Sentiment Analysis of vaccine tweets

Changyong Zhou

Abstract

By analyzing the dataset of "All COVID-19 Vaccines Tweets" from Kaggle.com, I tried to undestand which vaccine manufacture got the most positive reviews than others, and which manufacture got the most popularity.

Textblob , seaborn, matplotlib were used to analyze sentiment and visualize the findings.

After investigating 4 main vaccine manufactures, I identified that **Moderna** gained the most positive sentiment and most popularity.

Motivation

The purpose of my reseach is to find the best vaccine manufacture from tweets who owns the most popularity and most positive review. This can provide valuable information for health department of a government who has the privilege to procure vaccines from one producer instead of others.

Dataset

My dataset is "All COVID-19 Vaccines Tweets" from Kaggle.com, It collects recent tweets about the COVID-19 vaccines used in entire world, and includes 26539 records.

7 vaccine manufactures were involved in the dataset.

It was updated regularly.

Data preparation and cleaning

I removed null values and duplicated records from the dataset.

Cleaned the text field of the data before analyzing its sentiment.

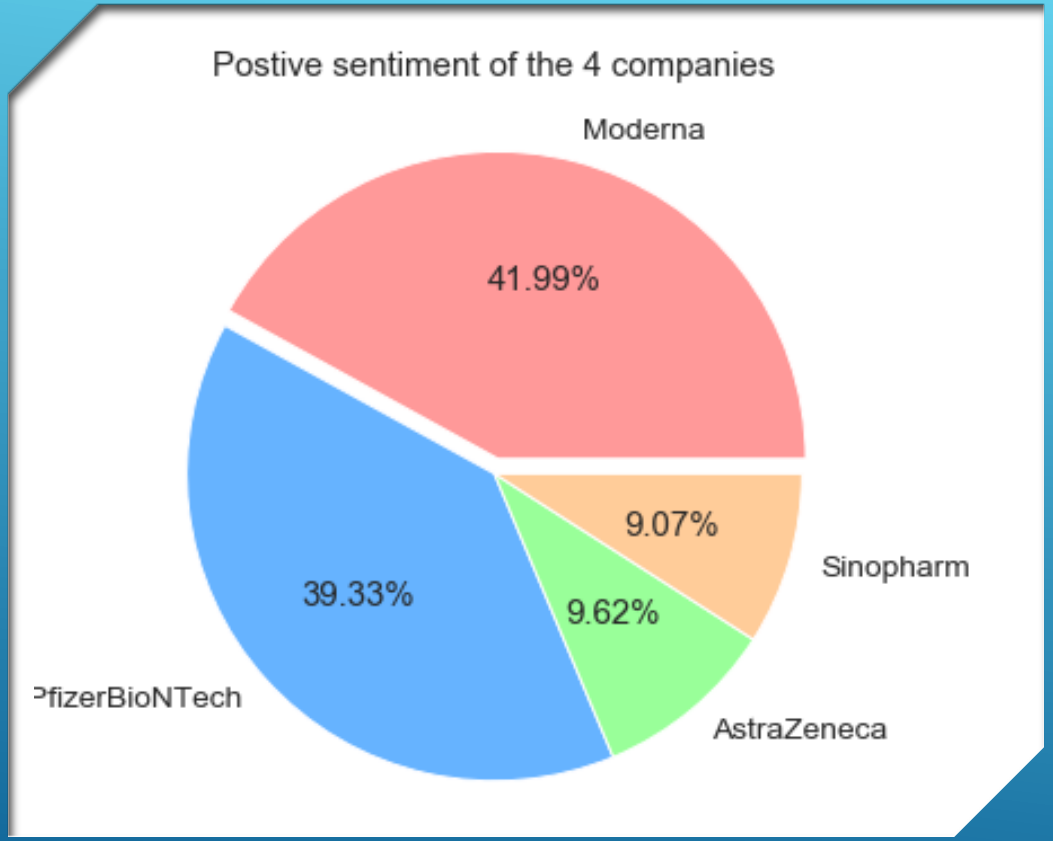
Research question(s)

Which vaccine manufacture got the most positive sentiments than others, and which manufacture got the most popularity.

Methods

I used **textblob** to handle the text and sentiment in the tweets, this library provided simple and effective APIs for NLP tasks, like sentiment analysis.

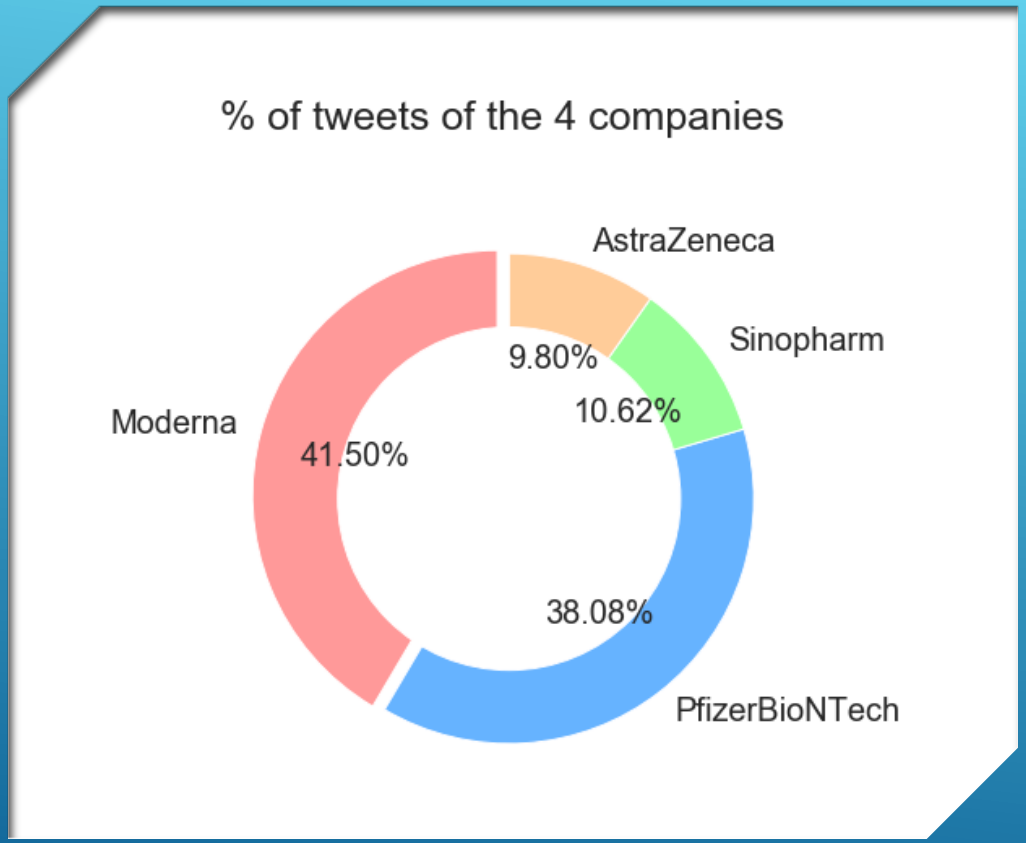
Matplotlib and **seaborn** are used for visualization of the findings.



Findings

Of all the 4 companies investigated, **Moderna** got 41.99% of positive sentiments.

Apparently **Moderna** is a winner regarding positive sentiments.



Findings

Of all the 4 companies investigated, **Moderna** got 41.50% of tweets.

Apparently **Moderna** is a winner regarding the popularity.

Limitations

The dataset is regularly updated, so the analysis can only reflect the sentiment trends I downloaded the dataset. The trends might change over times.

Conclusions

Moderna vaccine was the most popular vaccine been talked about, and it also had most positive sentiments than others.

Acknowledgements

I got the dataset form Kaggle.com, and Ariwan Sri Setya's notebook shedded great inspiration to my analysis.

References

I used Ariwan Sri Setya's notebook on Kaggle.com to help me with my analysis.

Jupyter notebook

Sentiment Analysis for COVID-19 Vaccines Tweets

In [362]:
```
import numpy as np
import pandas as pd
import seaborn as sns
import re
import matplotlib.pyplot as plt
from textblob import TextBlob
```

In [363]:
```
data = pd.read_csv('vaccination_all_tweets.csv')
```

In [364]:
```
data.head(5)
```

Out[364]:

on	user_description	user_created	user_followers	user_friends	user_favourites	user_verified	date	text	hashtags	source
a- :A	Aggregator of Asian American news; scanning di...	2009-04-08 17:52:46	405	1692	3247	False	2020-12-20 06:06:44	Same folks said daikon paste could treat a cyt...	['PfizerBioNTech']	Twitter for Android
an :A	Marketing dude, tech geek, heavy metal & '80s ...	2009-09-21 15:27:30	834	666	178	False	2020-12-13 16:27:13	While the world has been on the wrong side of ...	NaN	Twitter Web App
ed	heil, hydra 🖐 ☺	2020-06-25 23:30:28	10	88	155	False	2020-12-12 20:33:45	#coronavirus #SputnikV #AstraZeneca #PfizerBio...	['coronavirus', 'SputnikV', 'AstraZeneca', 'Pf...	Twitter for Android
er, da	Hosting "CharlesAdlerTonight" Global News Radi...	2008-09-10 11:28:53	49165	3933	21853	True	2020-12-12 20:23:59	Facts are immutable, Senator, even when you're...	NaN	Twitter Web App
								Explain to me		

In [366]:
```
data.columns
```

Out[366]:
```
Index(['id', 'user_name', 'user_location', 'user_description', 'user_created',
       'user_followers', 'user_friends', 'user_favourites', 'user_verified',
       'date', 'text', 'hashtags', 'source', 'retweets', 'favorites',
       'is_retweet'],
      dtype='object')
```

is there null data?

In [367]:
```
data.isnull().sum()
```

Out[367]:
```
id                    0
user_name             0
user_location      6162
user_description   1870
user_created          0
user_followers        0
user_friends          0
user_favourites       0
user_verified         0
date                  0
text                  0
hashtags           5590
```

clean null data

```
In [368]: data=data.dropna()

In [369]: data.shape
Out[369]: (15433, 16)

In [370]: len(data['text'].unique())
Out[370]: 15423
```

remove duplicates

```
In [371]: data = data.drop_duplicates('text')

In [372]: data.shape
Out[372]: (15423, 16)
```

15433-15423=10, so 10 duplicate rows were deleted

clean text data

In [373]:
```
def clean_data(text):
    text = re.sub(r'@\w+', '', text)
    text = re.sub(r'#', '', text)
    text = re.sub(r'RT[\s]+', '', text)
    text = re.sub(r'https?:\/\/\S+', '', text)
    text = text.lower()

    return text
```

In [374]:
```
data['text'] = data['text'].apply(clean_data)
```

Create funtions to get sentiment

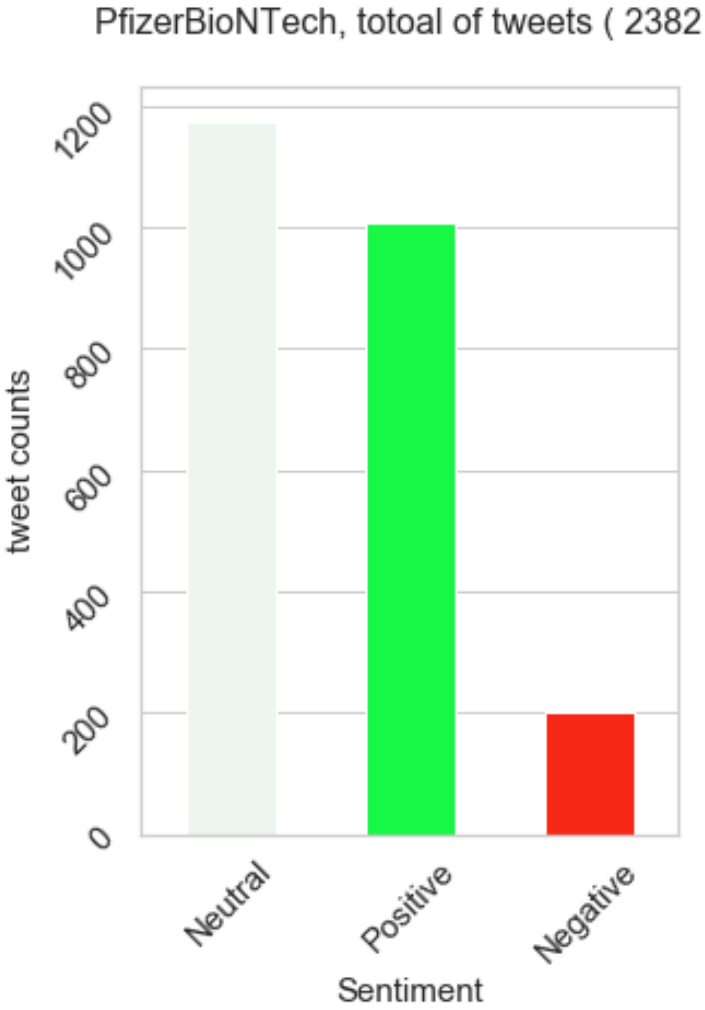
In [375]:
```
def get_polarity(text):
    return TextBlob(text).sentiment.polarity

def get_sentiment(score):
    if score > 0:
        return 'Positive'
    elif score == 0:
        return 'Neutral'
    else:
        return 'Negative'
```

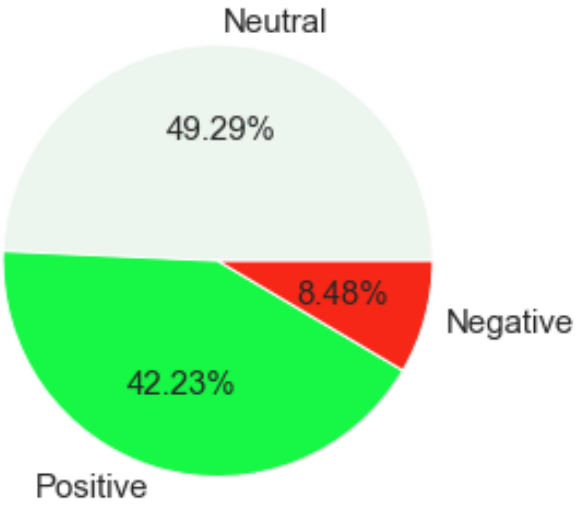
In [376]:
```
# apply different colors to different sentiment.
sentiment_color=['#Ecf5ee','#17f746','#F72717']
```

1. filter out tweets about PfizerBioNTech and get it's sentiment

In [388]:

```
PfizerBioNTech_filter = data['hashtags'].str.contains('PfizerBioNTech')
PfizerBioNTech_data = data[PfizerBioNTech_filter]

PfizerBioNTech_data['polarity'] = PfizerBioNTech_data['text'].apply(get_polarity)
PfizerBioNTech_data['sentiment'] = PfizerBioNTech_data['polarity'].apply(get_sentiment)
PfizerBioNTech_data["company"] = "PfizerBioNTech"

plt.subplot(1, 2, 1)
PfizerBioNTech_data['sentiment'].value_counts().plot.bar(color=sentiment_color);
plt.title(f"PfizerBioNTech, totoal of tweets ( {len(PfizerBioNTech_data)} )\n", fontsize = 16)
plt.xlabel("Sentiment",fontsize=15);
plt.ylabel("tweet counts",fontsize =15);
plt.xticks(fontsize=15, rotation=45)
plt.yticks(fontsize=15, rotation=45)
plt.grid(axis='x')

P_sentiment = PfizerBioNTech_data['sentiment'].value_counts()
P_sentiment_list=list(P_sentiment.index)

plt.subplot(1, 2, 2)
plt.pie(P_sentiment, pctdistance=0.6,labeldistance=1.1,
                 colors=sentiment_color,labels=P_sentiment_list,autopct='%1.2f%%',textprops={'fontsize': 15})

plt.show()
```

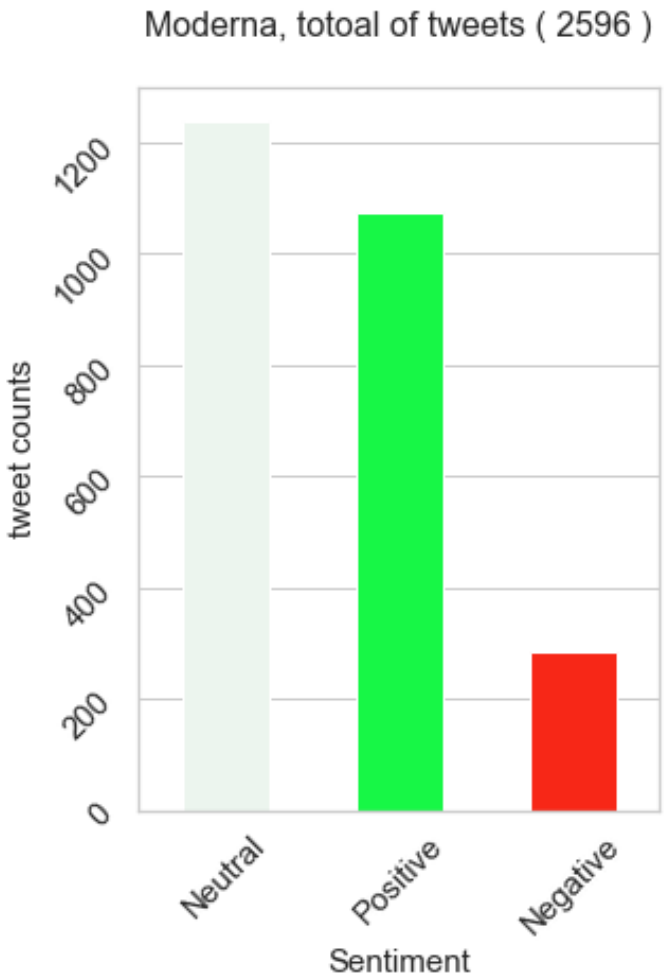

PfizerBioNTech, totoal of tweets (2382)
tweet counts
0
200
400
600
800
1000
1200
Neutral
Positive
Negative
Sentiment

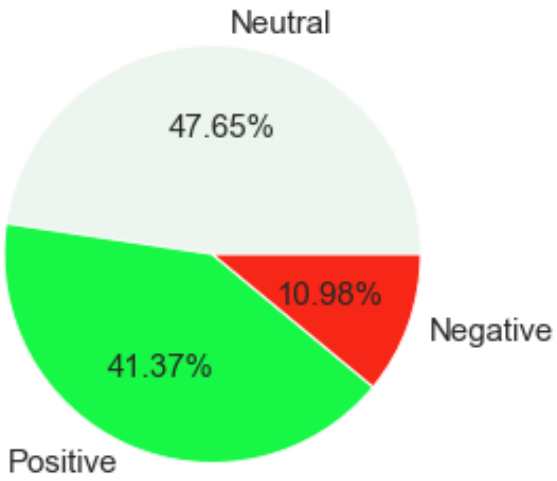

Neutral
49.29%
8.48%
Negative
42.23%
Positive

2. filter out tweets about Moderna and get it's sentiment

In [378]:

```
Moderna_filter = data['hashtags'].str.contains('Moderna')
Moderna_data = data[Moderna_filter]

Moderna_data['polarity'] = Moderna_data['text'].apply(get_polarity)
Moderna_data['sentiment'] = Moderna_data['polarity'].apply(get_sentiment)
Moderna_data["company"] = "Moderna"

plt.subplot(1, 2, 1)
Moderna_data['sentiment'].value_counts().plot.bar(color=sentiment_color);
plt.title(f"Moderna, totoal of tweets ( {len(Moderna_data)} )\n", fontsize = 16)
plt.xlabel("Sentiment",fontsize=15);
plt.ylabel("tweet counts",fontsize =15);
plt.xticks(fontsize=15, rotation=45)
plt.yticks(fontsize=15, rotation=45)
plt.grid(axis='x')

M_sentiment = Moderna_data['sentiment'].value_counts()
M_sentiment_list=list(M_sentiment.index)

plt.subplot(1, 2, 2)
plt.pie(M_sentiment, pctdistance=0.6,labeldistance=1.1,
                 colors=sentiment_color,labels=M_sentiment_list,autopct='%1.2f%%',textprops={'fontsize': 15})

plt.show()
```


Moderna, totoal of tweets (2596)
tweet counts
0
200
400
600
800
1000
1200
Neutral
Positive
Negative
Sentiment

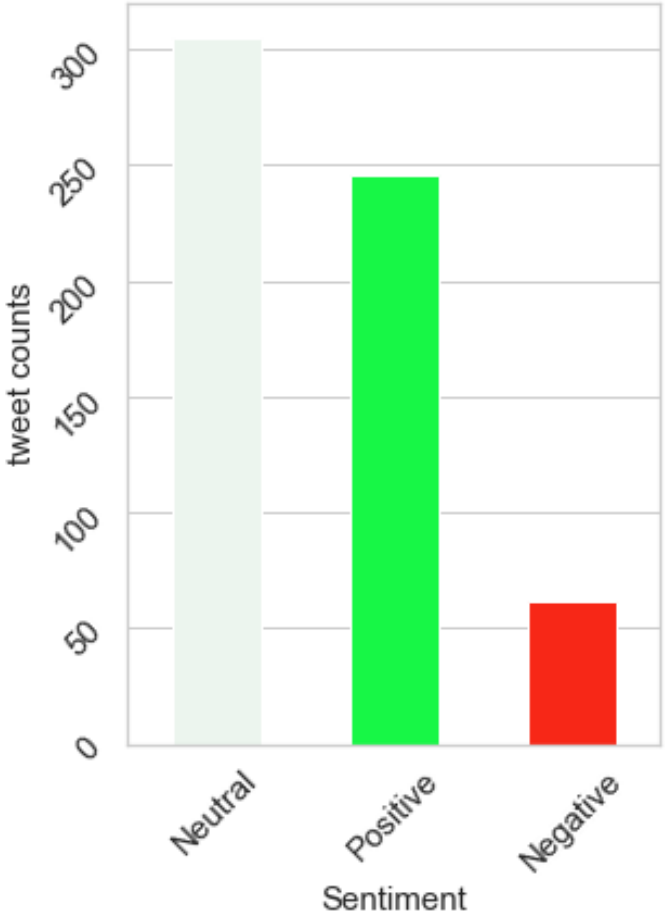

Neutral
47.65%
10.98%
Negative
41.37%
Positive

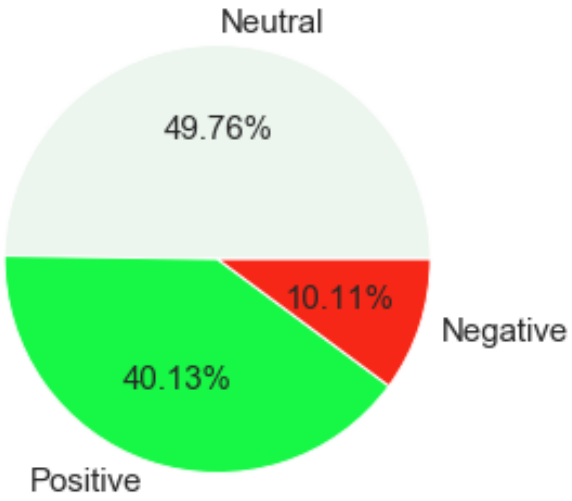
3. filter out tweets about AstraZeneca and get it's sentiment

In [379]:

```
AstraZeneca_filter = data['hashtags'].str.contains('AstraZeneca')
AstraZeneca_data = data[AstraZeneca_filter]

AstraZeneca_data['polarity'] = AstraZeneca_data['text'].apply(get_polarity)
AstraZeneca_data['sentiment'] = AstraZeneca_data['polarity'].apply(get_sentiment)
AstraZeneca_data["company"] = "AstraZeneca"

plt.subplot(1, 2, 1)
AstraZeneca_data['sentiment'].value_counts().plot.bar(color=sentiment_color);
plt.title(f"AstraZeneca, totoal of tweets ( {len(AstraZeneca_data)} )\n",fontsize = 16)
plt.xlabel("Sentiment",fontsize=15);
plt.ylabel("tweet counts",fontsize =15);
plt.xticks(fontsize=15, rotation=45)
plt.yticks(fontsize=15, rotation=45)
plt.grid(axis='x')

A_sentiment = AstraZeneca_data['sentiment'].value_counts()
A_sentiment_list=list(A_sentiment.index)

plt.subplot(1, 2, 2)
plt.pie(A_sentiment, pctdistance=0.6,labeldistance=1.1,
                 colors=sentiment_color,labels=A_sentiment_list,autopct='%1.2f%%',textprops={'fontsize': 15})

plt.show()
```

AstraZeneca, totoal of tweets (613)

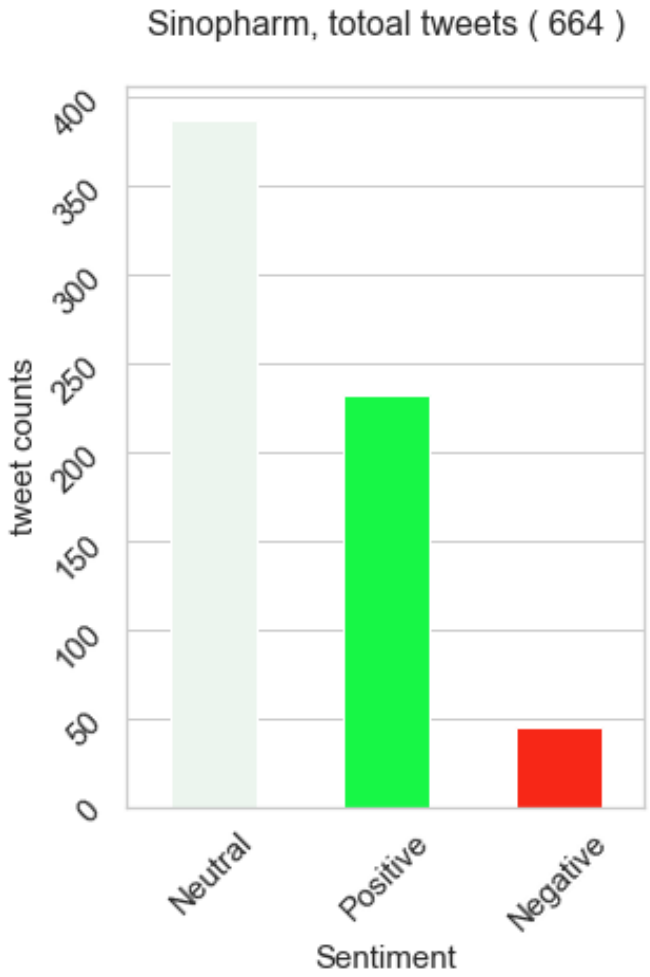


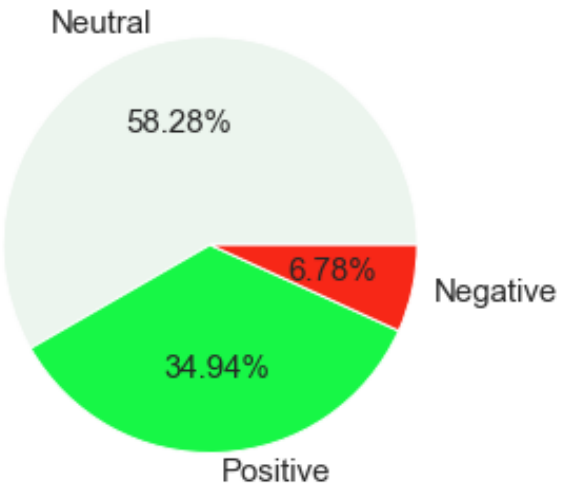


4. filter out tweets about Sinopharm and get it's sentiment

In [380]:

```
Sinopharm_filter = data['hashtags'].str.contains('Sinopharm')
Sinopharm_data = data[Sinopharm_filter]

Sinopharm_data['polarity'] = Sinopharm_data['text'].apply(get_polarity)
Sinopharm_data['sentiment'] = Sinopharm_data['polarity'].apply(get_sentiment)
Sinopharm_data["company"] = "Sinopharm"

plt.subplot(1, 2, 1)
Sinopharm_data['sentiment'].value_counts().plot.bar(color=sentiment_color);
plt.title(f"Sinopharm, totoal tweets ( {len(Sinopharm_data)} )\n",fontsize=16)
plt.xlabel("Sentiment",fontsize=15);
plt.ylabel("tweet counts",fontsize =15);
plt.xticks(fontsize=15, rotation=45)
plt.yticks(fontsize=15, rotation=45)
plt.grid(axis='x')

S_sentiment = Sinopharm_data['sentiment'].value_counts()
S_sentiment_list=list(S_sentiment.index)

plt.subplot(1, 2, 2)
plt.pie(S_sentiment, pctdistance=0.6,labeldistance=1.1,
                 colors=sentiment_color,labels=S_sentiment_list,autopct='%1.2f%%',textprops={'fontsize': 15})
#plt.legend(title = "Four company:")

plt.show()
```


Sinopharm, totoal tweets (664)
tweet counts
0
50
100
150
200
250
300
350
400
Neutral
Positive
Negative
Sentiment


Neutral
58.28%
6.78%
Negative
34.94%
Positive

now , put them together for comparison purpose

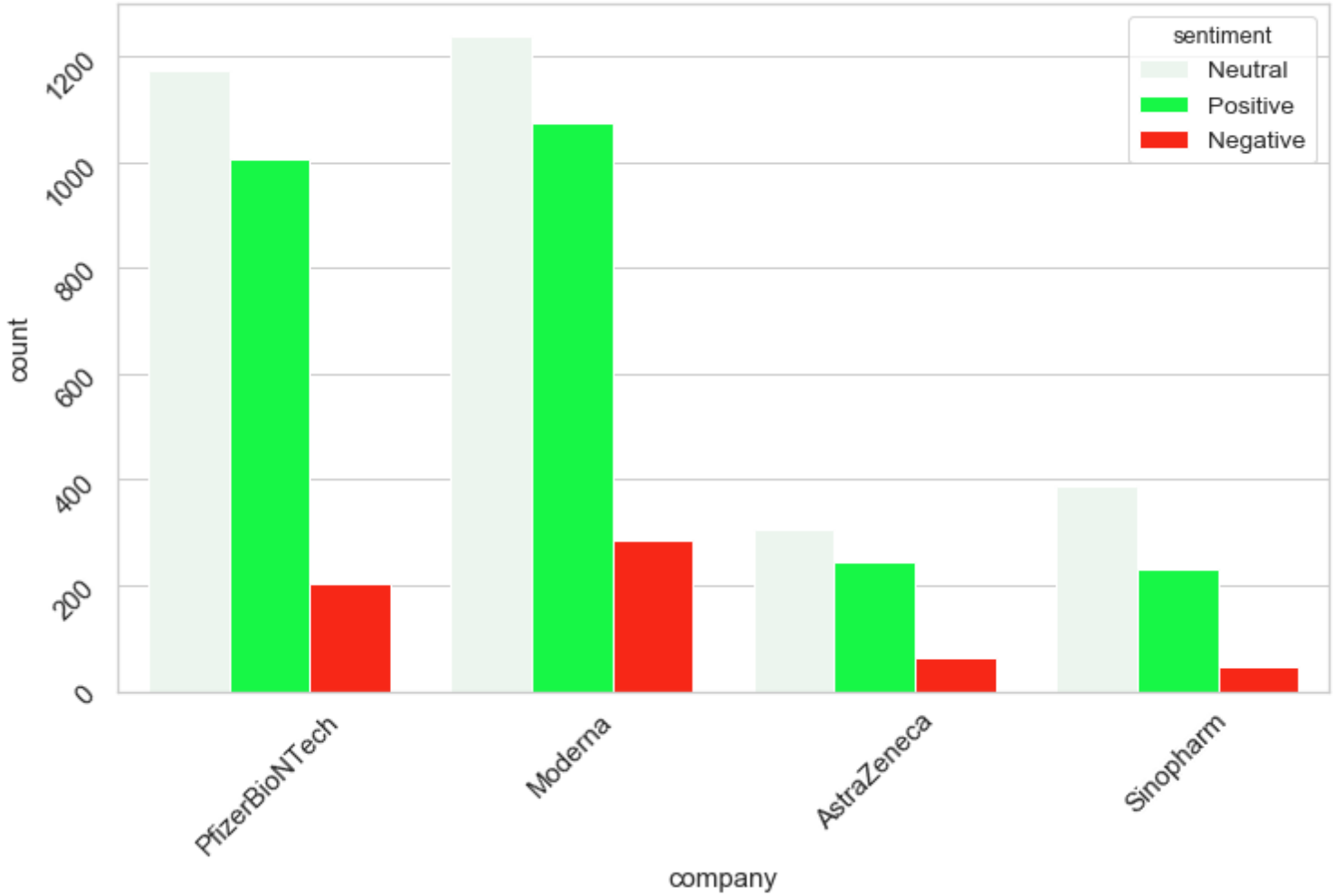
In [381]:
```
union1 = pd.concat([PfizerBioNTech_data, Moderna_data], ignore_index=True, sort=False)

union2 = pd.concat([AstraZeneca_data,Sinopharm_data], ignore_index=True,sort=False)

combined_data = pd.concat([union1,union2], ignore_index=True,sort=False)
```

In [384]:
```
len(combined_data)
```

Out[384]: 6255

plot all companies and their sentiments together

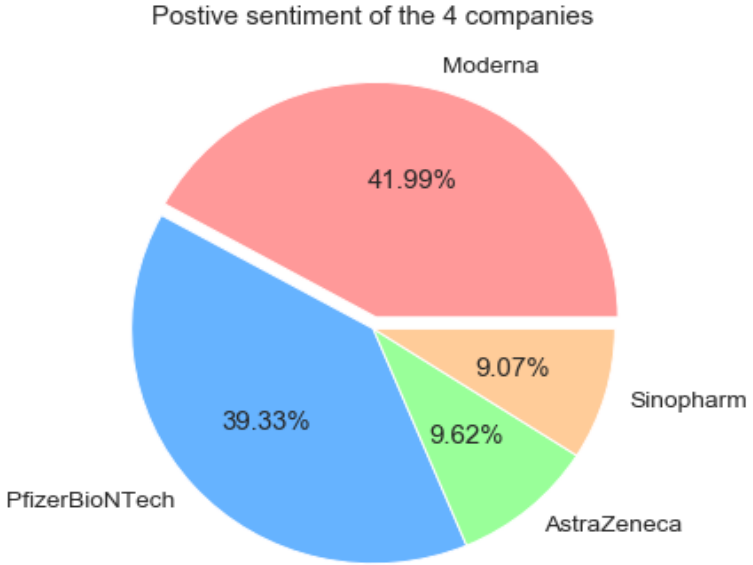
In [385]:
```
sns.set_style("whitegrid")
plt.figure(figsize=(12, 7))
#sns.set(font_scale=1.25)
sns.countplot(x='company',hue='sentiment',data=combined_data,palette=sentiment_color,saturation=9)
plt.xticks(fontsize=15, rotation=45)
plt.yticks(fontsize=15, rotation=45)
```


sentiment
Neutral
Positive
Negative
count
0
200
400
600
800
1000
1200
PfizerBioNTech
Moderna
AstraZeneca
Sinopharm
company

show positive review of 4 companies in a pie chart

In [398]:

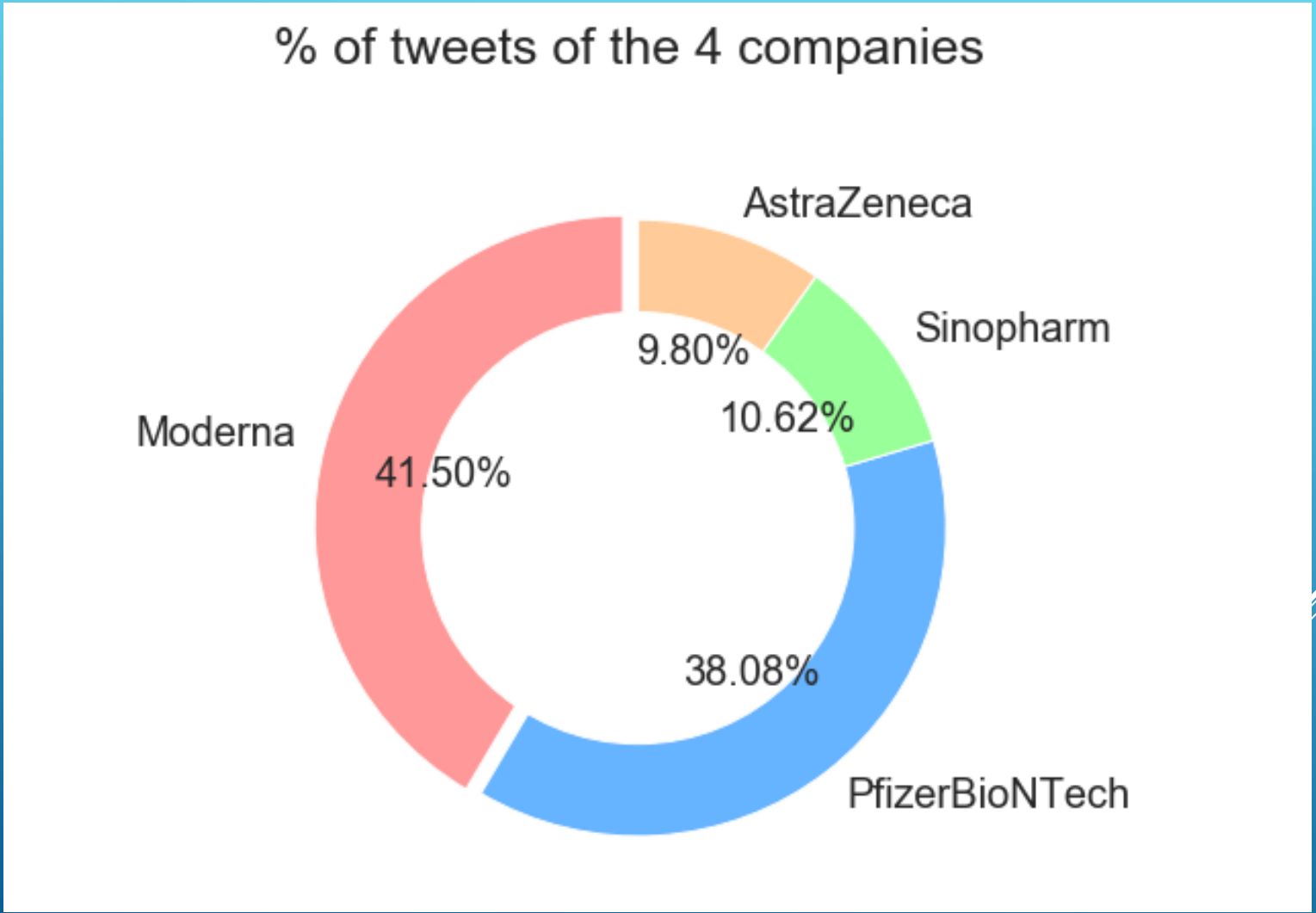
```
postive_sentiment_filter = combined_data['sentiment'] == "Positive"
postive_review = combined_data[postive_sentiment_filter]

company = postive_review['company'].value_counts()
company_list = list(company.index)
colors = ['#ff9999','#66b3ff','#99ff99','#ffcc99']
myexplode = [0.05, 0, 0, 0]
plt.pie(company, pctdistance=0.6,labeldistance=1.1,
                colors=colors,labels=company_list,autopct='%1.2f%%', explode=myexplode)
plt.title('Postive sentiment of the 4 companies')
plt.show()
```



show % of tweets of the 4 companies

In [387]:

```
#colors
colors = ['#ff9999','#66b3ff','#99ff99','#ffcc99']
myexplode = [0.05, 0, 0, 0]
fig1, ax1 = plt.subplots()
ax1.pie(company, colors = colors, labels=company_list, autopct='%1.2f%%', startangle=90,explode=myexplode,textprops={'fontsiz

#draw circle
centre_circle = plt.Circle((0,0),0.70,fc='white')
fig = plt.gcf()
fig.gca().add_artist(centre_circle)
# Equal aspect ratio ensures that pie is drawn as a circle
ax1.axis('equal')
ax1.set_title('% of tweets of the 4 companies\n\n',fontsize= 25)
plt.tight_layout()

plt.show()
```


% of tweets of the 4 companies
AstraZeneca
9.80%
Sinopharm
10.62%
Moderna
41.50%
38.08%
PfizerBioNTech