**Wrangle data report**

**By : Irfan Septiyana Putra**

In this project, I am doing data wrangling step on data from WeRateDogs twitter archive. All step of data wrangling such as gathering data, asses data and cleaning data is done. In this project gathering data is done by downloading data from udacity, from url and twitter api. After assessing data, found 9 quality issues and 2 issues in tidiness which will explain below. Each issue is cleaned with three steps : define problem, code and test. Those steps are really help data cleaning process since each issue is cleaned and tested one by one.

Gathering data is the first process in data wrangling. The first one ,tweeter archive data, I get by downloading it in udacity. I need to make request to url using get method in request library and then store it into image-prediction.tsv file. The final data is gotten from twitter api by helping from tweepy library. To get data from tweeter, authentication process is needed befor pulling data from twitter. Then, requesting data is done by requesting data with tweet\_id (from first data) and this process needs timer to make the pull request is not passing tweeter api limit request. This data stored as tweet\_json.txt.

The next step is asses data. First step is read stored data and make it on dataframe. Tweet archive and image prediction data is read using read\_csv method. I need to get data line per line using json load and store it to list of dictionaries, exracting tweet\_id, text, retweeted\_count and favorite count and change it into dataframe.

**Quality Issues Assesment**

First assesed is display each data head, info and describe. Gotten first issue in tweeter archive .Name column is floofer, this must be replace with floof.

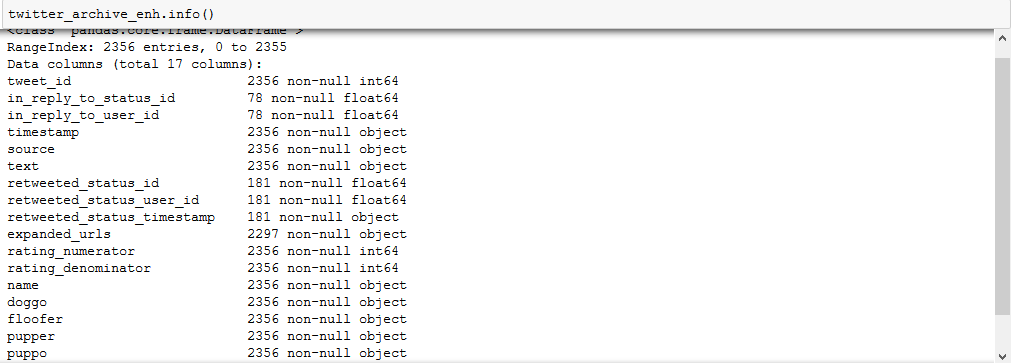


Figure 1 twitter archive info

Seeing twitter archive info, gotten timestamp in string type, must be indatetime type. in\_reply\_to\_status\_id and in\_reply\_to\_user\_id must be in integer instead of float. Checking retweet\_status\_id, gotten several retweet data or duplication data which is not desireable, fourth issue.

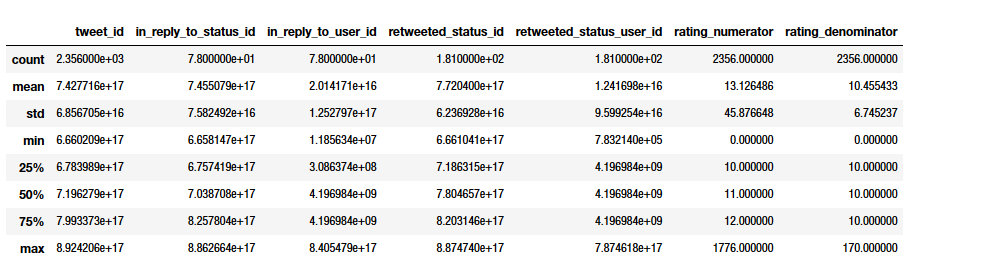


Figure 2 tweeter archive describe result

From statistic data above, it’s strange that rating numerator and denominator have big max number which is very far from mean and thirt quartile, adding additional column needed.

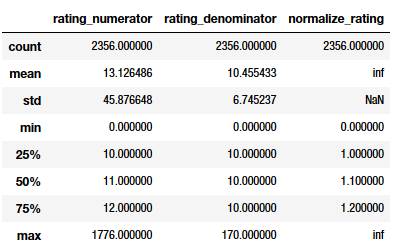


Figure 3 twiiter archive with normalize column

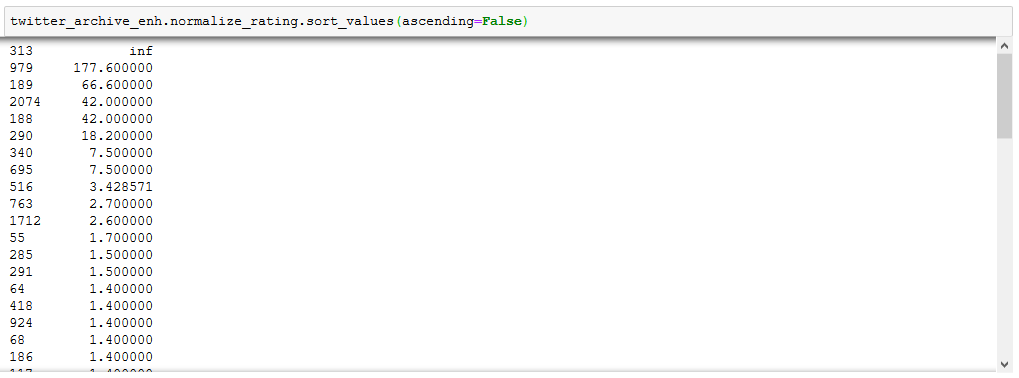
s

Figure 4 sort values normalize\_rating

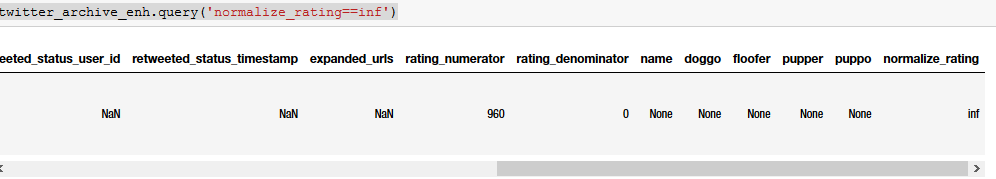


Figure 5 tweeter archive inf value row



Figure 6 inf value rows, text content

From four pictures about, concluded rating\_denominator and numerator is wrong.

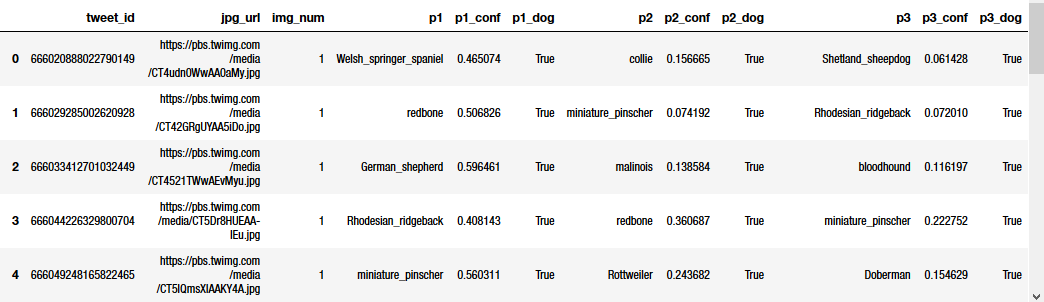


Figure 7 image prediction info

Inconsistent of using lower and uppercase and using ‘\_’ and ‘-’ as separator make dog prediction stage is not clear. Using value\_counts method, found several duplication data / same jpg\_url.

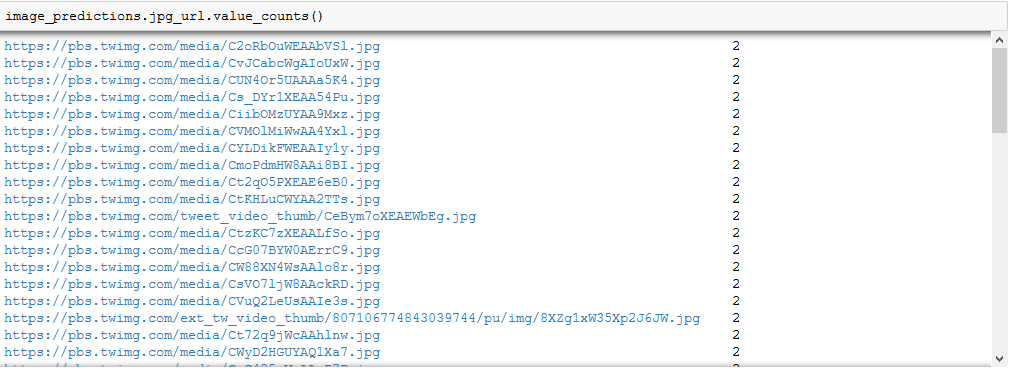


Figure 8 value\_counts result of jpg\_url column in image prediction

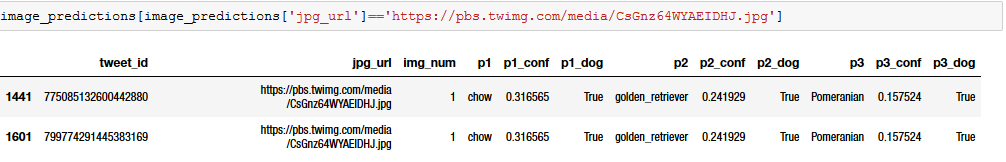


Figure 9 Duplication data example in image prediction

Analyze json data, duplication data found by extracting text that having ‘RT @’

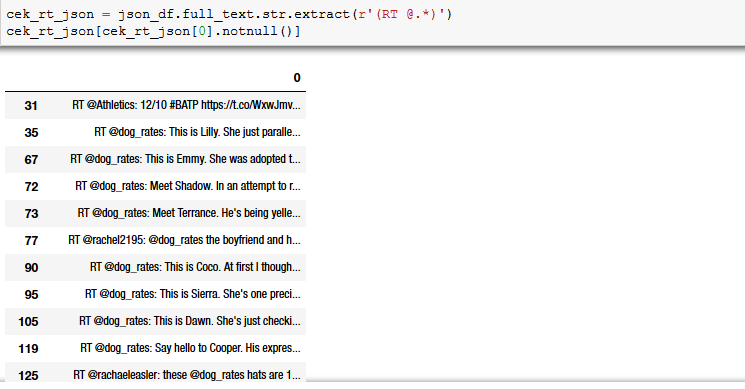


Figure 10 duplication data by extracting text

**Tidiness Data**

From figure 1, timestamp contain date and time. This makes data untidy, data must be separated.

Text contain both tweeter content and url, this need to be fixed too.

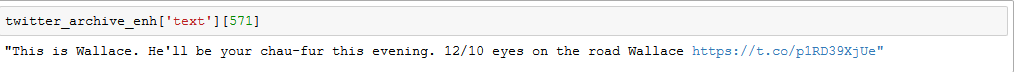


Figure 11 Content value example in tweeter archive

Cleaning data process will be explained below in per issue.

**Quality issues**

1. **timestamp and retweeted status timestamp has type string, which will good in date type**

Fixed using astype method for each column.



Figure 12 code change astype fro 2 columns

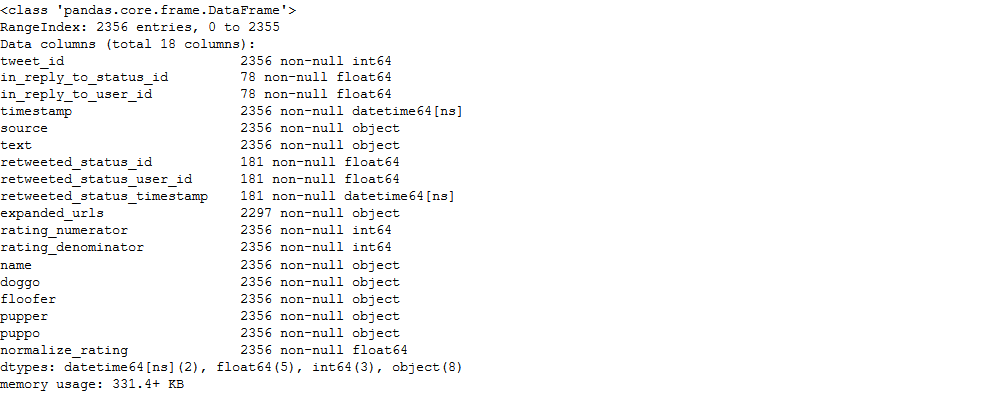


Figure 13 Test result, timestamp and retweet status timestamp type changed into datetime

1. **wrong column name, floofer must be floof**



Figure 14 changing column name

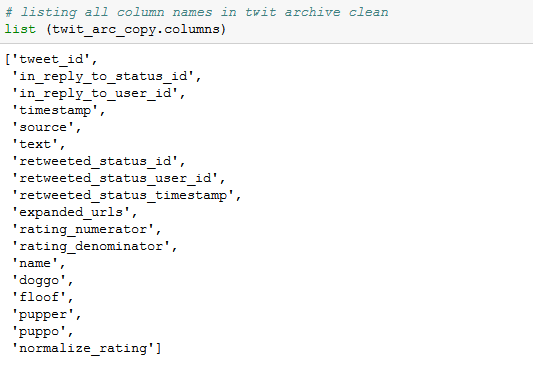
Changing column name using syntax above and recheck it by listing column name.

Figure 15 tweeter archive column name listing

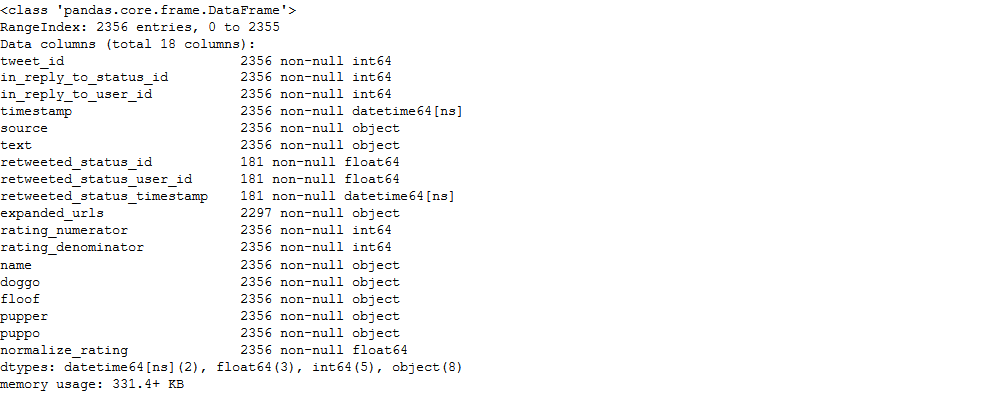
1. **in\_reply\_to\_status\_id and in\_reply\_to\_user\_id type is in float, but it must be in integer**



Figure 16 change type using astype

Test the result using info method, resulting in\_reply\_to\_status\_id and in\_reply\_to\_user\_id in

Figure 17 Info result tweeter archive

 Integer.

1. **rating\_numerator and rating\_denominator in row 313 value on normalize\_rating is 13/10 instead of 960/0**





Figure 18 Change rating nominator and denumerator value in row 313, and recalculate retweet favorite ratio

Directly change value in row 313 and checking the result using describe.Now there are no inf value in the last column.

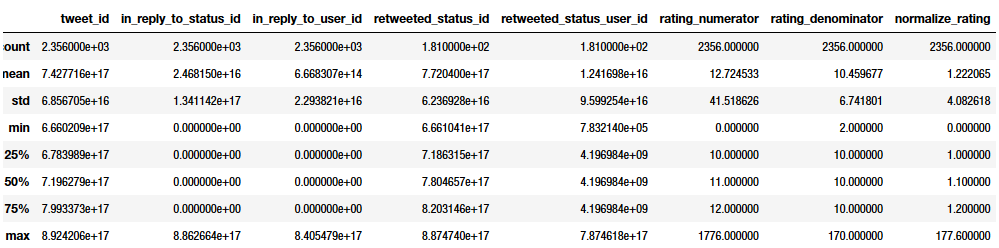


Figure 19 describe result tweeter archive

1. **Retweet data is in dataframe must be deleted**

Listing all duplication data with checking value retweeted status data, and deleted listed row.

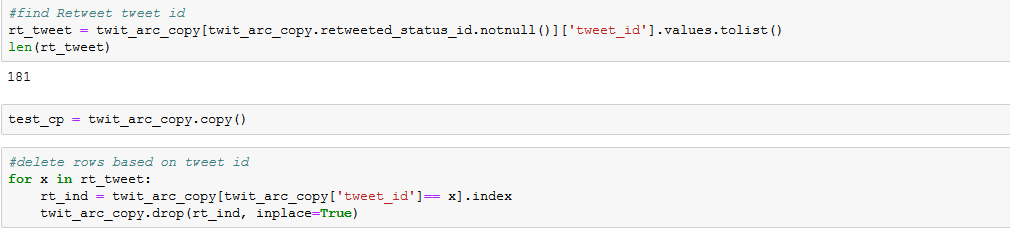


Figure 20 Listing duplicated row and delete it from dataframe



Figure 21 testing result by checking if there are still not nul value in retweeted status id

1. **dog prediction will be more clear by replacing '\_' separator into space**

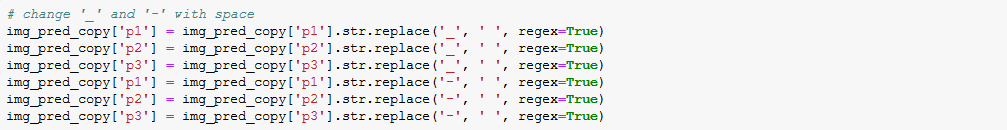


Figure 22 Changing char in value using str.replace methode

Using str.replace, we can change all unwanted char in column.

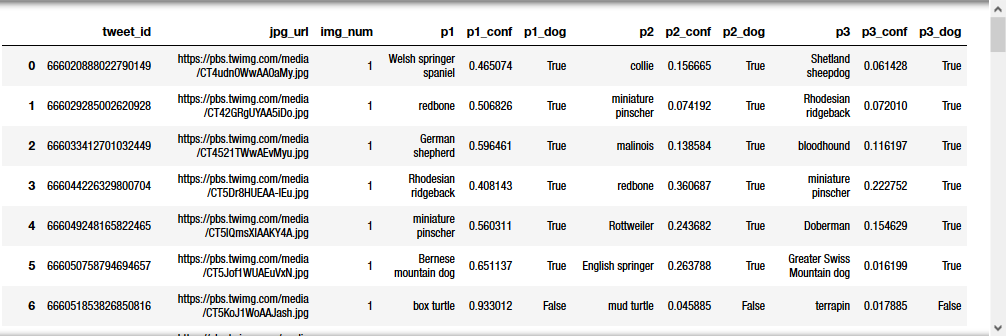


Figure 23 Testing result, there are no \_ and - as separator

1. **Several data duplication in image prediction stored with different tweet\_id**

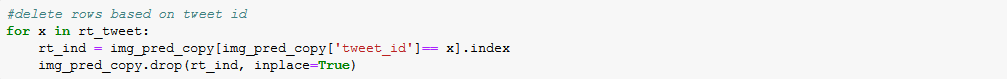


Figure 24 Deleting duplicated row in image prediction data

Deleting duplication by using listed tweet\_id from issue 5. In image prediction, testing duplication

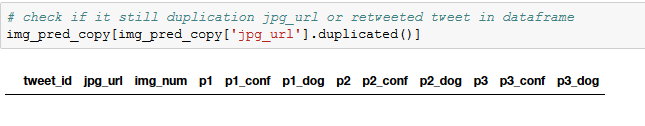
 can be done by see duplication in page\_url. Resulted empty series.

Figure 25 result of jpg\_url duplication search

1. **name of all dogs predictions are variate with upper and lower case**

Changing all char using str.lower() to make all char only in lower case.



Figure 26 Change value in p1,p2 and p3 into lower case using str.lower() method

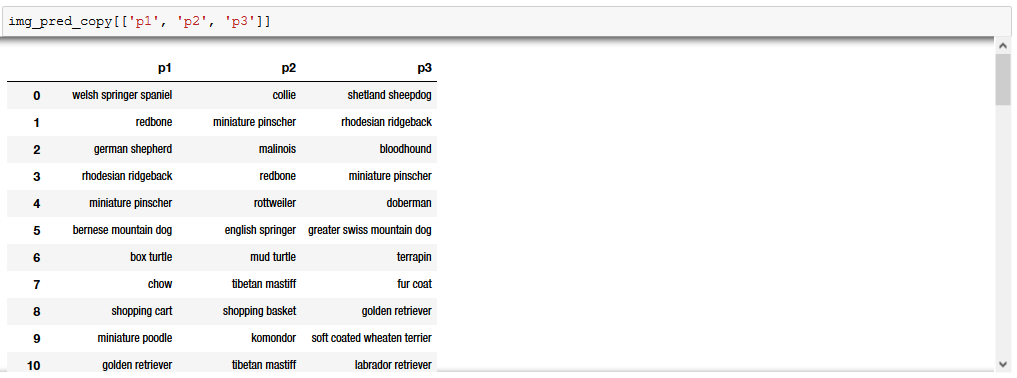
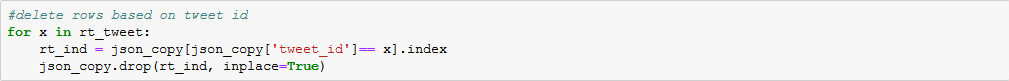


Figure 27 Test lowercasing result by displaying value in column p1,p2, and p3

1. **retweeted tweets is in data**

Deleting process same as issue 7, extracting ‘RT@’ in text used to test the result.



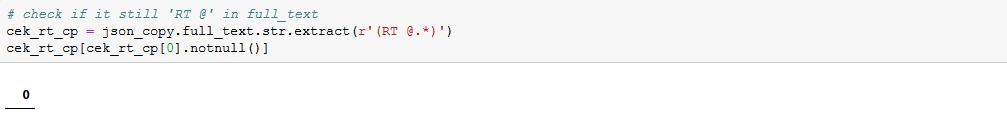
Figure 29 delete unwanted row from listed tweet\_id

Figure 28 Result null series from extracting 'RT @' in text column, indicating there are no duplication data anymore

**Tidiness Issues**

1. **timestamp has date and time in one column**

Parsing datetime is unique. Used pd.DatetimeIndex(twit\_arc\_copy['timestamp']).date to get date and pd.DatetimeIndex(twit\_arc\_copy['timestamp']).time to get time.



Figure 30 Parsing and storing date and time into different column

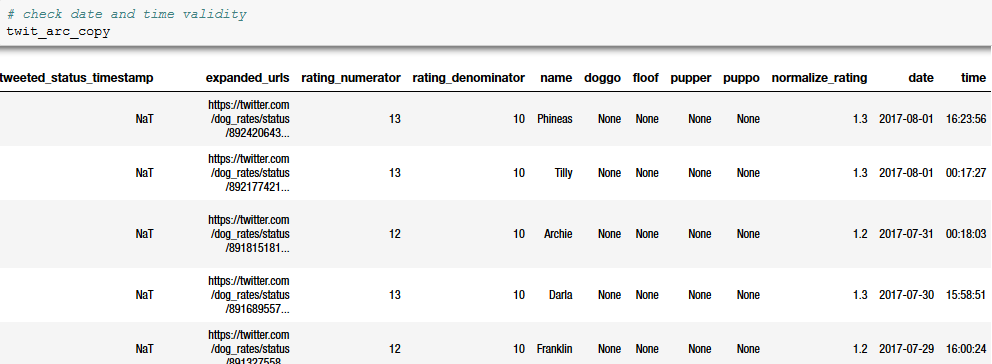


Figure 31Test by displaying data head

1. **text column value has two informations, there are tweet content and link**

This process needs help from str.extract() method using unique regex to get only content and url in text. Checked by visualizing data.

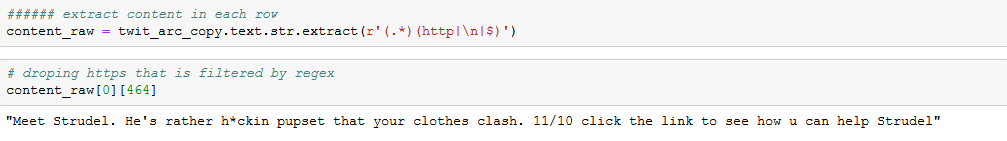


Figure 32 Extracting only content and example of extracted value

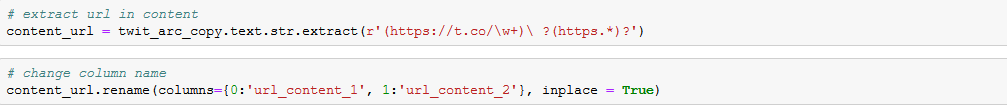
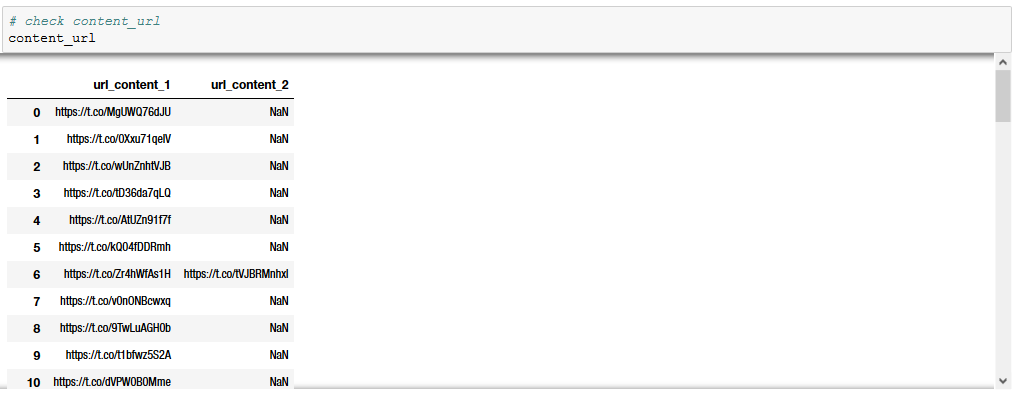


Figure 33 Extracting url, and example of extracted url data

Several contents have 2 url in content, and it makes only a few rows which have value in

url\_content\_2.

All in all, all wrangling data is finished and completing these processes with dropping unwanted column like timestamp, retweeted\_timestamp, retweeted\_status\_id, retweeted\_status\_user\_id in tweeter archive and full\_text in json data (stored as favorite retweeted filtered). Final step is storing cleaned data into all csv files.

Figure 34 Test result of content and url extraction