act_report

2018 年 12 月 14 日

```
In [ ]: import pandas as pd
        import numpy as np
        import json
        import tweepy
        import requests
        import matplotlib.pyplot as plt
        import seaborn as sns
        %matplotlib inline
```

1 收集数据

```
In [ ]: # 导入所需数据
        twitter_archive_enhanced = pd.read_csv('twitter_archive_enhanced.csv')
        image_predictions = pd.read_csv('image-predictions.tsv',sep='\t')
        tweet_json = pd.read_json('tweet_json.txt',lines=True)
```

```
In [ ]: #tweet_json 中仅取 id, retweet_count 和 favorite_count
        tweet_json = tweet_json[['id','retweet_count','favorite_count']]
        tweet_json.columns = ['tweet_id','retweet_count','favorite_count']
```

2 评估数据

```
In [ ]: # 设置列宽
        pd.set_option('max_colwidth',200)
```

```
In [ ]: # 建立副本
        tae_df = twitter_archive_enhanced.copy()
        im_df = image_predictions.copy()
        tj_df = tweet_json.copy()
```

```
In [ ]: tae_df.head()
```

```
In [ ]: tae_df.info()
```

```
In [ ]: tae_df.describe()
```

```
In [ ]: tae_df.expanded_urls.value_counts()
```

```
In [ ]: tae_df['name'].value_counts()
```

```
In [ ]: im_df.head()
```

```
In [ ]: im_df.info()
```

```
In [ ]: #jpg_url 列重复数量
        im_df.jpg_url.duplicated().sum()
```

```
In [ ]: tj_df.describe()
```

```
In [ ]: tj_df.info()
```

2.0.1 质量问题

twitter_archive_enhanced 表

- in_reply_to_status_id 和 in_reply_to_user_id 存在数据缺失；
- retweeted_status_id，retweeted_status_user_id 和 retweeted_status_timestamp 存在缺失；
- expanded_urls 列存在缺失；
- expanded_urls 中一列中同一种连接多次重复；
- rating_numerator 中出现异常大的数，远大于位置在 75% 的数；
- rating_denominator 均值和最大值都不为 10，分母出现了大于 10 的数;
- name 中有缺失且狗狗的名字有'a','the','an' 明显名字有误；
- source 列可只保留来源，删除链接地址；
- text 列中存在 RT 开头的转发信息，应考虑删除；
- timestamp 列格式有误；
- doggo,floorfer,pupper,puppo 列有缺失；

image_predictions 表

- jpg_url 列有 66 条链接重复；

2.0.2 整洁度问题

twitter_archive_enhanced 表

- doggo,floorfer,pupper,puppo 四类可以合并为一列；
- 三表可按照 id 进行合并整理；

3 清理数据

数据缺失 1.in_reply_to_status_id 和 in_reply_to_user_id 存在数据缺失，做删除

```
In [ ]: tae_df = tae_df.drop(['in_reply_to_status_id','in_reply_to_user_id'],axis=1)
```

2.jpg_url 列有 66 条链接重复，做删除

```
In [ ]: im_df = im_df[~im_df.jpg_url.duplicated()]
```

3.text 列中存在 RT 开头的转发信息，应考虑删除

```
In [ ]: tae_df = tae_df[tae_df['retweeted_status_id'].isnull()]
```

4.retweeted_status_id，retweeted_status_user_id 和 retweeted_status_timestamp 存在缺失，做删除

```
In [ ]: tae_df = tae_df.drop(['retweeted_status_id','retweeted_status_user_id','retweeted_statu
```

5.expanded_urls 列存在缺失，做删除

```
In [ ]: tae_df = tae_df[tae_df.expanded_urls.notnull()]
```

```
In [ ]: tae_df.info()
```

整洁度 6.doggo,floorfer,pupper,puppo 四类可以合并为一列

```
In [ ]: tae_df['category'] = tae_df['text'].str.lower().str.findall(r'doggo|floorfer|pupper|pu
        tae_df['category'] = tae_df['category'].apply(lambda x:','.join(set(x)))
        tae_df['category'] = tae_df['category'].replace('',np.nan)
```

```
In [ ]: tae_df['category'].value_counts()
```

7. 三表可按照 id 进行合并整理

```
In [ ]: df = tae_df.merge(tj_df,how='left',on='tweet_id').merge(im_df,how='inner',on='tweet_id
```

```
In [ ]: df.info()
```

质量 8.rating_denominator 均值和最大值都不为 10，分母出现了大于 10 的数

```
In [ ]: #8,9 问题可以综合考虑，如果分数确认无误，可以转换为分数进行计算
        # 可以先参看分母不为 10 的数据
        df[df['rating_denominator']!=10]
```

```
In [ ]: # 合计 18 条，其中 5 条明显错误手动可修改；
        df.loc[df['tweet_id']==666287406224695296,'rating_numerator']=9
        df.loc[df['tweet_id']==666287406224695296,'rating_denominator']=10
        df.loc[df['tweet_id']==740373189193256964,'rating_numerator']=14
        df.loc[df['tweet_id']==740373189193256964,'rating_denominator']=10
        df.loc[df['tweet_id']==682962037429899265,'rating_numerator']=10
        df.loc[df['tweet_id']==682962037429899265,'rating_denominator']=10
        df.loc[df['tweet_id']==722974582966214656,'rating_numerator']=13
        df.loc[df['tweet_id']==722974582966214656,'rating_denominator']=10
        df.loc[df['tweet_id']==716439118184652801,'rating_numerator']=11
        df.loc[df['tweet_id']==716439118184652801,'rating_denominator']=10
```

```
In [ ]: # 计算得分，用小数形式表示；
        df['rating'] = df['rating_numerator']/df['rating_denominator']
```

```
In [ ]: # 存在 177.6 和 42 两个过大的值
        df['rating'].value_counts()
```

9.rating_numerator 中出现异常大的数，远大于位置在 75% 的数

```
In [ ]: # 有 6 个值得分大于了 2，我们可以具体看看；
        df[df['rating']>2]
```

```
In [ ]: # 得分为 2.6,2.7，7.5 三列明显截取错误；
        df.loc[df['tweet_id']==680494726643068929,'rating']=1.126
        df.loc[df['tweet_id']==778027034220126208,'rating']=1.127
        df.loc[df['tweet_id']==786709082849828864,'rating']=0.975
```

```
In [ ]: # 另三个无截取错误，可能为记录时有误，我们初步手动修改为合理值；
        df.loc[df['tweet_id']==670842764863651840,'rating'] = 0.42
        df.loc[df['tweet_id']==749981277374128128,'rating'] = 1.776
        df.loc[df['tweet_id']==810984652412424192,'rating'] = 24/70
```

10.timestamp 列格式有误

```
In [ ]: df['timestamp'] = pd.to_datetime(df['timestamp'].str.split('+',expand=True)[0])
```

4 保存数据

```
In [ ]: # 删除对分析无用的列 source, expanded_urls 和 doggo,floorfer,pupper,puppo
        df = df.drop(['source','expanded_urls','doggo','floofer','pupper','puppo'],axis=1)
```

```
In [ ]: df.to_csv('twitter_archive_master.csv',index=False)
```

5 数据分析

5.0.1 问题：

1. 哪种类型的狗狗是最受欢迎的？2. 哪种类型的狗狗评分最高？3. 评级越高，关注度就越高，受欢迎的人数是不是越多呢？

1. 哪种类型的狗狗是最受欢迎的？

```
In [ ]: # 将图片预测为狗狗的名字依次筛选出来
        dog = []
        for i in df.index:
            if df.loc[i].p1_dog==True:
                dog.append(df.loc[i].p1)
            elif (df.loc[i].p1_dog==False)&(df.loc[i].p2_dog==True):
                dog.append(df.loc[i].p2)
            elif (df.loc[i].p1_dog==False)&(df.loc[i].p2_dog==False)&(df.loc[i].p3_dog==True):
                dog.append(df.loc[i].p3)
            else:
                dog.append(np.nan)
```

```
In [ ]: # 新增 dog 列，包含图片预测狗狗的名字并删除空值
        df['dog'] = dog
        df = df[df['dog'].notnull()]
```

```
In [ ]: # 查看受欢迎人数在前五的狗狗
        df.groupby('dog')['favorite_count'].sum().sort_values(ascending=False).head(5)
```

```
In [ ]: # 查看转推数在前五的狗狗
        df.groupby('dog')['retweet_count'].sum().sort_values(ascending=False).head(5)
```

受欢迎数最多的狗狗前五依次是 golden_retriever(金毛)，Labrador_retriever(拉布拉多)，Pembroke(柯基)，Chihuahua(吉娃娃) 和 Samoyed(萨摩耶)；而转推数最多依旧是这前五位，其中金毛人气最高；

2. 哪种类型的狗狗评分最高？

```
In [ ]: # 查看狗狗数量，点评分的总计、均值和极值
        df.groupby('dog')['rating'].aggregate(['count',np.mean,sum,max,min]).sort_values(by=['
```

金毛的数量最多评分总计也是最大的，均值排在第 11 位，但就平均评价分进行排序而言，最高的为 Bouvier_des_Flandres(波兰德斯布比野犬), 均分为 1.3 分，但在整个表中仅 1 只同类犬

3. 评级越高，关注度就越高，受欢迎的人数是不是越多呢？

```
In [ ]: df_relation = df[['favorite_count','rating']]
```

```
In [ ]: sns.lmplot('rating','favorite_count',data = df_relation)
```

从回归函数可以看出，评分大小和受欢迎程度呈正相关关系。排除大于 1.5 的异常值，在 1.25 至 1.5 之间的点赞人数达到峰值。

6 结论

狗狗的评分和狗狗的受欢迎是呈正相关关系狗狗，评价越高的狗狗自然也会受到更多推友的关注和点赞，评分对数据有影响（当然我们不能排除主人的拍照水平影响，这些没有在数据中体现）。我们还发现，实际上最受欢迎的狗狗都是人们饲养比例比较高的类型，比如金毛、拉布拉多，萨摩耶等，这些狗狗除了本身更受到人们的喜爱而饲养人群很多很多，在数据集中相较其他狗狗数量占比也更大，但评分结果参差不齐。

7 反思

客观上数据缺失，比如狗狗的分类约 2/3 的缺失，直接影响到了对这一列的使用；再者对图片的预测结果也有较大误差，很难注意检验数据是否准确；数据中存在极少部分的极端大评分，实际并无客观的错误，但考虑整体评分，最后在分析时可考虑删除这些评分。