Sentiment Analysis for TD

Shasha Liu

Dec 08 2019

Agenda

- Objective
- Quick Look at the Raw Data
- Overview for cleaned data - Frequency
  General Sentiment Analysis
- Overview for cleaned data - Monthly
- Overview for cleaned data - Sept
  Sentiment Analysis (2018-09-09)
- Overview for cleaned data - May
  Sentiment Analysis (2019-05-23)
- Conclusion 1
  Recall General Sentiment Analysis
- Negative tweets analysis
- Conclusion 2

Objective

- **What is the most common evaluation for TD?**

- **Which kind of suggestion can we find to improve TD?**

Data from https://td-canada-trust.pissedconsumer.com/review.html from July 2018 to July 2019.

Quick Look at the Raw Data

```
In [10]: df = df.dropna(subset=['tweet_ID','screenname','tweet'],axis=0)
         df.shape

Out[10]: (160840, 4)
```

```
In [12]: x = df['screenname'].value_counts()
         x

Out[12]: TDBank_US          9705
         TD_Canada          6204
         TopEmployers40     1950
         TD_Insurance       1336
         YoungEmployers      916
                            ...
         globalhalifax         1
         john_lambe            1
         irsp2017              1
         Brian_Mic             1
         thucnhi21             1
         Name: screenname, Length: 65071, dtype: int64
```

- TD had 160840 comments from July 2018 to July 2019.
- More than 6000 comments from TDBank_Us and TD_Canada. The highest frenquency tweets from offical ID and recruiter ID.

Assumption 1: If users' name contain 'TD', there is a high probability that users are TD's employees. Therefore we delete the data whose screenname include 'TD'

Quick Look at the Raw Data

```
In [16]: num['N'].value_counts()

Out[16]: 1      47904
         2       8883
         3       3137
         4       1507
         5        866
                 ...
         198        1
         102        1
         83         1
         92         1
         287        1
         Name: N, Length: 147, dtype: int64
```

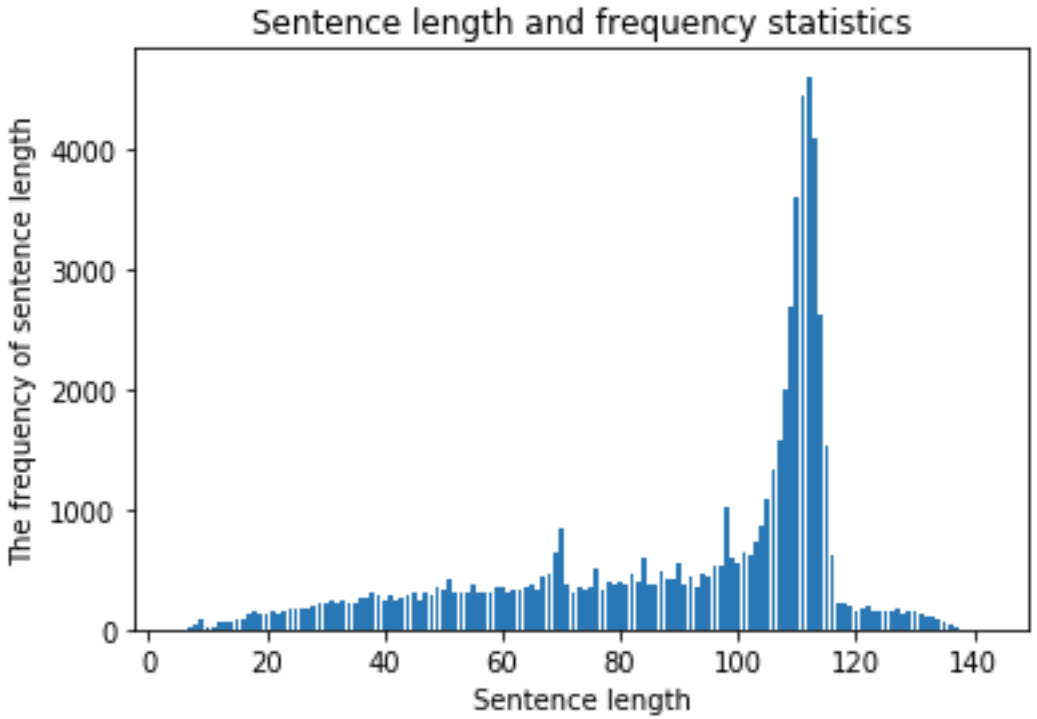
- Through the mode and median analysis, we can find most of people only send one tweet for TD a year.

```
In [18]: np.median(num)

Out[18]: 1.0
```

Assumption 2: Only analysis the tweets context which users send 1-2 tweets during a year

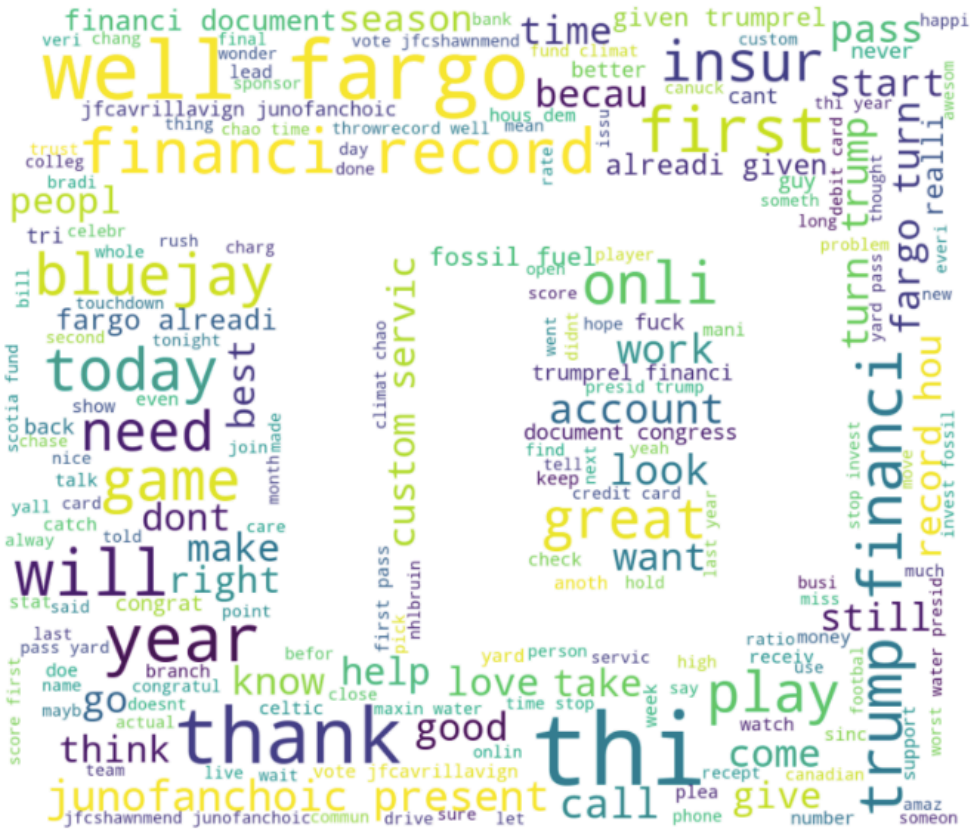
Overview for cleaned data - Frequency



length	frequency
112	4607
111	4449
113	4080

- The frequency of sentence length is 112

General Sentiment Analysis



Wordcloud (a year tweets)

- Overall, we can find ‘well’, ‘thanks’. It seems has more positive comments.

General Sentiment Analysis

In [77]:
```
from textblob import TextBlob
import re
def clean_tweet(tweet):
    '''
    Utility function to clean the text in a tweet by removing
    links and special characters using regex.
    '''
    return ' '.join(re.sub("(@[A-Za-z0-9]+)|([^0-9A-Za-z \t])|(\w+:\/\/\S+)", " ", tweet).split())
def analize_sentiment(tweet):
    '''
    Utility function to classify the polarity of a tweet
    using textblob.
    '''
    analysis = TextBlob(clean_tweet(tweet))
    if analysis.sentiment.polarity > 0:
        return 1
    elif analysis.sentiment.polarity == 0:
        return 0
    else:
        return -1
```

In [78]:
```
# We create a column with the result of the analysis:
text['SA'] = np.array([ analize_sentiment(tweet) for tweet in text['clean_tweet'] ])
```

In [79]:
```
pos_tweets = [ tweet for index, tweet in enumerate(text['clean_tweet']) if text['SA'][index] > 0]
neu_tweets = [ tweet for index, tweet in enumerate(text['clean_tweet']) if text['SA'][index] == 0]
neg_tweets = [ tweet for index, tweet in enumerate(text['clean_tweet']) if text['SA'][index] < 0]
```

In [80]:
```
print("Percentage of positive tweets: {}%".format(len(pos_tweets)*100/len(text['clean_tweet'])))
print("Percentage of neutral tweets: {}%".format(len(neu_tweets)*100/len(text['clean_tweet'])))
print("Percentage de negative tweets: {}%".format(len(neg_tweets)*100/len(text['clean_tweet'])))
```

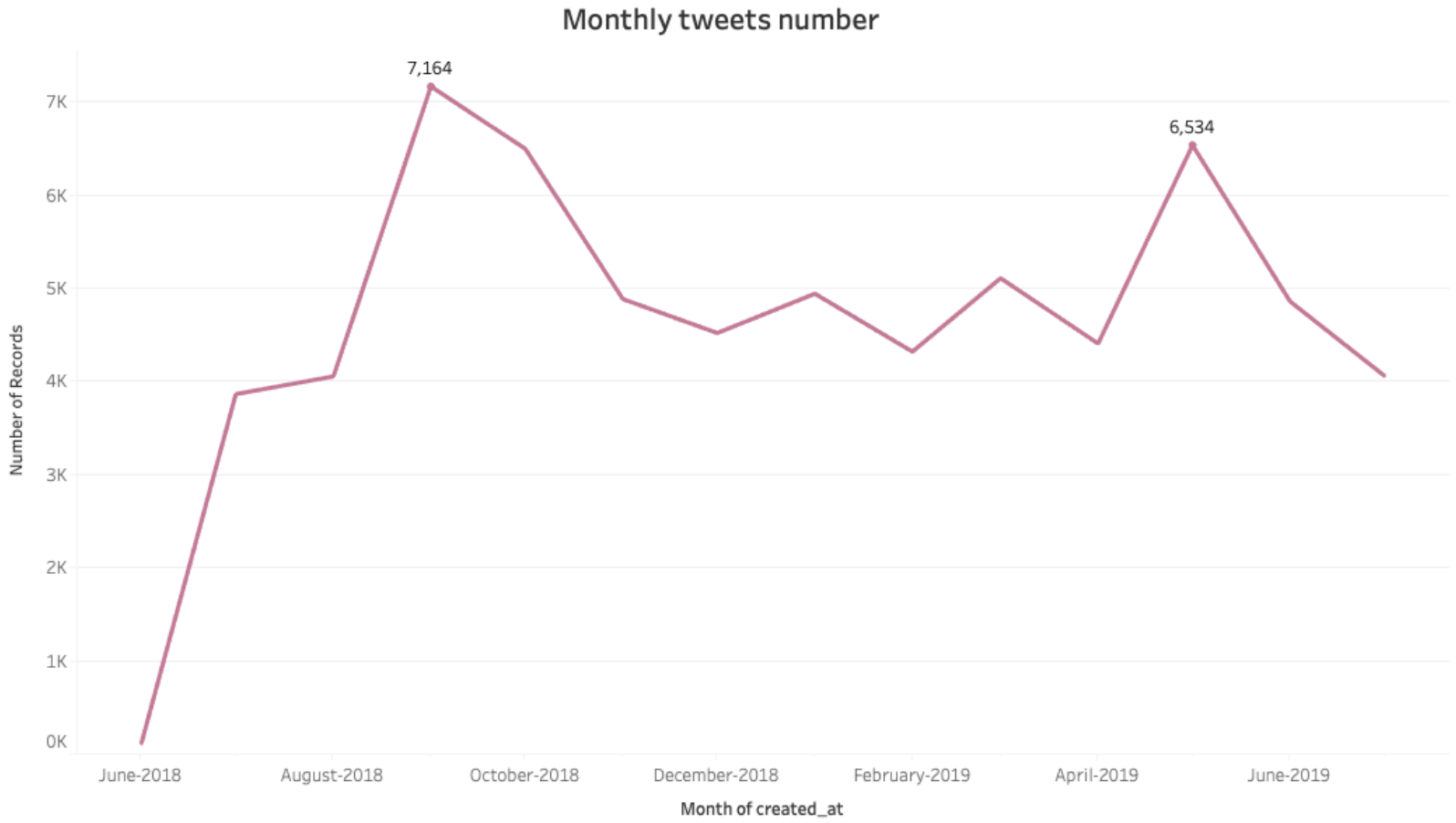
```
Percentage of positive tweets: 29.09425005742286%
Percentage of neutral tweets: 57.863869535257635%
Percentage de negative tweets: 13.041880407319502%
```

we can found that 57.86% tweets are neutral comments, around 29% tweets are positive comments, 13% are negative comments.

- we can found that 57.86% tweets are neutral comments, around 29% tweets are positive comments, 13% are negative comments.

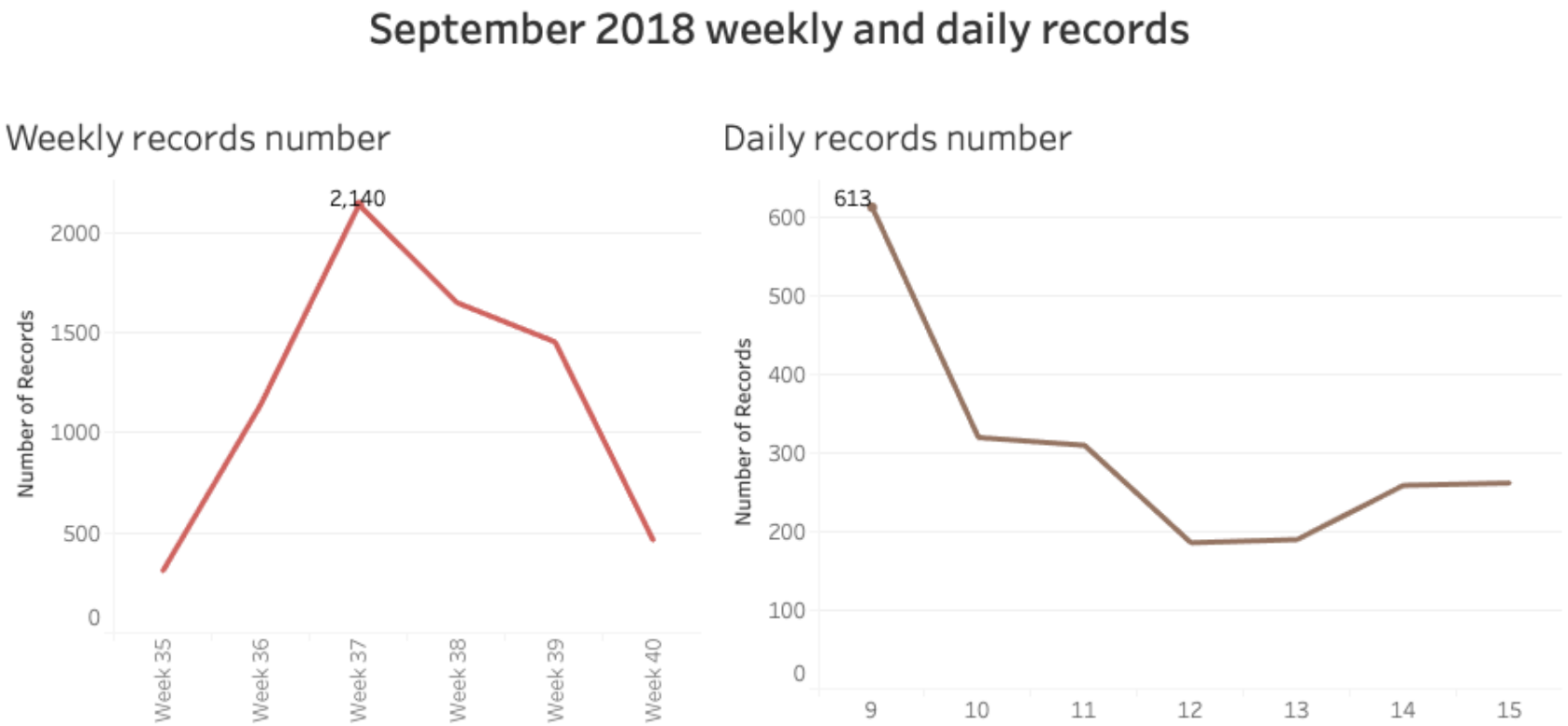
Sentiment analysis using TextBlob

Overview for cleaned data - Monthly



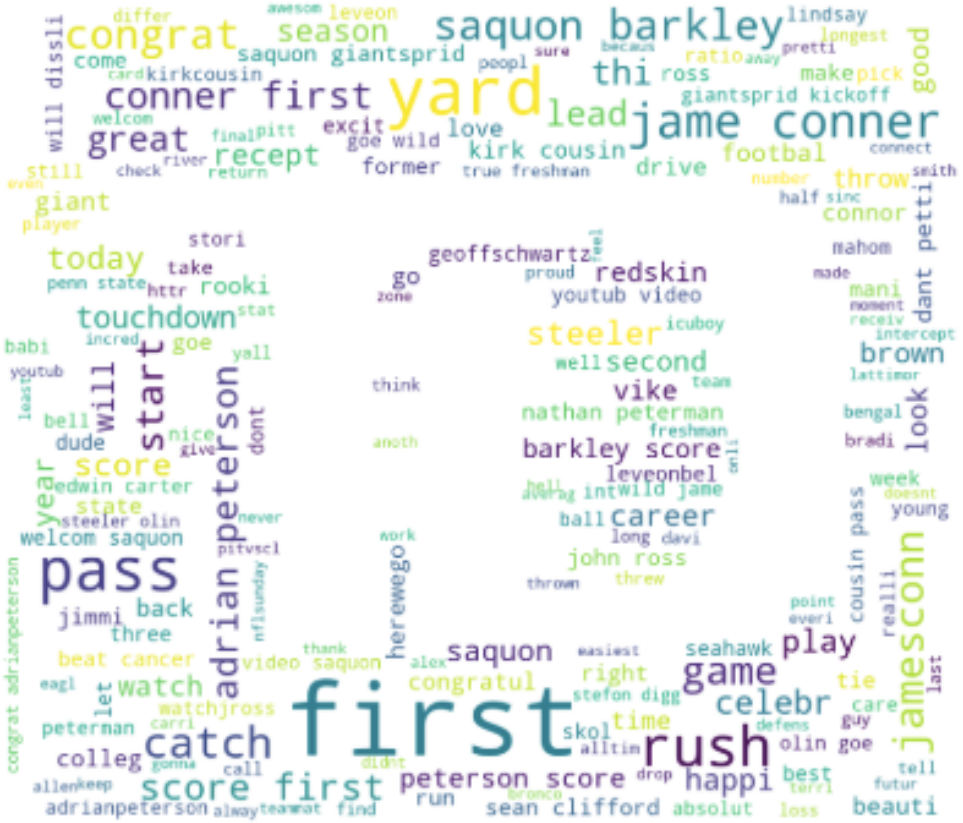
- Two peak happened on September 2018 and May 2019

Overview for cleaned data - Sept



- Peak September 2018 showed on 2018-09-09 (about 613 records)

Sentiment Analysis 2018-09-09



Wordcloud (2018-09-09)

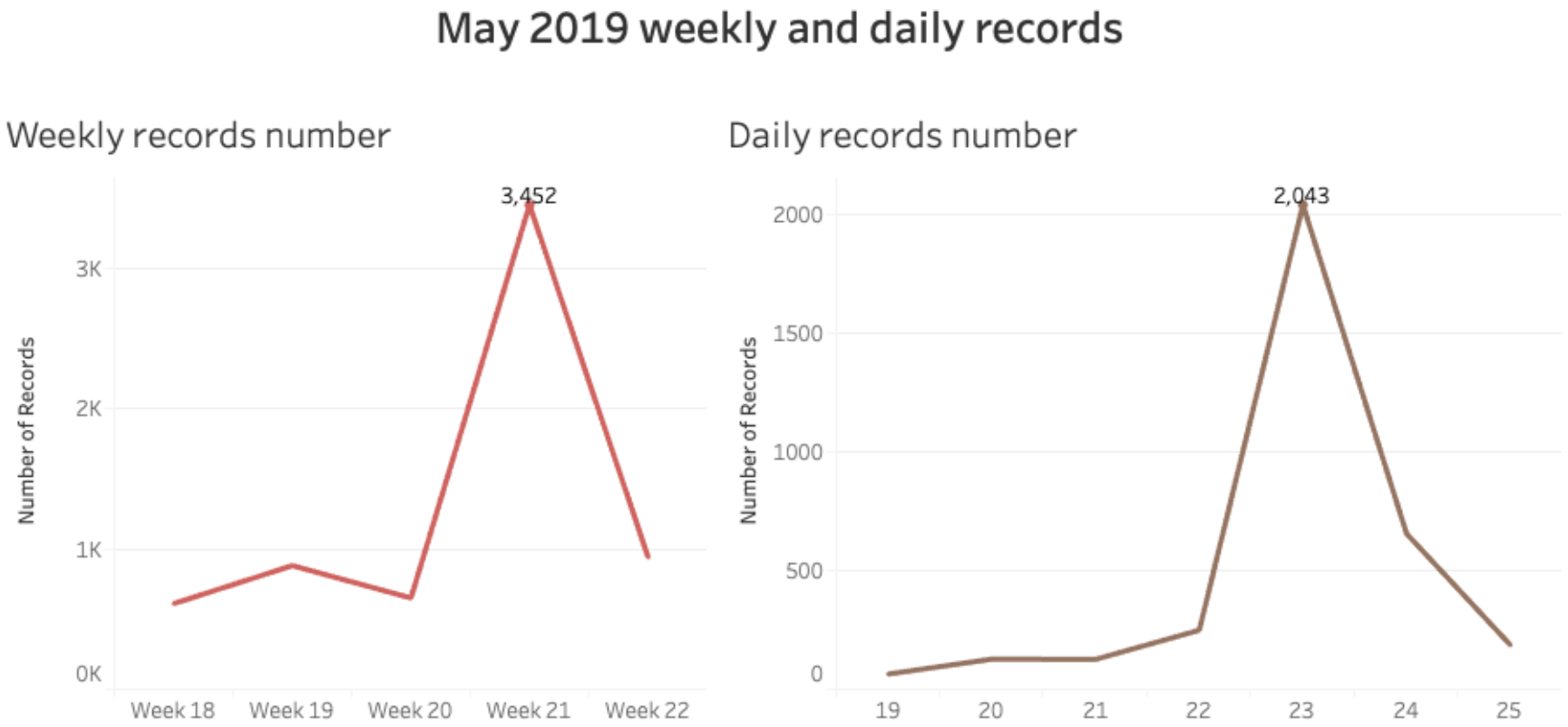
- we can found words like ‘first’, ‘great’,’congrat’ positive comments.

Sentiment Analysis 2018-09-09

```
Percentage of positive tweets: 56.76998368678 63%
Percentage of neutral tweets: 34.747145187601 96%
Percentage de negative tweets: 8.4828711256117 46%
```

- Here is more positive comments, only around 8.5% negative comments.

Overview for cleaned data- May



- Peak May 2019 showed on 2019-05-23 (about 2043 records)

Sentiment Analysis 2019-05-23

Wordcloud (2019-05-23)

- From the wordcloud we can find the word, such as ‘well’, but most words are neutral words.

Sentiment Analysis 2019-05-23

```
Percentage of positive tweets: 11.2090063631913 85%
Percentage of neutral tweets: 84.189916789035 73%
Percentage de negative tweets: 4.601076847772883%
```

- Here is more neutral tweets around 84.2%

Conclusion 1

- **What is the most common evaluation for TD?**

Through our sentiment analysis, we can find that people have more positive comments for TD.

Recall General Sentiment Analysis

In [77]:
```
from textblob import TextBlob
import re
def clean_tweet(tweet):
    '''
    Utility function to clean the text in a tweet by removing
    links and special characters using regex.
    '''
    return ' '.join(re.sub("(@[A-Za-z0-9]+)|([^0-9A-Za-z \t])|(\w+:\/\/\S+)", " ", tweet).split())
def analize_sentiment(tweet):
    '''
    Utility function to classify the polarity of a tweet
    using textblob.
    '''
    analysis = TextBlob(clean_tweet(tweet))
    if analysis.sentiment.polarity > 0:
        return 1
    elif analysis.sentiment.polarity == 0:
        return 0
    else:
        return -1
```

In [78]:
```
# We create a column with the result of the analysis:
text['SA'] = np.array([ analize_sentiment(tweet) for tweet in text['clean_tweet'] ])
```

In [79]:
```
pos_tweets = [ tweet for index, tweet in enumerate(text['clean_tweet']) if text['SA'][index] > 0]
neu_tweets = [ tweet for index, tweet in enumerate(text['clean_tweet']) if text['SA'][index] == 0]
neg_tweets = [ tweet for index, tweet in enumerate(text['clean_tweet']) if text['SA'][index] < 0]
```

In [80]:
```
print("Percentage of positive tweets: {}%".format(len(pos_tweets)*100/len(text['clean_tweet'])))
print("Percentage of neutral tweets: {}%".format(len(neu_tweets)*100/len(text['clean_tweet'])))
print("Percentage de negative tweets: {}%".format(len(neg_tweets)*100/len(text['clean_tweet'])))
```

```
Percentage of positive tweets: 29.09425005742286%
Percentage of neutral tweets: 57.863869535257635%
Percentage de negative tweets: 13.041880407319502%
```

we can found that 57.86% tweets are neutral comments, around 29% tweets are positive comments, 13% are negative comments.

- we can found that 57.86% tweets are neutral comments, around 29% tweets are positive comments, 13% are negative comments.

Sentiment analysis using TextBlob

Negative tweets analysis

- We can found these negative comments connect with service, and some bank products.

Conclusion 2

- **Which kind of suggestion can we find to improve TD?**

Through the analysis of negative tweets, we found that most comments related to the service and some products, which were general issues happened in retail industries.

Therefore, through the sentiment analysis, TD has a positive review. The suggestion we can give to help TD receive a higher score is improve their service and products.