07_LB_Transform_Tweets

July 13, 2021

1 Overall Twitter Dataset including tweets regarding Covid-19 from November 2020 - June 2021

1.1 Imports

[1]:
```
import pandas as pd
```

1.2 Reading Data

[2]:
```
#Dataset including Tweet ID and Sentiment
ids = pd.read_csv('../data/raw/Corona_Combined_Nov2020-June2021.csv',␣
 ↪header=None, encoding="utf-8", error_bad_lines=False)

#Hydrated Dataset of the above
hydrated = pd.read_csv('../data/raw/Hydrated_Tweets_Nov2020-Jun2021.csv')
```

```
b'Skipping line 11906: expected 2 fields, saw 3\nSkipping line 22310: expected 2
fields, saw 3\nSkipping line 22716: expected 2 fields, saw 3\nSkipping line
31791: expected 2 fields, saw 3\nSkipping line 32525: expected 2 fields, saw
3\nSkipping line 90716: expected 2 fields, saw 3\n'
```

1.3 Adjusting the Dataset adding the Sentiment Score to each Entry

[3]:
```
ids = ids.rename(columns={0:"id", 1:"sentiment"})
ids.id = ids.id.astype('str')
hydrated.id=hydrated.id.astype('str')
hydrated = hydrated.merge(ids, on="id", how="left")
```

1.3.1 Splitting coordinates in latitude and longitude

[4]:
```
df = hydrated
df[['longitude','latitude']] = df.coordinates.str.split(",",expand=True)
```

Export Dataset as CSV

[5]:
```
df.to_csv("../data/interim/Overall_Map_data.csv")
```

2 Using secondly hydrated Dataset with advanced Geo information

2.1 Dataset changes:

- Adding Sentiment Score to each Entry - Add new Column 'date' - Break down the timestamp to date - Drop unused Columns

[6]:
```
#Read hydrated Dataset including more detailed Geo information
data = pd.read_csv('../data/raw/Hydrated_Tweets_with_countries.csv')
```

```
c:\users\lukas\appdata\local\programs\python\python36\lib\site-
packages\IPython\core\interactiveshell.py:3063: DtypeWarning: Columns (31) have
mixed types.Specify dtype option on import or set low_memory=False.
  interactivity=interactivity, compiler=compiler, result=result)
```

[7]:
```
ids = ids.rename(columns={0:"id", 1:"sentiment"})
ids.id = ids.id.astype('float64')
data.id = data.id.astype('float64')
data = data.merge(ids, on="id", how="left")

data['date'] = ''
data['date'] = pd.to_datetime(data['created_at']).dt.strftime('%Y-%m-%d')

data.drop(columns=['Unnamed: 0', 'user', 'place', 'contributors',␣
↪'is_quote_status',
       'retweet_count', 'favorite_count', 'favorited', 'retweeted',
       'possibly_sensitive', 'lang', 'quoted_status_id','quoted_status_id_str',␣
↪'quoted_status_permalink', 'quoted_status',
       'extended_entities', 'withheld_in_countries'], inplace=True)
```

2.1.1 Drop all entries with empty date + adding month and year

[8]:
```
data = data.drop(data[data['date'].isnull()].index)
data['month'] = ''
data['year'] = ''

#Switched because american date format
data['month'] = pd.DatetimeIndex(data['date']).month
data['year'] = pd.DatetimeIndex(data['date']).year
```

2.1.2 Creating Column that includes Tweets per day for each country

[9]:
```
data['tweet_amount'] = ''
data.groupby(['date','country'])['id'].count().reset_index(name="tweet_amount")
```

[9]:

	date	country	tweet_amount
0	2020-11-01	Australia	12

```
1      2020-11-01          Austria              2
2      2020-11-01       Bangladesh              1
3      2020-11-01          Belgium              1
4      2020-11-01           Canada             50
…             …                …              …
8811   2021-06-13        Sri Lanka              1
8812   2021-06-13         Thailand              2
8813   2021-06-13   United Kingdom              2
8814   2021-06-13    United States             24
8815   2021-06-13          Vietnam              1

[8816 rows x 3 columns]
```

Export Dataset as CSV

[10]:
```
data.to_csv("../data/interim/Overall_data.csv")
```

2.2 Using Vaccine Dataset

[11]:
```
vaccine_data = pd.read_csv('../data/raw/country_vaccinations.csv')
```

2.2.1 Dropping Dates which are not included in the Overall Dataset

[12]:
```
vaccine_data = vaccine_data.drop(vaccine_data[vaccine_data['date'] ==␣
↪'2021-06-14'].index)
vaccine_data = vaccine_data.drop(vaccine_data[vaccine_data['date'] ==␣
↪'2021-06-15'].index)
vaccine_data = vaccine_data.drop(vaccine_data[vaccine_data['date'] ==␣
↪'2021-06-16'].index)
vaccine_data = vaccine_data.drop(vaccine_data[vaccine_data['date'] ==␣
↪'2021-06-17'].index)
vaccine_data = vaccine_data.drop(vaccine_data[vaccine_data['date'] ==␣
↪'2021-06-18'].index)
vaccine_data = vaccine_data.drop(vaccine_data[vaccine_data['date'] ==␣
↪'2021-06-19'].index)
vaccine_data = vaccine_data.drop(vaccine_data[vaccine_data['date'] ==␣
↪'2021-06-20'].index)
vaccine_data = vaccine_data.drop(vaccine_data[vaccine_data['date'] ==␣
↪'2021-06-21'].index)
vaccine_data = vaccine_data.drop(vaccine_data[vaccine_data['date'] ==␣
↪'2021-06-22'].index)
vaccine_data = vaccine_data.drop(vaccine_data[vaccine_data['date'] ==␣
↪'2021-06-23'].index)
vaccine_data = vaccine_data.drop(vaccine_data[vaccine_data['date'] ==␣
↪'2021-06-24'].index)
vaccine_data = vaccine_data.drop(vaccine_data[vaccine_data['date'] ==␣
↪'2021-06-25'].index)
```

```
vaccine_data = vaccine_data.drop(vaccine_data[vaccine_data['date'] ==␣
↪'2021-06-26'].index)
vaccine_data = vaccine_data.drop(vaccine_data[vaccine_data['date'] ==␣
↪'2021-06-27'].index)
vaccine_data = vaccine_data.drop(vaccine_data[vaccine_data['date'] ==␣
↪'2021-06-28'].index)
vaccine_data = vaccine_data.drop(vaccine_data[vaccine_data['date'] ==␣
↪'2021-06-29'].index)
vaccine_data = vaccine_data.drop(vaccine_data[vaccine_data['date'] ==␣
↪'2021-06-30'].index)
vaccine_data = vaccine_data.drop(vaccine_data[vaccine_data['date'] ==␣
↪'2021-07-01'].index)
```

Export Dataset as CSV

[13]:
```
vaccine_data.to_csv("../data/interim/Vaccine_data.csv")
```