In [ ]:

```
###necessary libraries
import pandas as pd
import glob
import os
from datetime import datetime, timezone
import re
import numpy as np
import itertools
from functools import reduce

# file where csv files of flair analysis lies
path_flair = r'C:\Users\victo\Master_Thesis\semanticanalysis\analysis_with_flair\audi\outcome_using_flair'
all_files_flair = glob.glob(os.path.join(path_flair, "*.csv"))

# read files to pandas frame
list_of_files_flair = []

for filename in all_files_flair:
    list_of_files_flair.append(pd.read_csv(filename,
                                           sep=',',
                                           )
                               )

# Concatenate all content of files into one DataFrames
concatenate_list_of_files_flair = pd.concat(list_of_files_flair,
                                            ignore_index=True,
                                            axis=0,
                                            )

# removing duplicates
cleaned_dataframe_flair = concatenate_list_of_files_flair.sort_values(by='url', ascending=False)
cleaned_dataframe_flair = cleaned_dataframe_flair.drop_duplicates(subset=["url"], keep='first', ignore_index=True)

#print(cleaned_dataframe_flair)

# file where csv files of vader analysis lies
path_vader = r'C:\Users\victo\Master_Thesis\semanticanalysis\analysis_with_vader\audi\outcome_using_vader'
all_files_vader = glob.glob(os.path.join(path_vader, "*.csv"))

# read files to pandas frame
list_of_files_vader = []

for filename in all_files_vader:
    list_of_files_vader.append(pd.read_csv(filename,
                                           sep=',',
                                           )
                               )

# Concatenate all content of files into one DataFrames
concatenate_list_of_files_vader = pd.concat(list_of_files_vader,
                                            ignore_index=True,
                                            axis=0,
                                            )

# removing duplicates
cleaned_dataframe_vader = concatenate_list_of_files_vader.sort_values(by='url', ascending=False)
cleaned_dataframe_vader = cleaned_dataframe_vader.drop_duplicates(subset=["url"], k
eep='first', ignore_index=True)

#print(cleaned_dataframe_vader)

# file where csv files of textblob analysis lies
path_textblob = r'C:\Users\victo\Master_Thesis\semanticanalysis\analysis_with_textb
lob\audi\outcome_using_texblob'
all_files_textblob = glob.glob(os.path.join(path_textblob, "*.csv"))

# read files to pandas frame
list_of_files_textblob = []

for filename in all_files_textblob:
    list_of_files_textblob.append(pd.read_csv(filename,
                                              sep=',',
                                              )
                                  )

# Concatenate all content of files into one DataFrames
concatenate_list_of_files_textblob = pd.concat(list_of_files_textblob,
                                               ignore_index=True,
                                               axis=0,
                                               )

# removing duplicates
cleaned_dataframe_textblob = concatenate_list_of_files_textblob.sort_values(by='url
', ascending=False)
cleaned_dataframe_textblob = cleaned_dataframe_textblob.drop_duplicates(subset=["ur
l"], keep='first', ignore_index=True)

#print(cleaned_dataframe_textblob)

##merging files together
merged_df = pd.merge(cleaned_dataframe_flair, cleaned_dataframe_vader, on=['url'])
#,'header','release time','article content','formatted date'])
merged_df = pd.merge(merged_df, cleaned_dataframe_textblob, on=['url'])#,'header','
release time','article content','formatted date'])
merged_df['formatted date'] = pd.to_datetime(merged_df['formatted date'])
merged_df.rename(columns={'formatted date': 'formatteddate'}, inplace=True)

#importing of stock price files
path_stockprices = r'C:\Users\victo\Master_Thesis\stockprice_data\audi\daily_stock_
pricefiles_with_return'

##filling empty cells with 0
#merged_df[['flair_sentiment_header_score', 'flair_sentiment_content_score', 'neg_v
ader_header', 'neu_vader_header',
#           'pos_vader_header', 'compound_vader_header', 'neg_vader_articel_content
', 'neu_vader_articel_content',
#           'pos_vader_articel_content', 'compound_vader_articel_content', 'polarit
y_textblob_sentiment_header',
#           'subjectivity_textblob_sentiment_header', 'polarity_textblob_sentiment_
content',
#           'subjectivity_textblob_sentiment_content'
#           ]] = merged_df[['flair_sentiment_header_score', 'flair_sentiment_conten
t_score', 'neg_vader_header', 'neu_vader_header',
#                           'pos_vader_header', 'compound_vader_header', 'neg_vader
_articel_content',
#                           'neu_vader_articel_content', 'pos_vader_articel_content
', 'compound_vader_articel_content',
#                           'polarity_textblob_sentiment_header', 'subjectivity_tex
tblob_sentiment_header',
#                           'polarity_textblob_sentiment_content', 'subjectivity_te
xtblob_sentiment_content'
```

```
#                            ]].fillna(0)

#creating new column with formatted date
dates = []
for date in merged_df['formatteddate']:
    matches = re.search('\d{4}-\d{2}-\d{2}', str(date))
    date_merged = matches.group()
    dates.append(date_merged)

merged_df['Date'] = dates
#print(merged_df.Date)

# new dataframe for merging later with stockprices
dates_merger = []
flair_sentiment_header_score = []
flair_sentiment_content_score = []
compound_vader_header = []
compound_vader_articel_content = []
polarity_textblob_sentiment_header = []
polarity_textblob_sentiment_content = []

for dates in merged_df['formatteddate']:
    matches2 = re.search('\d{4}-\d{2}-\d{2}', str(dates))
    date_merged2 = matches2.group()
    for index, row in merged_df.iterrows():
        if row['Date'] == date_merged2:
            dates_merger.append(row['Date'])
            #print(row['Date'])
            flair_sentiment_header_score.append(row['flair_sentiment_header_score
'])
            #print(row['flair_sentiment_header_score'])
            flair_sentiment_content_score.append(row['flair_sentiment_content_score
'])
            #print(row['flair_sentiment_content_score'])
            compound_vader_header.append(row['compound_vader_header'])
            #print(row['compound_vader_header'])
            compound_vader_articel_content.append(row['compound_vader_articel_conte
nt'])
            #print(row['compound_vader_articel_content'])
            polarity_textblob_sentiment_header.append(row['polarity_textblob_sentim
ent_header'])
            #print(row['polarity_textblob_sentiment_header'])
            polarity_textblob_sentiment_content.append(row['polarity_textblob_senti
ment_content'])
            #print(row['polarity_textblob_sentiment_content'])

merge_list = list(zip(dates_merger,
                      flair_sentiment_header_score,
                      flair_sentiment_content_score,
                      compound_vader_header,
                      compound_vader_articel_content,
                      polarity_textblob_sentiment_header,
                      polarity_textblob_sentiment_content))

new_merged_df = pd.DataFrame(data=merge_list,
                             columns=['Date',
                                      'flair_sentiment_header_score',
                                      'flair_sentiment_content_score',
                                      'compound_vader_header',
                                      'compound_vader_articel_content',
                                      'polarity_textblob_sentiment_header',
                                      'polarity_textblob_sentiment_content']
                             )
```

```
#new_merged_df['Date'] = pd.to_datetime(new_merged_df['Date'])

new_merged_df = new_merged_df.groupby('Date').mean()

print(new_merged_df)

for file in glob.iglob(path_stockprices + '\*.csv'):
    df_daily_stock_prices = pd.read_csv(file,
                                        sep=',',
                                        )
    #df_daily_stock_prices['Date'] = pd.to_datetime(df_daily_stock_prices['Date'])
    print(df_daily_stock_prices)
    new_df_daily_stockprices = df_daily_stock_prices.merge(new_merged_df,
                                                           left_on='Date',
                                                           right_on='Date',
                                                           how='left')

    new_df_daily_stockprices.to_csv(r'C:\Users\victo\Master_Thesis\merging_data\audi\daily\merged_files\daily_audiprices_with_semantics.csv', index=False)
    print('File has been saved!')
```