```
In [1]:
import psycopg2 as db
import pandas as pd
import geopandas as gpd
from dotenv import dotenv_values
from senti_preprocess.senti_preprocess.twitter import remove_urls, remove_mentions, se
```

```
In [2]:
conn_twitter = db.connect(**dotenv_values())
```

Sentiment analysis on Twitter and TripAdvisor

In this notebook, a sentiment analysis on a one month sample of POI tweets and TripAdvisor reviews located in London is performed. The Valence Aware Dictionary and sEntiment Reasoner will be used, which is a lexicon and rule based sentiment analyzer and was sepcifically developed for analyzing social media text data. First up: Twitter.

Twitter

1. Load data

```
In [3]:
tweet_sample_query = """
SELECT
    sq.id, sq.text, sq.place_full_name, sq.place_geom, sq.location_within_london
FROM
    (
        SELECT
            *
        FROM
            tweets_sample
        JOIN
            user_classification as uc
        ON
            tweets_sample.user_id = uc.user_id
    ) as sq, greater_london
WHERE
    sq.place_type = 'poi'
AND
    sq.lang = 'en'
AND ST_WITHIN(sq.place_geom, greater_london.geometry)
"""
```

```
In [4]:
df = pd.read_sql(tweet_sample_query, conn_twitter)
```

2. Preprocess tweets

Three preprocessing steps are performed on each tweet:

1. `remove_urls()` removes any urls from the tweet by matching `r"http\S+"` and replacing any matches with a space.
2. `remove_mentions()` removes any mentions from the tweet by matching `r"@[A-Za-z0-9]+"` and replacing any matches with a space.
3. `segment_hashtags()` finds all the hashtags in a tweet, and looks for those that contain medial capitals (also informally known as camel casing or camelCasing). These are separated into the single words, and the tweet with now segmented hashtags is returned.

```
In [5]:
def clean_tweet(tweet_string):
    tweet_string = remove_urls(tweet_string)
    tweet_string = remove_mentions(tweet_string)
    tweet_string = segment_hashtags(tweet_string)
    return tweet_string
```

```
In [6]:
df["cleaned_tweet"] = df["text"].apply(clean_tweet)
```

```
In [7]:
# Sample exploration
for i in range(20):
    print("{:-^50}".format(i))
    for column, descr in zip(["text", "cleaned_tweet"], ["UNCLEANED", "CLEANED"]):
        print(f'{descr}:\n {df.iloc[i][column]}')
```

```
------------------------0-------------------------
UNCLEANED:
 Merry Christmas from me and my lychee martini 🎄❤️ https://t.co/HbdX8SnBZc
CLEANED:
 Merry Christmas from me and my lychee martini 🎄❤️
------------------------1-------------------------
UNCLEANED:
 @missionswl #Emmanuel #God in #Christ #Jesus
is with us #ChristmasDay #yesterday #TOMORROWFOREVER #CMCWorship #AllareWelcome
Chelsea Fulham Methodist Church
Revd Mark Davenport minister https://t.co/RONFPHmgM7
CLEANED:
   Emmanuel  God in  Christ  Jesus
is with us   Christmas Day yesterday TOMORROWFOREVER CMC Worship  Allare Welcome
Chelsea Fulham Methodist Church
Revd Mark Davenport minister
------------------------2-------------------------
UNCLEANED:
 Ever wondered what Jeremy @theJeremyVine does on Christmas Day? Well, now you know...
#ChristmasDay #cycling #Battersea https://t.co/wtqZkybOAF
CLEANED:
 Ever wondered what Jeremy   does on Christmas Day? Well, now you know...
 Christmas Day cycling  Battersea
------------------------3-------------------------
UNCLEANED:
 A whopping 2545 runners this morning at @bushyparkrun, WOW! What a beautiful day it w
as and a fab way to start off Christmas. #loveparkrun https://t.co/mi0Nf2O2wi
CLEANED:
 A whopping 2545 runners this morning at  , WOW! What a beautiful day it was and a fab
way to start off Christmas. loveparkrun
------------------------4-------------------------
UNCLEANED:
 Art pic; Colours; Red.  Ladybug /Ladybird, Warren farm August 2019.  By @Neil_HW_

Staying mainly dry, possibly for a few days with a peppering of rain here and there.

#StormHour #ThePhotoHour
#ProtectWarrenFarm
#naturefirst https://t.co/OLb5ZTbsGI
CLEANED:
 Art pic; Colours; Red.  Ladybug /Ladybird, Warren farm August 2019.  By  _HW_

Staying mainly dry, possibly for a few days with a peppering of rain here and there.

 Storm Hour  The Photo Hour
 Protect Warren Farm
naturefirst
------------------------5-------------------------
UNCLEANED:
 @hammersmithandy the holidays are over. Get back to work and fix our roads properly.
Months and millions wasted by ineffective local labour? Seriously, who signed this of
f? Disgusting this is what you do to our local streets. Would you accept this if you l
ived on this street? https://t.co/VwmaaYH0qM
CLEANED:
   the holidays are over. Get back to work and fix our roads properly. Months and mill
ions wasted by ineffective local labour? Seriously, who signed this off? Disgusting th
is is what you do to our local streets. Would you accept this if you lived on this str
eet?
------------------------6-------------------------
UNCLEANED:
 Only bit of travelling I need to do this Christmas &amp; as I walk into station (at a
breathless pace with 3mins to spare) I hear the train is cancelled. Next one in an hou
r. I cant be bothered to walk back home  (I'll prob be almost late again if I do) so s
at watching world go by
CLEANED:
 Only bit of travelling I need to do this Christmas &amp; as I walk into station (at a
breathless pace with 3mins to spare) I hear the train is cancelled. Next one in an hou
r. I cant be bothered to walk back home  (I'll prob be almost late again if I do) so s
at watching world go by
------------------------7-------------------------
UNCLEANED:
 Seems the appropriate place to start the 50th birthday #theSmiths https://t.co/5mBZmN
bpSU
CLEANED:
 Seems the appropriate place to start the 50th birthday the Smiths
------------------------8-------------------------
UNCLEANED:
 Triple-parked on my 30th. Time for Las Vegas with @JessamineKate 🥳 https://t.co/CY97
4CDQVA
CLEANED:
 Triple-parked on my 30th. Time for Las Vegas with   🥳
------------------------9-------------------------
UNCLEANED:
 @RedbrickedSlums very lovely,  soothing, and hopeful,  thankyou
CLEANED:
   very lovely,  soothing, and hopeful,  thankyou
------------------------10------------------------
UNCLEANED:
 The Tricycle by Fernando Arrabal

Strangely light-hearted tale of poverty and murder starts this January 2020 in Barons
Court Theate in London

Follow us on https://t.co/4ZoSurj90E or https://t.co/ZPtRQsJu7A for more information!

#theatre #london https://t.co/xfnvmHsoca
CLEANED:
 The Tricycle by Fernando Arrabal

Strangely light-hearted tale of poverty and murder starts this January 2020 in Barons
Court Theate in London

Follow us on   or   for more information!

theatre london
------------------------11------------------------
UNCLEANED:
 @atptour @Cristiano @DjokerNole 2 legends ⚽🎾💪💪
CLEANED:
       2 legends ⚽🎾💪💪
------------------------12------------------------
UNCLEANED:
 @imaginecurve I have used the wrong card to make a transaction and cant go back in ti
me as its over £1k. Have emailed Curve Supprpt 24hrs ago but no response. Need advice
ASAP as it's an important payment
CLEANED:
   I have used the wrong card to make a transaction and cant go back in time as its ov
er £1k. Have emailed Curve Supprpt 24hrs ago but no response. Need advice ASAP as it's
an important payment
------------------------13------------------------
UNCLEANED:
 Sunnying shortly! https://t.co/P8OBmu9ZPB
CLEANED:
 Sunnying shortly!
------------------------14------------------------
UNCLEANED:
 If I was looking for a word to describe the Gents toilets at the Victoria and Albert
museum, that word would be pink.
#FridayThoughts #Kensington #toilets https://t.co/eHzKAHB5Sk
CLEANED:
 If I was looking for a word to describe the Gents toilets at the Victoria and Albert
museum, that word would be pink.
 Friday Thoughts  Kensington toilets
------------------------15------------------------
UNCLEANED:
 The weather is soo cold😿💗 https://t.co/yHW8rdop7V
CLEANED:
 The weather is soo cold😿💗
------------------------16------------------------
UNCLEANED:
 @joepike @BorisJohnson @itvcalendar How dare you @joepike try and make @BorisJohnson
engage with the people of this country and for him to go off script 😒 he doesn't want
to engage, it's better to keep the leavers on side by saying the same nonsense. #VoteN
otTory
CLEANED:
      How dare you   try and make   engage with the people of this country and for hi
m to go off script 😒 he doesn't want to engage, it's better to keep the leavers on si
de by saying the same nonsense.  Vote Not Tory
------------------------17------------------------
UNCLEANED:
 📣 The start of the #NATOEngages conference, the official event accompanying the #NAT
O Leaders Meeting in London.

Heads of state and experts gathered at Westminster to discuss the future of the Allian
ce.

Main topics: #Innovating the Alliance and #CyberSecurity 💻 https://t.co/wXPqU00zmA
CLEANED:
 📣 The start of the NATO Engages conference, the official event accompanying the NATO
Leaders Meeting in London.

Heads of state and experts gathered at Westminster to discuss the future of the Allian
ce.

Main topics:  Innovating the Alliance and  Cyber Security 💻
------------------------18------------------------
UNCLEANED:
 Great ideas being discussed at this morning's @TechDataASUK Healthcare ecosystem #des
ignthinkingworkshop. Partners and IBMers focused on Healthcare clients looking at how
to create best of breed solutions. https://t.co/oyMoxkYQYf
CLEANED:
 Great ideas being discussed at this morning's   Healthcare ecosystem designthinkingwo
rkshop. Partners and IBMers focused on Healthcare clients looking at how to create bes
t of breed solutions.
------------------------19------------------------
UNCLEANED:
 1pm. Improving crowd resilience :) https://t.co/epnCu6IyVB
CLEANED:
 1pm. Improving crowd resilience :)
```

The cleaned tweets look good, and the hashtag segmentation seems to work as expected. So let's feed the data into the analyzer

3. Sentiment analysis

```
In [8]:
import vaderSentiment.vaderSentiment as vader
```

```
In [9]:
senti_analyzer = vader.SentimentIntensityAnalyzer()
```

```
In [10]:
def calculate_sentiment(series):
    tweet = series["cleaned_tweet"]
    sentiment = senti_analyzer.polarity_scores(tweet)
    for key in sentiment.keys():
        series[key] = sentiment[key]
    return series
```

```
In [11]:
df = df.apply(calculate_sentiment, axis=1)
```

```
In [12]:
df
```

Out[12]:

	id	text	place_full_name	
0	1209819446329724929	Merry Christmas from me and my lychee martini ...	Mandarin Oriental Hyde Park	0101000020E610000094A5D6FB8D76C4BF
1	1209819577770762240	@missionswl #Emmanuel #God in #Christ #Jesus\n...	Chelsea Methodist Church	0101000020E6100000DA1EBDE13E72C5BF
2	1209820427528409089	Ever wondered what Jeremy @theJeremyVine does ...	Battersea Power Station	0101000020E6100000978C63247B84C2BF
3	1209820766486900736	A whopping 2545 runners this morning at @bushy...	Bushy parkrun	0101000020E6100000419DF2E84658D5BF
4	1210515867903123456	Art pic; Colours; Red. Ladybug /Ladybird, War...	Warren Farm	0101000020E6100000AFD2DD7536E4D6BF
...	...	...	...	...
23428	1209810083829428225	Merry Christmas everyone 🍻 https://t.co/oNube4...	The Archers	0101000020E6100000C6FD47A643A7C93F
23429	1209811961879441413	Happy and peaceful Christmas to all https://t....	The Church of Our Most Holy Redeemer	0101000020E6100000E2218C9FC6BDC5BF
23430	1209815028796469248	Merry Christmas from everyone here at @Burgean...	Burge and Gunson	0101000020E6100000B6D958897956C6BF
23431	1209816483687215104	MERRY CHRISTMAS EVERYONE!!	Bellingham Film Palace	0101000020E610000030D63730B95194BF
23432	1209816483687215104	MERRY CHRISTMAS EVERYONE!!	Bellingham Film Palace	0101000020E610000030D63730B95194BF

23433 rows × 10 columns

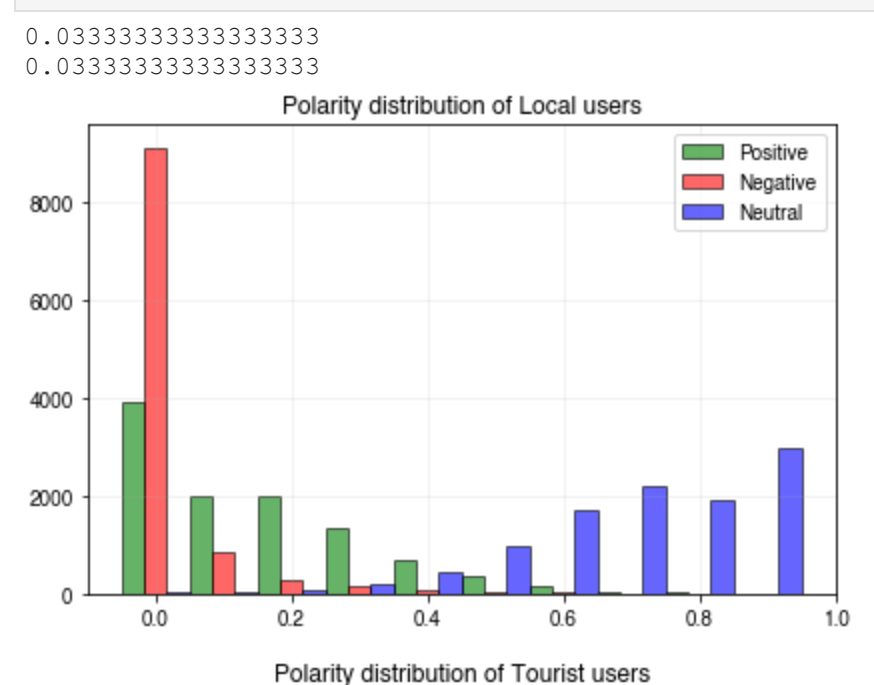
The sentiment analysis worked, so a deeper look at the results would be appropriate.

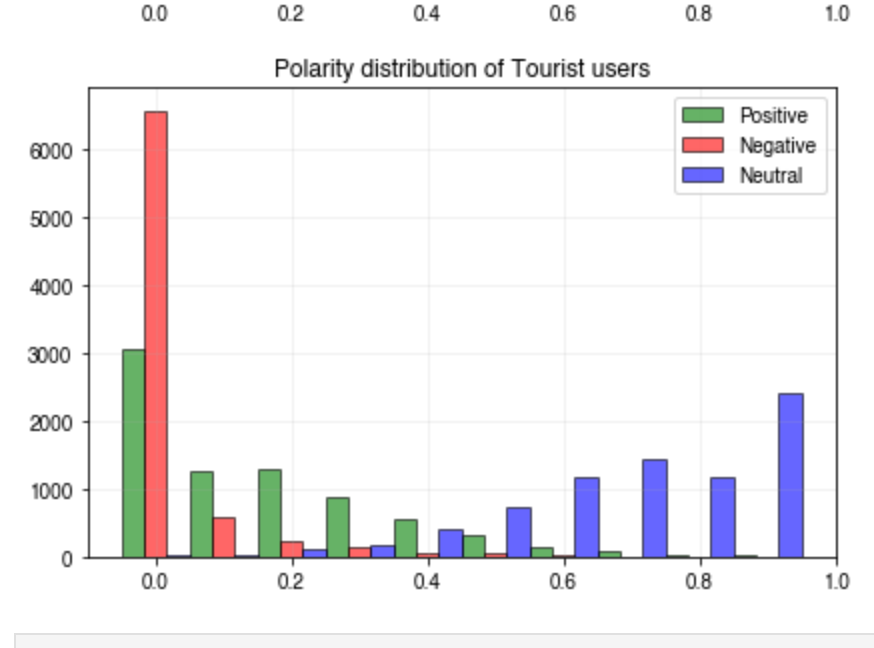
```
In [13]:
from matplotlib import pyplot as plt
import numpy as np
plt.rc('font', family='Helvetica')
```

```
In [143…
def pol_plot(df, user_classification_column):
    user_groups = ["local", "tourist"]
    group_names = ["Local users", "Tourist users"]
    for group, group_name in zip(user_groups, group_names):
        dframe = df.loc[df[user_classification_column] == group]
        edgecolor = "black"
        alpha = 0.6
        fig, ax = plt.subplots()
        columns = ["pos", "neg", "neu"]
        colors = ["green", "red", "blue"]
        legend_dict = dict(zip(columns, colors))

        labels = ["Positive", "Negative", "Neutral"]
        handles = [plt.Rectangle((0,0),1,1, facecolor=legend_dict[column], alpha=alpha

        for index, (col, color) in enumerate(zip(columns, colors)):
            if index == 0:
                hist_tuple_first = np.histogram(dframe[col])
                width = (hist_tuple_first[1][1] - hist_tuple_first[1][0])/(len(columns
                print(width)
                ax.bar(hist_tuple_first[1][:-1] - width, hist_tuple_first[0], facecolo
            else:
                hist_tuple = np.histogram(dframe[col], bins=hist_tuple_first[1])
                ax.bar(hist_tuple[1][:-1] + width * (index - 1), hist_tuple[0], facecc

    plt.legend(handles, labels)
    plt.grid(True, which="both", alpha=0.3, linewidth=0.5)
    plt.title(f"Polarity distribution of {group_name}")
    plt.tight_layout()
```

```
In [144…
pol_plot(df, "location_within_london")
```

```
0.03333333333333333
0.03333333333333333
```






```
In [15]:
from IPython.display import Markdown
```

```
In [88]:
def pretty_print_row(df, row_indices, color_dict, text_col):
    markdown = ""
    for i in row_indices:
        markdown = add_to_markdown(markdown, "{:-^75}".format(i))
        markdown = add_to_markdown(markdown, df.iloc[i][text_col])

        for key in color_dict.keys():
            markdown = add_to_markdown(markdown, df.iloc[i][key], color=color_dict[key
    return markdown

def add_to_markdown(str_1, str_2, color="black"):
    str_1 = str(str_1)
    str_2 = str(str_2)

    if color=="black":
        return str_1 + str_2 + "<br>"
    else:
        return str_1 + "<p style=color:{}>".format(color) + str_2 + "</p>"
```

```
In [89]:
def get_high_polarity_tweets(df, quantile=0.9, max_tweets=50):
    polarity_dict = {}
    for polarity in ["pos", "neg"]:
        quantile = df[polarity].quantile(quantile)
        polarity_dict[polarity] = df[polarity].sort_values(ascending=False).loc[df[po
    return polarity_dict
```

```
In [90]:
high_polarity_tweets = get_high_polarity_tweets(df=df, max_tweets=50)
markdown = pretty_print_row(df=df, row_indices=high_polarity_tweets["pos"], color_dict
Markdown(markdown)
```

Out[90]:

-----------------------------------15580-----------------------------------
@TheSun Classy

1.0

0.0

0.0

-----------------------------------12047-----------------------------------
@work1ngmensclub cheered https://t.co/ou9i42k5cb

1.0

0.0

0.0

-----------------------------------17875-----------------------------------

Gratitude. https://t.co/sv8AEx7GL5

1.0

0.0

0.0

-----------------------------------2131-----------------------------------
@mouthwaite @BlenheimPalace @cescloveslife @realDonaldTrump Cool! https://t.co/11fkj7sWhG

1.0

0.0

0.0

-----------------------------------3805-----------------------------------
@DJNoreenKhan Lovely @DJNoreenKhan @realDonaldTrump https://t.co/NKVma9rlm2

1.0

0.0

0.0

-----------------------------------2126-----------------------------------
@salmahayek @realDonaldTrump Cool! https://t.co/HJyfBzoNfl

1.0

0.0

0.0

-----------------------------------12045-----------------------------------
@FatWhiteFamily Cheers https://t.co/MIN39WJ359

1.0

0.0

0.0

-----------------------------------11000-----------------------------------

Lovely

1.0

0.0

0.0

-----------------------------------1613-----------------------------------
@Gelderdgary1979 @LUFC @FulhamFC Definitely

1.0

0.0

0.0

-----------------------------------11045-----------------------------------

love @clairo 💖💖 https://t.co/e97ouT4qJF

1.0

0.0

0.0

-----------------------------------6341-----------------------------------

Hope ✨ https://t.co/Ex1oRZUtSn

1.0

0.0

0.0

-----------------------------------10708-----------------------------------
@KieranNdl Nice

1.0

0.0

0.0

-----------------------------------6581-----------------------------------

Hope. https://t.co/DFRlK8mdAD

1.0

0.0

0.0

-----------------------------------7475-----------------------------------
@10DowningStreet @BorisJohnson @realDonaldTrump @BorisJohnson Congratulations https://t.co/DX3cSdNTcF

1.0

0.0

0.0

-----------------------------------9549-----------------------------------

beauty https://t.co/EVRDBDNYQM

1.0

0.0

0.0

-----------------------------------9245-----------------------------------

Focused https://t.co/zzYtdtkyuA

1.0

0.0

0.0

-----------------------------------9152-----------------------------------

pretty https://t.co/ydRuSH3Zye

1.0

0.0

0.0

-----------------------------------4867-----------------------------------
@realpreityzinta @realDonaldTrump cool! https://t.co/lKoXW2WNbv

1.0

0.0

0.0

-----------------------------------5448-----------------------------------

Classy https://t.co/8AnPU2MifH

1.0

0.0

0.0

-----------------------------------13308-----------------------------------

Cool https://t.co/FnmKa8CWKD

1.0

0.0

0.0

-----------------------------------1542-----------------------------------

Nice @DegenerateFoxUK https://t.co/3QJrf74mWX

1.0

0.0

0.0

-----------------------------------3070-----------------------------------

Thriller! https://t.co/kJH6pylVNY

1.0

0.0

0.0

-----------------------------------17553-----------------------------------

thanks @LewisCapaldi https://t.co/21eghK5VFx

1.0

0.0

0.0

-----------------------------------5022-----------------------------------

Help

1.0

0.0

0.0

-----------------------------------13752-----------------------------------
@Lloydstevens29 Beautiful

1.0

0.0

0.0

-----------------------------------22663-----------------------------------
@julietmusical Amazing!! https://t.co/eQyKRkYu56

1.0

0.0

0.0

-----------------------------------23039-----------------------------------

Heaven https://t.co/5bOFimGKs9

1.0

0.0

0.0

-----------------------------------20563-----------------------------------
@drjanaway Yes! Yes! YES!

1.0

0.0

0.0

-----------------------------------17063-----------------------------------

Excited!! https://t.co/UZJrqCwFfz

1.0

0.0

0.0

-----------------------------------7759-----------------------------------

Pretty https://t.co/wA6rXJFl7F

1.0

0.0

0.0

-----------------------------------5088-----------------------------------

breathtaking. @WinterWonderLDN https://t.co/xDhf1bjsbA

1.0

0.0

0.0

-----------------------------------1779-----------------------------------

fun fun fun https://t.co/10LU6KDApY

1.0

0.0

0.0

-----------------------------------7758-----------------------------------

Pretty https://t.co/wA6rXJFl7F

1.0

0.0

0.0

-----------------------------------19651-----------------------------------
@mseriesMMXIV Thanks

1.0

0.0

0.0

-----------------------------------10510-----------------------------------

Lovely Jubbly 👌 @dancafc80 https://t.co/iFFiKyHP01

0.909

0.0

0.091

-----------------------------------3009-----------------------------------
@thomasgfarrellj @babyleggs727 beautiful thankyou 👌

0.909

0.0

0.091

-----------------------------------14825-----------------------------------

Lovely Day 💗 https://t.co/h0LlXsE3xW

0.907

0.0

0.093

-----------------------------------14746-----------------------------------

Feeling loved 🖤 https://t.co/A8DSbLUKmA

0.906

0.0

0.094

-----------------------------------12338-----------------------------------

365 days 👌👌👌 https://t.co/CvbHkNv3Pp

0.902

0.0

0.098

-----------------------------------17453-----------------------------------
@226mf_ Loveyouuuu 💗💗💗

0.898

0.0

0.102

-----------------------------------10669-----------------------------------

pretty pretty london ✨ https://t.co/3L1mmlogyc

0.897

0.0

0.103

-----------------------------------8669-----------------------------------

Gorgeous 💛 https://t.co/tDkMwZih9a

0.891

0.0

0.109

-----------------------------------7102-----------------------------------

smile baby smile :) https://t.co/aVDV52dC6s

0.889

0.0

0.111

-----------------------------------11050-----------------------------------

With @GarthTroubadour

Great pleasure https://t.co/LPS9WL9g9n

0.886

0.0

0.114

-----------------------------------13176-----------------------------------

Happy Anniversary gorgeous. https://t.co/NuDlE3t8vV

0.885

0.0

0.115

-----------------------------------19961-----------------------------------

Yes! Great name! https://t.co/PoYcSvRl76

0.881

0.0

0.119

-----------------------------------6732-----------------------------------

✨ Enchanted Eltham ✨ https://t.co/tVL1IDUQs4

0.878

0.0

0.122

-----------------------------------8255-----------------------------------
Good night xoxo https://t.co/je1yHoJLwo
0.873

0.0

0.127

-----------------------------------18657-----------------------------------
happy place. 💙💗 https://t.co/ngkKXjgLP9
0.873

0.0

0.127

-----------------------------------153-----------------------------------
JUSTICE. LIBERTY. SCIENCE. https://t.co/p4OrrZP1Mq
0.872

0.0

0.128

In [19]:
```
from scipy.stats import ttest_ind
```

In [111…
```
def print_t_test(df, user_classification_column, **kwargs):
    columns = ["pos", "neg", "neu"]
    colors = ["green", "red", "blue"]
    labels = ["Positive", "Negative", "Neutral"]
    result = ""
    for col, col_name, color in zip([*columns, "compound"], [*labels, "Compound"], [*c
        t_test, p = ttest_ind(df[col].loc[df[user_classification_column] == "local"],
        result += f"t-test for <a style=color:{color}><b>{col_name}</b></a> polarity:
        result += f"t-test statistic: {round(t_test, 2)}<br>p-value: {round(p, 3)}"
        result += "<br><br>"
    return Markdown(result)
```

In [112…
```
print_t_test(df, "location_within_london", equal_var=False)
```

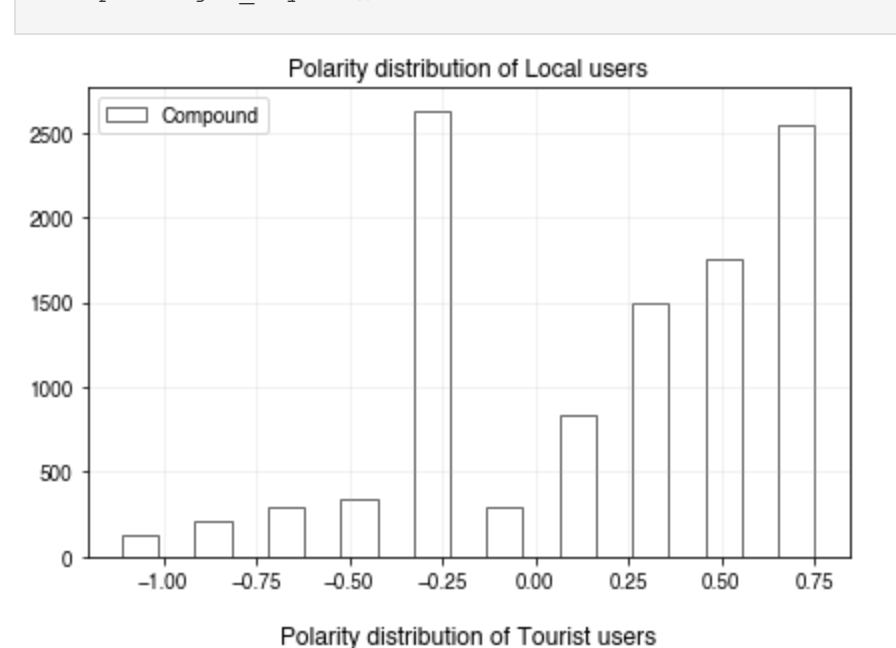
Out[112…
t-test for **Positive** polarity:
t-test statistic: -1.05
p-value: 0.293

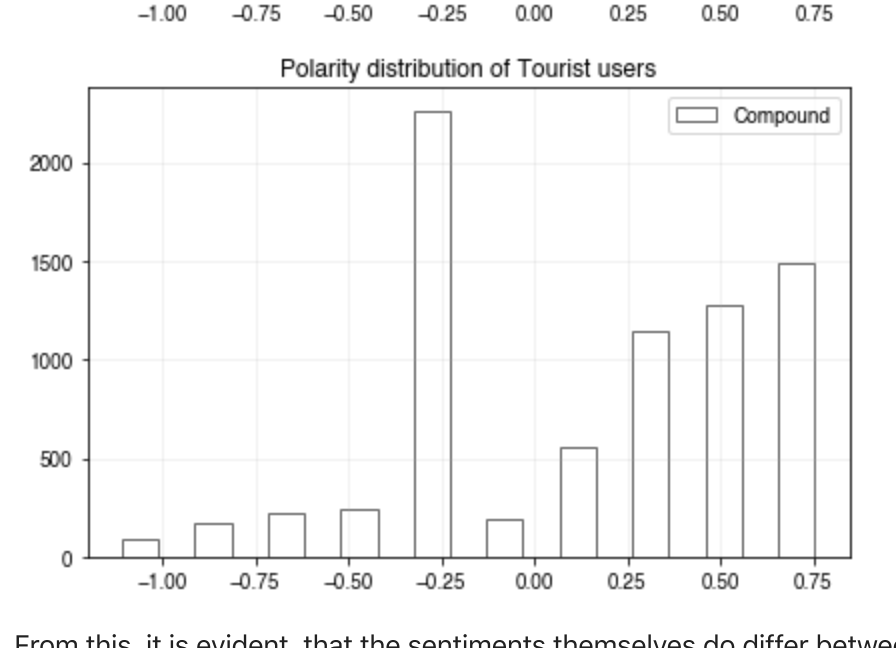
t-test for **Negative** polarity:
t-test statistic: -1.32
p-value: 0.188

t-test for **Neutral** polarity:
t-test statistic: 1.66
p-value: 0.097

t-test for **Compound** polarity:
t-test statistic: 6.1
p-value: 0.0

In [79]:
```
user_groups = ["local", "tourist"]
group_names = ["Local users", "Tourist users"]
user_classification_column = "location_within_london"
for group, group_name in zip(user_groups, group_names):
    dframe = df.loc[df[user_classification_column] == group]
    edgecolor = "black"
    alpha = 0.6
    fig, ax = plt.subplots()
    columns = ["compound"]
    colors = ["white"]
    legend_dict_comp = dict(zip(columns, colors))

    labels = ["Compound"]
    handles = [plt.Rectangle((0,0),1,1, facecolor=legend_dict_comp[column], alpha=alph

    for index, (col, color) in enumerate(zip(columns, colors)):
        if index == 0:
            hist_tuple_first = np.histogram(dframe[col])
            width = (hist_tuple_first[1][1] - hist_tuple_first[1][0])/(len(columns)+1)
            ax.bar(hist_tuple_first[1][:-1] - width, hist_tuple_first[0], facecolor=co
        else:
            hist_tuple = np.histogram(dframe[col], bins=hist_tuple_first[1])
            ax.bar(hist_tuple[1][:-1] + width * (index - 1), hist_tuple[0], facecolor=

    plt.legend(handles, labels)
    plt.grid(True, which="both", alpha=0.3, linewidth=0.5)
    plt.title(f"Polarity distribution of {group_name}")
    plt.tight_layout()
```





From this, it is evident, that the sentiments themselves do differ between tourists and locals, although the cause of this requires further investigation, for example whether a handful of very active users in the locals group leads to a contribution bias that explains the abundance of highly positive compound tweet sentiments.

TripAdvisor

1. Load data

In [43]:
```
env = dotenv_values()
env["dbname"] = "TripAdvisor"
conn_ta = db.connect(**env)
```

In [46]:
```
tripadvisor_sample_query = """
SELECT
    *
FROM
    (
    SELECT
        reviews.id, reviews.title, reviews.rating, reviews.date::date, reviews.full,
        reviews.user_profile, reviews.attr_id, case when ta_user_classification.class
    FROM
        reviews
    JOIN
        users
    ON
        users.profile = reviews.user_profile
    JOIN
        attractions
    ON
        reviews.attr_id = attractions.id
    JOIN
        ta_user_classification
    ON
        reviews.user_profile = ta_user_classification.profile
        ) as s
WHERE
    date BETWEEN SYMMETRIC '2019-11-30' AND '2020-01-01'
"""
```

In [47]:
```
df_ta = pd.read_sql(tripadvisor_sample_query, conn_ta)
```

In [48]:
```
df_ta
```

Out[48]:

	id	title	rating	date	full	user_profile	attr_id	case	
0	732868532	Court dinner	5	2019-12-15	An amazing venue , full of opulence and ...	/Profile/katrinad947	19041690	tourist	010100002
1	733257910	A striking and moving tribute to an icon of ou...	5	2019-12-17	The bust of Nelson Mandela now sits in a corri...	/Profile/richardpre	17326340	local	01010000
2	732271603	AMAZING!	5	2019-12-12	The best view ever! It's fantastic! We're thri...	/Profile/cmlz	15567100	tourist	010100002
3	730683726	Not a 5* experience	1	2019-12-04	Positives- beautiful view! It's truly an amazi...	/Profile/14georginal	15567100	tourist	010100002
4	733574536	Stadium tour	5	2019-12-19	Tommy and harry top job wicked day thanks full...	/Profile/403garethh	14985376	tourist	010100002
...	...	...	...	...	...	...	...	...	
3825	731974852	Fantastic day	5	2019-12-10	We had a brilliant day- it was a gift given to...	/Profile/X7517CYangelab	194299	local	010100002
3826	735180650	Pretty Market	5	2019-12-28	Beautiful market \nLots of shopping and food c...	/Profile/RParbhoo	189047	tourist	01010000
3827	734957020	One of the best spots at London	5	2019-12-27	A great place to see artists...to have a drink...	/Profile/PatriciaNare	189047	tourist	01010000
3828	734350491	Christmas tidings	3	2019-12-23	Christmas is only a day away and Covent Garden...	/Profile/Museuman	189047	local	01010000
3829	734293329	must visit it	5	2019-12-23	Very vibrant place with many events taking pla...	/Profile/dsotiroudis	189047	tourist	01010000

3830 rows × 9 columns

2. Preprocessing

Unlike tweets, TripAdvisor reviews are required to have at least 200 characters, but can be significantly longer. This leads to most reviews being written in a less informal way. Furthermore, TripAdvisor forbids including any URLs, while mentions and hashtags are not supported features on the platform. This makes the preprocessing steps taken for Twitter largely unnecessary.

Furthermore, the VADER SA eliminates the need for preprocessing steps typical to NLP problems, like stemming or lemmatization. Therefore, let's have a look at a review sample and check whether any other preprocessing might be necessary.

In [69]:
```
np.random.seed(0)
ta_sample = np.random.randint(0, df_ta.shape[0], 10)
```

In [70]:
```
for i in ta_sample:
    print(df_ta.iloc[i]["full"])
```

```
Booked to see The Man in the White Suit which unfortunately closed early so swapped th
e tickets for Curtains the replacement. The show itself was quite funny with good acti
ng and some very good tunes.  Jason Manford played the lead role very well.

The theatre is small, (the toilets in particular) but I was surprised the seats we cho
ose (up in the balcony) had such a good view and the seats were comfortable. The bar w
as average with the usual high theatre price for drinks and there could have been a fe
w more seats and tables.

Would definitely return to see another show at this theatre.
Offers lovely view - bollywood memories - must visit to relive all the scenes from bol
lywood movies.
Gallery lay out well done, narrative in each room was very informative. Excellent cove
rage of the different phases of this artist.
I visited the London Eye yesterday and had booked a time slot for the evening so that
I could see the London skyline lit up. I booked online and the queue moved fairly quic
kly. The staff were friendly and helpful. I was lucky enough to get a clear evening. T
he views were fantastic. The ride lasts around 30 minutes and I thoroughly enjoyed it.
I highly recommend the London Eye.
The world famous Big Ben is undergoing 4 years of restoration.  One of the faces has b
een uncovered but the rest of the tower remains under a mass of scaffolding so sadly n
o iconic photos
December is the best time of the year to visit London if you love Christmas. Carnaby s
treet is one of the spots you should visit.

Went for my birthday and had fab time .Staff was very helpful to tel the good location
for taking photos .
Had  drinks and ice cream ,the view from even the washroom is amazing .
Interesting place to visit, however it is very overcrowded, so get prepared for long w
aiting times and plan accordingly. Definitely a landmark to see.
The underground plies from the airport to many of the stations connecting the points o
f interest. The trains are busy peak hour but off peak hour it is much less. No need t
o rent a car when there is such a good connection. I travelled with the Visitor Oyster
card and it was really good value for money
This is about the Avanti First Class Lounge at Euston
Been using this for years however waited five minutes at the bar before asking to be s
erved
Two Havana Clubs with no mixer £20 sorry cheaper elsewhere
Don't bother apart from free tea and coffee and cookies
```

3. Sentiment Analysis

In [72]:
```
def calculate_review_sentiment(series):
    tweet = series["full"]
    sentiment = senti_analyzer.polarity_scores(tweet)
    for key in sentiment.keys():
        series[key] = sentiment[key]
    return series
```

In [73]:
```
df_ta = df_ta.apply(calculate_review_sentiment, axis=1)
```

In [96]:
```
high_pol_reviews = get_high_polarity_tweets(df_ta, max_tweets=10)
markdown = pretty_print_row(df_ta, high_pol_reviews["neg"], legend_dict, text_col="ful
Markdown(markdown)
```

Out[96]:
-----------------------------------2176-----------------------------------
Very disappointing. House drinks included with no name spirits, limited food, broken charging points. This is one of the worst lounges I've been in and deffo wouldn't come back.

0.0

0.381

0.619

-----------------------------------146-----------------------------------
Nothing really to say but it's overcrowded and not sure what the way forward is. But think unfair you pay more for such a poor service.
0.0

0.359

0.641

-----------------------------------3106-----------------------------------
We visited Hyde Park tk see winter wonderland. It was chaos. No organisation, massive fight broke out and it took 2 hours to get in. Shame as it is such a beautiful place.
0.086

0.318

0.596

-----------------------------------3265-----------------------------------
This was a great museum. The history of the prison was artfully presented and I can't believe how much we learned about this horrifying history of torture and mistreatment. Seeing the actual torture devices was fascinating and a bit terrifying too.
0.125

0.316

0.559

-----------------------------------447-----------------------------------
Really, don't bother ...,,,,,,,,,,,,,,,,,,,,Seats uncomfortable Sooooooo bad. .... dancing out of time...... tall skinny snowman when film is cute and round......
0.104

0.305

0.591

-----------------------------------3588-----------------------------------
London underground os old. Trains are old, stations are old. Very narrow, and scaring.

Not the best place to be in case of a Panic Situation.

0.0

0.299

0.701

-----------------------------------1580-----------------------------------
It is an expensive place to have a nap. Food is mediocre and the choise was underwhelming. But the worst of it was that I had to witness a racist rant coming from a British traveller toward a Muslim family. The staff managed it very poorly, and had the Muslim family removed rather then the racist man who had started the argument. I had to ask the staff to have him removed before anything happened to that racist man.
0.0

0.281

0.719

-----------------------------------759-----------------------------------
The afternoon tea was so bad it's not worth description Service was diabolical The Chef should be sacked terrible food It ruined our day in London I could make a better afternoon tea! Close it down
0.062

0.272

0.666

-----------------------------------1263-----------------------------------
Very disappointing where have all the market and craft stalls gone replaced with food chains lost its character
0.0

0.266

0.734

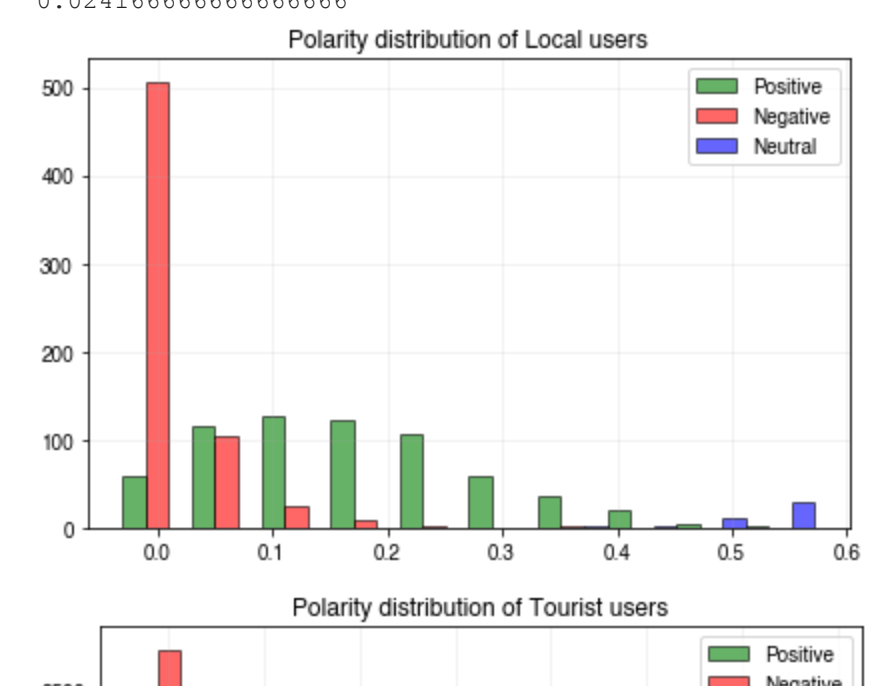
-----------------------------------3723-----------------------------------
Market disappointing - too repetitive ! Food stalls are ok and the food we had was hot and filled us up . Poor signage around the park so you struggle to find things. toilets are so disgusting ! No toilet paper and no paper towels. Even after asking an attendant they just shrugged it off . Seems a lot smaller than last year
0.028

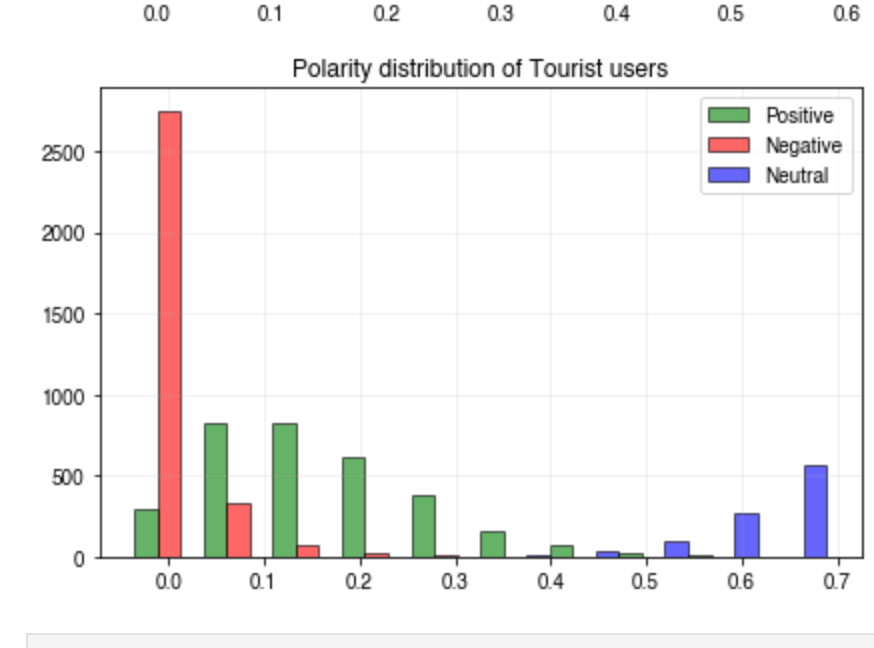
0.264

0.708

In [146…
```
pol_plot(df_ta, "case") # to do: fix x-axis
```

```
0.020066666666666667
0.024166666666666666
```





In [113…
```
print_t_test(df_ta, "case", equal_var=False)
```

Out[113…
t-test for **Positive** polarity:
t-test statistic: 0.27
p-value: 0.787

t-test for **Negative** polarity:
t-test statistic: 1.96
p-value: 0.05

t-test for **Neutral** polarity:
t-test statistic: -1.18
p-value: 0.237

t-test for **Compound** polarity:
t-test statistic: -1.38
p-value: 0.167

Not surprisingly, the sentiment distributions of Twitter and TripAdvisor are vastly different. Generally, TripAdvisor reviews are more sentiment-laden, which is expected, since they are longer and people use it explicitly to describe experiences.

That being said, there are differences between tourists' and locals' sentiment, but they differ across the two platforms: while on Twitter, a significant difference was found in the compound polarity, while on TripAdvisor the only significant difference was in negative polarity.

In [ ]: