探索Hacker News上的帖子

8207200203 计算机2006班 翟衍博

1. 了解和读取数据集

Hacker News是由Y Combinator创建的网站，其中用户提交的故事（称为“帖子”）被投票和评论。列的说明：

id：来自Hacker News的唯一标识符；

title：帖子的标题；

url：帖子链接到的URL，如果帖子有URL；

num_points：帖子获得的点数，计算的方法为支持的总票数减去反对的总票数；

num_comments：在帖子上发表的评论数；

author：提交帖子的人的用户名；

created_at：提交帖子的日期和时间

我们只关心以Ask HN或Show HN开头的帖子标题,我们将比较这两种类型的帖子，以确定以下内容：

Ask HN或Show HN平均会收到更多的评论吗?

在某个时间发布的帖子平均会收到更多评论吗?

读取数据集

In [59]:

```
opened_file=open('hacker_news.csv')
from csv import reader
read_file=reader(opened_file)
hn=list(read_file)
print(hn[:5])
```

```
[['id', 'title', 'url', 'num_points', 'num_comments', 'author', 'created_at'], ['122
24879', 'Interactive Dynamic Video', 'http://www.interactivedynamicvideo.com/', '38
6', '52', 'ne0phyte', '8/4/2016 11:52'], ['10975351', 'How to Use Open Source and Sh
ut the Fuck Up at the Same Time', 'http://hueniverse.com/2016/01/26/how-to-use-open-
source-and-shut-the-fuck-up-at-the-same-time/', '39', '10', 'josep2', '1/26/2016 19:
30'], ['11964716', "Florida DJs May Face Felony for April Fools' Water Joke", 'htt
p://www.thewire.com/entertainment/2013/04/florida-djs-april-fools-water-joke/6379
8/', '2', '1', 'vezycash', '6/23/2016 22:20'], ['11919867', 'Technology ventures: Fr
om Idea to Enterprise', 'https://www.amazon.com/Technology-Ventures-Enterprise-Thoma
s-Byers/dp/0073523429', '3', '1', 'hswarna', '6/17/2016 0:01']]
```

2.从二维列表中移除标题行

In [60]:

```
headers=hn[0]
hn=hn[1:]
print(headers)
print(hn[:5])
```

```
['id', 'title', 'url', 'num_points', 'num_comments', 'author', 'created_at']
[['12224879', 'Interactive Dynamic Video', 'http://www.interactivedynamicvideo.co
m/', '386', '52', 'ne0phyte', '8/4/2016 11:52'], ['10975351', 'How to Use Open Sourc
e and Shut the Fuck Up at the Same Time', 'http://hueniverse.com/2016/01/26/how-to-u
se-open-source-and-shut-the-fuck-up-at-the-same-time/', '39', '10', 'josep2', '1/26/
2016 19:30'], ['11964716', "Florida DJs May Face Felony for April Fools' Water Jok
e", 'http://www.thewire.com/entertainment/2013/04/florida-djs-april-fools-water-jok
e/63798/', '2', '1', 'vezycash', '6/23/2016 22:20'], ['11919867', 'Technology ventur
es: From Idea to Enterprise', 'https://www.amazon.com/Technology-Ventures-Enterprise
-Thomas-Byers/dp/0073523429', '3', '1', 'hswarna', '6/17/2016 0:01'], ['10301696',
'Note by Note: The Making of Steinway L1037 (2007)', 'http://www.nytimes.com/2007/1
1/07/movies/07stein.html?_r=0', '8', '2', 'walterbell', '9/30/2015 4:12']]
```

3. 提取Ask HN和Show HN帖子

In [61]:

```
#使用startswith(),lower()方法

ask_posts=[]
show_posts=[]
other_posts=[]
for row in hn:
    title=row[1]
    title=title.lower()
    if title.startswith('show hn') :
        show_posts.append(row)
    elif title.startswith('ask hn'):
        ask_posts.append(row)
    else:
        other_posts.append(row)

print(len(ask_posts))
print(len(show_posts))
print(len(other_posts))
```

```
1744
1162
17194
```

4.计算Ask HN和Show HN帖子的平均评论数量

In [62]:

```
total_ask_comments=0
for row in ask_posts:
    cmt=row[4]
    total_ask_comments+=int(cmt)
avg_ask_comments=total_ask_comments/len(ask_posts)
print(avg_ask_comments)
```

```
14.038417431192661
```

In [63]:

```
total_show_comments=0
for row in show_posts:
    cmt=row[4]
    total_show_comments+=int(cmt)
avg_show_comments=total_show_comments/len(show_posts)
print(avg_show_comments)
```

```
10.31669535283993
```

根据以上分析结果，我们可以得出“ask posts”类型会收获更多评论，这也符合常理，因为帖子是提问类型的，那么就会多一些评论回答。

5. 按小时计算所创建的ask posts和评论的数量

In [64]:

```
# 接下来，我们将确定在特定时间创建的ask posts是否更有可能吸引评论。
#使用datetime.strptime()
import datetime as dt
result_list=[]
for row in ask_posts:
    c_a=row[6]    #created_at列
    cmt=int(row[4])    #评论数量
    result_list.append([c_a,cmt])

counts_by_hour={}
comments_by_hour={}

for row in result_list:
    time=dt.datetime.strptime(row[0],'%m/%d/%Y %H:%M')#格式举例：'1/26/2016 19:30'
    hour=time.hour
    cmt=row[1]
    if hour not in counts_by_hour:
        counts_by_hour[hour]=1
        comments_by_hour[hour]=cmt
    else:
        counts_by_hour[hour]+=1
        comments_by_hour[hour]+=cmt
```

6. 按小时计算Ask HN帖子的平均评论数量

In [65]:

```
commments_pre_hour=[]
for hour in counts_by_hour:
    avg_cmt=comments_by_hour[hour]/counts_by_hour[hour]
    commments_pre_hour.append([hour,avg_cmt])
```

7. 对二维列表进行排序

In [66]:

```
#要根据平均评论数进行排序，因此要交换hour与avg_cmt的位置，
#然后使用sorted（）方法进行排序。
swap_commments_pre_hour=[];
for item in commments_pre_hour:
    swap_commments_pre_hour.append([item[1],item[0]])
sorted_swap=sorted(swap_commments_pre_hour,reverse=True);

print("Top 5 Hours for Ask Posts Comments")

for i in range(5):
    print(sorted_swap[i][0],sorted_swap[i][1])
```

```
Top 5 Hours for Ask Posts Comments
38.5948275862069 15
23.810344827586206 2
21.525 20
16.796296296296298 16
16.009174311926607 21
```

In [67]:

```
# 进行格式化
for i in range(5):
    print("{}:00: {:.2f} average comments per post".format(sorted_swap[i][1],sorted_swap[i][0]))
```

```
15:00: 38.59 average comments per post
2:00: 23.81 average comments per post
20:00: 21.52 average comments per post
16:00: 16.80 average comments per post
21:00: 16.01 average comments per post
```

结论：从以上分析可以得出，当地时间15：00时，Ask HN帖子的平均评论数量最大。