Problem: Using the content classification text result from Paul, I tried to match it with the tweet in the rachel_new_chess_community. It shows that there should be over 90,000 tweets being classified, however, when I extract the tweet and the corresponding ID from the original JSON file to merge with the classification, only about 4,000 tweets are being identified successfully.

```
# View the dataframe
print(content_data)
```

Description: df [99,859 × 2]

id <chr>	content <chr>
1627714970539462700	Chess Literature and Media
1625510551081169000	Chess Literature and Media
1626260865056968700	Chess Literature and Media
1625176847007858700	Chess Literature and Media
1624091169390047200	Chess Literature and Media
1623373387484594200	Chess Literature and Media
1622868023781597200	Chess Literature and Media
1622158253890093000	Chess Literature and Media
1621531657688813600	Chess Training and Improvement
1620890041512861700	Chess Literature and Media

```
print(all_tweet)
```

Description: df [3,968 × 7]

created_at <date>	retweet_user_id <chr>	type <chr>	content <chr>	rank <dbl>
2020-07-04	480444935	retweet_in	Social Media Impact on Chess	50
2020-07-17	23612012	retweet_in	Chess Legends and Personalities	2
2020-07-18	480444935	retweet_in	Chess Literature and Media	50
2020-07-18	4369711156	retweet_in	Chess Literature and Media	26
2020-08-04	4369711156	retweet_in	Chess Legends and Personalities	26
2020-08-11	617004214	retweet_in	Social Media Impact on Chess	43
2020-08-26	186797066	retweet_in	Chess Events and Tournaments	18
2020-08-26	29521967	retweet_in	Chess Legends and Personalities	30
2020-08-31	1651411087	retweet_in	Social Media Impact on Chess	22
2020-09-02	301042394	retweet_in	Online Chess Competitions	19

Steps:

1. For those tweets, identify the influencer in 2 ways:

- Users with the rank of less than 10
- Manually chosen the 10 influencers in the previous project

2. Generate tables that contain the tweets created by influencers and consumers with the content type (e.g. like the table below)

A	B	C	D
created_at	content	user_type	count
2020-06-29	Chess Events and Tournaments	consumer	1
2020-06-29	Political and Social Issues in Chess	consumer	1
2020-06-30	Chess Events and Tournaments	consumer	2
2020-07-01	Chess Literature and Media	consumer	1
2020-07-01	Social Media Impact on Chess	consumer	1
2020-07-02	Chess Training and Improvement	influencer	1
2020-07-02	Social Media Impact on Chess	consumer	2

3. Transform the table into 3 columns: the date, number of tweets by consumers, number of tweets by influencers. I have selected the top 2 classified content types: Chess Events and Tournaments & Chess Training and Improvement. Here is the sum of tweets in those 2 contents.
    - Problem: there are some dates that have 0 tweets generated by consumer nor influencer. Should we just not include such dates in the time series, or should we fill out with 0, or should we smooth the line to have a reasonable guess on such dates (like impute missing values with linear interpolation)? Here I just ignore such dates, to only use those dates with non-sparse date to form the time series.
    -

A	B	C
created_at	**consumer**	**influencer**
2020-06-29	1	0
2020-06-30	2	0
2020-07-02	0	1
2020-07-03	1	0
2020-07-10	1	0
2020-07-11	1	0
2020-07-12	0	1
2020-07-13	0	1

4. Create time series on the dataset and perform the Granger test. The result is not so good since I got p-values larger than 0.05 for every time lag I tested (lag from 1 to 10)

```
grangertest(influencer_ts ~ consumer_ts, order = 1)
```

```
Granger causality test

Model 1: influencer_ts ~ Lags(influencer_ts, 1:1) + Lags(consumer_ts, 1:1)
Model 2: influencer_ts ~ Lags(influencer_ts, 1:1)
  Res.Df Df      F Pr(>F)
1    551
2    552 -1 1.1941  0.275
```

Improvement: Here the first experiment I did is for all the influencers with top 10 rank aggregated. I haven't tested the result for a single influencer yet. Also, I can do the same experiment for the other influencer identification (manually selected the influencer id in the previous project). The missing values in the time series might also affect the results.

Experienment 2:

Select the top 10 content types with the supply from the top 2 core agents. (by rank). The Result is not good. Every p value (1-10 lag) is larger than 0.05.

Experiment 3:

Select single core agent (the top core agent manually selected by user id) with the top 10 content types. The relationship is not significant: every p value is larger than 0.05.

Experienment 4:

Select top 10 rank as an influencer, get the aggregated tweet counts for consumer ang influencers. Get the panel data with index: content type, the best result was lag = 7 (p=0.14).

Experiment 5:

Select 2 influencers (rank at 2 and 4), and get the top 10 content types tweet data. The result shows at lag 2 the causality is significant but in other lags the p values are larger than 0.05.

```
grangertest(influencer_count ~ consumer_count, data = df.pd, order = 2L)
```

```
Granger causality test

Model 1: influencer_count ~ Lags(influencer_count, 1:2) + Lags(consumer_count, 1:2)
Model 2: influencer_count ~ Lags(influencer_count, 1:2)
  Res.Df Df      F  Pr(>F)
1   2778
2   2780 -2 3.0856 0.04586 *
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```