_count'].values
train_real_pred['cp']=result['comment_count'].values
train_real_pred['lp']=result['like_count'].values
train_real_pred=train_real_pred.round()
print ("Score on the training set:{0:.2f}%".format(precision(train_real_pred.values)
*100))
```

```
G:\Anaconda\lib\site-packages\ipykernel_launcher.py:2: SettingWithCopyWarning:
A value is trying to be set on a copy of a slice from a DataFrame.
Try using .loc[row_indexer,col_indexer] = value instead

See the caveats in the documentation: http://pandas.pydata.org/pandas-docs/stable/ind
exing.html#indexing-view-versus-copy

G:\Anaconda\lib\site-packages\ipykernel_launcher.py:3: SettingWithCopyWarning:
A value is trying to be set on a copy of a slice from a DataFrame.
Try using .loc[row_indexer,col_indexer] = value instead

See the caveats in the documentation: http://pandas.pydata.org/pandas-docs/stable/ind
exing.html#indexing-view-versus-copy
  This is separate from the ipykernel package so we can avoid doing imports until
G:\Anaconda\lib\site-packages\ipykernel_launcher.py:4: SettingWithCopyWarning:
A value is trying to be set on a copy of a slice from a DataFrame.
Try using .loc[row_indexer,col_indexer] = value instead

See the caveats in the documentation: http://pandas.pydata.org/pandas-docs/stable/ind
exing.html#indexing-view-versus-copy
  after removing the cwd from sys.path.
```

```
Score on the training set:22.47%
```

Team 5A09 DMA Course Project :

Sina Weibo Interaction Prediction Challenge

Modelling Phase

1.
   - Team ID - 5A09
     - Sem - 5TH
     - Div - 'A'
     - School - KLE Technological university

1.
   - Topic ID - 5ADMACP14
   - Project Title - Sina Weibo Intereaction Prediction

1. Problem Statement - To predict the user behaviors such as forwarding, commenting and liking on Sina Weibo Microblogging site.

1.
   - Team Leader - Deepti Nadkarni - 01FE16BCS062 (Roll no-58)
   - Members
     - Apoorva Malemath - 01FE16BCS041 (Roll no-39)
     - Arundati Dixit - 01FE16BCS046 (Roll no-44)
     - Ashish Kar - 01FE16BCS047 (Roll no-45)

----------Pre-Processing Highlights----------

1. TRANSLATION TO ENGLISH CONTENT

In [2]:

```
import pandas as pd
import numpy as np
from sklearn import linear_model
from sklearn.model_selection import train_test_split
from sklearn.metrics import accuracy_score
from matplotlib import pyplot as plt
import statsmodels.api as sm
import import_ipynb
from evaluation import precision
```

```
importing Jupyter notebook from evaluation.ipynb
```

In [5]:

```
df1=pd.read_csv("G://DMA_PROJECT//preprocessed_1.csv")
df2=pd.read_csv("G://DMA_PROJECT//preprocessed_2.csv")
frames=[df1,df2]
traintrans=pd.concat(frames)
```

In [4]:

```
traintrans.head(5)
```

Out[4]:

	u_id	m_id	forward_count	comment_count	like_count

	u_id	m_id	forward_count	comment_count	like_count
0	d38e9bed5d98110dc2489d0d1cac3c2a	7d45833d9865727a88b960b0603c19f6	0.0	0.0	0.0
1	fa13974743d3fe6ff40d21b872325e9e	8169f1d45051e08ef213bf1106b1225d	0.0	0.0	0.0
2	da534fe87e7a52777bee5c30573ed5fd	68cd0258c31c2c525f94febea2d9523b	0.0	0.0	0.0
3	e06a22b7e065e559a1f0bf7841a85c51	00b9f86b4915aedb7db943c54fd19d59	0.0	4.0	3.0
4	f9828598f9664d4e347ef2048ce17734	c7f6f66044c0c5a3330e2c5371be6824	0.0	0.0	0.0

In [8]:

```
traintrans.columns
```

Out[8]:

```
Index(['u_id', 'm_id', 'forward_count', 'comment_count', 'like_count',
       'content', 'date', 'time', 'content_media_count', 'content_spchar',
       'non_emoji_content', 'en_content', 'Unnamed: 1'],
      dtype='object')
```

2. BOW as factor

Processes Involved for Text Processing:

REMOVAL OF NOISE - URL

REMOVAL OF STOPWORDS

STEMMING

LEMMATIZATION

CONVERSION TO LOWERCASE

REMOVAL OF NUMBERS

REMOVAL OF PUNCTUATION

In [60]:

```
trainpp=pd.read_csv("G://DMA_PROJECT//preprocessed.csv")
```

In [61]:

```
trainpp.head(10)
```

Out[61]:

	u_id	m_id	forward_count	comment_count	like_count
0	ef132857ae5c47ff0aa2ce251436258c	a09fd98a3fddd174f281e0b56d14c6fc	0	0	0
1	97775929a27fdb7e0da1e8c64bf796b4	c44d9db6c197a921eb2635017b733d41	0	0	0
2	875a4a77b339d93f819e2c4de5bd0b57	423c182a9ab1a2ba71f97721717607dc	0	0	0
3	935dd42bcab833225d96eb826e2fb959	119426d163f5f77fe626d3e3701289e4	0	1	0
4	c9ef6a4615183d652a777771599dcfbe	b135c8123b51ccd53c26dc76fa1d3ed4	0	0	0

	u_id	m_id	forward_count	comment_count	like_count
5	ca1010cf23e9327e9a68358f5a0f7484	a3dafc001f7e8d8a2169a2625d698cc3	0	1	0
6	b1de85c455b9a42fb1bdf8e44792a50e	d2500bae5669ad1cfbcbaaeb1384a338	0	0	0
7	63c0b7f38fbd83add57273d1ec907551	1e037545970e9983b94acae7bc9ac2c2	1	0	0
8	d38e9bed5d98110dc2489d0d1cac3c2a	abf00ba5489ed889f0ce35b7eb586941	0	0	0
9	dd20701e6bb5bd4eae4df9ef7fcd7103	cb441ee83d8c1cc005f8b4f66748c211	2	2	0

10 rows × 29 columns

In [62]:

```
trainpp.columns
```

Out[62]:

```
Index(['u_id', 'm_id', 'forward_count', 'comment_count', 'like_count',
       'content', 'date', 'time', 'content_media_count', 'content_spchar',
```

```
       'non_emoji_content', 'en_content', 'Unnamed: 1', 'en_contenturl',
       'url_rem', 'en_contentsw', 'Stopwrod_removed', 'Stopword_removed',
       'en_contentst', 'Stemming', 'Stemmingle', 'lemmatization',
       'lemmatizationtl', 'lower', 'lowerrmum', 'no_num', 'lowerrnum',
       'no_numrp', 'no_punc'],
      dtype='object')
```

3. UID Stats as factor

In [6]:

```
stat=pd.read_csv("train_uid_stat.csv")
```

In [7]:

```
stat.head(10)
```

Out[7]:

	u_id	forward_min	forward_max	forward_median	forward_mean	comment_min	comm
0	000127c6126e2b0019f255ed21ac1cb7	0	1	0	0	0	0
1	0001565a5edece1669577e2ace9a6a3d	0	0	0	0	0	1
2	00033a6513b86b2705de9ffa9d37ffb6	0	0	0	0	0	0
3	0004fe2742507420eaa73e119dc83ac5	0	6	0	0	0	1
4	000c663a24a2f91f4ba156fcd4f8b9f2	0	1	0	0	0	7
5	000ce19d2fccb1f22421bec50bf25b08	0	0	0	0	0	0
6	000d7bf7406392b2212dfb4fe907d946	0	0	0	0	0	0
7	0012edb614365800e901c7f2b47e9129	0	0	0	0	0	4
8	001349a053bdecf1a71960f29288ced1	0	0	0	0	0	1
9	0015c42ec93854687a258a7f170c6acf	0	0	0	0	0	0

In [9]:

```
stat.columns
```

Out[9]:

```
Index(['u_id', 'forward_min', 'forward_max', 'forward_median', 'forward_mean',
       'comment_min', 'comment_max', 'comment_median', 'comment_mean',
       'like_min', 'like_max', 'like_median', 'like_mean'],
      dtype='object')
```

4. Initial Predictions without Model and Analysis

Putting known values of stats in predict dataset without any computation and finding accuracy

Best Statistical Factors and Default Value

Analysis

1. The Best Default Value is 0 1 1 (F C L) which can be used for new users in predict dataset or as a default value

2. The Highest Accuracy is for Median factor which is considerable as per current standings in Sina Weibo leaderbord. (Top

Accuracy: 41.73%)

3. We further wanted to add more factors from content like emoji and media, time etc. with these factors to better our accuracy

----------Factors Considered For Modelling----------

In [10]:

```
df1=pd.read_csv("G://DMA_PROJECT//weibo_train1_cptsd.csv")
df2=pd.read_csv("G://DMA_PROJECT//weibo_train2_cptsd.csv")
frames=[df1,df2]
train_dataset=pd.concat(frames)
predict_dataset=pd.read_csv("G://DMA_PROJECT//weibo_predict_cptsd.csv")
```

In [12]:

```
train_dataset.head(5)
```

Out[12]:

	u_id	m_id	forward_count	comment_count	like_count
0	d38e9bed5d98110dc2489d0d1cac3c2a	7d45833d9865727a88b960b0603c19f6	0	0	0
1	d38e9bed5d98110dc2489d0d1cac3c2a	00755196c77936bf44656ada98291c59	0	0	0
2	d38e9bed5d98110dc2489d0d1cac3c2a	4fedf3888b1e16592f0e0bdc8b393845	0	0	0
3	d38e9bed5d98110dc2489d0d1cac3c2a	91be0b8612265aae32725cd4fa80b222	0	0	0
4	d38e9bed5d98110dc2489d0d1cac3c2a	bd2af99ecf1298f5539f0ddfcdd3ed64	0	0	0

5 rows × 33 columns

In [13]:

```
train_dataset.columns
```

Out[13]:

```
Index(['u_id', 'm_id', 'forward_count', 'comment_count', 'like_count',
       'content', 'date', 'time', 'content_media_count', 'content_#_count',
       'content_@_count', 'content_?_count', 'content_!_count',
       'content_length', 'content_emoji_count', 'hour', 'min', 'sec',
       'forward_min', 'forward_max', 'forward_median', 'forward_mean',
       'comment_min', 'comment_max', 'comment_median', 'comment_mean',
       'like_min', 'like_max', 'like_median', 'like_mean', 'default_forward',
       'default_comment', 'default_like'],
      dtype='object')
```

1. Content Factors

1a) content_media_count

1b) content_#_count

1c) content_@_count

1d) content_?_count

1e) content_!_count

1f) content_length

1g) content_emoji_count

2. Time Factors

2a) hour

2b) min

2c) sec

3. Statistical Factors

3a) forward_min

3b) forward_max

3c) forward_median

3d) forward_mean

3e) comment_min

3f) comment_max

3g) comment_median

3h) comment_mean

3i) like_min

3j) like_max

3k) like_median

3l) like_mean

4. Default Values

4a) default_forward

4b) default_comment

4c) default_like

Total Factors: 24

----------Model Building Pre requisite----------

In [34]:

```
train_month1=pd.read_csv("G://DMA_PROJECT//weibo_train_feb_cptsd.csv")
train_month2=pd.read_csv("G://DMA_PROJECT//weibo_train_march_cptsd.csv")
train_month3=pd.read_csv("G://DMA_PROJECT//weibo_train_april_cptsd.csv")
train_month4=pd.read_csv("G://DMA_PROJECT//weibo_train_may_cptsd.csv")
train_month5=pd.read_csv("G://DMA_PROJECT//weibo_train_june_cptsd.csv")
train_month6=pd.read_csv("G://DMA_PROJECT//weibo_train_july_cptsd.csv")
```

In [35]:

```
frames1=[train_month1,train_month2,train_month3,train_month4,train_month5]
train=pd.concat(frames1)
predict=train_month6
```

Library used for modelling: sklearn (linear model)

----------Modelling----------

1. Some Models and Inferences

1a) New Factor Analysis

Model 1: (Factors: Media, #, @, ?, !, Length, Emoji)

In [17]:

```
X_train=train[["content_media_count","content_#_count","content_@_count","content_?_count","conten
t_!_count","content_length","content_emoji_count"]]
Y_train=train[["forward_count","comment_count","like_count"]]
X_test=predict[["content_media_count","content_#_count","content_@_count","content_?_count","conte
nt_!_count","content_length","content_emoji_count"]]
Y_test=predict[["forward_count","comment_count","like_count"]]

print(X_train.shape,Y_train.shape)
print(X_test.shape,Y_test.shape)

pd.options.mode.use_inf_as_na = True
X_train.fillna(X_train.max(),inplace=True)
X_test.fillna(X_test.max(),inplace=True)

lm=linear_model.LinearRegression()
model=lm.fit(X_train,Y_train)
pred=lm.predict(X_test)
pred=pred.round()
pred=(np.maximum(pred,0.))
```

```
(1044681, 7) (1044681, 3)
(184937, 7) (184937, 3)
```

```
G:\Anaconda\lib\site-packages\pandas\core\generic.py:5430: SettingWithCopyWarning:
A value is trying to be set on a copy of a slice from a DataFrame

See the caveats in the documentation: http://pandas.pydata.org/pandas-
docs/stable/indexing.html#indexing-view-versus-copy
  self._update_inplace(new_data)
```

In [18]:

```
print(pred[0:5])
print(model.coef_)
print(model.intercept_)
```

```
[[ 4.  1.  2.]
 [ 3.  2.  3.]
 [ 3.  1.  1.]
 [ 7.  1.  2.]
 [ 9.  2.  3.]]
[[-2.13937044 -0.36595303 -0.29844433 -0.04270781 -0.20884535  0.05323583
  -0.11373958]
 [-1.46356558 -0.15223682 -0.02946097 -0.00326344 -0.04880305  0.00769467
   0.1747673 ]
 [-2.6330917  -0.13749083 -0.32811723 -0.01394959 -0.071054    0.01544652
   0.26583545]]
[ 1.31437859  1.76293348  2.98461306]
```

In [19]:

```
np.savetxt("G://DMA_PROJECT//weibo_predict_result2.csv",pred,delimiter=',',header="forward_count,cc
mment_count,like_count",comments="")
result=pd.read_csv("G://DMA_PROJECT//weibo_predict_result2.csv")
```

In [20]:

```
train_real_pred = Y_test
forward=result['forward_count'].values
comment=result['forward_count'].values
like=result['forward_count'].values
train_real_pred['fp'],train_real_pred['cp'],train_real_pred['lp'] = forward,comment,like
print ("Score on the training set:{0:.2f}%".format(precision(train_real_pred.values)*100))
```

```
G:\Anaconda\lib\site-packages\ipykernel_launcher.py:5: SettingWithCopyWarning:
A value is trying to be set on a copy of a slice from a DataFrame.
Try using .loc[row_indexer,col_indexer] = value instead

See the caveats in the documentation: http://pandas.pydata.org/pandas-
docs/stable/indexing.html#indexing-view-versus-copy
  """
```

```
Score on the training set:13.69%
```

Model 2: (Factors: Media, Length, Emoji)

In [22]:

```
X_train=train[["content_media_count","content_length","content_emoji_count"]]
Y_train=train[["forward_count","comment_count","like_count"]]
X_test=predict[["content_media_count","content_length","content_emoji_count"]]
Y_test=predict[["forward_count","comment_count","like_count"]]

print(X_train.shape,Y_train.shape)
print(X_test.shape,Y_test.shape)

pd.options.mode.use_inf_as_na = True
X_train.fillna(X_train.max(),inplace=True)
X_test.fillna(X_test.max(),inplace=True)

lm=linear_model.LinearRegression()
model=lm.fit(X_train,Y_train)
pred=lm.predict(X_test)
pred=pred.round()
pred=(np.maximum(pred,0.))
```

```
(1044681, 3) (1044681, 3)
(184937, 3) (184937, 3)
```

```
G:\Anaconda\lib\site-packages\pandas\core\generic.py:5430: SettingWithCopyWarning:
```

```
A value is trying to be set on a copy of a slice from a DataFrame

See the caveats in the documentation: http://pandas.pydata.org/pandas-
docs/stable/indexing.html#indexing-view-versus-copy
  self._update_inplace(new_data)
```

In [23]:

```
print(pred[0:5])
print(model.coef_)
print(model.intercept_)
```

```
[[ 4.  1.  2.]
 [ 3.  2.  3.]
 [ 3.  1.  1.]
 [ 7.  1.  2.]
 [ 8.  1.  3.]]
[[-2.19780299  0.05049652 -0.14777505]
 [-1.47377848  0.00679751  0.16018448]
 [-2.68756793  0.01390734  0.25398113]]
[ 1.23425639  1.73473692  2.94322868]
```

In [24]:

```
np.savetxt("G://DMA_PROJECT//weibo_predict_result3.csv",pred,delimiter=',',header="forward_count,cc
mment_count,like_count",comments="")
result=pd.read_csv("G://DMA_PROJECT//weibo_predict_result3.csv")
```

In [25]:

```
train_real_pred = Y_test
forward=result['forward_count'].values
comment=result['forward_count'].values
like=result['forward_count'].values
train_real_pred['fp'],train_real_pred['cp'],train_real_pred['lp'] = forward,comment,like
print ("Score on the training set:{0:.2f}%".format(precision(train_real_pred.values)*100))
```

```
G:\Anaconda\lib\site-packages\ipykernel_launcher.py:5: SettingWithCopyWarning:
A value is trying to be set on a copy of a slice from a DataFrame.
Try using .loc[row_indexer,col_indexer] = value instead

See the caveats in the documentation: http://pandas.pydata.org/pandas-
docs/stable/indexing.html#indexing-view-versus-copy
  """
```

```
Score on the training set:13.79%
```

Model 3: (Factors:Time: (Hour, Min, Sec), Media, Length, Emoji)

In [27]:

```
X_train=train[["content_media_count","content_length","content_emoji_count","hour","min","sec"]]
Y_train=train[["forward_count","comment_count","like_count"]]
X_test=predict[["content_media_count","content_length","content_emoji_count","hour","min","sec"]]
Y_test=predict[["forward_count","comment_count","like_count"]]
print(X_train.shape,Y_train.shape)
print(X_test.shape,Y_test.shape)

pd.options.mode.use_inf_as_na = True
X_train.fillna(X_train.max(),inplace=True)
X_test.fillna(X_test.max(),inplace=True)

lm=linear_model.LinearRegression()
model=lm.fit(X_train,Y_train)
pred=lm.predict(X_test)
pred=pred.round()
pred=(np.maximum(pred,0.))
```

```
(1044681, 6) (1044681, 3)
```

```
(184937, 6) (184937, 3)
```

```
G:\Anaconda\lib\site-packages\pandas\core\generic.py:5430: SettingWithCopyWarning:
A value is trying to be set on a copy of a slice from a DataFrame

See the caveats in the documentation: http://pandas.pydata.org/pandas-
docs/stable/indexing.html#indexing-view-versus-copy
  self._update_inplace(new_data)
```

In [28]:

```
print(pred[0:5])
print(model.coef_)
print(model.intercept_)
```

```
[[  4.   1.   1.]
 [  1.   2.   3.]
 [  2.   1.   1.]
 [  4.   1.   1.]
 [ 10.   2.   4.]]
[[-2.16126691  0.04995623 -0.14079719  0.09097968 -0.05802816 -0.03386975]
 [-1.46714342  0.0068272   0.15966872  0.02589544 -0.00483482 -0.00418282]
 [-2.6652836   0.0137497   0.25624217  0.07054347 -0.03242168 -0.01633076]]
[ 2.62859063  1.63033587  3.36618167]
```

In [29]:

```
np.savetxt("G://DMA_PROJECT//weibo_predict_result4.csv",pred,delimiter=',',header="forward_count,cc
mment_count,like_count",comments="")
result=pd.read_csv("G://DMA_PROJECT//weibo_predict_result4.csv")
```

In [30]:

```
train_real_pred = Y_test
forward=result['forward_count'].values
comment=result['forward_count'].values
like=result['forward_count'].values
train_real_pred['fp'],train_real_pred['cp'],train_real_pred['lp'] = forward,comment,like
print ("Score on the training set:{0:.2f}%".format(precision(train_real_pred.values)*100))
```

```
G:\Anaconda\lib\site-packages\ipykernel_launcher.py:5: SettingWithCopyWarning:
A value is trying to be set on a copy of a slice from a DataFrame.
Try using .loc[row_indexer,col_indexer] = value instead

See the caveats in the documentation: http://pandas.pydata.org/pandas-
docs/stable/indexing.html#indexing-view-versus-copy
  """
```

```
Score on the training set:15.41%
```

Analysis: New Factors not yeilding very satisfactory results

1 b) Combination with old factors

Model 4: (Factors: Median,Time: (Hour, Min, Sec), Media, Length, Emoji)

In [40]:

```
X_train=train[["content_media_count","content_length","content_emoji_count","hour","min","sec","for
ward_median","comment_median","like_median"]]
Y_train=train[["forward_count","comment_count","like_count"]]
X_test=predict[["content_media_count","content_length","content_emoji_count","hour","min","sec","fo
rward_median","comment_median","like_median"]]
Y_test=predict[["forward_count","comment_count","like_count"]]
print(X_train.shape,Y_train.shape)
print(X_test.shape,Y_test.shape)
```

```
pd.options.mode.use_inf_as_na = True
X_train.fillna(X_train.max(),inplace=True)
X_test.fillna(X_test.max(),inplace=True)

lm=linear_model.LinearRegression()
model=lm.fit(X_train,Y_train)
pred=lm.predict(X_test)
pred=pred.round()
pred=(np.maximum(pred,0.))
```

```
(1044681, 9) (1044681, 3)
(184937, 9) (184937, 3)
```

```
G:\Anaconda\lib\site-packages\pandas\core\generic.py:5430: SettingWithCopyWarning:
A value is trying to be set on a copy of a slice from a DataFrame

See the caveats in the documentation: http://pandas.pydata.org/pandas-
docs/stable/indexing.html#indexing-view-versus-copy
  self._update_inplace(new_data)
```

In [41]:

```
print(pred[0:5])
print(model.coef_)
print(model.intercept_)
```

```
[[  3.   4.   3.]
 [  3.   4.   3.]
 [  3.   4.   3.]
 [ 13.   7.   8.]
 [  2.   2.   2.]]
[[ -2.63364425e-01   1.91174376e-02   5.54642973e-02   3.76150971e-02
    5.27858190e-04   1.68362209e-03   1.58279691e+00   1.67729136e+00
   -2.93017304e-01]
 [ -6.60961017e-01   2.55079540e-03   1.00504045e-01   1.24809468e-02
    2.35146358e-03   1.20243289e-03   7.17940397e-02   1.55348700e+00
    4.45266999e-03]
 [ -8.70474303e-01   9.23346652e-04   5.98607519e-02   2.34643687e-02
    4.45920014e-05   1.52298289e-03   1.45839466e-01   3.91354761e-01
    1.06591334e+00]]
[-1.04897282  0.36245024  0.43019058]
```

Analysis: Weights assigned to Statistical Factors is higher

In [42]:

```
np.savetxt("G://DMA_PROJECT//weibo_predict_result5.csv",pred,delimiter=',',header="forward_count,cc
mment_count,like_count",comments="")
result=pd.read_csv("G://DMA_PROJECT//weibo_predict_result5.csv")
```

In [44]:

```
train_real_pred = Y_test
train_real_pred['fp']=result['forward_count'].values
train_real_pred['cp']=result['comment_count'].values
train_real_pred['lp']=result['like_count'].values
train_real_pred=train_real_pred.round()
print ("Score on the training set:{0:.2f}%".format(precision(train_real_pred.values)*100))
```

```
G:\Anaconda\lib\site-packages\ipykernel_launcher.py:2: SettingWithCopyWarning:
A value is trying to be set on a copy of a slice from a DataFrame.
Try using .loc[row_indexer,col_indexer] = value instead

See the caveats in the documentation: http://pandas.pydata.org/pandas-
docs/stable/indexing.html#indexing-view-versus-copy

G:\Anaconda\lib\site-packages\ipykernel_launcher.py:3: SettingWithCopyWarning:
A value is trying to be set on a copy of a slice from a DataFrame.
Try using .loc[row_indexer,col_indexer] = value instead
```

```
See the caveats in the documentation: http://pandas.pydata.org/pandas-
docs/stable/indexing.html#indexing-view-versus-copy
  This is separate from the ipykernel package so we can avoid doing imports until
G:\Anaconda\lib\site-packages\ipykernel_launcher.py:4: SettingWithCopyWarning:
A value is trying to be set on a copy of a slice from a DataFrame.
Try using .loc[row_indexer,col_indexer] = value instead

See the caveats in the documentation: http://pandas.pydata.org/pandas-
docs/stable/indexing.html#indexing-view-versus-copy
  after removing the cwd from sys.path.
```

```
Score on the training set:32.47%
```

Analysis:

1. Median factor boosted accuracy yet again

2. Separate predictions are necessory for F C L as in cases like Forward won't need Like Median and Comment Median. Also Each of F C L can have different best factors

1 c) Separate prediction Models for FCL and Final Model

Model 5: (Factors: Median, Min, Max, Media, Emoji)

In [45]:

```
X_train1=train[["forward_median","forward_mean","forward_min","content_media_count","content_emoji_
count"]]
Y_train1=train[["forward_count"]]
X_test1=predict[["forward_median","forward_mean","forward_min","content_media_count","content_emoji
_count"]]
Y_test1=predict[["forward_count"]]

X_train2=train[["comment_median","comment_mean","comment_min","content_media_count","content_emoji_
count"]]
Y_train2=train[["comment_count"]]
X_test2=predict[["comment_median","comment_mean","comment_min","content_media_count","content_emoji
_count"]]
Y_test2=predict[["comment_count"]]

X_train3=train[["like_median","like_mean","like_min","content_media_count","content_emoji_count"]]
Y_train3=train[["like_count"]]
X_test3=predict[["like_median","like_mean","like_min","content_media_count","content_emoji_count"]
]
Y_test3=predict[["like_count"]]

pd.options.mode.use_inf_as_na = True
X_train1.fillna(X_train1.max(),inplace=True)
X_test1.fillna(X_test1.max(),inplace=True)
X_train2.fillna(X_train2.max(),inplace=True)
X_test2.fillna(X_test2.max(),inplace=True)
X_train3.fillna(X_train3.max(),inplace=True)
X_test3.fillna(X_test3.max(),inplace=True)

print(X_train1.shape,Y_train1.shape)
print(X_test1.shape,Y_test1.shape)

print(X_train2.shape,Y_train2.shape)
print(X_test2.shape,Y_test2.shape)

print(X_train3.shape,Y_train3.shape)
print(X_test3.shape,Y_test3.shape)
```

```
G:\Anaconda\lib\site-packages\pandas\core\generic.py:5430: SettingWithCopyWarning:
A value is trying to be set on a copy of a slice from a DataFrame

See the caveats in the documentation: http://pandas.pydata.org/pandas-
```

```
See the caveats in the documentation: http://pandas.pydata.org/pandas-
docs/stable/indexing.html#indexing-view-versus-copy
  self._update_inplace(new_data)
```

```
(1044681, 5) (1044681, 1)
(184937, 5) (184937, 1)
(1044681, 5) (1044681, 1)
(184937, 5) (184937, 1)
(1044681, 5) (1044681, 1)
(184937, 5) (184937, 1)
```

In [46]:

```
lm1=linear_model.LinearRegression()
model1=lm1.fit(X_train1,Y_train1)
pred1=lm1.predict(X_test1)
pred1=pred1.round()
pred1=(np.maximum(pred1,0.))

lm2=linear_model.LinearRegression()
model2=lm2.fit(X_train2,Y_train2)
pred2=lm2.predict(X_test2)
pred2=pred2.round()
pred2=(np.maximum(pred2,0.))

lm3=linear_model.LinearRegression()
model3=lm3.fit(X_train3,Y_train3)
pred3=lm3.predict(X_test3)
pred3=pred3.round()
pred3=(np.maximum(pred3,0.))
```

In [48]:

```
print(pred1[0:5])
print(model1.coef_)
print(model1.intercept_)

print(pred2[0:5])
print(model2.coef_)
print(model2.intercept_)

print(pred3[0:5])
print(model3.coef_)
print(model3.intercept_)
```

```
[[  2.]
 [  2.]
 [  2.]
 [ 13.]
 [ 12.]]
[[ 0.44920013  0.84888874 -0.28999774  0.32885539  0.01281363]]
[-0.20643995]
[[ 4.]
 [ 4.]
 [ 4.]
 [ 4.]
 [ 3.]]
[[-0.23456946  1.16942801  0.04188341 -0.33471336  0.02751557]]
[ 0.19200623]
[[ 3.]
 [ 3.]
 [ 3.]
 [ 7.]
 [ 4.]]
[[ 0.13417819  0.93515754 -0.20071856 -0.64595136 -0.11687872]]
[ 0.47901386]
```

In [47]:

```
np.savetxt("G://DMA_PROJECT//weibo_predict_resultc1.csv",pred1,delimiter=',',header="forward_count"
,comments="")
result1=pd.read_csv("G://DMA_PROJECT//weibo_predict_resultc1.csv")
np.savetxt("G://DMA_PROJECT//weibo_predict_resultc2.csv",pred2,delimiter=',',header="comment_count"
,comments="")
```

```
,comments="")
result2=pd.read_csv("G://DMA_PROJECT//weibo_predict_resultc2.csv")
np.savetxt("G://DMA_PROJECT//weibo_predict_resultc3.csv",pred3,delimiter=',',header="like_count",c
omments="")
result3=pd.read_csv("G://DMA_PROJECT//weibo_predict_resultc3.csv")
```

In [49]:

```
train_real_pred = pd.concat([Y_test1,Y_test2,Y_test3],axis=1)
train_real_pred['fp']=result1['forward_count']
train_real_pred['cp']=result2['comment_count']
train_real_pred['lp']=result3['like_count']
train_real_pred=train_real_pred.round()
print ("Score on the training set:{0:.4f}%".format(precision(train_real_pred.values)*100))
```

```
Score on the training set:38.0378%
```

Analysis: This is the best accuracy we got out of all the models we tried

We got the same accuracy for another model....

Model 6: (Factors: Median, Min, Max, Media, Emoji, Default Values)

In [50]:

```
X_train1=train[["forward_median","forward_mean","forward_min","content_media_count","content_emoji_
count","default_forward"]]
Y_train1=train[["forward_count"]]
X_test1=predict[["forward_median","forward_mean","forward_min","content_media_count","content_emoji
_count","default_forward"]]
Y_test1=predict[["forward_count"]]

X_train2=train[["comment_median","comment_mean","comment_min","content_media_count","content_emoji_
count","default_comment"]]
Y_train2=train[["comment_count"]]
X_test2=predict[["comment_median","comment_mean","comment_min","content_media_count","content_emoji
_count","default_comment"]]
Y_test2=predict[["comment_count"]]

X_train3=train[["like_median","like_mean","like_min","content_media_count","content_emoji_count","
default_like"]]
Y_train3=train[["like_count"]]
X_test3=predict[["like_median","like_mean","like_min","content_media_count","content_emoji_count",
"default_like"]]
Y_test3=predict[["like_count"]]

pd.options.mode.use_inf_as_na = True
X_train1.fillna(X_train1.max(),inplace=True)
X_test1.fillna(X_test1.max(),inplace=True)
X_train2.fillna(X_train2.max(),inplace=True)
X_test2.fillna(X_test2.max(),inplace=True)
X_train3.fillna(X_train3.max(),inplace=True)
X_test3.fillna(X_test3.max(),inplace=True)

print(X_train1.shape,Y_train1.shape)
print(X_test1.shape,Y_test1.shape)

print(X_train2.shape,Y_train2.shape)
print(X_test2.shape,Y_test2.shape)

print(X_train3.shape,Y_train3.shape)
print(X_test3.shape,Y_test3.shape)
```

```
G:\Anaconda\lib\site-packages\pandas\core\generic.py:5430: SettingWithCopyWarning:
A value is trying to be set on a copy of a slice from a DataFrame

See the caveats in the documentation: http://pandas.pydata.org/pandas-
docs/stable/indexing.html#indexing-view-versus-copy
  self._update_inplace(new_data)
```

```
(1044681, 6) (1044681, 1)
(184937, 6) (184937, 1)
(1044681, 6) (1044681, 1)
(184937, 6) (184937, 1)
(1044681, 6) (1044681, 1)
(184937, 6) (184937, 1)
```

In [51]:

```
lm1=linear_model.LinearRegression()
model1=lm1.fit(X_train1,Y_train1)
pred1=lm1.predict(X_test1)
pred1=pred1.round()
pred1=(np.maximum(pred1,0.))

lm2=linear_model.LinearRegression()
model2=lm2.fit(X_train2,Y_train2)
pred2=lm2.predict(X_test2)
pred2=pred2.round()
pred2=(np.maximum(pred2,0.))

lm3=linear_model.LinearRegression()
model3=lm3.fit(X_train3,Y_train3)
pred3=lm3.predict(X_test3)
pred3=pred3.round()
pred3=(np.maximum(pred3,0.))
```

In [52]:

```
print(pred1[0:5])
print(model1.coef_)
print(model1.intercept_)

print(pred2[0:5])
print(model2.coef_)
print(model2.intercept_)

print(pred3[0:5])
print(model3.coef_)
print(model3.intercept_)
```

```
[[  2.]
 [  2.]
 [  2.]
 [ 13.]
 [ 12.]]
[[ 0.44920013  0.84888874 -0.28999774  0.32885539  0.01281363  0.        ]]
[-0.20643995]
[[ 4.]
 [ 4.]
 [ 4.]
 [ 4.]
 [ 3.]]
[[-0.23456946  1.16942801  0.04188341 -0.33471336  0.02751557  0.        ]]
[ 0.19200623]
[[ 3.]
 [ 3.]
 [ 3.]
 [ 7.]
 [ 4.]]
[[ 0.13417819  0.93515754 -0.20071856 -0.64595136 -0.11687872  0.        ]]
[ 0.47901386]
```

In [53]:

```
np.savetxt("G://DMA_PROJECT//weibo_predict_resultd1.csv",pred1,delimiter=',',header="forward_count"
,comments="")
result1=pd.read_csv("G://DMA_PROJECT//weibo_predict_resultd1.csv")
np.savetxt("G://DMA_PROJECT//weibo_predict_resultd2.csv",pred2,delimiter=',',header="comment_count"
,comments="")
result2=pd.read_csv("G://DMA_PROJECT//weibo_predict_resultd2.csv")
np.savetxt("G://DMA_PROJECT//weibo_predict_resultd3.csv",pred3,delimiter=',',header="like_count",c
omments="")
```

```
result3=pd.read_csv("G://DMA_PROJECT//weibo_predict_resultd3.csv")
```

In [54]:

```
train_real_pred = pd.concat([Y_test1,Y_test2,Y_test3],axis=1)
train_real_pred['fp']=result1['forward_count']
train_real_pred['cp']=result2['comment_count']
train_real_pred['lp']=result3['like_count']
train_real_pred=train_real_pred.round()
print ("Score on the training set:{0:.4f}%".format(precision(train_real_pred.values)*100))
```

```
Score on the training set:38.0378%
```

Analysis: Since Model with less factors is more preferred so Model 5 is the Final Model

----------Final Model Analysis----------

Factors for forward: "forward_median","forward_mean","forward_min","content_media_count","con

Factors for comment: "comment_median","comment_mean","comment_min","content_media_count"

Factors for like: "like_median","like_mean","like_min","content_media_count","content_emoji_c

Linear Equations:

forward_count = 0.44 x forward_median + 0.84 x forward_mean - 0.28 x forward_min + 0.32 x content_media_count + 0.01 x content_emoji_count - 0.2

comment_count = 0.23 x comment_median + 1.16 x comment_mean - 0.04 x comment_min - 0.33 x content_media_count + 0.02 x content_emoji_count + 0.19

like_count = 0.13 x like_median + 0.93 x like_mean - 0.2 x like_min - 0.64 x content_media_count - 0.11 x content_emoji_count + 0.47

Final Precision: 38.04%

Graphical Analysis of final Model

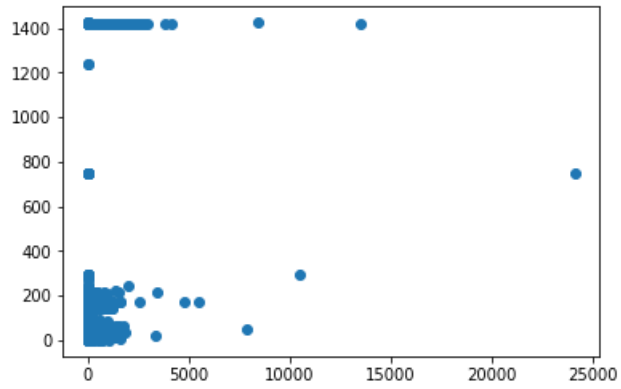
Forward Predictions

In [57]:

```
plt.scatter(Y_test1,pred1)
```

Out[57]:

```
<matplotlib.collections.PathCollection at 0x5eefb6fc18>
```



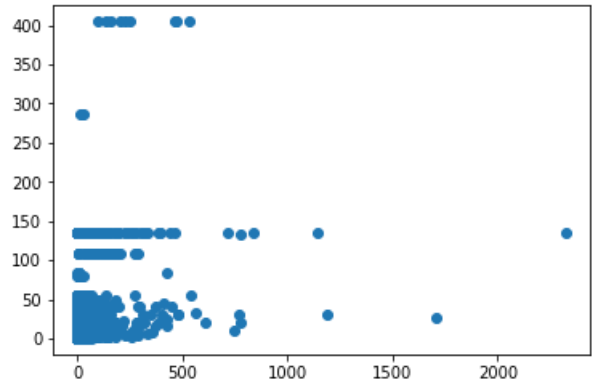
Comment Predictions

In [58]:

```
plt.scatter(Y_test2,pred2)
```

Out[58]:

```
<matplotlib.collections.PathCollection at 0x5eefae5668>
```



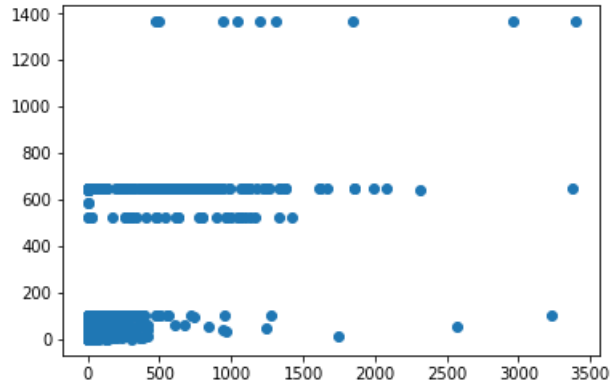
Like Predictions

In [59]:

```
plt.scatter(Y_test3,pred3)
```

Out[59]:

```
<matplotlib.collections.PathCollection at 0x5eefb960b8>
```



Analysis: Better prediction for lower F C L

Scope and Learnings

1. Though we got satisfactory results from our above factors, Sina Weibo Prediction consists of vast factors for prediction and further new factors are very much possible to find like BOW and polarity which can increase the accuracy. More In depth analysis for these microblog site factors can lead to more better results.

2. Due to some factors hard to process like BOW and Polarity for relatively large data we might not have used them for selected models used here but these can also be used for better results.

3. We have considered BOW and polarity for approx first 10K tuples and computed accuracy.This work can further be carried out.

Limitation

Prediction for higher F C L is less accurate

In [2]:

```
import numpy as np
import pandas as pd
import matplotlib.pyplot as plt
import seaborn as sns
%pylab inline
import copy
from googletrans import Translator
import pandas as pd
import numpy as np
import csv
import re
import jieba
import time
import json
from sklearn.feature_extraction.text import CountVectorizer
from sklearn import linear_model
from sklearn.externals import joblib
from nltk.corpus import stopwords as e_stopwords
from datetime import datetime, timedelta
import jieba
import sys

from nltk.corpus import stopwords
from nltk.tokenize import word_tokenize
```

```
Populating the interactive namespace from numpy and matplotlib
```

```
C:\Users\DELL\Anaconda3\lib\site-packages\IPython\core\magics\pylab.py:160: UserWarni
ng: pylab import has clobbered these variables: ['copy', 'datetime']
`%matplotlib` prevents importing * from pylab and numpy
  "\n`%matplotlib` prevents importing * from pylab and numpy"
```

Random Forest Regressor for Statistical Factors

In [3]:

```
import import_ipynb
from evaluation import precision
from runTime import runTime
```

```
importing Jupyter notebook from evaluation.ipynb
importing Jupyter notebook from runTime.ipynb
```

In [4]:

```
#Reading data from document
import pandas as pd
df_pre=pd.read_csv("E:\\DMA_PRE\\PREPROCESSED.csv")
```

In [21]:

```
df_pre.shape[0]
```

Out[21]:

```
37263
```

In [17]:

```
train=df_pre[0:8000]
cv=df_pre[8001:10000]
```

In [19]:

```
df_stat=pd.read_csv("E:\\DMA_PRE\\train_uid_stat.csv")
```

In [20]:

```
df_stat.shape
```

Out[20]:

```
(37263, 13)
```

In [62]:

```
df_stat.head(5)
```

Out[62]:

	u_id	forward_min	forward_max	forward_median	for
0	000127c6126e2b0019f255ed21ac1cb7	0	1	0	0
1	0001565a5edece1669577e2ace9a6a3d	0	0	0	0
2	00033a6513b86b2705de9ffa9d37ffb6	0	0	0	0
3	0004fe2742507420eaa73e119dc83ac5	0	6	0	0
4	000c663a24a2f91f4ba156fcd4f8b9f2	0	1	0	0

In [4]:

```
train_all1=pd.read_csv('E:\\5th-Sem\\DMA Project\\Model
Evaluation\\weibo_train1_cp.csv')
train_all2=pd.read_csv('E:\\5th-Sem\\DMA Project\\Model
Evaluation\\weibo_train2_cp.csv')
frames=[train_all1,train_all2]
train_all=pd.concat(frames)
```

In [71]:

```
train_all.shape[0]
```

Out[71]:

```
1229618
```

In [73]:

```
df_merge = pd.merge(train_all, df_stat, how='left', on=['u_id'])
```

In [5]:

```
df1=pd.read_csv("E:\\5th-Sem\\DMA Project\\Project\\weibo_train1_cpts.csv")
df2=pd.read_csv("E:\\5th-Sem\\DMA Project\\Project\\weibo_train2_cpts.csv")
frames=[df1,df2]
train_all=pd.concat(frames)
```

In [6]:

```
X=train_all[["content_media_count","content_#_count","content_length","content_emoji_
count","forward_median","comment_median","like_median"]]
y=train_all[['forward_count', 'comment_count', 'like_count']]
```

In [7]:

```
from sklearn import cross_validation
## Spliting of training dataset into 70% training data and 30% testing data randomly
features_train, features_test, labels_train, labels_test = cross_validation.train_tes
t_split(X, y, test_size=0.3, random_state=1)
```

```
C:\Users\DELL\Anaconda3\lib\site-packages\sklearn\cross_validation.py:41: Deprecation
Warning: This module was deprecated in version 0.18 in favor of the model_selection m
odule into which all the refactored classes and functions are moved. Also note that t
he interface of the new CV iterators are different from that of this module. This
module will be removed in 0.20.
  "This module will be removed in 0.20.", DeprecationWarning)
```

Model 1 (Predicting all 3 values together)

In [ ]:

```
from sklearn.ensemble import RandomForestRegressor

x = features_train
y = labels_train
x1 = features_test
y1 = labels_test



regr = RandomForestRegressor(max_depth=30, random_state=0,n_estimators=100)
regr.fit(x, y)
y11_predict = regr.predict(x1)
```

```
y11_predict=y11_predict.round()
y11_predict=(np.maximum(y11_predict,0.))
#print(r2_score(y1, y11_predict) ) #Random forest regressor
```

In [16]:

```
np.savetxt("E:\\weibo_predict_result.csv",y11_predict,delimiter=',',header="forward_count,comment_count,like_count",comments="")
result=pd.read_csv("E:\\weibo_predict_result.csv")
```

In [20]:

```
train_real_pred = labels_test
forward=result['forward_count'].values
comment=result['comment_count'].values
like=result['like_count'].values
train_real_pred['fp'],train_real_pred['cp'],train_real_pred['lp'] = forward,comment,like
print ("Score on the training set:{0:.2f}%".format(precision(train_real_pred.values)*100))
```

```
Score on the training set:30.04%
```

In [24]:

```
train_all.head(1).T
```

Out[24]:

	0
u_id	d38e9bed5d98110dc2489d0d1cac3c2a
m_id	7d45833d9865727a88b960b0603c19f6
forward_count	0
comment_count	0
like_count	0
content	丽江旅游(sz002033)#股票##炒股##财经##理财##投资#推荐包赢股，盈利对半分成...
date	2015-02-23
time	17:41:29
content_media_count	0
content_#_count	10
content_@_count	0
content_?_count	0
content_!_count	0
content_length	62

~~content_length~~	~~62~~
~~content_emoji_count~~	~~0~~
hour	17
min	41
sec	29
forward_min	0
forward_max	114
forward_median	0
forward_mean	1
comment_min	0
comment_max	48
comment_median	0
comment_mean	0
like_min	0
like_max	5
like_median	0
like_mean	0

Model 2 - Constructing 3 individual models and concatinating the results

In [7]:

```
X=train_all[["forward_median","forward_mean","forward_min","content_media_count","con
tent_emoji_count"]]
y=train_all['forward_count']
from sklearn import cross_validation
## Spliting of training dataset into 70% training data and 30% testing data randomly
features_train, features_test, labels_train, labels_test = cross_validation.train_tes
t_split(X, y, test_size=0.3, random_state=1)
from sklearn.ensemble import RandomForestRegressor

x = features_train
y = labels_train
x1 = features_test
y1 = labels_test



regr = RandomForestRegressor(max_depth=30, random_state=0,n_estimators=100)
regr.fit(x, y)
y11_predict = regr.predict(x1)
```

```
#print(r2_score(y1, y11_predict) ) #Random forest regressor
np.savetxt("E:\\weibo_predict_result1.csv",y11_predict,delimiter=',',header="forward_
count",comments="")
result1=pd.read_csv("E:\\weibo_predict_result1.csv")
```

```
C:\Users\DELL\Anaconda3\lib\site-packages\sklearn\cross_validation.py:41: Deprecation
Warning: This module was deprecated in version 0.18 in favor of the model_selection m
odule into which all the refactored classes and functions are moved. Also note that t
he interface of the new CV iterators are different from that of this module. This
module will be removed in 0.20.
  "This module will be removed in 0.20.", DeprecationWarning)
```

In [11]:

```
X=train_all[["comment_median","comment_mean","comment_min","content_media_count","con
tent_emoji_count"]]
y=train_all['comment_count']
from sklearn import cross_validation
## Spliting of training dataset into 70% training data and 30% testing data randomly
features_train, features_test, labels_train, labels_test = cross_validation.train_tes
t_split(X, y, test_size=0.3, random_state=1)
from sklearn.ensemble import RandomForestRegressor

x = features_train
y = labels_train
x1 = features_test
y2 = labels_test



regr = RandomForestRegressor(max_depth=30, random_state=0,n_estimators=100)
regr.fit(x, y)
y11_predict = regr.predict(x1)
#print(r2_score(y1, y11_predict) ) #Random forest regressor
np.savetxt("E:\\weibo_predict_result2.csv",y11_predict,delimiter=',',header="comment_
count",comments="")
result2=pd.read_csv("E:\\weibo_predict_result2.csv")
```

In [12]:

```
X=train_all[["like_median","like_mean","like_min","content_media_count","content_emoj
i_count"]]
y=train_all['like_count']
from sklearn import cross_validation
## Spliting of training dataset into 70% training data and 30% testing data randomly
features_train, features_test, labels_train, labels_test = cross_validation.train_tes
t_split(X, y, test_size=0.3, random_state=1)
from sklearn.ensemble import RandomForestRegressor

x = features_train
y = labels_train
x1 = features_test
```

```
y3 = labels_test



regr = RandomForestRegressor(max_depth=30, random_state=0,n_estimators=100)
regr.fit(x, y)
y11_predict = regr.predict(x1)
#print(r2_score(y1, y11_predict) ) #Random forest regressor
np.savetxt("E:\\weibo_predict_result3.csv",y11_predict,delimiter=',',header="like_cou
nt",comments="")
result3=pd.read_csv("E:\\weibo_predict_result3.csv")
```

In [13]:

```
train_real_pred = pd.concat([y1,y2,y3],axis=1)
train_real_pred['fp']=result1['forward_count']
train_real_pred['cp']=result2['comment_count']
train_real_pred['lp']=result3['like_count']
train_real_pred=train_real_pred.round()
print ("Score on the training set:{0:.4f}%".format(precision(train_real_pred.values)
*100))
```

```
Score on the training set:12.1019%
```

In [1]:

```
import pandas as pd
import csv
```

In [2]:

```
import _pickle as cPickle
import import_ipynb
import pandas as pd
import numpy as np
from genUidStat import loadData,genUidStat
from evaluation import precision
from runTime import runTime
from pathos.pools import _ProcessPool
from multiprocess.pool import Pool
```

```
importing Jupyter notebook from evaluation.ipynb
importing Jupyter notebook from runTime.ipynb
```

In [3]:

```
df1=pd.read_csv("weibo_train1.csv")
df2=pd.read_csv("weibo_train2.csv")
frames=[df1,df2]
traindata=pd.concat(frames)
```

In [4]:

```
def splitDataFrameIntoSmaller(df, chunkSize = 10000):
    listOfDf = list()
    numberChunks = len(df) // chunkSize + 1
    for i in range(numberChunks):
        listOfDf.append(df[i*chunkSize:(i+1)*chunkSize])
    return listOfDf
```

In [5]:

```
uid_stat=pd.read_csv("train_uid_stat.csv")
```

In [6]:

```
uid = splitDataFrameIntoSmaller(uid_stat, chunkSize = 500)
```

In [7]:

```
uid[0].shape[0]
```

Out[7]:

```
500
```

In [8]:

```
def search_all_uid(stat_dic,file):
 import pandas as pd
 import numpy as np
 def _deviation(predict, real, kind):
  t = 5.0 if kind=='f' else 3.0
  return abs(predict - real) / (real + t)
 def _precision_i(fp, fr, cp, cr, lp, lr):
  return 1 - 0.5 * _deviation(fp, fr, 'f') - 0.25 * _deviation(cp, cr, 'c') - 0.25 *
_deviation(lp, lr, 'l')
 def _sgn(x):
  return 1 if x>0 else 0
 def _count_i(fr, cr, lr):
  x = fr + cr + lr
  return 101 if x>100 else (x+1)
 def precision(real_and_predict):
  numerator,denominator = 0.0,0.0
  for  fr, cr, lr,fp, cp, lp in real_and_predict:
   numerator += _count_i(fr, cr, lr) * _sgn(_precision_i(fp, fr, cp, cr, lp, lr) - 0.
8)
   denominator += _count_i(fr, cr, lr)
  return (numerator / denominator)
 def score(uid_data,pred):
  """
  uid_data:
   pd.DataFrame
  pred:
   list, [fp,cp,lp]
  """
  uid_real_pred = uid_data[['forward_count','comment_count','like_count']]
  uid_real_pred['fp'] = pred[0]
  uid_real_pred['cp'] = pred[1]
  uid_real_pred['lp'] = pred[2]
  return precision(uid_real_pred.values)
 def search(uid_data,target,args):
  args = list(args)
  target_index = ['forward_count','comment_count','like_count'].index(target)
  target_min,target_median,target_max = args[3*target_index:3*target_index+3]
  del args[3*target_index:3*target_index+3]
  pred = (args[1],args[4])

  best_num = [target_median]
  best_pred = list(pred)
  best_pred.insert(target_index,target_median)
  best_score = score(uid_data,best_pred)
  for num in range(target_min,target_max+1):
   this_pred = list(pred)
   this_pred.insert(target_index,num)
   this_score = score(uid_data,this_pred)
   if this_score >= best_score:
    if this_score > best_score:
     best_num = [num]
     best_score = this_score
    else:
```

```
    else:
     best_num.append(num)

  return best_num[np.array([abs(i - target_median) for i in best_num]).argmin()]
 uid_best_pred = {}
 pool = _ProcessPool()
 uids,f,c,l = [],[],[],[]
 m=1
 for uid in stat_dic:
  print ("search uid:{}".format(uid),m)
  m=m+1
  uid_data = traindata[traindata.u_id == uid]
  arguments = stat_dic[uid][['forward_min','forward_median','forward_max','comment_mi
n',\
      'comment_median','comment_max','like_min','like_median','like_max']]
  arguments = tuple([int(i) for i in arguments])
  f.append(pool.apply_async(search,args=(uid_data,'forward_count',arguments)))
  c.append(pool.apply_async(search,args=(uid_data,'comment_count',arguments)))
  l.append(pool.apply_async(search,args=(uid_data,'like_count',arguments)))
  uids.append(uid)
 pool.close()
 pool.join()
 f = [i.get() for i in f]
 c = [i.get() for i in c]
 l = [i.get() for i in l]
 for i in range(len(uids)):
  uid_best_pred[uids[i]] = [f[i],c[i],l[i]]
 #cPickle.dump(uid_best_pred,open('uid_best_pred'+str(file)+'.pkl','ab'))
 label = ['forward_count','comment_count','like_count']
 pd.DataFrame.from_dict(data=uid_best_pred,orient='index').to_csv("G:\\Anconda Prog\\
BestPred\\weibo_uidbest"+str(file)+".csv",header=label)
 print("Written to file")
```

In [ ]:

```
uid_stat=pd.read_csv("train_uid_stat.csv")
uid_stat=uid_stat.set_index('u_id')
uid = splitDataFrameIntoSmaller(uid_stat, chunkSize = 100)
n=8
while n<75:
    df=uid[n].T
    stat=df.to_dict('series')
    n=n+1
    search_all_uid(stat,n)
```

```
search uid:059e69f515a8ccae9005d9184082e1a7 1
search uid:059f1af2f4d0cc0c11f7c2333fb56df9 2
search uid:05a17e58a7d0318c02ca957ea287c63a 3
search uid:05a3e00d1bf81123eaa16d9140781814 4
search uid:05a86ee2ef6ca329c2d7bd29a0bd43a2 5
search uid:05aa7401f543aff4eaa9804faf94fe5d 6
search uid:05abd155225287be6ccc9e851743c33d 7
search uid:05ac93f1cea114840ec678882df58bde 8
```

search uid:05ac976dd6c437b557aacc0f8bf95820 9
search uid:05ad91adac4f36090687b821074a6839 10
search uid:05b00857a652495ecd61ff287eefa0fa 11
search uid:05b0ed6b6c5a3c7ec0ee133658afc455 12
search uid:05ba03da99f9a8fd86ce5cedabb74eaa 13
search uid:05ba689e3f3d89f4f1de80b156f09c51 14
search uid:05bab1fcb0bef33fd6ab88f718d5d41d 15
search uid:05bc20ee20f50b744c00d948da2ee82f 16
search uid:05bc29c0517afb7cb63379e447646db2 17
search uid:05bc52673524ae6b2342f8c00e815aa8 18
search uid:05bd6785b6c3ca20728be79ed7e2fd73 19
search uid:05be804ef16a4d1b3b442cc1668c15ca 20
search uid:05bf76486b0c1eac3fab100d88678514 21
search uid:05c1a7c2eecd8568015fcb7245aba5d8 22
search uid:05c324e0a53b7b8b548163168e5c1763 23
search uid:05ca5e5ab3bc016c4056bebfd971af90 24
search uid:05cb634bc84a59d89c3747b9684cee56 25
search uid:05cc9da72ceb5ad3e8fcbd5e3178f70c 26
search uid:05ce6400d4ecf6b8c89cc281689da137 27
search uid:05cfb9c7126bf9aa546686673c01eaee 28
search uid:05d081c07b61499dd9647ebf883472e9 29
search uid:05d219d1ce32acb7591dca2ca181bb51 30
search uid:05d2b10375eaa12878f398267c2bedf9 31
search uid:05d43f272ed3dfce12a22a4d4c509fcc 32
search uid:05d79a718397917cfbe233968f069345 33
search uid:05dc5aac901219181ccbd75f50afd0b4 34
search uid:05df0f5c8252ae78d8bfd2e62290a7fa 35
search uid:05e101976ec8f0d28ae4f25f5f5a4df4 36
search uid:05e12cb9f7db55e84bf89006906e0f60 37
search uid:05e18af6ed9b5a23b1642bf118e740d0 38
search uid:05e4492aeefc260d3c8d0fbd70523c48 39
search uid:05e60e17f6f7d473d2b0fc6c471f5af9 40
search uid:05eab8ab56fc145464ae489d5e5ca9c1 41
search uid:05ed33dbc866e3ce5d597e650bb99d65 42
search uid:05edcddbfd2fcc147f7fb1bc09d3c0d9 43
search uid:05ef9ec5b807b59c6788626408c44754 44
search uid:05f0eee4768432862a7661085ed6533d 45
search uid:05f57a02ea3698e84d45e10383e44d63 46
search uid:05f974da06803245c215a2e13e059160 47
search uid:05fec84bb78c23377204a76411be1be7 48
search uid:05fff005c1d91f6ed477dbcd1c34111c 49
search uid:0604bdbdc3d0c12ee5ed75f300f99f73 50
search uid:0606698e2eb897452aa862bdac512726 51
search uid:06079f6c6929fe184ecc805679066bed 52
search uid:0609a0d54851bf22862d754ceda87b2b 53
search uid:060b9e5107e8f29baed68962f6f8eaff 54
search uid:060d776b251d79f434b75c69c4940462 55
search uid:060f5a4f7e058bb2ff9c9de92cfad542 56
search uid:061589a44fc0b6a40cf3e9807c510b6c 57
search uid:0615c6631b18572c9b6387eb90d85ab5 58
search uid:061a5a5ef25b09e39620d0408e7fec31 59
search uid:061b887b096b9d808a474bff00473fc1 60
search uid:061bd14b8a30dcc2231a6a610730a5b2 61

```
search uid:061bd14b8a30dca2331a6c619739a5b3 61
search uid:061c29516f8c179782a5546f47b47317 62
search uid:061d1cbc09f676d9be499136844ecc16 63
search uid:061dcf335c3c77727b8bdacb1eddc2f9 64
search uid:061eff81086b93a607360775a2497998 65
search uid:061f989b85ac194f5b4d845644d4310a 66
search uid:062087b718dcf91fc9ccdb3164778d46 67
search uid:0621633270b62ed5bd67cf932f3415f5 68
search uid:0621b7682d67938d9b63c745fe2fd401 69
search uid:0621eaff870b91d9c5177bb9a0534470 70
search uid:06229bb279bedc152ba845da65da941b 71
search uid:06240eed2b2c20c30c29c948d6ab5b73 72
search uid:0625890a7a1cdef6336cffadd84f9e29 73
search uid:0626b7ac922a88e8b6782ce9d3de2605 74
search uid:062731665285bdb5c8e9e0a11a71c85e 75
search uid:06275826c69fa270b3979f50368f4ecc 76
search uid:06276dceec3540eeef29cc488c2a4b45 77
search uid:06294a4f2333bedf4b33f9a0357cc4ce 78
search uid:0629ef303d35298e48fb02d4424ab909 79
search uid:062b899802ceff6a713702887d9d2e90 80
search uid:0630665026540a028878b4cc753a356a 81
search uid:0631e635b7a1eb8e3275ed767b2188cb 82
search uid:063306849f9adab61d662f67f90a5d23 83
search uid:063371083b2bb8cfaca892db2a703b2b 84
search uid:06348c1b6f53b5434b57b786d50a0104 85
search uid:0635ff2e769fc70d4f643dc639b4e9cc 86
search uid:06371151752b567f1ca4de664faf3250 87
search uid:06377961d878de50acea93b23219e7fe 88
search uid:063a41bd09fbc1790b43263cc6107d2f 89
search uid:063be5817b395ae8f439de921443179c 90
search uid:063e88517736cab464595cf98676ede3 91
search uid:063ee4a5343d064c2e49dd1cc9b45a6b 92
search uid:064275777ee18e51251f78abfbd4e1ce 93
search uid:0642a5738b33696fa17e69f3e46eaa96 94
search uid:06437d2b81aa54e843967bbc8684fb2c 95
search uid:064507c048c4185dcac8b4e9af81af5c 96
search uid:0645345d36c461a0d2b53278cd926c99 97
search uid:064978a78f30bf1637bcd794a6bc66b0 98
search uid:064a46a128786589314717 88f3eecff7 99
search uid:064b5927535578d5b54b5f533a490819 100
```

FILES GENERATED

In [92]:

```
@runTime
def predict_by_search(submission=True):
 traindata,testdata = loadData()
 #concat all frames here
```

```
 ub=pd.read_csv("train_best_pred.csv")
 ub=ub.set_index('u_id')
 df=ub.T
 uid_best_pred=df.to_dict('series')
 #uid_best_pred = search_all_uid()
 #print ("search done,now predict on traindata and testdata...")

 #predict traindata with uid's best fp,cp,lp
 forward,comment,like = [],[],[]
 for uid in traindata['u_id']:
  if uid in uid_best_pred:
   forward.append(int(uid_best_pred[uid][0]))
   comment.append(int(uid_best_pred[uid][1]))
   like.append(int(uid_best_pred[uid][2]))
  else:
   forward.append(0)
   comment.append(0)
   like.append(0)
 #score on the traindata
 train_real_pred = traindata[['forward_count','comment_count','like_count']]
 train_real_pred['fp'],train_real_pred['cp'],train_real_pred['lp'] = forward,comment,
like
 print ("Score on the training set:{0:.2f}%".format(precision(train_real_pred.values
)*100))
 if submission:
  test_pred = testdata[['u_id','m_id']]
  forward,comment,like = [],[],[]
  for uid in testdata['u_id']:
   if uid in uid_best_pred:
    forward.append(int(uid_best_pred[uid][0]))
    comment.append(int(uid_best_pred[uid][1]))
    like.append(int(uid_best_pred[uid][2]))
   else:
    forward.append(0)
    comment.append(0)
    like.append(0)
  test_pred['fp'],test_pred['cp'],test_pred['lp'] = forward,comment,like

  #generate submission file
  result = []
  filename = "weibo_predict_search.txt"
  for _,row in test_pred.iterrows():
   result.append("{0}\t{1}\t{2},{3},{4}\n".format(row[0],row[1],row[2],row[3],row[4])
)
  f = open(filename,'w')
  f.writelines(result)
  f.close()
  print ('generate submission file "{}"'.format(filename))
```

In [93]:

```
predict_by_search()
```

```
search uid:24b621c98f2594b698c0b1d60c9ae6db
search uid:d38e9bed5d98110dc2489d0d1cac3c2a
search uid:d80f3d3c5c1d658e82b837a4dd1af849
search uid:da534fe87e7a52777bee5c30573ed5fd
search uid:e06a22b7e065e559a1f0bf7841a85c51
search uid:e44d81d630e4f382f657e72aa4b685da
search uid:f349a67d1cd7c8683c5bbc5f8486e193
search uid:f9828598f9664d4e347ef2048ce17734
search uid:fa13974743d3fe6ff40d21b872325e9e
search uid:fbe6c953632e1b3dda66cf6118b6ab12
1
Before loop

search done,now predict on traindata and testdata...

C:\Users\DELL\Anaconda3\lib\site-packages\ipykernel_launcher.py:20: SettingWithCopyWa
rning:
A value is trying to be set on a copy of a slice from a DataFrame.
Try using .loc[row_indexer,col_indexer] = value instead

See the caveats in the documentation: http://pandas.pydata.org/pandas-docs/stable/ind
exing.html#indexing-view-versus-copy

Score on the training set:55.56%

C:\Users\DELL\Anaconda3\lib\site-packages\ipykernel_launcher.py:34: SettingWithCopyWa
rning:
A value is trying to be set on a copy of a slice from a DataFrame.
Try using .loc[row_indexer,col_indexer] = value instead

See the caveats in the documentation: http://pandas.pydata.org/pandas-docs/stable/ind
exing.html#indexing-view-versus-copy

generate submission file "weibo_predict_search.txt"
predict_by_search run time: 72.50s
```

In [1]:

```
import pandas as pd
import numpy as np
from sklearn import linear_model
from sklearn.model_selection import train_test_split
from sklearn.metrics import accuracy_score
from matplotlib import pyplot as plt
import statsmodels.api as sm
import import_ipynb
from evaluation import precision
```

```
importing Jupyter notebook from evaluation.ipynb
```

In [4]:

```
train_month1=pd.read_csv("E:\5th Sem\DMA Project\Model
Evaluation\weibo_train_feb_cpts.csv")
train_month2=pd.read_csv("E:\5th Sem\DMA Project\Model
Evaluation\weibo_train_march_cpts.csv")
train_month3=pd.read_csv("E:\5th Sem\DMA Project\Model
Evaluation\weibo_train_april_cpts.csv")
train_month4=pd.read_csv("E:\5th Sem\DMA Project\Model
Evaluation\weibo_train_may_cpts.csv")
train_month5=pd.read_csv("E:\5th Sem\DMA Project\Model
Evaluation\weibo_train_june_cpts.csv")
train_month6=pd.read_csv("E:\5th Sem\DMA Project\Model
Evaluation\weibo_train_july_cpts.csv")
```

In [4]:

```
frames1=[train_month1,train_month2,train_month3,train_month4,train_month5]
train=pd.concat(frames1)
predict=train_month6
```

In [5]:

```
X_train1=train[["forward_median","forward_mean","forward_min","content_media_count",
"content_emoji_count"]]
Y_train1=train[["forward_count"]]
X_test1=predict[["forward_median","forward_mean","forward_min","content_media_count",
"content_emoji_count"]]
Y_test1=predict[["forward_count"]]

X_train2=train[["comment_median","comment_mean","comment_min","content_media_count",
"content_emoji_count"]]
Y_train2=train[["comment_count"]]
X_test2=predict[["comment_median","comment_mean","comment_min","content_media_count",
"content_emoji_count"]]
Y_test2=predict[["comment_count"]]

X_train3=train[["like_median","like_mean","like_min","content_media_count","content_e
moji_count"]]
```

```
Y_train3=train[["like_count"]]
X_test3=predict[["like_median","like_mean","like_min","content_media_count","content_
emoji_count"]]
Y_test3=predict[["like_count"]]

pd.options.mode.use_inf_as_na = True
X_train1.fillna(X_train1.max(),inplace=True)
X_test1.fillna(X_test1.max(),inplace=True)
X_train2.fillna(X_train2.max(),inplace=True)
X_test2.fillna(X_test2.max(),inplace=True)
X_train3.fillna(X_train3.max(),inplace=True)
X_test3.fillna(X_test3.max(),inplace=True)

print(X_train1.shape,Y_train1.shape)
print(X_test1.shape,Y_test1.shape)

print(X_train2.shape,Y_train2.shape)
print(X_test2.shape,Y_test2.shape)

print(X_train3.shape,Y_train3.shape)
print(X_test3.shape,Y_test3.shape)
```

```
G:\Anaconda\lib\site-packages\pandas\core\generic.py:5430: SettingWithCopyWarning:
A value is trying to be set on a copy of a slice from a DataFrame

See the caveats in the documentation: http://pandas.pydata.org/pandas-docs/stable/ind
exing.html#indexing-view-versus-copy
  self._update_inplace(new_data)
```

```
(1044681, 5) (1044681, 1)
(184937, 5) (184937, 1)
(1044681, 5) (1044681, 1)
(184937, 5) (184937, 1)
(1044681, 5) (1044681, 1)
(184937, 5) (184937, 1)
```

In [6]:

```
model1=sm.OLS(Y_train1,X_train1).fit()
pred1=model1.predict(X_test1)
pred1=pred1.round()
pred1=(np.maximum(pred1,0.))
print(model1.summary())

model2=sm.OLS(Y_train2,X_train2).fit()
pred2=model2.predict(X_test2)
pred2=pred2.round()
pred2=(np.maximum(pred2,0.))
print(model2.summary())

model3=sm.OLS(Y_train3,X_train3).fit()
pred3=model3.predict(X_test3)
pred3=pred3.round()
```

```
pred3=(np.maximum(pred3,0.))
print(model3.summary())
```

```
                            OLS Regression Results
==============================================================================
Dep. Variable:          forward_count   R-squared:                       0.155
Model:                            OLS   Adj. R-squared:                  0.155
Method:                 Least Squares   F-statistic:                 3.837e+04
Date:                Thu, 08 Nov 2018   Prob (F-statistic):               0.00
Time:                        18:51:04   Log-Likelihood:            -6.0303e+06
No. Observations:             1044681   AIC:                         1.206e+07
Df Residuals:                 1044676   BIC:                         1.206e+07
Df Model:                           5
Covariance Type:            nonrobust
==============================================================================
=
                         coef    std err          t      P>|t|      [0.025      0.97
]
------------------------------------------------------------------------------
-
forward_median           0.4510      0.011     41.482      0.000       0.430
0.472
forward_mean             0.8472      0.008    104.955      0.000       0.831
0.863
forward_min             -0.2918      0.224     -1.301      0.193      -0.731       0.1
8
content_media_count      0.1469      0.089      1.651      0.099      -0.028       0.3
21
content_emoji_count     -0.0135      0.329     -0.041      0.967      -0.658       0.6
31
==============================================================================
Omnibus:                  5350284.603   Durbin-Watson:                   2.008
Prob(Omnibus):                  0.000   Jarque-Bera (JB):   394619723036624.188
Skew:                         234.725   Prob(JB):                         0.00
Kurtosis:                   95216.359   Cond. No.                         152.
==============================================================================

Warnings:
[1] Standard Errors assume that the covariance matrix of the errors is correctly spec
ified.
                            OLS Regression Results
==============================================================================
Dep. Variable:          comment_count   R-squared:                       0.141
Model:                            OLS   Adj. R-squared:                  0.141
Method:                 Least Squares   F-statistic:                 3.437e+04
Date:                Thu, 08 Nov 2018   Prob (F-statistic):               0.00
Time:                        18:51:05   Log-Likelihood:            -4.5502e+06
No. Observations:             1044681   AIC:                         9.100e+06
Df Residuals:                 1044676   BIC:                         9.101e+06
Df Model:                           5
Covariance Type:            nonrobust
==============================================================================
```

```
=
                            coef    std err          t      P>|t|      [0.025      0.97
]
------------------------------------------------------------------------------------
-
comment_median         -0.2502      0.017    -14.783      0.000      -0.283      -
0.217
comment_mean            1.1808      0.010    119.894      0.000       1.161
1.200
comment_min             0.0641      0.160      0.401      0.689      -0.249       0.3
8
content_media_count    -0.1673      0.022     -7.733      0.000      -0.210      -0.1
25
content_emoji_count     0.0507      0.080      0.635      0.525      -0.106       0.2
07
==============================================================================
Omnibus:                  5213202.872   Durbin-Watson:                     1.956
Prob(Omnibus):                  0.000   Jarque-Bera (JB):   199884674609233.969
Skew:                         214.327   Prob(JB):                          0.00
Kurtosis:                   67766.344   Cond. No.                          73.3
==============================================================================

Warnings:
[1] Standard Errors assume that the covariance matrix of the errors is correctly spec
ified.
                            OLS Regression Results
==============================================================================
Dep. Variable:             like_count   R-squared:                       0.370
Model:                            OLS   Adj. R-squared:                  0.370
Method:                 Least Squares   F-statistic:                 1.229e+05
Date:                Thu, 08 Nov 2018   Prob (F-statistic):               0.00
Time:                        18:51:06   Log-Likelihood:            -5.1616e+06
No. Observations:             1044681   AIC:                         1.032e+07
Df Residuals:                 1044676   BIC:                         1.032e+07
Df Model:                           5
Covariance Type:            nonrobust
==============================================================================
=
                            coef    std err          t      P>|t|      [0.025      0.97
]
------------------------------------------------------------------------------------
-
like_median             0.1165      0.014      8.225      0.000       0.089       0.1
4
like_mean               0.9511      0.012     78.935      0.000       0.927       0.9
5
like_min               -0.1539      0.252     -0.610      0.542      -0.649       0.3
1
content_media_count    -0.2240      0.039     -5.791      0.000      -0.300      -0.1
48
content_emoji_count    -0.0588      0.143     -0.411      0.681      -0.340       0.2
22
```

```
==============================================================================
Omnibus:                  4825971.004   Durbin-Watson:                   1.943
Prob(Omnibus):                  0.000   Jarque-Bera (JB):   88805152128512.828
Skew:                         163.871   Prob(JB):                         0.00
Kurtosis:                   45170.018   Cond. No.                         247.
==============================================================================

Warnings:
[1] Standard Errors assume that the covariance matrix of the errors is correctly spec
ified.
```

In [7]:

```
print(pred1[0:5])

print(pred2[0:5])

print(pred3[0:5])
```

```
0     3.0
1     3.0
2     3.0
3    13.0
4    12.0
dtype: float64
0    4.0
1    4.0
2    4.0
3    4.0
4    3.0
dtype: float64
0    3.0
1    3.0
2    3.0
3    6.0
4    4.0
dtype: float64
```

In [8]:

```
np.savetxt("G://DMA_PROJECT//weibo_predict_resulto1.csv",pred1,delimiter=',',header="
forward_count",comments="")
result1=pd.read_csv("G://DMA_PROJECT//weibo_predict_resulto1.csv")
np.savetxt("G://DMA_PROJECT//weibo_predict_resulto2.csv",pred2,delimiter=',',header="
comment_count",comments="")
result2=pd.read_csv("G://DMA_PROJECT//weibo_predict_resulto2.csv")
np.savetxt("G://DMA_PROJECT//weibo_predict_resulto3.csv",pred3,delimiter=',',header="
like_count",comments="")
result3=pd.read_csv("G://DMA_PROJECT//weibo_predict_resulto3.csv")
```

In [9]:

```
train_real_pred = pd.concat([Y_test1,Y_test2,Y_test3],axis=1)
```

```
train_real_pred['fp']=result1['forward_count']
train_real_pred['cp']=result2['comment_count']
train_real_pred['lp']=result3['like_count']
train_real_pred=train_real_pred.round()
print ("Score on the training set:{0:.2f}%".format(precision(train_real_pred.values)
*100))
```

```
Score on the training set:27.86%
```

In [1]:

```
import pandas as pd
import numpy as np
import re
from sklearn import linear_model
from sklearn.linear_model import Lasso
from sklearn.model_selection import train_test_split
from sklearn.metrics import mean_squared_error
from matplotlib import pyplot as plt
from textblob import TextBlob
import statsmodels.api as sm
```

In [2]:

```
import import_ipynb
from evaluation import precision
from runTime import runTime
```

```
importing Jupyter notebook from evaluation.ipynb
importing Jupyter notebook from runTime.ipynb
```

In [3]:

```
df1=pd.read_csv("E:\\5th-Sem\\DMA Project\\Project\\weibo_train1_cpts.csv")
df2=pd.read_csv("E:\\5th-Sem\\DMA Project\\Project\\weibo_train2_cpts.csv")
frames=[df1,df2]
train_all=pd.concat(frames)
```

In [5]:

```
train_all.shape[0]
```

Out[5]:

```
1229618
```

In [6]:

```
train=train_all[0:983694]
predict=train_all[983695:1229618]
```

In [8]:

```
train.columns
```

Out[8]:

```
Index(['u_id', 'm_id', 'forward_count', 'comment_count', 'like_count',
       'content', 'date', 'time', 'content_media_count', 'content_#_count',
       'content_@_count', 'content_?_count', 'content_!_count',
       'content_length', 'content_emoji_count', 'hour', 'min', 'sec',
       'forward_min', 'forward_max', 'forward_median', 'forward_mean',
       'comment_min', 'comment_max', 'comment_median', 'comment_mean',
       'like_min', 'like_max', 'like_median', 'like_mean'],
      dtype='object')
```

Model 1 - Linear Regression

In [14]:

```
X_train=train[["content_media_count","content_length","content_emoji_count","hour","min","sec","for
ward_median","comment_median","like_median"]]
Y_train=train[["forward_count","comment_count","like_count"]]
X_test=predict[["content_media_count","content_length","content_emoji_count","hour","min","sec","fo
rward_median","comment_median","like_median"]]
```

```
fward_median", "comment_median", "like_median"]]
Y_test=predict[["forward_count","comment_count","like_count"]]
print(X_train.shape,Y_train.shape)
print(X_test.shape,Y_test.shape)

pd.options.mode.use_inf_as_na = True
X_train.fillna(X_train.max(),inplace=True)
X_test.fillna(X_test.max(),inplace=True)

lm=linear_model.LinearRegression()
model=lm.fit(X_train,Y_train)
pred1=lm.predict(X_test)
```

```
(983694, 9) (983694, 3)
(245923, 9) (245923, 3)
```

```
C:\Users\DELL\Anaconda3\lib\site-packages\pandas\core\generic.py:5430: SettingWithCopyWarning:
A value is trying to be set on a copy of a slice from a DataFrame

See the caveats in the documentation: http://pandas.pydata.org/pandas-
docs/stable/indexing.html#indexing-view-versus-copy
  self._update_inplace(new_data)
```

In [15]:

```
temp = pd.DataFrame.from_records(pred1)
temp=temp.round()
temp=(np.maximum(temp,0))
train_real_pred=Y_test
train_real_pred['fp']=temp[0].values
train_real_pred['cp']=temp[1].values
train_real_pred['lp']=temp[2].values
print("Score:{0:.2f}%".format(precision(train_real_pred.values)*100))
```

```
C:\Users\DELL\Anaconda3\lib\site-packages\ipykernel_launcher.py:5: SettingWithCopyWarning:
A value is trying to be set on a copy of a slice from a DataFrame.
Try using .loc[row_indexer,col_indexer] = value instead

See the caveats in the documentation: http://pandas.pydata.org/pandas-
docs/stable/indexing.html#indexing-view-versus-copy
  """
C:\Users\DELL\Anaconda3\lib\site-packages\ipykernel_launcher.py:6: SettingWithCopyWarning:
A value is trying to be set on a copy of a slice from a DataFrame.
Try using .loc[row_indexer,col_indexer] = value instead

See the caveats in the documentation: http://pandas.pydata.org/pandas-
docs/stable/indexing.html#indexing-view-versus-copy

C:\Users\DELL\Anaconda3\lib\site-packages\ipykernel_launcher.py:7: SettingWithCopyWarning:
A value is trying to be set on a copy of a slice from a DataFrame.
Try using .loc[row_indexer,col_indexer] = value instead

See the caveats in the documentation: http://pandas.pydata.org/pandas-
docs/stable/indexing.html#indexing-view-versus-copy
  import sys
```

```
Score:22.83%
```

In [ ]:

```
from sklearn.model_selection import KFold
kf = KFold(n_splits=10)
kf.get_n_splits(X)
KFold(n_splits=10, random_state=None, shuffle=False)
for train_index, test_index in kf.split(X):
print("TRAIN:", train_index, "TEST:", test_index)
X_train, X_test = X[train_index], X[test_index]
y_train, y_test = y[train_index], y[test_index]
```

Model 2 - Random Forest

In [9]:

```
from sklearn.ensemble import RandomForestRegressor
```

In [111]:

```
## Spliting of training dataset into 70% training data and 30% testing data randomly
features_train=train[["content_media_count","content_#_count","content_length","content_emoji_count
","forward_median","comment_median","like_median"]]
features_test=predict[["content_media_count","content_#_count","content_length","content_emoji_cour
t","forward_median","comment_median","like_median"]]
labels_train=train[['forward_count', 'comment_count', 'like_count']]
labels_test=predict[['forward_count', 'comment_count', 'like_count']]

x = features_train
y = labels_train
x1 = features_test
y1 = labels_test



regr = RandomForestRegressor(max_depth=50, random_state=0,n_estimators=100)
regr.fit(x, y)
pred2 = regr.predict(x1)
```

In [112]:

```
temp = pd.DataFrame.from_records(pred2)
temp=temp.round()
temp=(np.maximum(temp,0))
temp=temp.abs()
temp=temp.astype(int)
train_real_pred=Y_test
train_real_pred['fp']=temp[0].values
train_real_pred['cp']=temp[1].values
train_real_pred['lp']=temp[2].values
print("Score:{0:.2f}%".format(precision(train_real_pred.values)*100))
```

```
C:\Users\DELL\Anaconda3\lib\site-packages\ipykernel_launcher.py:7: SettingWithCopyWarning:
A value is trying to be set on a copy of a slice from a DataFrame.
Try using .loc[row_indexer,col_indexer] = value instead

See the caveats in the documentation: http://pandas.pydata.org/pandas-
docs/stable/indexing.html#indexing-view-versus-copy
  import sys
C:\Users\DELL\Anaconda3\lib\site-packages\ipykernel_launcher.py:8: SettingWithCopyWarning:
A value is trying to be set on a copy of a slice from a DataFrame.
Try using .loc[row_indexer,col_indexer] = value instead

See the caveats in the documentation: http://pandas.pydata.org/pandas-
docs/stable/indexing.html#indexing-view-versus-copy

C:\Users\DELL\Anaconda3\lib\site-packages\ipykernel_launcher.py:9: SettingWithCopyWarning:
A value is trying to be set on a copy of a slice from a DataFrame.
Try using .loc[row_indexer,col_indexer] = value instead

See the caveats in the documentation: http://pandas.pydata.org/pandas-
docs/stable/indexing.html#indexing-view-versus-copy
  if __name__ == '__main__':
```

```
Score:30.01%
```

In [118]:

```
#train_real_pred
```

Model 3 - OLS

In [38]:

```
model3=sm.OLS(Y_train,X_train).fit()
pred3=model3.predict(X_test)
```

Model 4- Ridge

In [39]:

```
lm1=linear_model.Ridge(alpha=3)
model4=lm1.fit(X_train,Y_train)
pred4=lm1.predict(X_test)
```

Model 5- Lasso

In [40]:

```
lm1=Lasso(alpha=0.01)
model5=lm1.fit(X_train,Y_train)
pred5=lm1.predict(X_test)
```

Ensemble - Averaging

In [41]:

```
pred=(pred1+pred2+pred3+pred4+pred5)/5
```

In [46]:

```
pred=pred.round()
pred=(np.maximum(pred,0))
```

In [58]:

```
pred=pred.abs()
pred1=pred.astype(int)
```

In [61]:

```
pred1
```

Out[61]:

	0	1	2
368886	1	0	0
368887	1	0	0
368888	1	0	0
368889	1	0	0
368890	0	0	0
368891	106	22	91
368892	1	0	0
368893	1	2	2
368894	6	3	3
368895	1	0	0
368896	1	0	0
368897	1	0	0
368898	0	1	1

	0	1	2
368898	0	1	1
368899	2	1	1
368900	1	0	0
368901	1	2	2
368902	0	1	1
368903	1	0	0
368904	0	0	0
368905	1	0	0
368906	2	0	0
368907	4	3	3
368908	0	1	1
368909	0	1	1
368910	45	131	495
368911	0	1	1
368912	1	0	0
368913	4	2	3
368914	0	0	0
368915	0	0	0
...	...	...	...
614779	0	1	1
614780	2	0	0
614781	16	4	5
614782	3	0	0
614783	0	0	0
614784	0	0	0
614785	7	1	2
614786	0	1	1
614787	1	0	0
614788	1	0	0
614789	103	51	66
614790	3	0	1
614791	0	0	0
614792	1	0	0
614793	0	0	0
614794	1	0	0
614795	2	1	4
614796	6	3	3
614797	2	1	2
614798	0	1	0
614799	0	1	1
614800	2	0	0
614801	0	1	1
614802	0	1	1
614803	1	0	0
614804	1	0	0
614805	0	1	1

	0	1	2
614806	0	1	1
614807	1	0	0
614808	2	0	0

245923 rows × 3 columns

In [59]:

```
#np.savetxt("E://DMA_PRED//result_Ensemble.csv",pred1,delimiter=',',header="forward_count,comment_
,like_count",comments="")
#result1=pd.read_csv("E://DMA_PRED//result_Ensemble.csv")
```

In [63]:

```
#result1
```

In [85]:

```
train_real_pred=Y_test
train_real_pred['fp']=pred1[0]
train_real_pred['cp']=pred1[1]
train_real_pred['lp']=pred1[2]
print("Score:{0:.2f}%".format(precision(train_real_pred.values)*100))
```

```
C:\Users\DELL\Anaconda3\lib\site-packages\ipykernel_launcher.py:2: SettingWithCopyWarning:
A value is trying to be set on a copy of a slice from a DataFrame.
Try using .loc[row_indexer,col_indexer] = value instead

See the caveats in the documentation: http://pandas.pydata.org/pandas-
docs/stable/indexing.html#indexing-view-versus-copy

C:\Users\DELL\Anaconda3\lib\site-packages\ipykernel_launcher.py:3: SettingWithCopyWarning:
A value is trying to be set on a copy of a slice from a DataFrame.
Try using .loc[row_indexer,col_indexer] = value instead

See the caveats in the documentation: http://pandas.pydata.org/pandas-
docs/stable/indexing.html#indexing-view-versus-copy
  This is separate from the ipykernel package so we can avoid doing imports until
C:\Users\DELL\Anaconda3\lib\site-packages\ipykernel_launcher.py:4: SettingWithCopyWarning:
A value is trying to be set on a copy of a slice from a DataFrame.
Try using .loc[row_indexer,col_indexer] = value instead

See the caveats in the documentation: http://pandas.pydata.org/pandas-
docs/stable/indexing.html#indexing-view-versus-copy
  after removing the cwd from sys.path.
```

```
Score:24.22%
```

In [113]:

```
#train_real_pred
```

XG Boost

In [116]:

```
x = features_train
y = labels_train
x1 = features_test
y1 = labels_test
import xgboost as xgb
```

```
C:\Users\DELL\Anaconda3\lib\site-packages\sklearn\cross_validation.py:41: DeprecationWarning: This
module was deprecated in version 0.18 in favor of the model_selection module into which all the re
factored classes and functions are moved. Also note that the interface of the new CV iterators are
different from that of this module. This module will be removed in 0.20.
```

```
  "This module will be removed in 0.20.", DeprecationWarning)
```

In [119]:

```
#T_train_xgb = xgb.DMatrix(x, y)
#params = {"objective": "reg:linear", "booster":"gblinear"}
#gbm = xgb.train(dtrain=T_train_xgb,params=params)
#Y_pred = gbm.predict(xgb.DMatrix(x1))
#print(r2_score(y1, Y_pred) )  #xgboost
```

Mapping Uid

In [140]:

```
unique_id=train_all['u_id'].unique().tolist()
```

In [141]:

```
uid_df = pd.DataFrame({'u_id':unique_id})
```

In [143]:

```
uid_df.shape[0]
```

Out[143]:

```
37263
```

In [160]:

```
from sklearn import preprocessing
le = preprocessing.LabelEncoder()
le.fit(unique_id)
#list(le.classes_)
```

Out[160]:

```
LabelEncoder()
```

In [161]:

```
l=[]
l=le.transform(unique_id)
df = pd.DataFrame({'u_id':l})
uid_df['id']=df['u_id']
```

In [163]:

```
train_all=train_all.set_index('u_id').join(uid_df.set_index('u_id'))
```

In [168]:

```
train=train_all[0:983694]
predict=train_all[983695:1229618]
```

In [169]:

```
X_train=train[["id","content_media_count","content_length","content_emoji_count","hour","min","sec"
,"forward_median","comment_median","like_median"]]
Y_train=train[["forward_count","comment_count","like_count"]]
X_test=predict[["id","content_media_count","content_length","content_emoji_count","hour","min","sec
","fo
                rward_median","comment_median","like_median"]]
Y_test=predict[["forward_count","comment_count","like_count"]]
print(X_train.shape,Y_train.shape)
print(X_test.shape,Y_test.shape)
```

```
print(X_test.shape,Y_test.shape)

pd.options.mode.use_inf_as_na = True
X_train.fillna(X_train.max(),inplace=True)
X_test.fillna(X_test.max(),inplace=True)

lm=linear_model.LinearRegression()
model=lm.fit(X_train,Y_train)
pred1=lm.predict(X_test)
```

```
(983694, 10) (983694, 3)
(245923, 10) (245923, 3)
```

```
C:\Users\DELL\Anaconda3\lib\site-packages\pandas\core\generic.py:5430: SettingWithCopyWarning:
A value is trying to be set on a copy of a slice from a DataFrame

See the caveats in the documentation: http://pandas.pydata.org/pandas-
docs/stable/indexing.html#indexing-view-versus-copy
  self._update_inplace(new_data)
```

In [170]:

```
temp = pd.DataFrame.from_records(pred1)
temp=temp.round()
temp=(np.maximum(temp,0))
temp=temp.abs()
temp=temp.astype(int)
train_real_pred=Y_test
train_real_pred['fp']=temp[0].values
train_real_pred['cp']=temp[1].values
train_real_pred['lp']=temp[2].values
print("Score:{0:.2f}%".format(precision(train_real_pred.values)*100))
```

```
C:\Users\DELL\Anaconda3\lib\site-packages\ipykernel_launcher.py:7: SettingWithCopyWarning:
A value is trying to be set on a copy of a slice from a DataFrame.
Try using .loc[row_indexer,col_indexer] = value instead

See the caveats in the documentation: http://pandas.pydata.org/pandas-
docs/stable/indexing.html#indexing-view-versus-copy
  import sys
C:\Users\DELL\Anaconda3\lib\site-packages\ipykernel_launcher.py:8: SettingWithCopyWarning:
A value is trying to be set on a copy of a slice from a DataFrame.
Try using .loc[row_indexer,col_indexer] = value instead

See the caveats in the documentation: http://pandas.pydata.org/pandas-
docs/stable/indexing.html#indexing-view-versus-copy

C:\Users\DELL\Anaconda3\lib\site-packages\ipykernel_launcher.py:9: SettingWithCopyWarning:
A value is trying to be set on a copy of a slice from a DataFrame.
Try using .loc[row_indexer,col_indexer] = value instead

See the caveats in the documentation: http://pandas.pydata.org/pandas-
docs/stable/indexing.html#indexing-view-versus-copy
  if __name__ == '__main__':
```

```
Score:25.69%
```

In [173]:

```
## Spliting of training dataset into 70% training data and 30% testing data randomly
features_train=train[["id","content_media_count","content_#_count","content_length","content_emoji_
count","forward_median","comment_median","like_median"]]
features_test=predict[["id","content_media_count","content_#_count","content_length","content_emoji
_count","forward_median","comment_median","like_median"]]
labels_train=train[['forward_count', 'comment_count', 'like_count']]
labels_test=predict[['forward_count', 'comment_count', 'like_count']]

x = features_train
y = labels_train
x1 = features_test
y1 = labels_test
```

```
regr = RandomForestRegressor(max_depth=50, random_state=0,n_estimators=100)
regr.fit(x, y)
pred2 = regr.predict(x1)
```

In [174]:

```
temp = pd.DataFrame.from_records(pred2)
temp=temp.round()
temp=(np.maximum(temp,0))
temp=temp.abs()
temp=temp.astype(int)
train_real_pred=Y_test
train_real_pred['fp']=temp[0].values
train_real_pred['cp']=temp[1].values
train_real_pred['lp']=temp[2].values
print("Score:{0:.2f}%".format(precision(train_real_pred.values)*100))
```

```
C:\Users\DELL\Anaconda3\lib\site-packages\ipykernel_launcher.py:7: SettingWithCopyWarning:
A value is trying to be set on a copy of a slice from a DataFrame.
Try using .loc[row_indexer,col_indexer] = value instead

See the caveats in the documentation: http://pandas.pydata.org/pandas-
docs/stable/indexing.html#indexing-view-versus-copy
  import sys
C:\Users\DELL\Anaconda3\lib\site-packages\ipykernel_launcher.py:8: SettingWithCopyWarning:
A value is trying to be set on a copy of a slice from a DataFrame.
Try using .loc[row_indexer,col_indexer] = value instead

See the caveats in the documentation: http://pandas.pydata.org/pandas-
docs/stable/indexing.html#indexing-view-versus-copy

C:\Users\DELL\Anaconda3\lib\site-packages\ipykernel_launcher.py:9: SettingWithCopyWarning:
A value is trying to be set on a copy of a slice from a DataFrame.
Try using .loc[row_indexer,col_indexer] = value instead

See the caveats in the documentation: http://pandas.pydata.org/pandas-
docs/stable/indexing.html#indexing-view-versus-copy
  if __name__ == '__main__':
```

```
Score:20.74%
```

In [4]:

```
import pandas as pd
import numpy as np
import re
from sklearn import linear_model
from sklearn.linear_model import Lasso
from sklearn.model_selection import train_test_split
from sklearn.metrics import mean_squared_error
from matplotlib import pyplot as plt
from textblob import TextBlob
import statsmodels.api as sm
```

In [5]:

```
import import_ipynb
from evaluation import precision
from runTime import runTime
```

```
importing Jupyter notebook from evaluation.ipynb
importing Jupyter notebook from runTime.ipynb
```

----------Polarity as a factor----------

Sentences with positive meaning have positive polarity and negative meaning ones have negative polarity and neutral ones have zero as polarity

In [7]:

```
dfpol=pd.read_csv("C:/Users/user/Downloads/weibo_polarity.csv")
```

In [8]:

```
dfpol.head(10)
```

Out[8]:

	Unnamed: 0	u_id	m_id	forward_count	comment_count	l
0	0	d38e9bed5d98110dc2489d0d1cac3c2a	7d45833d9865727a88b960b0603c19f6	0	0	(
1	1	d38e9bed5d98110dc2489d0d1cac3c2a	00755196c77936bf44656ada98291c59	0	0	(
2	2	d38e9bed5d98110dc2489d0d1cac3c2a	4fedf3888b1e16592f0e0bdc8b393845	0	0	(
3	3	d38e9bed5d98110dc2489d0d1cac3c2a	91be0b8612265aae32725cd4fa80b222	0	0	(

	Unnamed: 0	u_id	m_id	forward_count	comment_count	l
4	4	d38e9bed5d98110dc2489d0d1cac3c2a	bd2af99ecf1298f5539f0ddfcdd3ed64	0	0	0
5	5	d38e9bed5d98110dc2489d0d1cac3c2a	182078c5a409834f2128b3c9c2c289c3	0	0	0
6	6	d38e9bed5d98110dc2489d0d1cac3c2a	2c9697e5d6f1d9d479540173c4c374cb	0	0	0
7	7	d38e9bed5d98110dc2489d0d1cac3c2a	0ce5d103d7712b398ee2e81f83f49751	0	0	0
8	8	d38e9bed5d98110dc2489d0d1cac3c2a	a651facd0523d2a85a0717b83928c6c8	0	0	0
9	9	d38e9bed5d98110dc2489d0d1cac3c2a	3e1895f6017e0214f7392013552ac96a	0	0	0

10 rows × 39 columns

In [9]:

```
dfpol.columns
```

Out[9]:

```
Index(['Unnamed: 0', 'u_id', 'm_id', 'forward_count', 'comment_count',
       'like_count', 'content', 'date', 'time', 'content_media_count',
       'content_#_count', 'content_@_count', 'content_?_count',
       'content_!_count', 'content_length', 'content_emoji_count', 'hour',
       'min', 'sec', 'forward_min', 'forward_max', 'forward_median',
       'forward_mean', 'comment_min', 'comment_max', 'comment_median',
       'comment_mean', 'like_min', 'like_max', 'like_median', 'like_mean',
       'Unnamed: 0.1', 'content_spchar', 'non_emoji_content', 'en_content',
       'Unnamed: 1', 'url_rem', 'contentwurl', 'polarity'],
      dtype='object')
```

In [10]:

```
dfpol['date']=pd.to_datetime(dfpol['date'],errors='coerce')
train_month=[g for n, g in dfpol.groupby(pd.Grouper(key='date',freq='M'))]
```

In [11]:

```
train_month[0]=pd.read_csv("C:/Users/user/Downloads/weibo_train_feb_cpts10000.csv")
train_month[1]=pd.read_csv("C:/Users/user/Downloads/weibo_train_march_cpts10000.csv")
train_month[2]=pd.read_csv("C:/Users/user/Downloads/weibo_train_april_cpts10000.csv")
train_month[3]=pd.read_csv("C:/Users/user/Downloads/weibo_train_may_cpts10000.csv")
train_month[4]=pd.read_csv("C:/Users/user/Downloads/weibo_train_june_cpts10000.csv")
train_month[5]=pd.read_csv("C:/Users/user/Downloads/weibo_train_july_cpts10000.csv")
```

In [12]:

```
frames1=[train_month[0],train_month[1],train_month[2],train_month[3],train_month[4]]
train=pd.concat(frames1)
predict=train_month[5]
```

Model 7: (Factors: Media, Length, Emoji, Median,Polarity)

In [9]:

```
X_train1=train[["content_media_count","content_length","forward_median","comment_median","like_medi
an","polarity"]]
Y_train1=train[["forward_count","comment_count","like_count"]]
X_test1=predict[["content_media_count","content_length","forward_median","comment_median","like_med
ian","polarity"]]
Y_test1=predict[["forward_count","comment_count","like_count"]]

pd.options.mode.use_inf_as_na = True
X_train1.fillna(X_train1.max(),inplace=True)
X_test1.fillna(X_test1.max(),inplace=True)
```

```
C:\Users\user\Anaconda3\lib\site-packages\pandas\core\generic.py:5430: SettingWithCopyWarning:
A value is trying to be set on a copy of a slice from a DataFrame

See the caveats in the documentation: http://pandas.pydata.org/pandas-
docs/stable/indexing.html#indexing-view-versus-copy
  self._update_inplace(new_data)
```

In [10]:

```
lm1=linear_model.LinearRegression()
model1=lm1.fit(X_train1,Y_train1)
pred1=lm1.predict(X_test1)
pred1=pred1.round()
pred1=(np.maximum(pred1,0.))
```

In [11]:

```
print(model1.coef_)
print(model1.intercept_)
```

```
[[ -3.60345081e-01   6.74510609e-03  -9.03871388e+00  -5.69877934e+00
    1.46618263e+01  -3.00702607e-01]
 [ -4.52559256e-01   2.88854313e-04  -2.58935392e+00   3.20973326e-01
    2.56664104e+00   1.78063868e-01]
 [ -1.70607807e-01  -1.73336140e-03  -2.49319057e+00  -7.66383744e-01
    3.84877692e+00   1.34944972e-01]]
[ 0.09961692  0.27754213  0.23556883]
```

In [12]:

```
np.savetxt("C:/Users/user/Downloads/weibo_predict_result51.csv",pred1,delimiter=',',header="forward
_count,comment_count,like_count",comments="")
result1=pd.read_csv("C:/Users/user/Downloads/weibo_predict_result51.csv")
```

In [13]:

```
print(mean_squared_error(Y_test1,result1))
```

```
21.2745912995
```

In [14]:

```
train_real_pred=Y_test1
train_real_pred['fp']=result1['forward_count']
train_real_pred['cp']=result1['comment_count']
train_real_pred['lp']=result1['like_count']
```

```
print("Score:{0:.2f}%".format(precision(train_real_pred.values)*100))
```

```
C:\Users\user\Anaconda3\lib\site-packages\ipykernel_launcher.py:2: SettingWithCopyWarning:
A value is trying to be set on a copy of a slice from a DataFrame.
Try using .loc[row_indexer,col_indexer] = value instead

See the caveats in the documentation: http://pandas.pydata.org/pandas-
docs/stable/indexing.html#indexing-view-versus-copy

C:\Users\user\Anaconda3\lib\site-packages\ipykernel_launcher.py:3: SettingWithCopyWarning:
A value is trying to be set on a copy of a slice from a DataFrame.
Try using .loc[row_indexer,col_indexer] = value instead

See the caveats in the documentation: http://pandas.pydata.org/pandas-
docs/stable/indexing.html#indexing-view-versus-copy
  This is separate from the ipykernel package so we can avoid doing imports until
C:\Users\user\Anaconda3\lib\site-packages\ipykernel_launcher.py:4: SettingWithCopyWarning:
A value is trying to be set on a copy of a slice from a DataFrame.
Try using .loc[row_indexer,col_indexer] = value instead

See the caveats in the documentation: http://pandas.pydata.org/pandas-
docs/stable/indexing.html#indexing-view-versus-copy
  after removing the cwd from sys.path.
```

```
Score:35.39%
```

Analysis: Result with Polarity as factor are satisfactory considering the data used for train. This might prove to be a good factor for whole dataset prediction.

Model 8: (Factors: Media, Length,Median,Polarity)

In [15]:

```
X_train1=train[["content_media_count","content_length","forward_mean","comment_mean","like_mean","p
olarity"]]
Y_train1=train[["forward_count","comment_count","like_count"]]
X_test1=predict[["content_media_count","content_length","forward_mean","comment_mean","like_mean",
"polarity"]]
Y_test1=predict[["forward_count","comment_count","like_count"]]

pd.options.mode.use_inf_as_na = True
X_train1.fillna(X_train1.max(),inplace=True)
X_test1.fillna(X_test1.max(),inplace=True)
```

```
C:\Users\user\Anaconda3\lib\site-packages\pandas\core\generic.py:5430: SettingWithCopyWarning:
A value is trying to be set on a copy of a slice from a DataFrame

See the caveats in the documentation: http://pandas.pydata.org/pandas-
docs/stable/indexing.html#indexing-view-versus-copy
  self._update_inplace(new_data)
```

In [16]:

```
lm1=linear_model.LinearRegression()
model1=lm1.fit(X_train1,Y_train1)
pred1=lm1.predict(X_test1)
pred1=pred1.round()
pred1=(np.maximum(pred1,0.))
```

In [17]:

```
print(model1.coef_)
print(model1.intercept_)
```

```
[[  1.84153944e-01   1.16499672e-02   1.23769730e+00   5.03985664e-01
   -5.30125012e-01   2.40824642e-01]
 [ -4.34205066e-01   2.65588900e-03   4.65693627e-02   8.21537197e-01
```

```
   -4.63148333e-03   8.38895495e-02]
 [ -1.07997853e-01  -6.08462330e-04   1.24279057e-01  -2.25635181e-02
    7.47271891e-01   1.46482193e-01]]
[-1.685958    0.04881615  0.00796132]
```

In [18]:

```
np.savetxt("C:/Users/user/Downloads/weibo_predict_result51.csv",pred1,delimiter=',',header="forward
_count,comment_count,like_count",comments="")
result1=pd.read_csv("C:/Users/user/Downloads/weibo_predict_result51.csv")
```

In [19]:

```
print(mean_squared_error(Y_test1,result1))
```

```
21.0421169299
```

In [20]:

```
train_real_pred=Y_test1
train_real_pred['fp']=result1['forward_count']
train_real_pred['cp']=result1['comment_count']
train_real_pred['lp']=result1['like_count']
print("Score:{0:.2f}%".format(precision(train_real_pred.values)*100))
```

```
Score:34.58%
```

```
C:\Users\user\Anaconda3\lib\site-packages\ipykernel_launcher.py:2: SettingWithCopyWarning:
A value is trying to be set on a copy of a slice from a DataFrame.
Try using .loc[row_indexer,col_indexer] = value instead

See the caveats in the documentation: http://pandas.pydata.org/pandas-
docs/stable/indexing.html#indexing-view-versus-copy

C:\Users\user\Anaconda3\lib\site-packages\ipykernel_launcher.py:3: SettingWithCopyWarning:
A value is trying to be set on a copy of a slice from a DataFrame.
Try using .loc[row_indexer,col_indexer] = value instead

See the caveats in the documentation: http://pandas.pydata.org/pandas-
docs/stable/indexing.html#indexing-view-versus-copy
  This is separate from the ipykernel package so we can avoid doing imports until
C:\Users\user\Anaconda3\lib\site-packages\ipykernel_launcher.py:4: SettingWithCopyWarning:
A value is trying to be set on a copy of a slice from a DataFrame.
Try using .loc[row_indexer,col_indexer] = value instead

See the caveats in the documentation: http://pandas.pydata.org/pandas-
docs/stable/indexing.html#indexing-view-versus-copy
  after removing the cwd from sys.path.
```

Model 9: (Factors: Media, Length, Emoji, Median,Mean,Polarity)

In [28]:

```
X_train1=train[["content_media_count","content_length","forward_mean","forward_median","comment_mea
n","comment_median","like_mean","like_median","polarity"]]
Y_train1=train[["forward_count","comment_count","like_count"]]
X_test1=predict[["content_media_count","content_length","forward_mean","forward_median","comment_me
an","comment_median","like_mean","like_median","polarity"]]
Y_test1=predict[["forward_count","comment_count","like_count"]]

pd.options.mode.use_inf_as_na = True
X_train1.fillna(X_train1.max(),inplace=True)
X_test1.fillna(X_test1.max(),inplace=True)
```

```
C:\Users\user\Anaconda3\lib\site-packages\pandas\core\generic.py:5430: SettingWithCopyWarning:
A value is trying to be set on a copy of a slice from a DataFrame

See the caveats in the documentation: http://pandas.pydata.org/pandas-
```

```
docs/stable/indexing.html#indexing-view-versus-copy
  self._update_inplace(new_data)
```

In [29]:

```
lm1=linear_model.LinearRegression()
model1=lm1.fit(X_train1,Y_train1)
pred1=lm1.predict(X_test1)
pred1=pred1.round()
pred1=(np.maximum(pred1,0.))
```

In [30]:

```
print(model1.coef_)
print(model1.intercept_)
```

```
[[  4.24002976e-01   1.39830856e-02   1.77218462e+00   5.28953775e+00
    5.32073405e-01   2.72797599e+00   9.08606236e-02  -8.60889360e+00
    4.57317719e-01]
 [ -3.94970258e-01   2.85864515e-03   1.34655712e-01   5.14556038e-01
    7.61232739e-01   4.96108919e-01   2.61640134e-02  -1.03251918e+00
    1.30303697e-01]
 [ -2.52656284e-01  -1.69108114e-03  -1.95657316e-01  -2.51251894e+00
    9.14012006e-02  -1.71242671e+00   4.71094851e-01   4.48754073e+00
   -7.29299942e-04]]
[-2.47378947 -0.06752787  0.45657126]
```

In [31]:

```
np.savetxt("C:/Users/user/Downloads/weibo_predict_result51.csv",pred1,delimiter=',',header="forward
_count,comment_count,like_count",comments="")
result1=pd.read_csv("C:/Users/user/Downloads/weibo_predict_result51.csv")
```

In [32]:

```
print(mean_squared_error(Y_test1,result1))
```

```
20.9952895539
```

In [33]:

```
train_real_pred=Y_test1
train_real_pred['fp']=result1['forward_count']
train_real_pred['cp']=result1['comment_count']
train_real_pred['lp']=result1['like_count']
print("Score:{0:.2f}%".format(precision(train_real_pred.values)*100))
```

```
Score:34.22%
```

```
C:\Users\user\Anaconda3\lib\site-packages\ipykernel_launcher.py:2: SettingWithCopyWarning:
A value is trying to be set on a copy of a slice from a DataFrame.
Try using .loc[row_indexer,col_indexer] = value instead

See the caveats in the documentation: http://pandas.pydata.org/pandas-
docs/stable/indexing.html#indexing-view-versus-copy

C:\Users\user\Anaconda3\lib\site-packages\ipykernel_launcher.py:3: SettingWithCopyWarning:
A value is trying to be set on a copy of a slice from a DataFrame.
Try using .loc[row_indexer,col_indexer] = value instead

See the caveats in the documentation: http://pandas.pydata.org/pandas-
docs/stable/indexing.html#indexing-view-versus-copy
  This is separate from the ipykernel package so we can avoid doing imports until
C:\Users\user\Anaconda3\lib\site-packages\ipykernel_launcher.py:4: SettingWithCopyWarning:
A value is trying to be set on a copy of a slice from a DataFrame.
Try using .loc[row_indexer,col_indexer] = value instead

See the caveats in the documentation: http://pandas.pydata.org/pandas-
docs/stable/indexing.html#indexing-view-versus-copy
  after removing the cwd from sys.path
```

```
  after removing the cwd from sys.path.
```

Model10: (Factors: Media, Length, Emoji, Median,Polarity) with OLS

OLS is a type of linear least sqaures methods for estimating parameters in a linear regression model

In [13]:

```
X_train1=train[["content_media_count","content_length","forward_median","comment_median","like_medi
an","polarity"]]
Y_train1=train[["forward_count","comment_count","like_count"]]
X_test1=predict[["content_media_count","content_length","forward_median","comment_median","like_med
ian","polarity"]]
Y_test1=predict[["forward_count","comment_count","like_count"]]

pd.options.mode.use_inf_as_na = True
X_train1.fillna(X_train1.max(),inplace=True)
X_test1.fillna(X_test1.max(),inplace=True)
```

```
C:\Users\user\Anaconda3\lib\site-packages\pandas\core\generic.py:5430: SettingWithCopyWarning:
A value is trying to be set on a copy of a slice from a DataFrame

See the caveats in the documentation: http://pandas.pydata.org/pandas-
docs/stable/indexing.html#indexing-view-versus-copy
  self._update_inplace(new_data)
```

In [15]:

```
model1=sm.OLS(Y_train1,X_train1).fit()
pred1=model1.predict(X_test1)
pred1=pred1.round()
pred1=(np.maximum(pred1,0.))
```

In [16]:

```
np.savetxt("C:/Users/user/Downloads/weibo_predict_result52.csv",pred1,delimiter=',',header="forward
_count,comment_count,like_count",comments="")
result1=pd.read_csv("C:/Users/user/Downloads/weibo_predict_result52.csv")
```

In [17]:

```
train_real_pred=Y_test1
train_real_pred['fp']=result1['forward_count']
train_real_pred['cp']=result1['comment_count']
train_real_pred['lp']=result1['like_count']
print("Score:{0:.2f}%".format(precision(train_real_pred.values)*100))
```

```
C:\Users\user\Anaconda3\lib\site-packages\ipykernel_launcher.py:2: SettingWithCopyWarning:
A value is trying to be set on a copy of a slice from a DataFrame.
Try using .loc[row_indexer,col_indexer] = value instead

See the caveats in the documentation: http://pandas.pydata.org/pandas-
docs/stable/indexing.html#indexing-view-versus-copy

C:\Users\user\Anaconda3\lib\site-packages\ipykernel_launcher.py:3: SettingWithCopyWarning:
A value is trying to be set on a copy of a slice from a DataFrame.
Try using .loc[row_indexer,col_indexer] = value instead

See the caveats in the documentation: http://pandas.pydata.org/pandas-
docs/stable/indexing.html#indexing-view-versus-copy
  This is separate from the ipykernel package so we can avoid doing imports until
C:\Users\user\Anaconda3\lib\site-packages\ipykernel_launcher.py:4: SettingWithCopyWarning:
A value is trying to be set on a copy of a slice from a DataFrame.
Try using .loc[row_indexer,col_indexer] = value instead

See the caveats in the documentation: http://pandas.pydata.org/pandas-
docs/stable/indexing.html#indexing-view-versus-copy
```

```
  after removing the cwd from sys.path.
```

```
Score:35.39%
```

Model11: (Factors: Media, Length, Emoji, Median,Polarity) with Ridge regression

Ridge regression is used to prevent multicollinearity among variables by shrinking the parameters

In [18]:

```
X_train1=train[["content_media_count","content_length","forward_median","comment_median","like_medi
an","polarity"]]
Y_train1=train[["forward_count","comment_count","like_count"]]
X_test1=predict[["content_media_count","content_length","forward_median","comment_median","like_med
ian","polarity"]]
Y_test1=predict[["forward_count","comment_count","like_count"]]

pd.options.mode.use_inf_as_na = True
X_train1.fillna(X_train1.max(),inplace=True)
X_test1.fillna(X_test1.max(),inplace=True)
```

```
C:\Users\user\Anaconda3\lib\site-packages\pandas\core\generic.py:5430: SettingWithCopyWarning:
A value is trying to be set on a copy of a slice from a DataFrame

See the caveats in the documentation: http://pandas.pydata.org/pandas-
docs/stable/indexing.html#indexing-view-versus-copy
  self._update_inplace(new_data)
```

In [19]:

```
lm1=linear_model.Ridge(alpha=3)
model1=lm1.fit(X_train1,Y_train1)
pred1=lm1.predict(X_test1)
pred1=pred1.round()
pred1=(np.maximum(pred1,0.))
```

In [20]:

```
print(model1.coef_)
print(model1.intercept_)
```

```
[[ -3.86510208e-01   6.41054941e-03  -8.34912342e+00  -5.36638332e+00
    1.39393510e+01  -2.63639792e-01]
 [ -4.56306722e-01   1.98158232e-04  -2.45402009e+00   3.77596491e-01
    2.43239453e+00   1.84454477e-01]
 [ -1.76866329e-01  -1.82770299e-03  -2.31640230e+00  -6.84262708e-01
    3.66606306e+00   1.43628492e-01]]
[ 0.12872141  0.28577674  0.2444117 ]
```

In [21]:

```
np.savetxt("C:/Users/user/Downloads/weibo_predict_result53.csv",pred1,delimiter=',',header="forward
_count,comment_count,like_count",comments="")
result1=pd.read_csv("C:/Users/user/Downloads/weibo_predict_result53.csv")
```

In [22]:

```
train_real_pred=Y_test1
train_real_pred['fp']=result1['forward_count']
train_real_pred['cp']=result1['comment_count']
train_real_pred['lp']=result1['like_count']
print("Score:{0:.2f}%".format(precision(train_real_pred.values)*100))
```

```
Score:35.48%
```

```
C:\Users\user\Anaconda3\lib\site-packages\ipykernel_launcher.py:2: SettingWithCopyWarning:
A value is trying to be set on a copy of a slice from a DataFrame.
Try using .loc[row_indexer,col_indexer] = value instead

See the caveats in the documentation: http://pandas.pydata.org/pandas-
docs/stable/indexing.html#indexing-view-versus-copy

C:\Users\user\Anaconda3\lib\site-packages\ipykernel_launcher.py:3: SettingWithCopyWarning:
A value is trying to be set on a copy of a slice from a DataFrame.
Try using .loc[row_indexer,col_indexer] = value instead

See the caveats in the documentation: http://pandas.pydata.org/pandas-
docs/stable/indexing.html#indexing-view-versus-copy
  This is separate from the ipykernel package so we can avoid doing imports until
C:\Users\user\Anaconda3\lib\site-packages\ipykernel_launcher.py:4: SettingWithCopyWarning:
A value is trying to be set on a copy of a slice from a DataFrame.
Try using .loc[row_indexer,col_indexer] = value instead

See the caveats in the documentation: http://pandas.pydata.org/pandas-
docs/stable/indexing.html#indexing-view-versus-copy
  after removing the cwd from sys.path.
```

Model 12: (Factors: Media, Length, Emoji, Median,Polarity) with Lasso regression

Lasso regression does automatic feature selection that means if some features are correlated then lasso will pick only one feature

In [23]:

```
X_train1=train[["content_media_count","content_length","forward_median","comment_median","like_medi
an","polarity"]]
Y_train1=train[["forward_count","comment_count","like_count"]]
X_test1=predict[["content_media_count","content_length","forward_median","comment_median","like_med
ian","polarity"]]
Y_test1=predict[["forward_count","comment_count","like_count"]]

pd.options.mode.use_inf_as_na = True
X_train1.fillna(X_train1.max(),inplace=True)
X_test1.fillna(X_test1.max(),inplace=True)
```

```
C:\Users\user\Anaconda3\lib\site-packages\pandas\core\generic.py:5430: SettingWithCopyWarning:
A value is trying to be set on a copy of a slice from a DataFrame

See the caveats in the documentation: http://pandas.pydata.org/pandas-
docs/stable/indexing.html#indexing-view-versus-copy
  self._update_inplace(new_data)
```

In [24]:

```
lm1=Lasso(alpha=0.01)
model1=lm1.fit(X_train1,Y_train1)
pred1=lm1.predict(X_test1)
pred1=pred1.round()
pred1=(np.maximum(pred1,0.))
```

In [25]:

```
print(model1.coef_)
print(model1.intercept_)
```

```
[[ -3.93885876e-01   5.30762986e-03  -7.01981756e+00  -4.71960564e+00
    1.25490926e+01  -8.17241037e-02]
 [ -3.98749279e-01  -1.15532108e-03  -1.29700987e+00   7.65902250e-01
    1.36959059e+00   8.67988313e-02]
 [ -1.33179860e-01  -2.99999206e-03  -8.19004244e-01  -0.00000000e+00
    2.14881763e+00   4.81255664e-02]]
[ 0.09786835  0.4897481   0.43776902]
```

In [26]:

```
np.savetxt("C:/Users/user/Downloads/weibo_predict_result54.csv",pred1,delimiter=',',header="forward
_count,comment_count,like_count",comments="")
result1=pd.read_csv("C:/Users/user/Downloads/weibo_predict_result54.csv")
```

In [27]:

```
train_real_pred=Y_test1
train_real_pred['fp']=result1['forward_count']
train_real_pred['cp']=result1['comment_count']
train_real_pred['lp']=result1['like_count']
print("Score:{0:.2f}%".format(precision(train_real_pred.values)*100))
```

```
Score:35.58%
```

```
C:\Users\user\Anaconda3\lib\site-packages\ipykernel_launcher.py:2: SettingWithCopyWarning:
A value is trying to be set on a copy of a slice from a DataFrame.
Try using .loc[row_indexer,col_indexer] = value instead

See the caveats in the documentation: http://pandas.pydata.org/pandas-
docs/stable/indexing.html#indexing-view-versus-copy

C:\Users\user\Anaconda3\lib\site-packages\ipykernel_launcher.py:3: SettingWithCopyWarning:
A value is trying to be set on a copy of a slice from a DataFrame.
Try using .loc[row_indexer,col_indexer] = value instead

See the caveats in the documentation: http://pandas.pydata.org/pandas-
docs/stable/indexing.html#indexing-view-versus-copy
  This is separate from the ipykernel package so we can avoid doing imports until
C:\Users\user\Anaconda3\lib\site-packages\ipykernel_launcher.py:4: SettingWithCopyWarning:
A value is trying to be set on a copy of a slice from a DataFrame.
Try using .loc[row_indexer,col_indexer] = value instead

See the caveats in the documentation: http://pandas.pydata.org/pandas-
docs/stable/indexing.html#indexing-view-versus-copy
  after removing the cwd from sys.path.
```

In [1]:

```
import pandas as pd
```

Normalizing polarity over complete dataset

In [175]:

```
p0=pd.read_csv("polarityL0.csv")
p1=pd.read_csv("polarityL1.csv")
p2=pd.read_csv("polarityL2.csv")
p3=pd.read_csv("polarityL3.csv")
p4=pd.read_csv("polarityL4.csv")
p5=pd.read_csv("polarityL5.csv")
p6=pd.read_csv("polarityL6.csv")
p7=pd.read_csv("polarityL7.csv")
p8=pd.read_csv("polarityL8.csv")
p9=pd.read_csv("polarityL9.csv")
p10=pd.read_csv("polarityL10.csv")
p11=pd.read_csv("polarityL11.csv")
```

In [176]:

```
frames1=[p0,p1,p2,p3,p4,p5,p6,p7,p8,p9,p10,p11]
```

In [177]:

```
polarity=pd.concat(frames1)
```

```
/home/shashwat/anaconda3/lib/python3.6/site-packages/ipykernel_launcher.py:1: FutureW
arning: Sorting because non-concatenation axis is not aligned. A future version
of pandas will change to not sort by default.

To accept the future behavior, pass 'sort=False'.

To retain the current behavior and silence the warning, pass 'sort=True'.

  """Entry point for launching an IPython kernel.
```

In [178]:

```
polarity.shape[0]
```

Out[178]:

```
1223517
```

In [179]:

```
preprocess1=pd.read_csv("preprocessed_1.csv")
```

In [180]:

```
preprocess2=pd.read_csv("preprocessed_2.csv")
```

In [181]:

```
frames=[preprocess1,preprocess2]
```

In [182]:

```
train=pd.concat(frames)
```

In [183]:

```
train.shape[0]
```

Out[183]:

```
1223517
```

In [187]:

```
train=train.reset_index()
```

In [188]:

```
polarity.shape[0]
```

Out[188]:

```
1223517
```

In [197]:

```
res=pd.merge(polarity, train,left_index=True, right_index=True)
```

In [198]:

```
res.shape[0]
```

Out[198]:

```
1223517
```

In [202]:

```
res=res.rename(columns={'u_id_x':'u_id'})
```

In [203]:

```
uid_stat=pd.read_csv("train_uid_stat.csv")
```

In [204]:

```
uid_stat.size
```

Out[204]:

```
484419
```

In [205]:

```
dfmerge=pd.merge(res,uid_stat, on=['u_id'],how='left')
```

In [206]:

```
dfmerge.columns
```

Out[206]:

```
Index(['Stemming', 'Stemmingle', 'Stopword_removed', 'Stopwrod_removed',
       'Unnamed: 0', 'Unnamed: 1_x', 'comment_count_x', 'content_x',
       'content_media_count_x', 'content_spchar_x', 'date_x', 'en_content_x',
       'en_contentst', 'en_contenturl', 'forward_count_x', 'lemmatization',
       'lemmatizationtl', 'lemmatizationtlp', 'like_count_x', 'lower',
       'm_id_x', 'no_num', 'no_num.1', 'no_punc', 'non_emoji_content_x',
       'polarity', 'time_x', 'u_id', 'url_rem', 'url_rem.1', 'index', 'u_id_y',
       'm_id_y', 'forward_count_y', 'comment_count_y', 'like_count_y',
       'content_y', 'date_y', 'time_y', 'content_media_count_y',
       'content_spchar_y', 'non_emoji_content_y', 'en_content_y',
       'Unnamed: 1_y', 'forward_min', 'forward_max', 'forward_median',
       'forward_mean', 'comment_min', 'comment_max', 'comment_median',
       'comment_mean', 'like_min', 'like_max', 'like_median', 'like_mean'],
      dtype='object')
```

In [207]:

```
dfmerge.shape[0]
```

Out[207]:

```
1223517
```

In [208]:

```
dfmerge.columns
```

Out[208]:

```
Index(['Stemming', 'Stemmingle', 'Stopword_removed', 'Stopwrod_removed',
       'Unnamed: 0', 'Unnamed: 1_x', 'comment_count_x', 'content_x',
       'content_media_count_x', 'content_spchar_x', 'date_x', 'en_content_x',
       'en_contentst', 'en_contenturl', 'forward_count_x', 'lemmatization',
       'lemmatizationtl', 'lemmatizationtlp', 'like_count_x', 'lower',
       'm_id_x', 'no_num', 'no_num.1', 'no_punc', 'non_emoji_content_x',
       'polarity', 'time_x', 'u_id', 'url_rem', 'url_rem.1', 'index', 'u_id_y',
       'm_id_y', 'forward_count_y', 'comment_count_y', 'like_count_y',
       'content_y', 'date_y', 'time_y', 'content_media_count_y',
       'content_spchar_y', 'non_emoji_content_y', 'en_content_y',
       'Unnamed: 1_y', 'forward_min', 'forward_max', 'forward_median',
       'forward_mean', 'comment_min', 'comment_max', 'comment_median',
       'comment_mean', 'like_min', 'like_max', 'like_median', 'like_mean'],
      dtype='object')
```

In [ ]:

```
dfmerge=dfmerge.drop(['Unnamed: 1_x','Unnamed: 0','lemmatizationtlp'],axis=1)
```

In [211]:

```
import pandas as pd
import numpy as np
import re
from sklearn import linear_model
from sklearn.linear_model import Lasso
from sklearn.model_selection import train_test_split
from sklearn.metrics import mean_squared_error
from matplotlib import pyplot as plt
#from textblob import TextBlob
import statsmodels.api as sm
import import_ipynb
from evaluation import precision
from runTime import runTime
```

In [212]:

```
import sklearn
```

In [213]:

```
df1=dfmerge
```

In [214]:

```
def normalize(df):
    result = df.copy()
    for feature_name in df.columns:

        result[feature_name] = (df[feature_name] - min_value) / (max_value - min_valu
e)
    return result
```

In [215]:

```
max_value = df1['polarity'].max()
min_value = df1['polarity'].min()
df1['pnorm'] = (df1['polarity'] - min_value) / (max_value - min_value)
```

In [216]:

```
df1 = sklearn.utils.shuffle(df1)
```

In [217]:

```
df1=df1.fillna(0)
```

In [218]:

```
train=df1[0:110000]
predict=df1[110000:]
```

In [221]:

```
from sklearn.ensemble import RandomForestRegressor
features_train=train[['content_media_count_y','pnorm','forward_min', 'forward_max', '
forward_median', 'forward_mean',
       'comment_min', 'comment_max', 'comment_median', 'comment_mean',
       'like_min', 'like_max', 'like_median', 'like_mean']]
features_test=predict[['content_media_count_y','pnorm','forward_min', 'forward_max',
'forward_median', 'forward_mean',
       'comment_min', 'comment_max', 'comment_median', 'comment_mean',
       'like_min', 'like_max', 'like_median', 'like_mean']]
labels_train=train[['forward_count_x', 'comment_count_x', 'like_count_x']]
labels_test=predict[['forward_count_x', 'comment_count_x', 'like_count_x']]

x = features_train
y = labels_train
x1 = features_test
y1 = labels_test
```

In [222]:

```
regr = RandomForestRegressor(max_depth=50, random_state=0,n_estimators=100)
regr.fit(x, y)
pred2 = regr.predict(x1)
temp = pd.DataFrame.from_records(pred2)
temp=temp.round()
temp=(np.maximum(temp,0))
temp=temp.abs()
temp=temp.astype(int)
```

In [223]:

```
train_real_pred=y1
train_real_pred['fp']=temp[0].values
train_real_pred['cp']=temp[1].values
train_real_pred['lp']=temp[2].values
print("Score:{0:.2f}%".format(precision(train_real_pred.values)*100))
```

```
/home/shashwat/anaconda3/lib/python3.6/site-packages/ipykernel_launcher.py:2: Setting
WithCopyWarning:
A value is trying to be set on a copy of a slice from a DataFrame.
Try using .loc[row_indexer,col_indexer] = value instead

See the caveats in the documentation: http://pandas.pydata.org/pandas-docs/stable/ind
exing.html#indexing-view-versus-copy

/home/shashwat/anaconda3/lib/python3.6/site-packages/ipykernel_launcher.py:3: Setting
WithCopyWarning:
```

```
A value is trying to be set on a copy of a slice from a DataFrame.
Try using .loc[row_indexer,col_indexer] = value instead

See the caveats in the documentation: http://pandas.pydata.org/pandas-docs/stable/indexing.html#indexing-view-versus-copy
  This is separate from the ipykernel package so we can avoid doing imports until
/home/shashwat/anaconda3/lib/python3.6/site-packages/ipykernel_launcher.py:4: SettingWithCopyWarning:
A value is trying to be set on a copy of a slice from a DataFrame.
Try using .loc[row_indexer,col_indexer] = value instead

See the caveats in the documentation: http://pandas.pydata.org/pandas-docs/stable/indexing.html#indexing-view-versus-copy
  after removing the cwd from sys.path.
```

Score:78.98%

Predicting for the test data and generating submission file

In [224]:

```
pred1=pd.read_csv("weibo_predict_cpts.csv")
```

In [225]:

```
p=pd.read_csv("polarity_pred.csv")
```

In [226]:

```
p.shape[0]
```

Out[226]:

123517

In [227]:

```
res=pd.merge(p, pred1,left_index=True, right_index=True)
```

In [228]:

```
res.shape[0]
```

Out[228]:

123517

In [229]:

```
predict=res
```

In [230]:

```
predict.shape[0]
```

Out[230]:

```
123517
```

In [232]:

```
train=df1
```

In [233]:

```
train.shape[0]
```

Out[233]:

```
1223517
```

In [234]:

```
train.columns
```

Out[234]:

```
Index(['Stemming', 'Stemmingle', 'Stopword_removed', 'Stopwrod_removed',
       'Unnamed: 0', 'Unnamed: 1_x', 'comment_count_x', 'content_x',
       'content_media_count_x', 'content_spchar_x', 'date_x', 'en_content_x',
       'en_contentst', 'en_contenturl', 'forward_count_x', 'lemmatization',
       'lemmatizationtl', 'lemmatizationtlp', 'like_count_x', 'lower',
       'm_id_x', 'no_num', 'no_num.1', 'no_punc', 'non_emoji_content_x',
       'polarity', 'time_x', 'u_id', 'url_rem', 'url_rem.1', 'index', 'u_id_y',
       'm_id_y', 'forward_count_y', 'comment_count_y', 'like_count_y',
       'content_y', 'date_y', 'time_y', 'content_media_count_y',
       'content_spchar_y', 'non_emoji_content_y', 'en_content_y',
       'Unnamed: 1_y', 'forward_min', 'forward_max', 'forward_median',
       'forward_mean', 'comment_min', 'comment_max', 'comment_median',
       'comment_mean', 'like_min', 'like_max', 'like_median', 'like_mean',
       'pnorm'],
      dtype='object')
```

In [199]:

```
#pred2.shape[0]
```

In [235]:

```
predict=predict.fillna(0)
```

In [243]:

```
max_value = predict['polarity'].max()
min_value = predict['polarity'].min()
predict['pnorm'] = (predict['polarity'] - min_value) / (max_value - min_value)
```

In [244]:

```
predict['forward_count']=0
predict['comment_count']=0
predict['like_count']=0
```

In [245]:

```
predict.columns
```

Out[245]:

```
Index(['Unnamed: 0', 'lemmatizationtlp', 'no_punc', 'polarity', 'u_id', 'm_id',
       'content', 'date', 'time', 'content_media_count', 'content_#_count',
       'content_@_count', 'content_?_count', 'content_!_count',
       'content_length', 'content_emoji_count', 'hour', 'min', 'sec',
       'forward_min', 'forward_max', 'forward_median', 'forward_mean',
       'comment_min', 'comment_max', 'comment_median', 'comment_mean',
       'like_min', 'like_max', 'like_median', 'like_mean', 'forward_count',
       'comment_count', 'like_count', 'pnorm'],
      dtype='object')
```

In [246]:

```
from sklearn.ensemble import RandomForestRegressor
features_train=train[['content_media_count_y','pnorm','forward_min', 'forward_max', '
forward_median', 'forward_mean',
       'comment_min', 'comment_max', 'comment_median', 'comment_mean',
       'like_min', 'like_max', 'like_median', 'like_mean']]
features_test=predict[['content_media_count','pnorm','forward_min', 'forward_max', 'f
orward_median', 'forward_mean',
       'comment_min', 'comment_max', 'comment_median', 'comment_mean',
       'like_min', 'like_max', 'like_median', 'like_mean']]
labels_train=train[['forward_count_x', 'comment_count_x', 'like_count_x']]
labels_test=predict[['forward_count', 'comment_count', 'like_count']]

x = features_train
y = labels_train
x1 = features_test
y1 = labels_test
```

In [247]:

```
regr = RandomForestRegressor(max_depth=50, random_state=0,n_estimators=100)
regr.fit(x, y)
pred2 = regr.predict(x1)
temp = pd.DataFrame.from_records(pred2)
temp=temp.round()
temp=(np.maximum(temp,0))
temp=temp.abs()
temp=temp.astype(int)
```

In [248]:

```
predict['forward_count']=temp[0].values
predict['comment_count']=temp[1].values
predict['like_count']=temp[2].values
```

In [249]:

```
predictf=predict[['u_id','m_id','forward_count','comment_count','like_count']].copy()
```

In [250]:

```
result = []
filename = "SUBMISSION_POLARITY.txt"
#predictf1=predictf.head(10000)
for _,row in predictf.iterrows():
    result.append("{0}\t{1}\t{2},{3},{4}\n".format(row[0],row[1],row[2],row[3],row[4]
))
f = open(filename,'w')
f.writelines(result)
f.close()
print ('generate submission file'.format(filename))
```

```
generate submission file
```