Importing Libraries

In [1]:

```
import pandas as pd
import plotly.express as px
import plotly.figure_factory as ff
import warnings
import itertools

from IPython.display import display

warnings.simplefilter("ignore", UserWarning)
warnings.simplefilter("ignore", FutureWarning)
```

graphs won't be shown here, can use nbviewer to view the plots

- The brief carefully it states that the client wanted to see “An analysis of their content categories showing the top 5 categories with the largest popularity”.
- As explained in the data model, popularity is quantified by the “Score” given to each reaction type.
- We therefore need data showing the content ID, category, content type, reaction type, and reaction score.
- So, to figure out popularity, we’ll have to add up which content categories have the largest score.

Plotting Function

In [2]:

```
def reaction_count(content_reactions_types_df_rc,row_count):
    for cols_ in enumerate(reaction_sentiment_list):
        content_reactions_types_df_rc_sort=content_reactions_types_df_rc.sort_values([co
        fig = px.bar(x=content_reactions_types_df_rc_sort[row_count], y=content_reaction
        fig.update_layout(
            title="Count of "+str(cols_[1])+" "+"("+str(cols_[0]+1)+")",
            xaxis_title=str(cols_[1]),
            yaxis_title="Count",
            font=dict(family="Courier New, monospace",size=18,color="RebeccaPurple")
            )
        fig.show()
```

Importing Data

In [3]:

```
content_df=pd.read_csv("Content.csv")
reactions_df=pd.read_csv("Reactions.csv")
reaction_type_df=pd.read_csv("ReactionTypes.csv")
```

In [4]:

```
# Merging Dataframes to List of Dataframes
df_list = [content_df,reactions_df,reaction_type_df]
```

In [5]:

```
for i_df in df_list:
#     getting names of dataframes
    name =[x for x in globals() if globals()[x] is i_df][0]
    print(name)
    display(i_df.head())
    display(str(name)+" Size",i_df.shape)
    null_columns=i_df.columns[i_df.isnull().any()]
    display(i_df[i_df.isnull().any(axis=1)][null_columns])
    col_obj=i_df.columns[i_df.dtypes=='object']
    col_obj=list(col_obj)
    for i_df_columns in i_df:
        if i_df_columns in col_obj:
            display(i_df[i_df_columns].value_counts())
```

```
content_df
```

	Unnamed: 0	Content ID	User ID	Type	Category	UR
0	0	97522e57-d9ab-4bd6-97bf-c24d952602d2	8d3cd87d-8a31-4935-9a4f-b319bfe05f31	photo	Studying	https://socialbuzz.cdn.com/content/storage/975
1	1	9f737e0a-3cdd-4d29-9d24-753f4e3be810	beb1f34e-7870-46d6-9fc7-2e12eb83ce43	photo	healthy eating	https://socialbuzz.cdn.com/content/storage/9f7
2	2	230c4e4d-70c3-461d-b42c-ec09396efb3f	a5c65404-5894-4b87-82f2-d787cbee86b4	photo	healthy eating	https://socialbuzz.cdn.com/content/storage/230
3	3	356fff80-da4d-4785-9f43-bc1261031dc6	9fb4ce88-fac1-406c-8544-1a899cee7aaf	photo	technology	https://socialbuzz.cdn.com/content/storage/356
4	4	01ab84dd-6364-4236-abbb-3f237db77180	e206e31b-5f85-4964-b6ea-d7ee5324def1	video	food	https://socialbuzz.cdn.com/content/storage/01a

```
'content_df Size'
(1000, 6)
```

	URL
5	NaN
10	NaN
15	NaN
20	NaN
25	NaN
...	...
975	NaN
980	NaN
985	NaN
990	NaN
995	NaN

199 rows × 1 columns

```
97522e57-d9ab-4bd6-97bf-c24d952602d2    1
d0c1d7f4-7735-49a9-ab6b-2cdb2338a609    1
22cdf77b-00c8-41c3-ad59-333c3e751e2c    1
28b8c278-a125-4295-98dd-5f8d45c3200b    1
e1792fe6-28e0-49fc-8b01-0e027461d8b5    1
```

```
                                       ..
16fb5dcb-4349-4831-acf2-8c116ad7dae5    1
0d5eb9fd-879d-4716-aebc-840f7b1b7e9f    1
e4827a5c-c604-4aad-b7ed-b396b3601b74    1
abecd821-ad3d-43b0-a550-aadd9d267072    1
75d6b589-7fae-4a6d-b0d0-752845150e56    1
Name: Content ID, Length: 1000, dtype: int64
72d2587e-8fae-4626-a73d-352e6465ba0f    8
3956593b-7739-426a-b7a5-e841c95a5df9    7
b473e898-b7b0-4a57-959d-484bf4cc4483    7
47def058-01cc-478f-9830-eaddcccac633    6
13f0db8a-152a-496f-a6e8-1ed6a90b8788    6
                                       ..
49f49bcf-17fe-4edd-990d-16c3d1df931b    1
b76ebf8d-3f04-4e7f-aec8-22575f68d9e2    1
6978b891-dea2-4217-8bc8-47d5aa25e743    1
9b6d35f9-5e15-4cd0-a8d7-b1f3340e02c4    1
fb4654ff-ce23-4a77-b52a-50f8948b5664    1
Name: User ID, Length: 446, dtype: int64
photo    261
video    259
GIF      244
audio    236
Name: Type, dtype: int64
technology           71
animals              67
travel               67
culture              63
science              63
fitness              61
food                 61
healthy eating       61
cooking              60
soccer               58
tennis               58
education            57
dogs                 56
studying             55
veganism             48
public speaking      48
Fitness               5
Animals               4
Science               4
"soccer"              3
"culture"             3
Soccer                3
"dogs"                2
Education             2
Studying              2
Travel                2
Food                  2
"veganism"            1
"public speaking"     1
Public Speaking       1
"technology"          1
"cooking"             1
Healthy Eating        1
"studying"            1
"food"                1
Culture               1
"tennis"              1
Technology            1
"animals"             1
Veganism              1
```

```
"science"           1
Name: Category, dtype: int64
https://socialbuzz.cdn.com/content/storage/97522e57-d9ab-4bd6-97bf-c24d952602d2    1
https://socialbuzz.cdn.com/content/storage/f3e8d168-6fb9-48b9-8347-b35595162c1d    1
https://socialbuzz.cdn.com/content/storage/28b8c278-a125-4295-98dd-5f8d45c3200b    1
https://socialbuzz.cdn.com/content/storage/0c91753a-0bb4-4b77-a919-96c2f998ad91    1
https://socialbuzz.cdn.com/content/storage/fd0a3090-6b89-4c4e-9192-1aa4db213a3e    1
                                                                                  ..
https://socialbuzz.cdn.com/content/storage/a39e8a86-63e3-4dcc-8561-4a0b7006df53    1
https://socialbuzz.cdn.com/content/storage/16fb5dcb-4349-4831-acf2-8c116ad7dae5    1
https://socialbuzz.cdn.com/content/storage/0d5eb9fd-879d-4716-aebc-840f7b1b7e9f    1
https://socialbuzz.cdn.com/content/storage/abecd821-ad3d-43b0-a550-aadd9d267072    1
https://socialbuzz.cdn.com/content/storage/75d6b589-7fae-4a6d-b0d0-752845150e56    1
Name: URL, Length: 801, dtype: int64
reactions_df
```

	Unnamed: 0	Content ID	User ID	Type	Datetime
0	0	97522e57-d9ab-4bd6-97bf-c24d952602d2	NaN	NaN	2021-04-22 15:17:15
1	1	97522e57-d9ab-4bd6-97bf-c24d952602d2	5d454588-283d-459d-915d-c48a2cb4c27f	disgust	2020-11-07 09:43:50
2	2	97522e57-d9ab-4bd6-97bf-c24d952602d2	92b87fa5-f271-43e0-af66-84fac21052e6	dislike	2021-06-17 12:22:51
3	3	97522e57-d9ab-4bd6-97bf-c24d952602d2	163daa38-8b77-48c9-9af6-37a6c1447ac2	scared	2021-04-18 05:13:58
4	4	97522e57-d9ab-4bd6-97bf-c24d952602d2	34e8add9-0206-47fd-a501-037b994650a2	disgust	2021-01-06 19:13:01

```
'reactions_df Size'
(25553, 5)
```

	User ID	Type
0	NaN	NaN
10	NaN	love
20	NaN	intrigued
30	NaN	intrigued
40	NaN	dislike
...	...	...
25519	NaN	love
25529	NaN	dislike
25539	NaN	cherish
25540	NaN	NaN
25550	NaN	interested

3019 rows × 2 columns

```
4b2d0fff-3b4f-43ca-a7df-c430479cb9ba    49
697af362-e84b-4429-b4ea-4123c6ab44ba    49
d706b190-216c-4103-9107-fb7304766d68    49
36d36f19-7a10-4d7d-a3ab-a3f2cbbfcf4a    49
4dd4da35-453e-466d-95ca-b1a7710fac1f    49
                                        ..
daaae2e1-3090-4f64-9d43-b4e4ffbb5c8b     1
5d915af1-3cc3-4d44-a0ff-d170a008a5d5     1
9dd95c34-8b39-4776-a232-412512329c3f     1
```

```
0f1fce4d-78a3-4e0e-8a7b-ebd5f97c305e     1
f04de5da-e42f-4d89-a79a-3dff16f7d422     1
Name: Content ID, Length: 980, dtype: int64
c76c3393-88e2-47b0-ac37-dc4f2053f5a5    65
68724f58-bc4d-4ab0-a4e1-60cdd5e95e7d    65
0871bb31-3d6e-4e4c-ab19-95a262cac0d4    63
d1a89d23-7d17-4949-9e1a-637317141f3d    62
4fe1900d-5e78-41a3-88ed-18e6889c6c77    62
                                        ..
3663e3e6-3d5c-4ed9-a6af-1e680ec5f34b    31
b4a6b3ac-b6af-4525-8d59-7afc00ff279d    30
90898216-e580-46c0-8e79-f2df84a9676d    30
e57c1d53-11ce-4df6-bb4b-85647776fd6d    30
a710ab29-b72a-42c8-a79b-42e63d4a8bfd    29
Name: User ID, Length: 500, dtype: int64
heart           1622
scared          1572
peeking         1559
hate            1552
interested      1549
dislike         1548
adore           1548
want            1539
love            1534
disgust         1526
like            1520
super love      1519
indifferent     1512
cherish         1501
worried         1497
intrigued       1475
Name: Type, dtype: int64
2020-10-29 20:51:08    2
2020-09-10 06:59:59    2
2020-08-10 18:01:52    2
2021-01-07 14:49:14    2
2020-12-13 17:37:25    2
                      ..
2020-07-06 12:18:09    1
2021-05-11 10:32:28    1
2021-03-05 13:48:27    1
2021-01-26 03:26:19    1
2021-01-04 04:55:11    1
Name: Datetime, Length: 25542, dtype: int64
reaction_type_df
```

	Unnamed: 0	Type	Sentiment	Score
0	0	heart	positive	60
1	1	want	positive	70
2	2	disgust	negative	0
3	3	hate	negative	5
4	4	interested	positive	30

```
'reaction_type_df Size'
(16, 4)
```

```
heart          1
want           1
disgust        1
hate           1
interested     1
indifferent    1
```

```
love          1
super love    1
cherish       1
adore         1
like          1
dislike       1
intrigued     1
peeking       1
scared        1
worried       1
Name: Type, dtype: int64
positive    9
negative    5
neutral     2
Name: Sentiment, dtype: int64
```

Cleaning data accordingly

In [6]:

```python
content_df['Category']=content_df['Category'].apply(lambda x: x.replace('"', ''))
content_df['Category']=content_df['Category'].apply(lambda x: x.replace('Studying', 'stu
```

1. Removing rows that have values which are missing (According to details provided)

In [7]:

```python
for i_df in df_list:
    name =[x for x in globals() if globals()[x] is i_df][0]
    print(name)
    before_null=i_df.shape[0]
    display(str(name)+" Size",i_df.shape)
    display(i_df.head())
    i_df.dropna(inplace=True)
    i_df.drop(['Unnamed: 0'],axis=1,inplace=True)
    after_null=i_df.shape[0]
    print("Removed ",str(before_null-after_null),"for ",name)
    display(str(name)+" Size",i_df.shape)
```

```
content_df
'content_df Size'
(1000, 6)
```

	Unnamed: 0	Content ID	User ID	Type	Category	UR
0	0	97522e57-d9ab-4bd6-97bf-c24d952602d2	8d3cd87d-8a31-4935-9a4f-b319bfe05f31	photo	studying	https://socialbuzz.cdn.com/content/storage/975
1	1	9f737e0a-3cdd-4d29-9d24-753f4e3be810	beb1f34e-7870-46d6-9fc7-2e12eb83ce43	photo	healthy eating	https://socialbuzz.cdn.com/content/storage/9f7
2	2	230c4e4d-70c3-461d-b42c-ec09396efb3f	a5c65404-5894-4b87-82f2-d787cbee86b4	photo	healthy eating	https://socialbuzz.cdn.com/content/storage/230
3	3	356fff80-da4d-4785-9f43-bc1261031dc6	9fb4ce88-fac1-406c-8544-1a899cee7aaf	photo	technology	https://socialbuzz.cdn.com/content/storage/356
4	4	01ab84dd-6364-4236-	e206e31b-5f85-4964-	video	food	https://socialbuzz.cdn.com/content/storage/01a

abbb- 3f237db77180	b6ea- d7ee5324def1

```
Removed  199 for  content_df
'content_df Size'
(801, 5)
reactions_df
'reactions_df Size'
(25553, 5)
```

	Unnamed: 0	Content ID	User ID	Type	Datetime
0	0	97522e57-d9ab-4bd6-97bf-c24d952602d2	NaN	NaN	2021-04-22 15:17:15
1	1	97522e57-d9ab-4bd6-97bf-c24d952602d2	5d454588-283d-459d-915d-c48a2cb4c27f	disgust	2020-11-07 09:43:50
2	2	97522e57-d9ab-4bd6-97bf-c24d952602d2	92b87fa5-f271-43e0-af66-84fac21052e6	dislike	2021-06-17 12:22:51
3	3	97522e57-d9ab-4bd6-97bf-c24d952602d2	163daa38-8b77-48c9-9af6-37a6c1447ac2	scared	2021-04-18 05:13:58
4	4	97522e57-d9ab-4bd6-97bf-c24d952602d2	34e8add9-0206-47fd-a501-037b994650a2	disgust	2021-01-06 19:13:01

```
Removed  3019 for  reactions_df
'reactions_df Size'
(22534, 4)
reaction_type_df
'reaction_type_df Size'
(16, 4)
```

	Unnamed: 0	Type	Sentiment	Score
0	0	heart	positive	60
1	1	want	positive	70
2	2	disgust	negative	0
3	3	hate	negative	5
4	4	interested	positive	30

```
Removed  0 for  reaction_type_df
'reaction_type_df Size'
(16, 3)
```

2. Changing the data type of some values within a column (According to details provided)

In [8]:

```
for i_df in df_list:
    name =[x for x in globals() if globals()[x] is i_df][0]
    print(name)
    print(i_df.dtypes,"\n")
```

```
content_df
Content ID    object
User ID       object
Type          object
Category      object
URL           object
dtype: object

reactions_df
Content ID    object
```

```
User ID         object
Type            object
Datetime        object
dtype: object

reaction_type_df
Type           object
Sentiment      object
Score           int64
dtype: object
```

In [9]:

```
reactions_df["Datetime"] = reactions_df["Datetime"].apply(pd.to_datetime)
reactions_df.dtypes
```

Out[9]:

```
Content ID                object
User ID                   object
Type                      object
Datetime          datetime64[ns]
dtype: object
```

In [10]:

```
content_df.rename(columns={'Type':'Content_Type'}, inplace=True)
reactions_df.rename(columns={'Type':'Reaction_Type'}, inplace=True)
reaction_type_df.rename(columns={'Type':'Reaction_Type'}, inplace=True)
```

3. Removing columns which are not relevant to this task (According to details provided)

Think about how each column might be relevant to the business question you're investigating. If you can't think of why a column may be useful, it may not be worth including it.

Carefully looking at the columns of the dataframes it looks like only URL in the content_df should be removed as it is Unique Identifier and not that useful in analysis for top categories compared to other variables.

In [11]:

```
content_df.drop(['URL'],axis=1,inplace=True)
content_df.drop(['User ID'],axis=1,inplace=True)
reactions_df.drop(['User ID'],axis=1,inplace=True)
display(content_df.head())
display(reactions_df.head())
```

	Content ID	Content_Type	Category
0	97522e57-d9ab-4bd6-97bf-c24d952602d2	photo	studying
1	9f737e0a-3cdd-4d29-9d24-753f4e3be810	photo	healthy eating
2	230c4e4d-70c3-461d-b42c-ec09396efb3f	photo	healthy eating
3	356fff80-da4d-4785-9f43-bc1261031dc6	photo	technology
4	01ab84dd-6364-4236-abbb-3f237db77180	video	food

	Content ID	Reaction_Type	Datetime
1	97522e57-d9ab-4bd6-97bf-c24d952602d2	disgust	2020-11-07 09:43:50
2	97522e57-d9ab-4bd6-97bf-c24d952602d2	dislike	2021-06-17 12:22:51
3	97522e57-d9ab-4bd6-97bf-c24d952602d2	scared	2021-04-18 05:13:58
4	97522e57-d9ab-4bd6-97bf-c24d952602d2	disgust	2021-01-06 19:13:01
5	97522e57-d9ab-4bd6-97bf-c24d952602d2	interested	2020-08-23 12:25:58

Joining Tables (Content, Reactions, Reaction_Types)

In [12]:

```
content_reactions_df=pd.merge(content_df,reactions_df, how='left', on=['Content ID'])
content_reactions_df
```

Out[12]:

	Content ID	Content_Type	Category	Reaction_Type	Datetime
0	97522e57-d9ab-4bd6-97bf-c24d952602d2	photo	studying	disgust	2020-11-07 09:43:50
1	97522e57-d9ab-4bd6-97bf-c24d952602d2	photo	studying	dislike	2021-06-17 12:22:51
2	97522e57-d9ab-4bd6-97bf-c24d952602d2	photo	studying	scared	2021-04-18 05:13:58
3	97522e57-d9ab-4bd6-97bf-c24d952602d2	photo	studying	disgust	2021-01-06 19:13:01
4	97522e57-d9ab-4bd6-97bf-c24d952602d2	photo	studying	interested	2020-08-23 12:25:58
...	...	...	...	...	...
18408	75d6b589-7fae-4a6d-b0d0-752845150e56	audio	technology	worried	2020-10-31 04:50:14
18409	75d6b589-7fae-4a6d-b0d0-752845150e56	audio	technology	dislike	2020-06-27 09:46:48
18410	75d6b589-7fae-4a6d-b0d0-752845150e56	audio	technology	intrigued	2021-02-16 17:17:02
18411	75d6b589-7fae-4a6d-b0d0-752845150e56	audio	technology	worried	2020-11-04 20:08:31
18412	75d6b589-7fae-4a6d-b0d0-752845150e56	audio	technology	cherish	2021-01-04 04:55:11

18413 rows × 5 columns

In [13]:

```
content_reactions_types_df=pd.merge(content_reactions_df,reaction_type_df, how='left', o
content_reactions_types_df
```

Out[13]:

	Content ID	Content_Type	Category	Reaction_Type	Datetime	Sentiment	Score
0	97522e57-d9ab-4bd6-97bf-c24d952602d2	photo	studying	disgust	2020-11-07 09:43:50	negative	0.0
1	97522e57-d9ab-4bd6-97bf-c24d952602d2	photo	studying	dislike	2021-06-17 12:22:51	negative	10.0
2	97522e57-d9ab-4bd6-97bf-c24d952602d2	photo	studying	scared	2021-04-18 05:13:58	negative	15.0
3	97522e57-d9ab-4bd6-97bf-c24d952602d2	photo	studying	disgust	2021-01-06 19:13:01	negative	0.0
4	97522e57-d9ab-4bd6-97bf-c24d952602d2	photo	studying	interested	2020-08-23 12:25:58	positive	30.0
...	...	...	...	...	...	...	...
18408	75d6b589-7fae-4a6d-	audio	technology	worried	2020-10-	negative	12.0

	b0d0-752845150e56				31 04:50:14		
18409	75d6b589-7fae-4a6d-b0d0-752845150e56	audio	technology	dislike	2020-06-27 09:46:48	negative	10.0
18410	75d6b589-7fae-4a6d-b0d0-752845150e56	audio	technology	intrigued	2021-02-16 17:17:02	positive	45.0
18411	75d6b589-7fae-4a6d-b0d0-752845150e56	audio	technology	worried	2020-11-04 20:08:31	negative	12.0
18412	75d6b589-7fae-4a6d-b0d0-752845150e56	audio	technology	cherish	2021-01-04 04:55:11	positive	70.0

18413 rows × 7 columns

```
In [14]: content_reactions_types_df.isnull().sum()
```

```
Out[14]: Content ID        0
         Content_Type      0
         Category          0
         Reaction_Type    29
         Datetime         29
         Sentiment        29
         Score            29
         dtype: int64
```

```
In [15]: null_columns=content_reactions_types_df.columns[content_reactions_types_df.isnull().any(
         null_columns.tolist()
```

```
Out[15]: ['Reaction_Type', 'Datetime', 'Sentiment', 'Score']
```

```
In [16]: content_reactions_types_df[content_reactions_types_df.isnull().any(axis=1)][content_reac
```

Out[16]:

	Content ID	Content_Type	Category	Reaction_Type	Datetime	Sentiment	Score
184	46fb701d-6c26-458e-ada3-2ebe5dbba01f	audio	public speaking	NaN	NaT	NaN	NaN
1105	9dd95c34-8b39-4776-a232-412512329c3f	audio	fitness	NaN	NaT	NaN	NaN
1814	b1ba68bc-fa4c-4a36-98a1-4d4a381ef873	video	animals	NaN	NaT	NaN	NaN
2517	6efd3911-1705-49dc-aa7b-994ce83a7387	photo	fitness	NaN	NaT	NaN	NaN
2931	9fd8c6fc-1c8f-4a1d-86ec-cd1c71e044e1	video	public speaking	NaN	NaT	NaN	NaN

```
In [17]: content_reactions_types_df.dropna(inplace=True)
         content_reactions_types_df.isnull().sum()
```

```
Out[17]: Content ID       0
         Content_Type     0
         Category         0
         Reaction_Type    0
         Datetime         0
         Sentiment        0
         Score            0
         dtype: int64
```

In [18]:
```
content_reactions_types_df.sort_values(['Content ID', 'Datetime'], ascending=[True, True
```

Out[18]:

	Content ID	Content_Type	Category	Reaction_Type	Datetime	Sentiment	Score
14777	004e820e-49c3-4ba2-9d02-62db0065410c	audio	tennis	heart	2021-03-09 08:50:44	positive	60.0
11589	00d0cdf9-5919-4102-bf84-ebde253c3cd2	audio	healthy eating	indifferent	2020-06-21 10:18:13	neutral	20.0
11580	00d0cdf9-5919-4102-bf84-ebde253c3cd2	audio	healthy eating	interested	2020-06-23 09:36:11	positive	30.0
11586	00d0cdf9-5919-4102-bf84-ebde253c3cd2	audio	healthy eating	hate	2020-06-24 11:46:02	negative	5.0
11584	00d0cdf9-5919-4102-bf84-ebde253c3cd2	audio	healthy eating	cherish	2020-07-03 06:23:44	positive	70.0
...	...	...	...	...	...	...	...
3870	ff883828-a610-492d-8635-8a777eaad25f	photo	education	love	2021-05-08 08:06:23	positive	65.0
3888	ff883828-a610-492d-8635-8a777eaad25f	photo	education	worried	2021-05-17 07:17:46	negative	12.0
3885	ff883828-a610-492d-8635-8a777eaad25f	photo	education	heart	2021-06-04 06:09:23	positive	60.0
3877	ff883828-a610-492d-8635-8a777eaad25f	photo	education	like	2021-06-07 17:04:06	positive	50.0
3853	ff883828-a610-492d-8635-8a777eaad25f	photo	education	want	2021-06-14 08:39:48	positive	70.0

18384 rows × 7 columns

In [19]:
```
content_reactions_types_df.sort_values(['Content ID', 'Datetime'], ascending=[True, Fals
```

Out[19]:

	Content ID	Content_Type	Category	Reaction_Type	Datetime	Sentiment	Score
14777	004e820e-49c3-4ba2-9d02-62db0065410c	audio	tennis	heart	2021-03-09 08:50:44	positive	60.0
11561	00d0cdf9-5919-4102-bf84-ebde253c3cd2	audio	healthy eating	dislike	2021-06-18 02:00:03	negative	10.0
11587	00d0cdf9-5919-4102-bf84-ebde253c3cd2	audio	healthy eating	cherish	2021-05-29 12:27:00	positive	70.0
11556	00d0cdf9-5919-4102-bf84-ebde253c3cd2	audio	healthy eating	worried	2021-05-14 06:14:07	negative	12.0
11558	00d0cdf9-5919-4102-bf84-ebde253c3cd2	audio	healthy eating	scared	2021-05-09 17:16:40	negative	15.0

...	...	...	...	...	...	...	...
3854	ff883828-a610-492d-8635-8a777eaad25f	photo	education	hate	2020-08-04 13:24:12	negative	5.0
3864	ff883828-a610-492d-8635-8a777eaad25f	photo	education	intrigued	2020-08-02 01:06:33	positive	45.0
3868	ff883828-a610-492d-8635-8a777eaad25f	photo	education	intrigued	2020-07-21 21:44:12	positive	45.0
3865	ff883828-a610-492d-8635-8a777eaad25f	photo	education	disgust	2020-07-21 12:55:02	negative	0.0
3866	ff883828-a610-492d-8635-8a777eaad25f	photo	education	hate	2020-07-21 07:39:29	negative	5.0

18384 rows × 7 columns

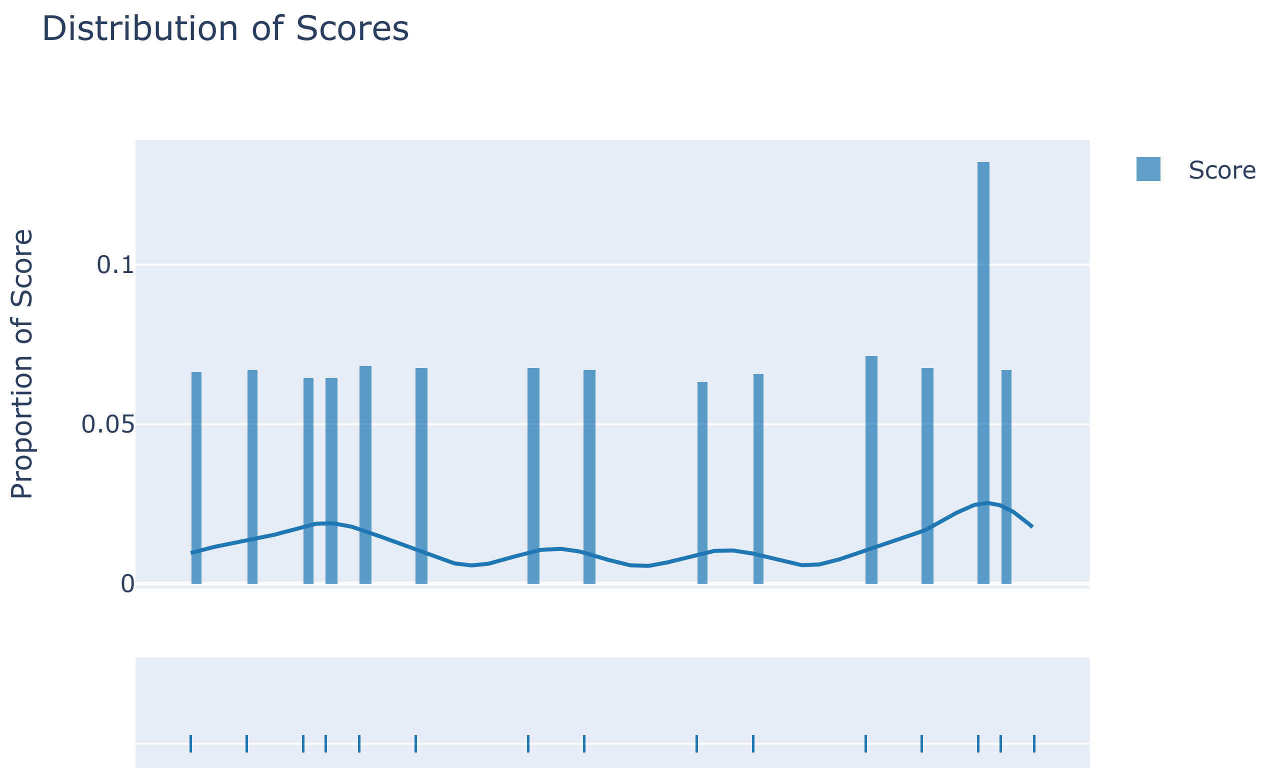
Score Distplot

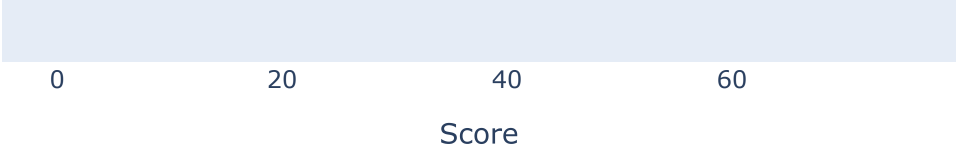
In [20]:

```
fig = ff.create_distplot([content_reactions_types_df['Score']], ['Score']).update_layout
    xaxis_title="Score", yaxis_title="Proportion of Score",
    title="Distribution of Scores"
)
fig.show()


rts = content_reactions_types_df['Score'].value_counts()
rtsp = content_reactions_types_df['Score'].value_counts(normalize=True)

rtsp_df=pd.DataFrame({"count of Score":rts,"% of Score":rtsp*100})
rtsp_df
```





Out[20]:

	count of Score	% of Score
70.0	2273	12.364012
60.0	1225	6.663403
15.0	1174	6.385988
30.0	1169	6.358790
20.0	1167	6.347911
75.0	1167	6.347911
65.0	1165	6.337032
35.0	1157	6.293516
5.0	1153	6.271758
72.0	1148	6.244560
0.0	1142	6.211923
50.0	1132	6.157528
12.0	1112	6.048738
10.0	1109	6.032419
45.0	1091	5.934508

Grouping w.r.t to Content ID and Category Reactions

In [21]:

```
gr = content_reactions_types_df.groupby(['Content ID','Category','Content_Type'])
gg_reactions_types_df = pd.concat([
    # equivalent to count the contents and category in df
    gr.size().to_frame(name='Count of Content ID & Category'),
    # equivalent to dummy then sum the dummy reaction types columns
    gr['Reaction_Type'].value_counts().unstack(fill_value=0).add_suffix('_reactions'),
    # equivalent to dummy then sum the dummy sentiment columns
    gr['Sentiment'].value_counts().unstack(fill_value=0).add_suffix('_sentiments'),
    # sum the score to get the total score
    gr['Score'].sum().to_frame(name='Total Score')], axis=1)
gg_reactions_types_df.reset_index(inplace=True)
gg_reactions_types_df
```

Out[21]:

	Content ID	Category	Content_Type	Count of Content ID & Category	adore_reactions	cherish_reactions	disgust_reacti
0	004e820e-49c3-4ba2-9d02-62db0065410c	tennis	audio	1	0	0	
1	00d0cdf9-5919-4102-bf84-ebde253c3cd2	healthy eating	audio	42	3	3	

2	01396602-c759-4a17-90f0-8f9b3ca11b30	tennis	GIF	36	0	1
3	01ab84dd-6364-4236-abbb-3f237db77180	food	video	1	0	0
4	01aff5ec-2aa8-412e-99ec-526f0f9a6d5e	fitness	video	39	4	3
...	...	...	...	...	...	...
767	fdca8d15-966b-4825-8133-1fafc5c1f9fc	dogs	video	25	3	2
768	fe06b730-b1f8-4f55-af1a-52487d8f1ec6	animals	video	23	2	4
769	fea8d77c-fd0b-4678-868f-fbae567642f3	science	GIF	6	1	1
770	fea9077f-2fe7-43bd-aaef-dc2619988d94	culture	photo	6	0	1
771	ff883828-a610-492d-8635-8a777eaad25f	education	photo	44	2	4

772 rows × 24 columns

In [22]:

```
display(gg_reactions_types_df.head())
display(print("No of Rows",gg_reactions_types_df.shape[0],", No of Columns",gg_reactions
```

	Content ID	Category	Content_Type	Count of Content ID & Category	adore_reactions	cherish_reactions	disgust_reaction
0	004e820e-49c3-4ba2-9d02-62db0065410c	tennis	audio	1	0	0	
1	00d0cdf9-5919-4102-bf84-ebde253c3cd2	healthy eating	audio	42	3	3	
2	01396602-c759-4a17-90f0-8f9b3ca11b30	tennis	GIF	36	0	1	
3	01ab84dd-6364-4236-abbb-3f237db77180	food	video	1	0	0	

4	01aff5ec-2aa8-412e-99ec-526f0f9a6d5e	fitness	video	39	4	3

5 rows × 24 columns

```
No of Rows 772 , No of Columns 24
None
```

Merging Content and Reaction Types Data

In [23]:

```
# Reaction Sentiment Score List
reaction_sentiment_list=['adore_reactions',
 'cherish_reactions',
 'disgust_reactions',
 'dislike_reactions',
 'hate_reactions',
 'heart_reactions',
 'indifferent_reactions',
 'interested_reactions',
 'intrigued_reactions',
 'like_reactions',
 'love_reactions',
 'peeking_reactions',
 'scared_reactions',
 'super love_reactions',
 'want_reactions',
 'worried_reactions',
 'negative_sentiments',
 'neutral_sentiments',
 'positive_sentiments',
 'Total Score']
```

Grouping and getting Sum by Categories content and reactions and score

In [24]:

```
content_reactions_types_df_cat=gg_reactions_types_df.groupby(['Category'])[reaction_sent
content_reactions_types_df_cat
```

Out[24]:

	Category	adore_reactions	cherish_reactions	disgust_reactions	dislike_reactions	hate_reactions	he
14	travel	85	85	97	96	77	
9	science	91	77	85	81	72	
7	healthy eating	87	87	74	78	88	
0	animals	78	84	80	72	91	
1	cooking	79	67	88	77	82	
2	culture	66	80	74	74	79	
6	food	63	86	74	68	83	
12	technology	81	67	75	63	71	
4	education	66	68	55	68	76	
13	tennis	59	73	85	84	77	
10	soccer	71	58	66	62	61	
5	fitness	68	63	74	76	58	

3	dogs	65	55	67	59	70
15	veganism	61	62	47	51	63
8	public speaking	68	47	51	57	46
11	studying	60	60	50	43	59

16 rows × 21 columns

```
In [25]: content_reactions_types_df_cat[["Category","Total Score"]][:5]
```

Out[25]:

	Category	Total Score
14	travel	53935.0
9	science	53657.0
7	healthy eating	52745.0
0	animals	52443.0
1	cooking	49681.0

Conclusion

Based on the above table, it looks like travel has the highest total score with 53,935 score followed by science, healthy eating, animals and cooking. It does make sense based on the following reasons

- Travel:

    - Travel content is highly popular on social media because it offers a glimpse into new cultures and beautiful destinations. Travel posts often feature stunning photography or video of picturesque locations, which can be highly shareable and visually engaging. Additionally, travel content often provides educational value, with bloggers and influencers sharing tips and recommendations for travel planning, packing, and budgeting. Travel content also has emotional appeal, as followers may feel inspired or nostalgic when seeing beautiful photos or hearing about new experiences.
- Science:

    - Science content can be highly engaging on social media due to its educational value and potential for creating wonder and amazement. Science posts can include images of outer space or microscopic views of cells and organisms, which can be visually stunning and highly shareable. Science content can also be highly informative, with bloggers and scientists sharing new discoveries and insights on topics ranging from medicine to climate change. Additionally, science content can help people better understand the world around them, creating a sense of curiosity and excitement.
- Healthy Eating:

    - Healthy eating content has become increasingly popular on social media, as more people focus on living a healthy lifestyle. Healthy eating posts often feature beautiful images of colorful, nutrient-dense meals, which can be highly visually appealing and shareable. Additionally, healthy eating content provides educational value, with bloggers and nutritionists sharing recipes, cooking tips, and information on healthy eating habits. This type of content can also inspire followers to make positive changes in their own lives, leading to increased engagement and loyalty.

- Animals:
    - Animal content has been a staple of social media for many years, and for good reason. Animal posts often feature adorable or funny images or videos of pets or wildlife, which can be highly shareable and engaging. Additionally, animal content can help create a sense of emotional connection, as people may feel compassion, empathy, or joy when seeing animals. Animal content can also provide educational value, with bloggers and animal experts sharing information on animal behavior, care, and conservation.
- Cooking:
    - Cooking content has become increasingly popular on social media, with many people turning to online platforms for recipe inspiration and cooking tutorials. Cooking posts often feature beautiful images or videos of delicious dishes, which can be highly visually engaging and shareable. Additionally, cooking content provides educational value, with bloggers and chefs sharing recipes, cooking tips, and information on different cuisines and ingredients. This type of content can also inspire followers to get in the kitchen and try new things, creating a sense of community and engagement.

In summary, travel, science, healthy eating, animals, and cooking categories are popular on social media due to their broad appeal, educational value, visual appeal, emotional engagement, and personal connection. By providing valuable and inspiring content, creators in these categories can build loyal and engaged followers.

How many unique categories are there?

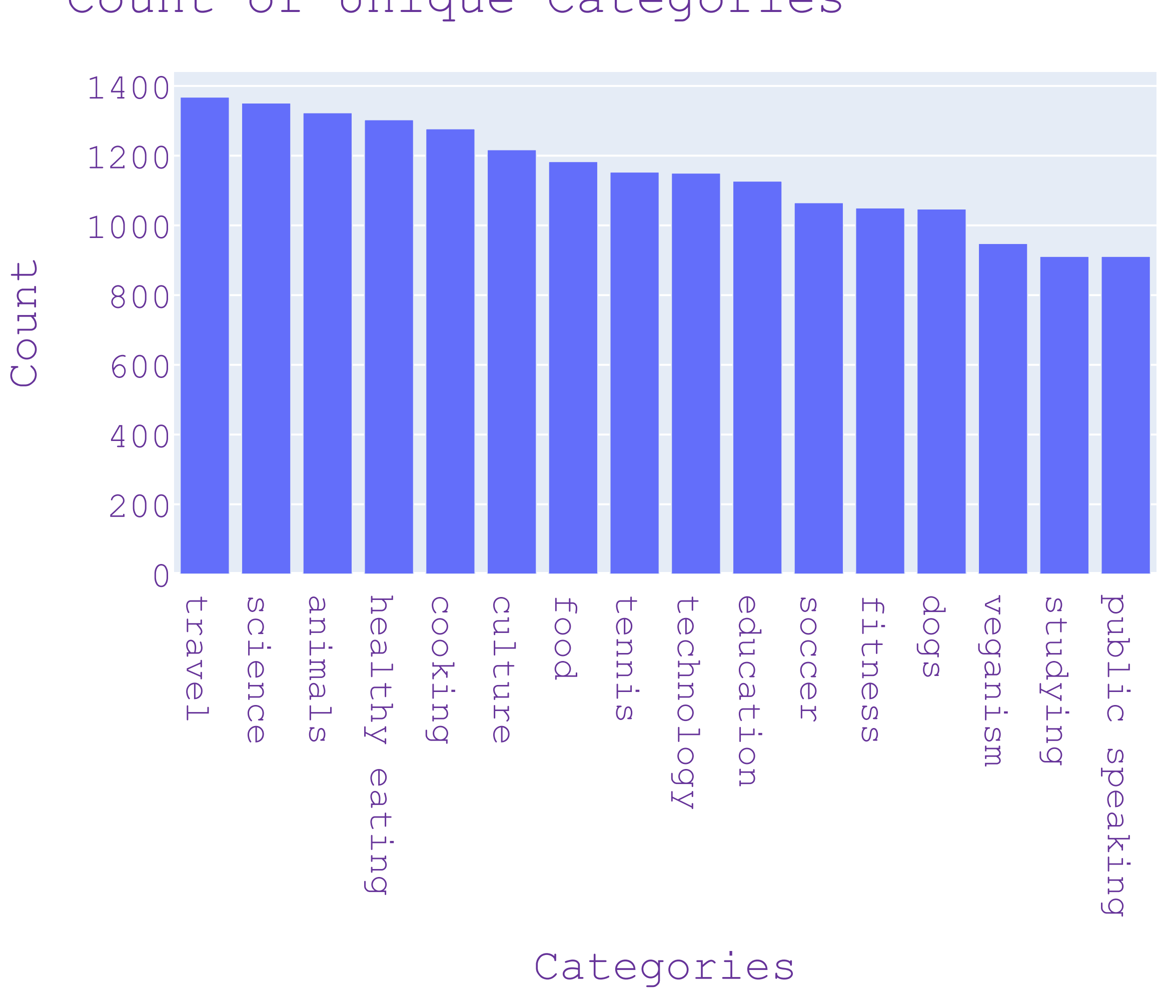
In [26]:

```
display(list(content_reactions_types_df['Category'].unique()))
display(len(content_reactions_types_df['Category'].unique()))
```

```
['studying',
 'healthy eating',
 'technology',
 'food',
 'dogs',
 'soccer',
 'public speaking',
 'tennis',
 'travel',
 'education',
 'science',
 'veganism',
 'cooking',
 'animals',
 'fitness',
 'culture']
16
```

In [27]:

```
fig = px.bar(x=content_reactions_types_df['Category'].value_counts().index.values, y=con
fig.update_layout(
title="Count of Unique Categories",
xaxis_title="Categories",
yaxis_title="Count",
font=dict(family="Courier New, monospace",size=18,color="RebeccaPurple"))
fig.show()
```



How many reactions are there to the most popular category?

In [28]:

```
content_reactions_types_df_cat_index=content_reactions_types_df_cat.set_index('Category'
content_reactions_types_df_cat_index.sort_values(by="travel",axis=1,ascending=False).ilo
```

Out[28]:

	disgust_reactions	dislike_reactions	want_reactions	like_reactions	peeking_reactions	heart_rea
Category						
travel	97	96	94	92	87	

In [29]:

```
viz_r_c=content_reactions_types_df_cat_index.sort_values(by="travel",axis=1,ascending=Fa
viz_r_c
```

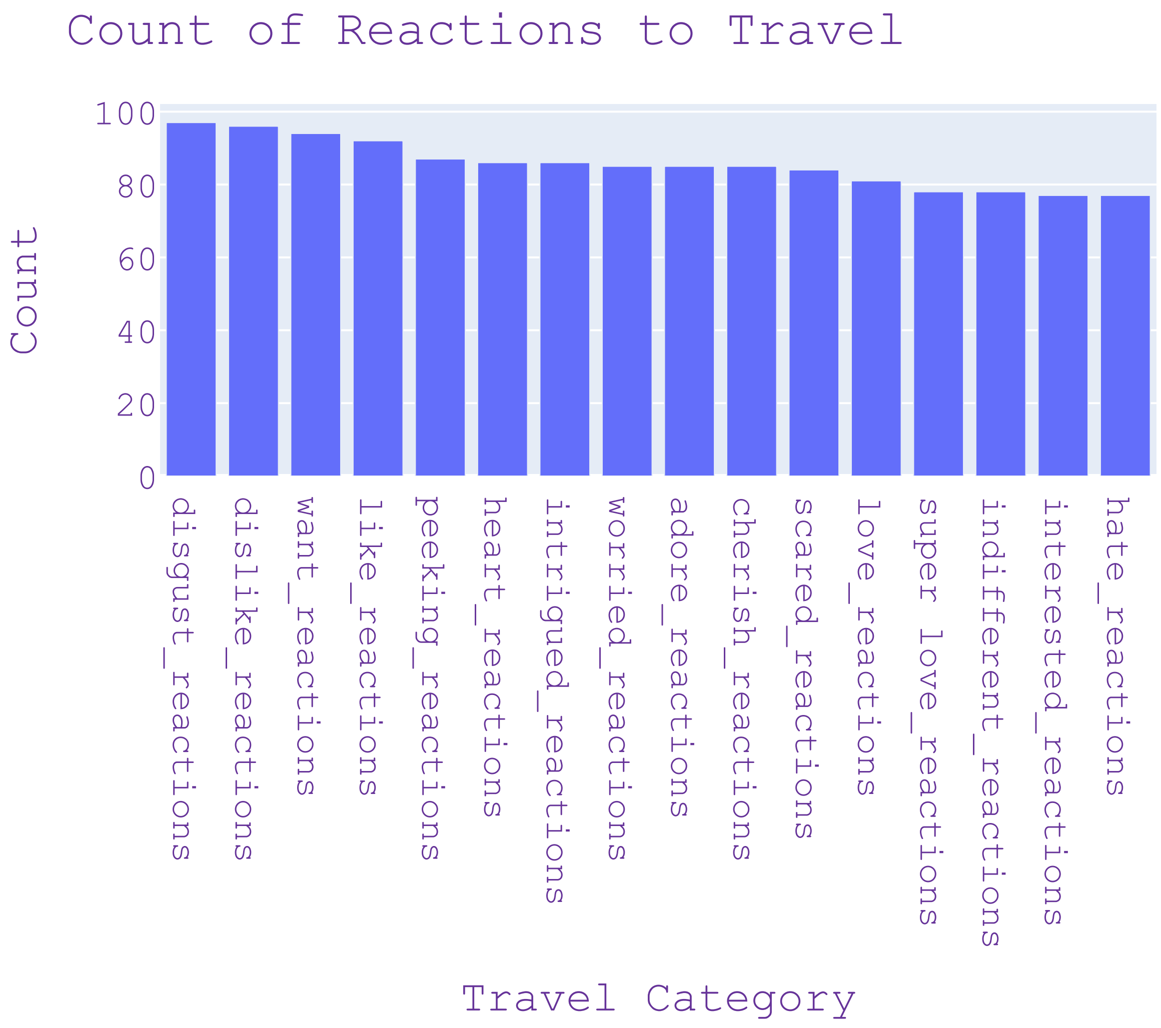
Out[29]:

Category	travel
disgust_reactions	97
dislike_reactions	96
want_reactions	94
like_reactions	92
peeking_reactions	87
heart_reactions	86
intrigued_reactions	86
worried_reactions	85
adore_reactions	85

cherish_reactions	85
scared_reactions	84
love_reactions	81
super love_reactions	78
indifferent_reactions	78
interested_reactions	77
hate_reactions	77

In [30]:

```
fig = px.bar(x=viz_r_c.index.values, y=list(itertools.chain(*viz_r_c.values.tolist())))
fig.update_layout(
title="Count of Reactions to Travel",
xaxis_title="Travel Category",
yaxis_title="Count",
font=dict(family="Courier New, monospace",size=18,color="RebeccaPurple"))
fig.show()
```



What was the month with the most posts?

In [31]:

```
content_reactions_types_df["Month_Year"] = content_reactions_types_df['Datetime'].dt.to_
content_reactions_types_df["Month_Year"] = content_reactions_types_df["Month_Year"].asty
content_reactions_types_df["Month_Year"].value_counts()
```

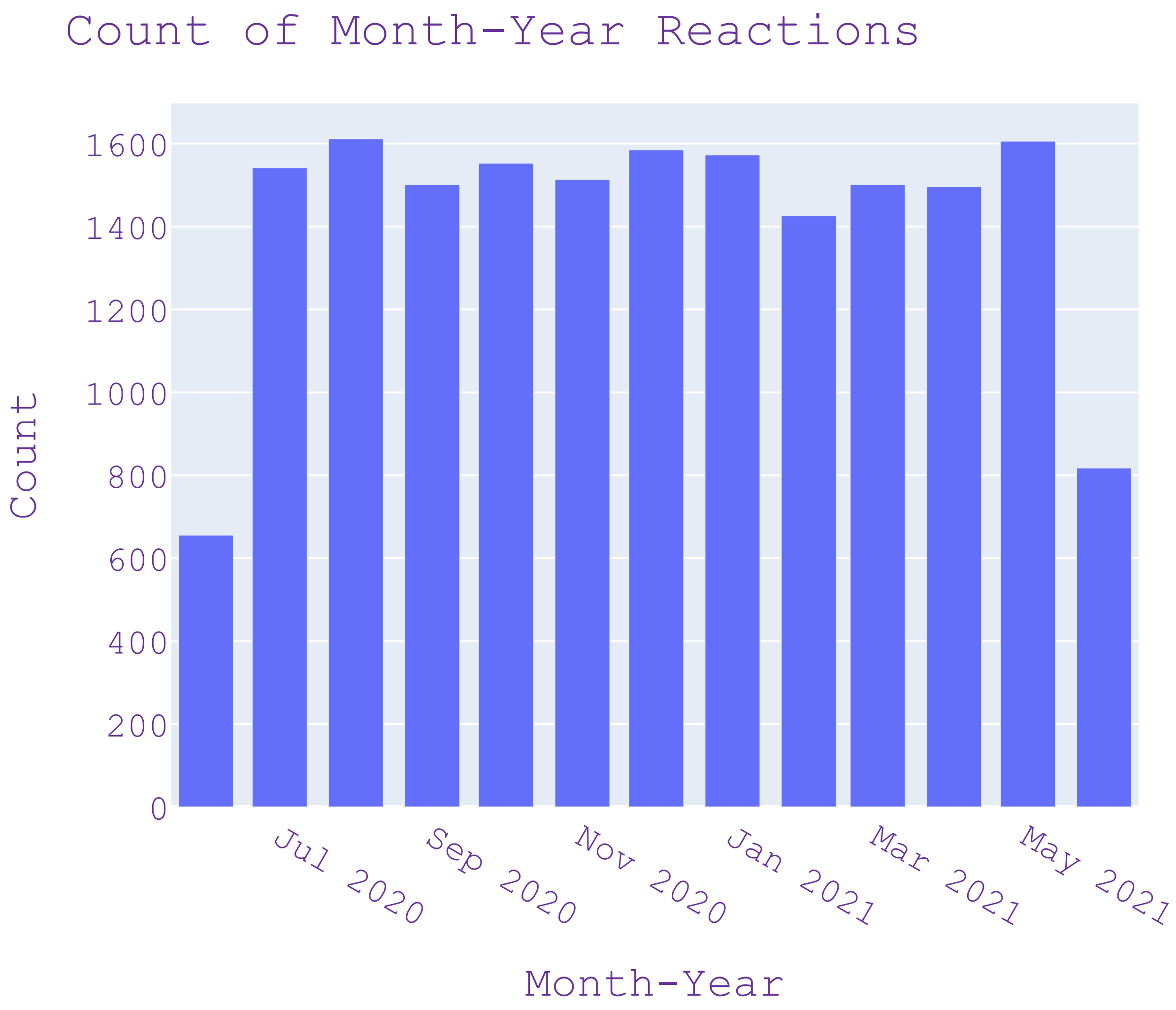
Out[31]:

```
2020-08    1612
2021-05    1606
2020-12    1585
```

```
2021-01    1573
2020-10    1553
2020-07    1542
2020-11    1514
2021-03    1502
2020-09    1501
2021-04    1496
2021-02    1426
2021-06     818
2020-06     656
Name: Month_Year, dtype: int64
```

In [32]:

```
fig = px.bar(x=content_reactions_types_df["Month_Year"].value_counts().index.values, y=c
fig.update_layout(
title="Count of Month-Year Reactions",
xaxis_title="Month-Year",
yaxis_title="Count",
font=dict(family="Courier New, monospace",size=18,color="RebeccaPurple"))
fig.show()
content_reactions_types_df["Month_Year"].value_counts()
```



Out[32]:

```
2020-08    1612
2021-05    1606
2020-12    1585
2021-01    1573
2020-10    1553
2020-07    1542
2020-11    1514
2021-03    1502
2020-09    1501
2021-04    1496
2021-02    1426
2021-06     818
```

```
2020-06     656
Name: Month_Year, dtype: int64
```

Extras and Data Visualization

Grouping and getting Sum by Content_Type content, reactions and score

In [33]:
```
content_reactions_types_df_cont_type=gg_reactions_types_df.groupby(['Content_Type'])[rea
content_reactions_types_df_cont_type
```

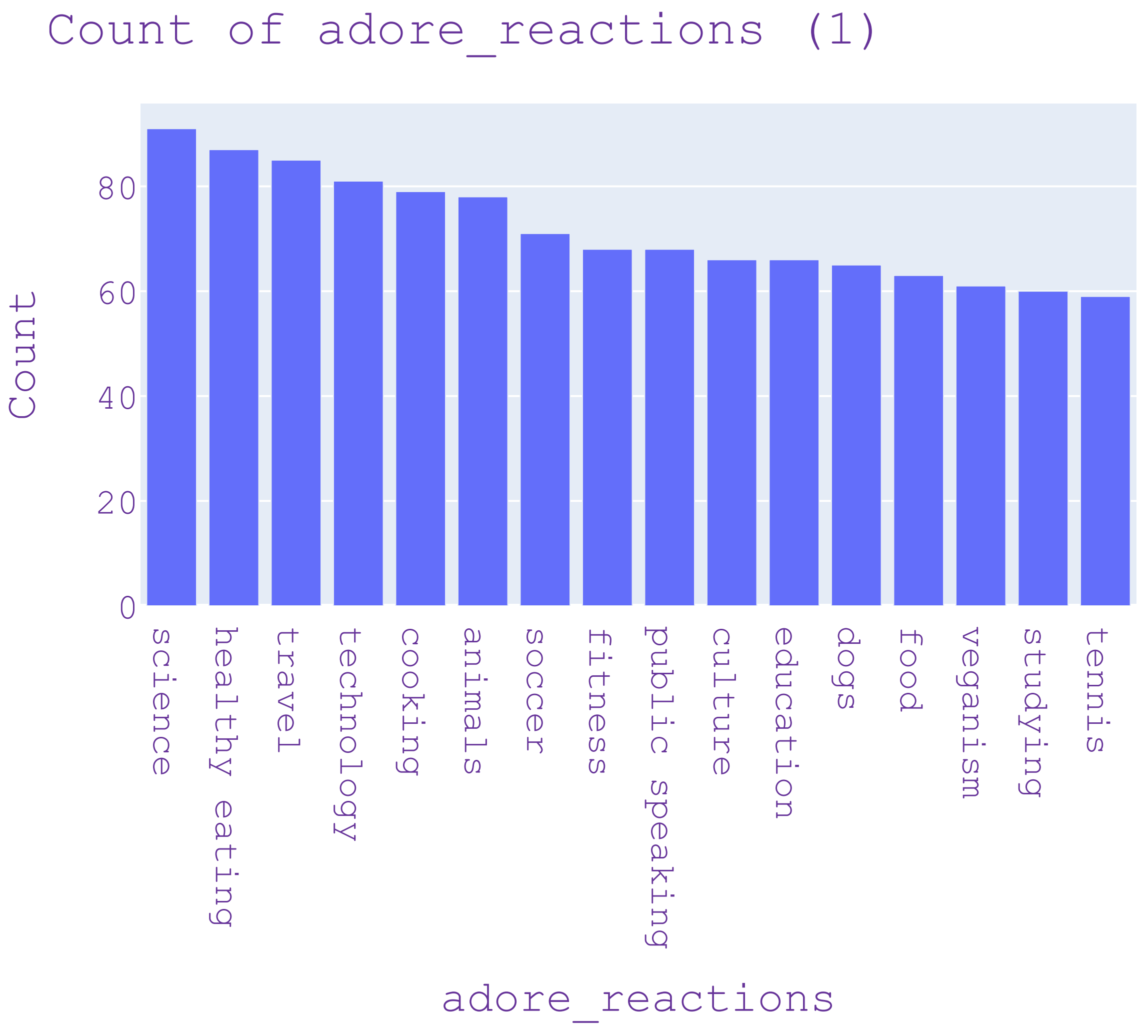
Out[33]:

	Content_Type	adore_reactions	cherish_reactions	disgust_reactions	dislike_reactions	hate_reactions
2	photo	324	301	321	280	312
3	video	284	284	287	301	268
0	GIF	285	261	274	268	313
1	audio	255	273	260	260	260

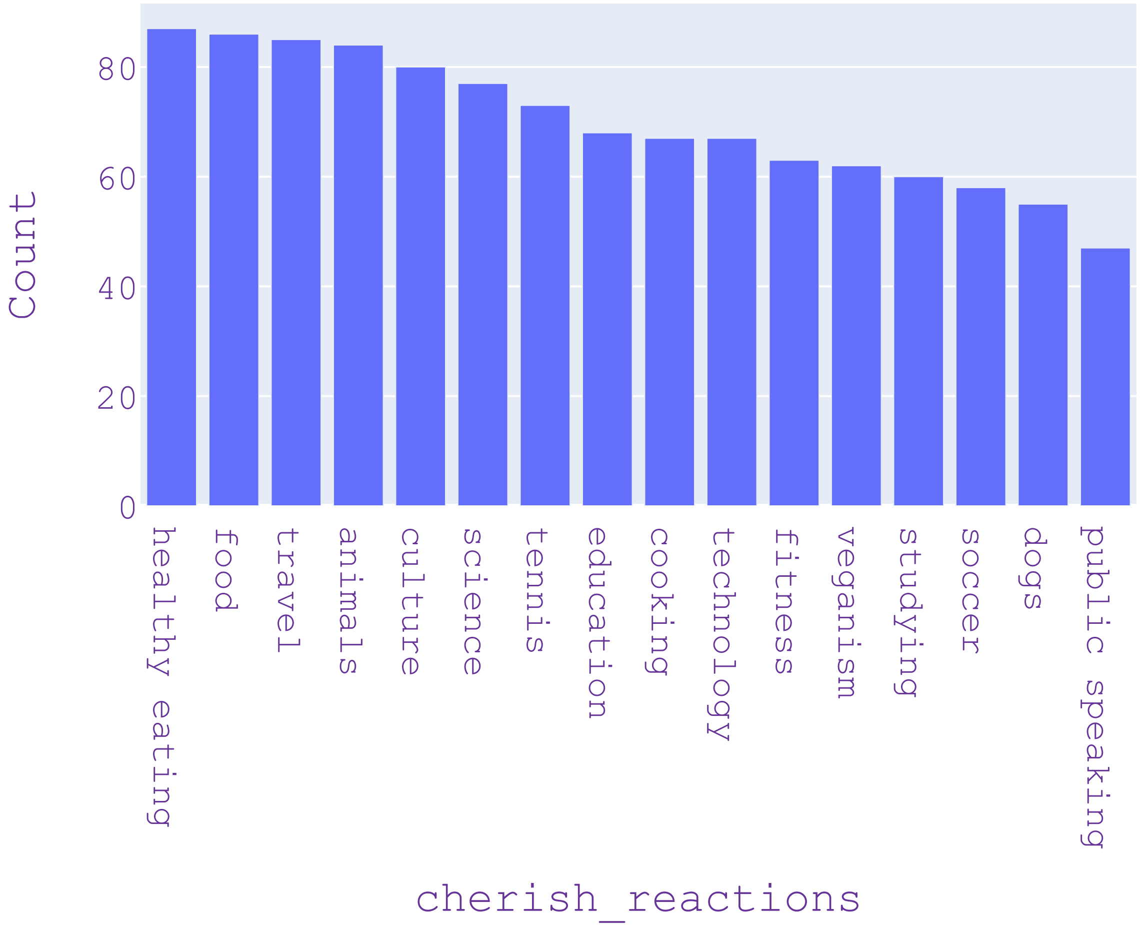
4 rows × 21 columns

Displaying above data visually for Category

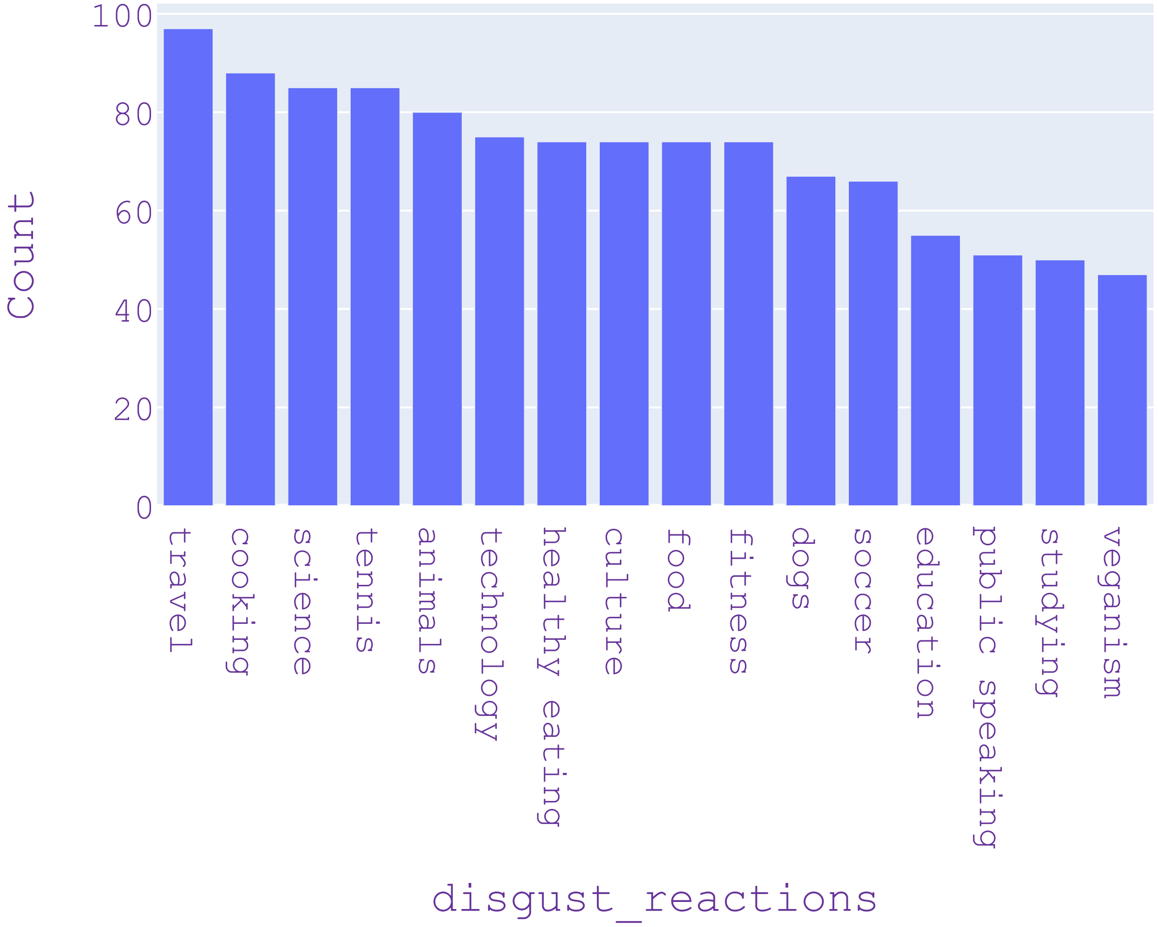
In [34]:
```
reaction_count(content_reactions_types_df_cat,'Category')
```


Count of adore_reactions (1)

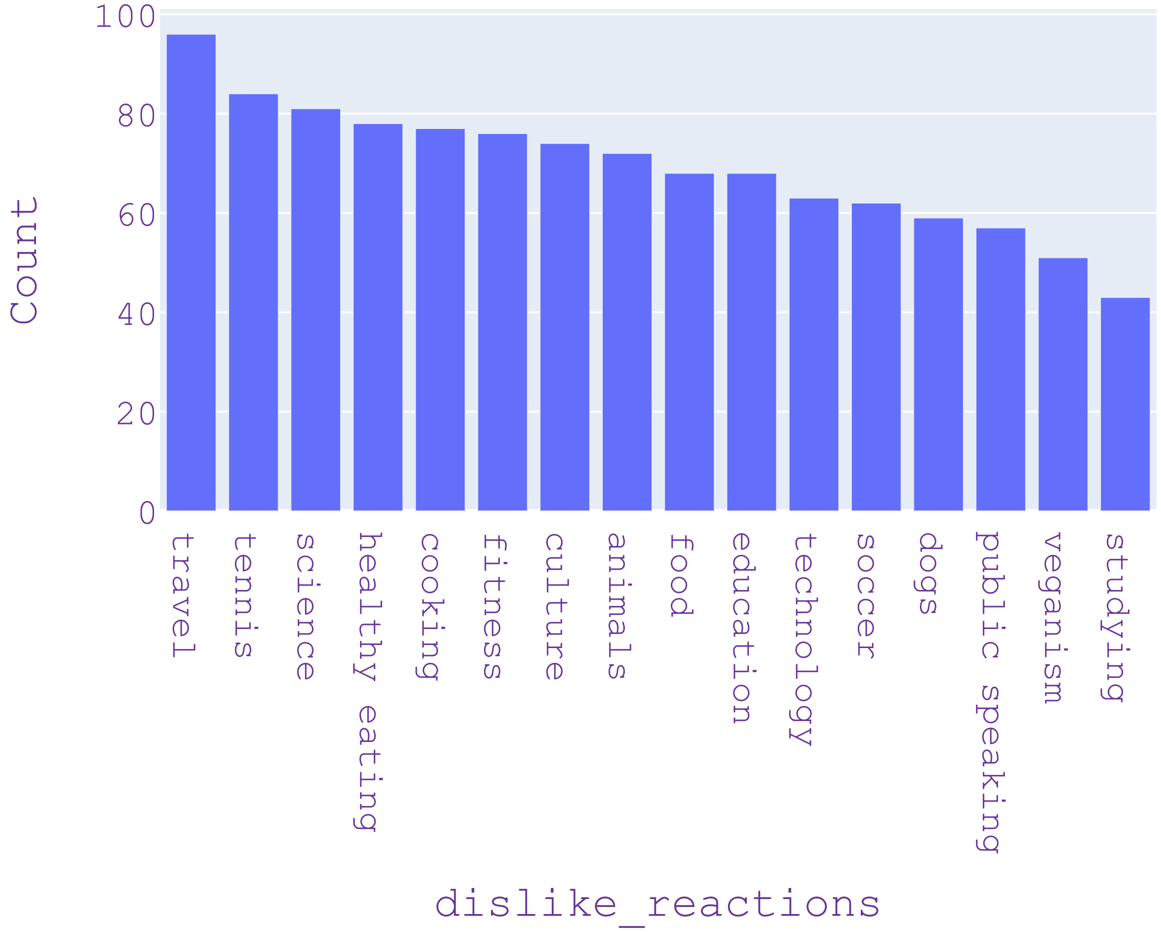
Count of cherish_reactions (2)



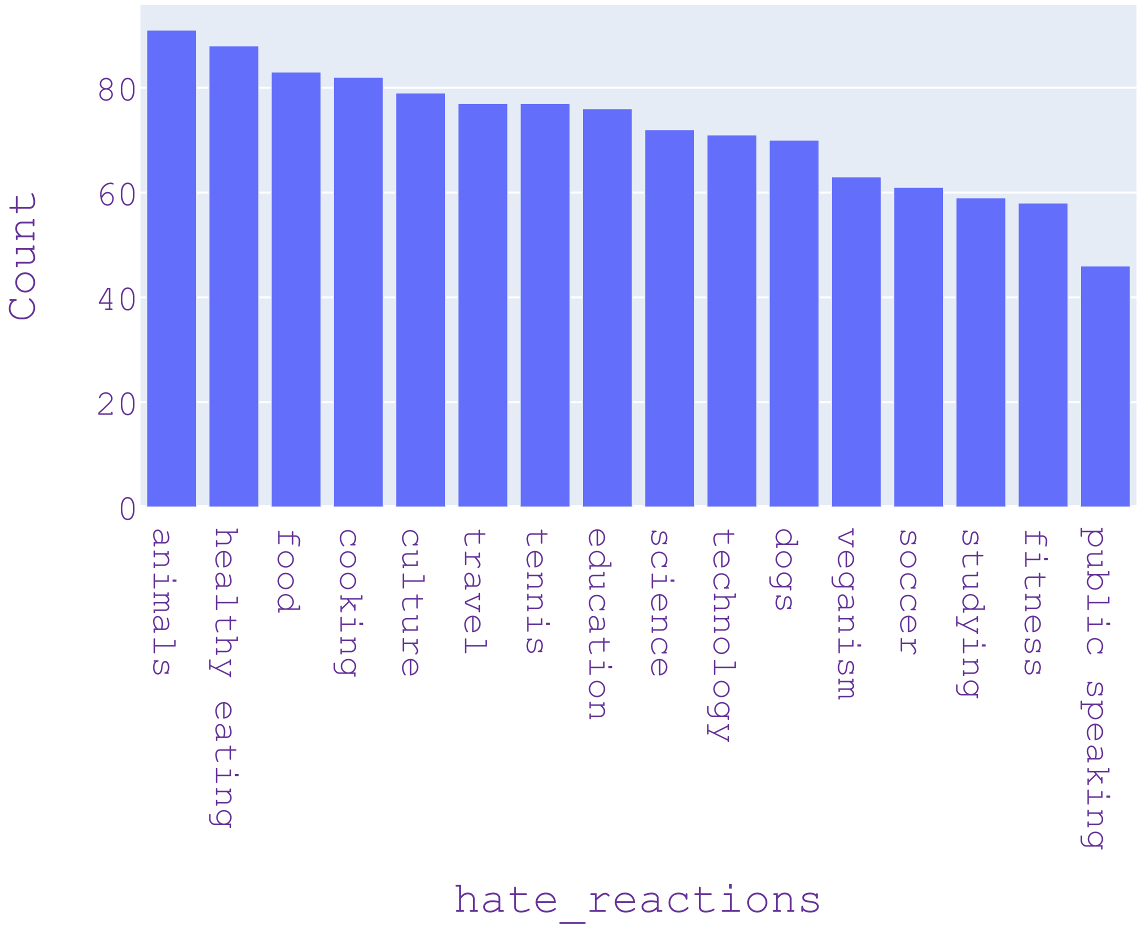
Count of disgust_reactions (3)



Count of dislike_reactions (4)

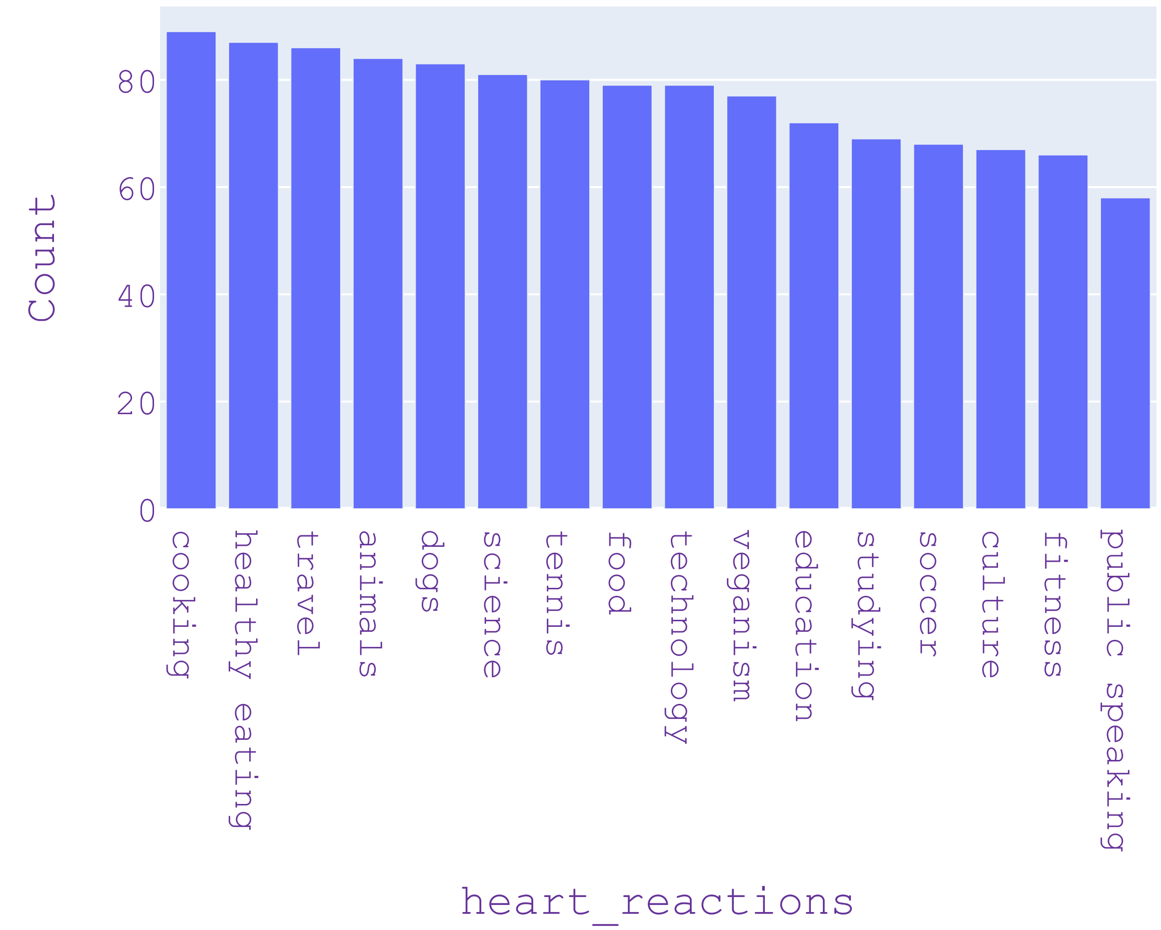


Count of hate_reactions (5)

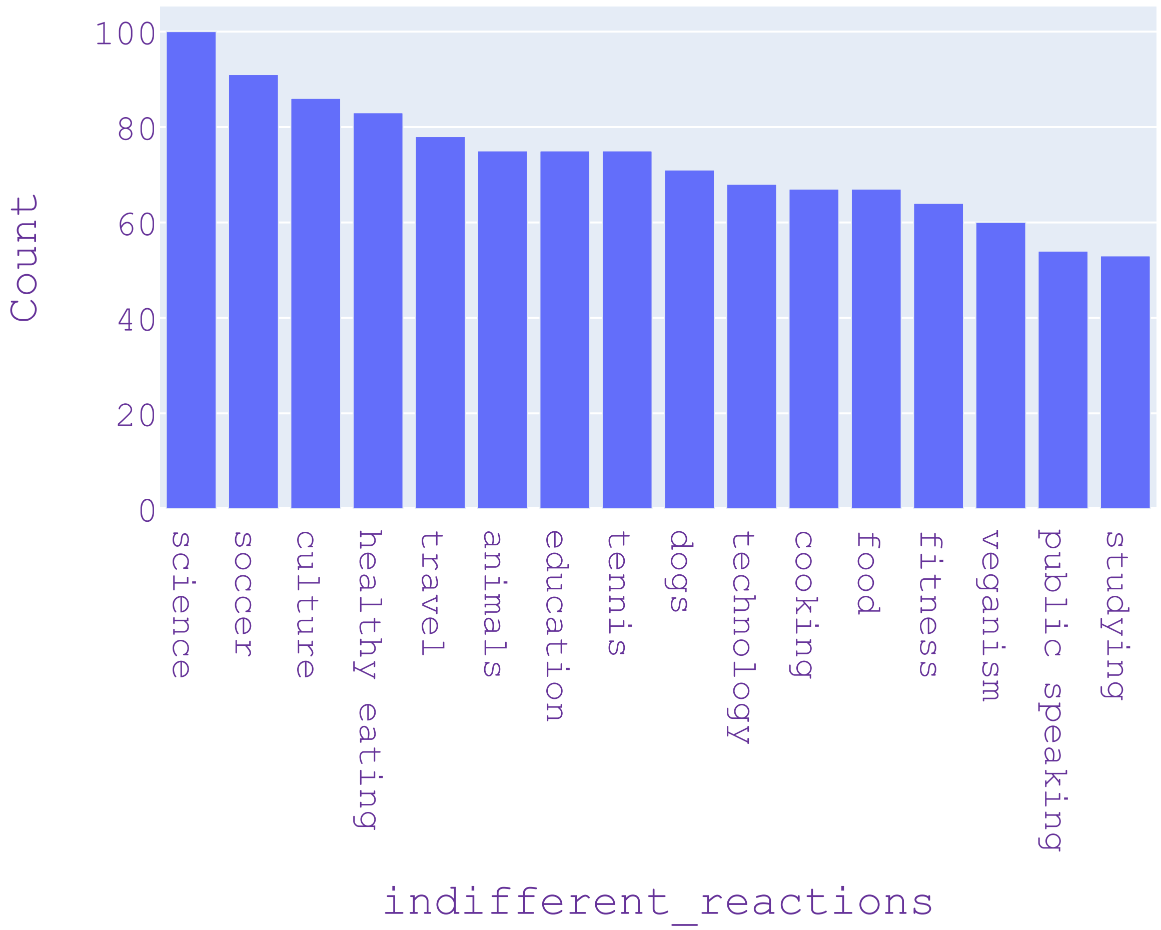


Count of heart_reactions (6)

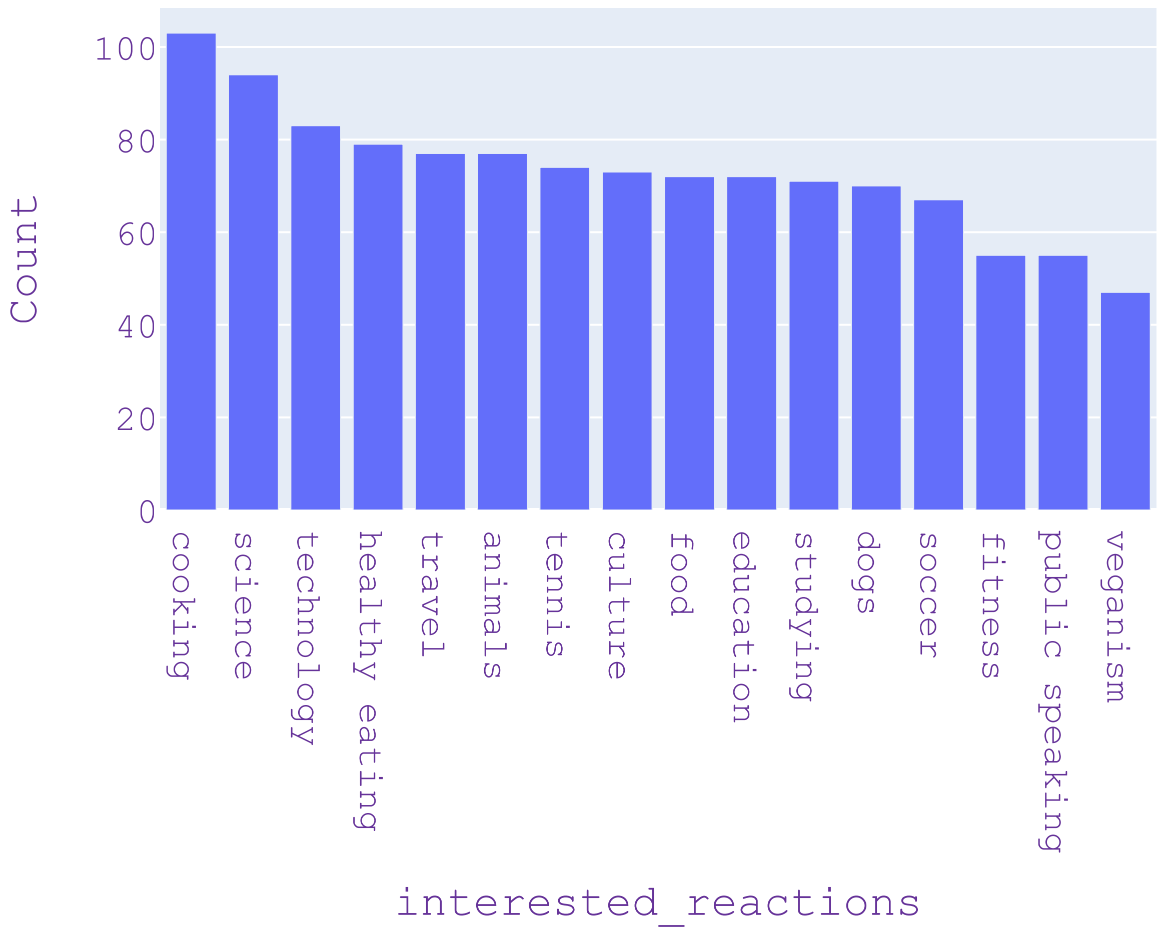
Count of heart_reactions (6)



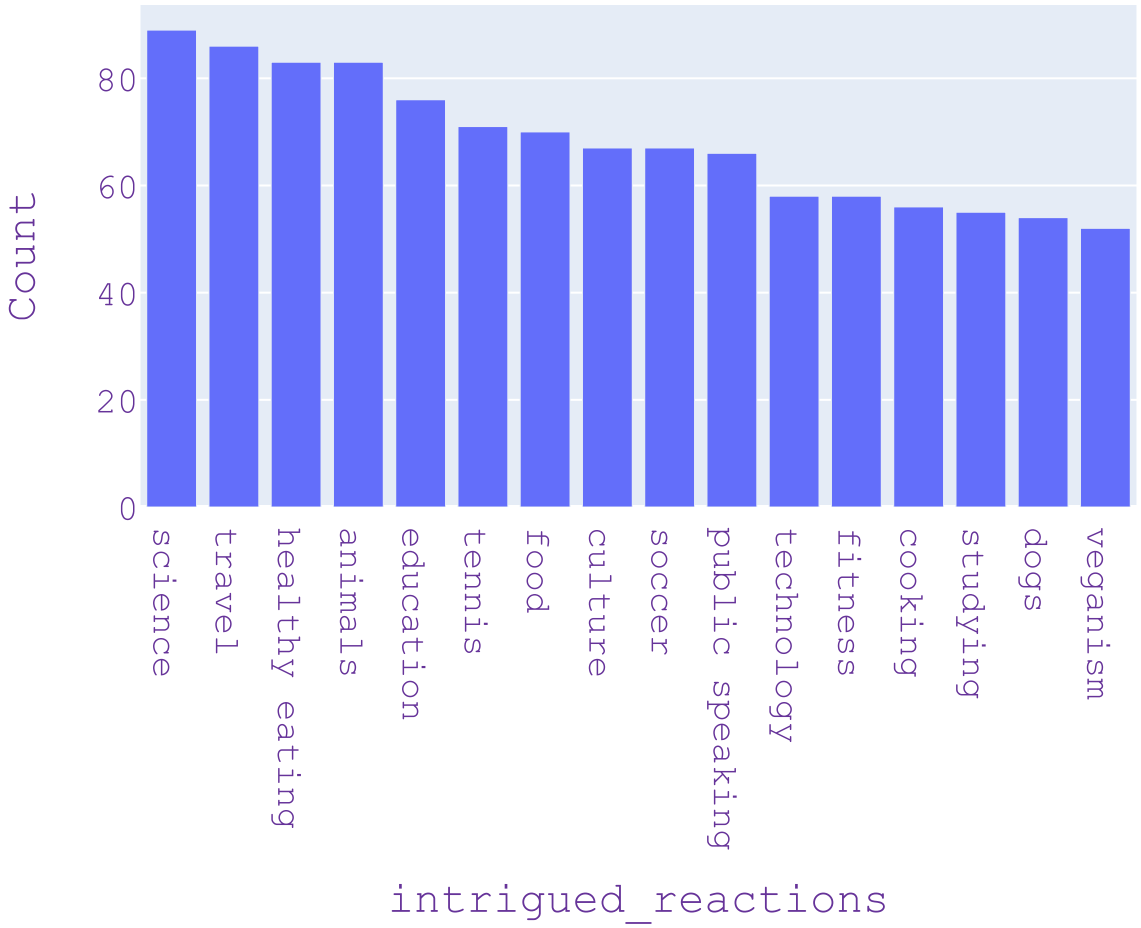
Count of indifferent_reactions (7)



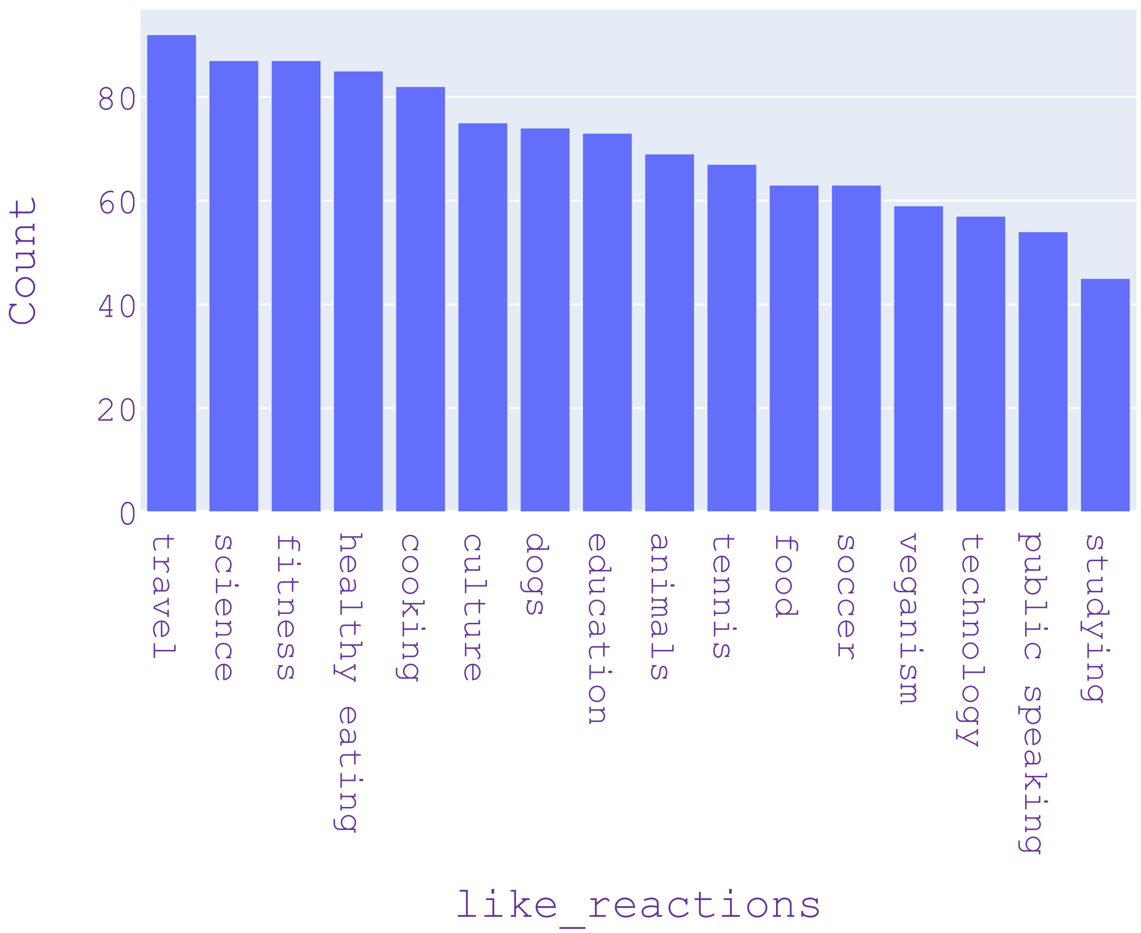
Count of interested_reactions (8)



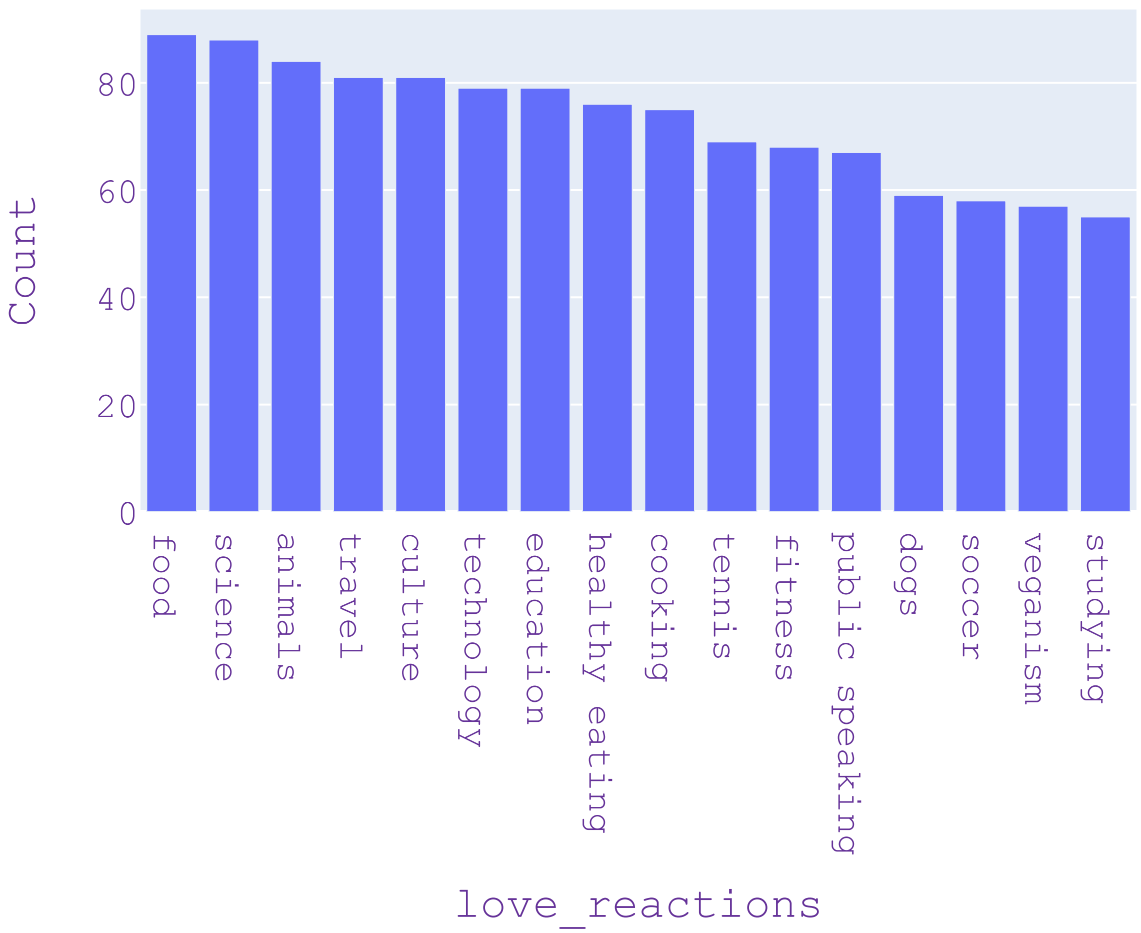
Count of intrigued_reactions (9)



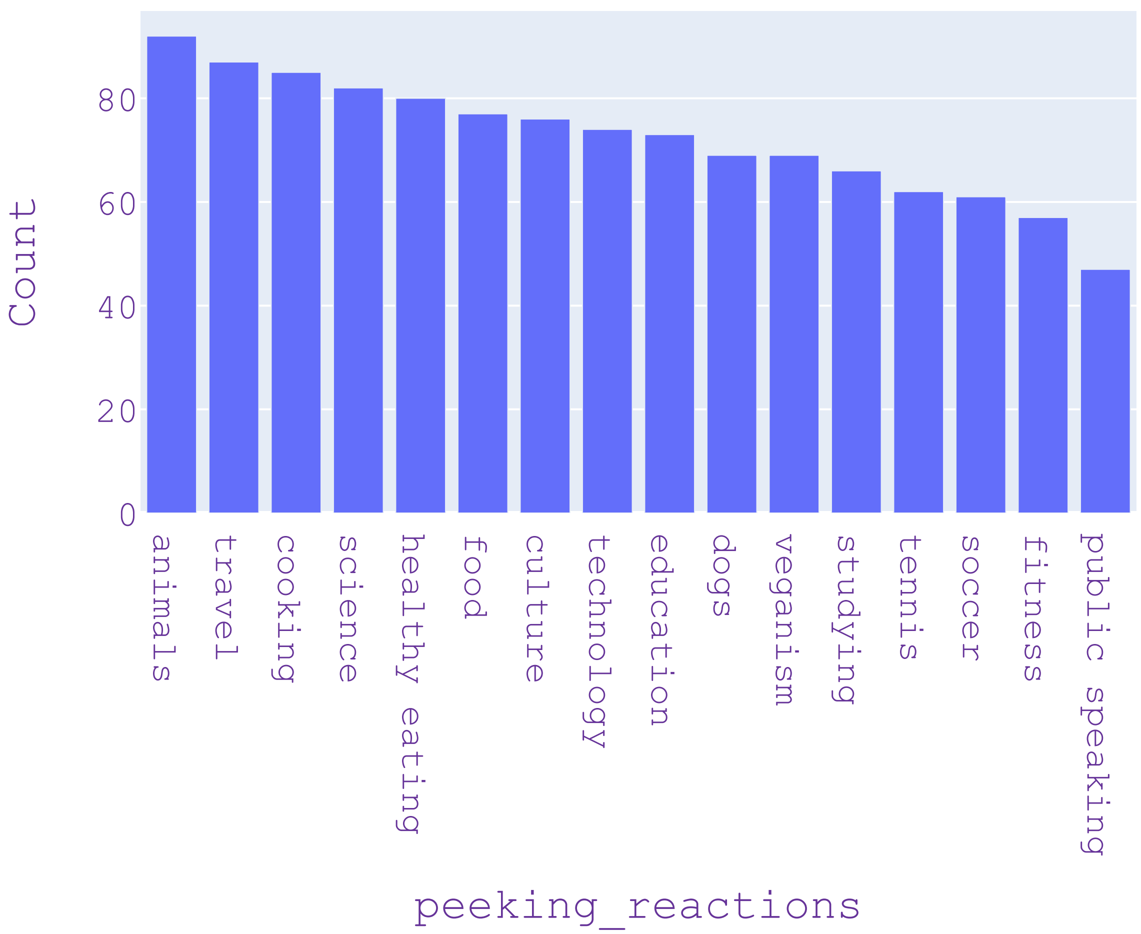
Count of like_reactions (10)



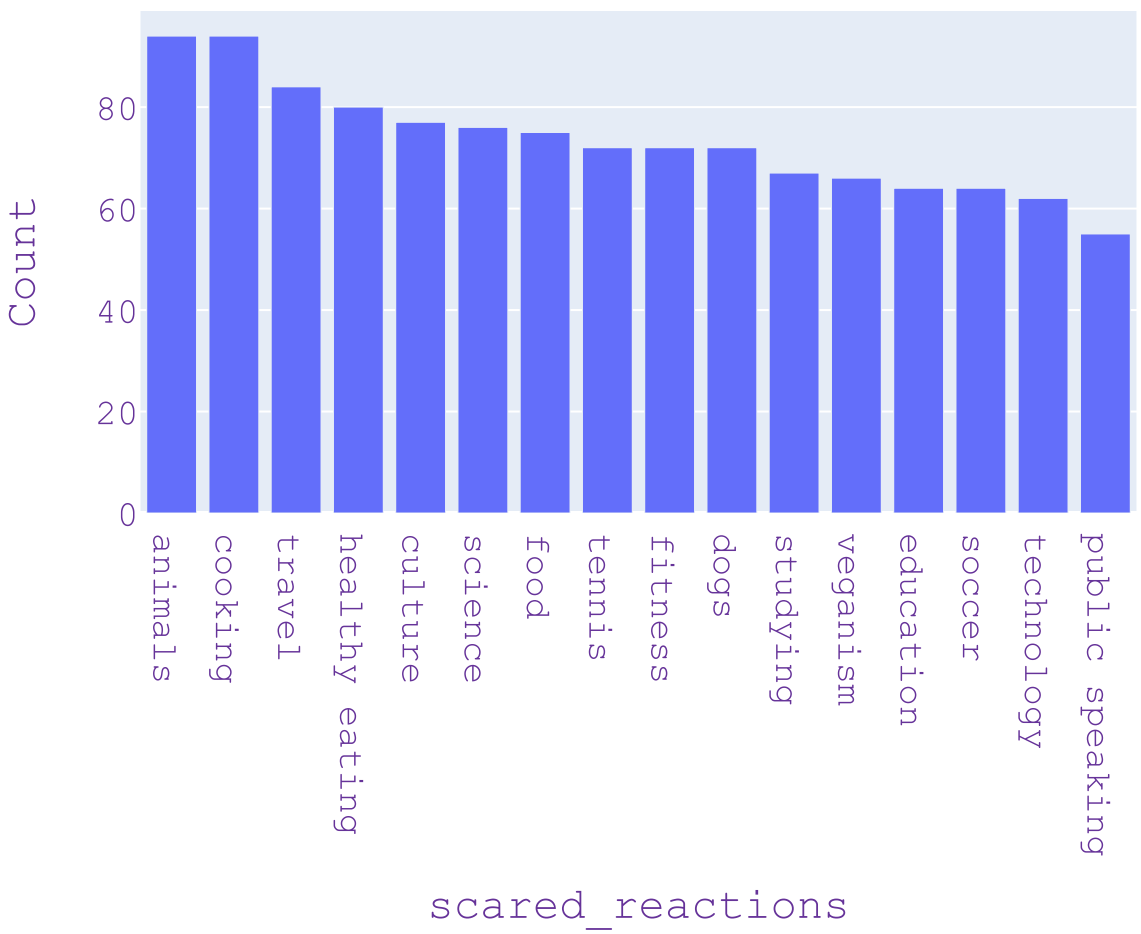
Count of love_reactions (11)

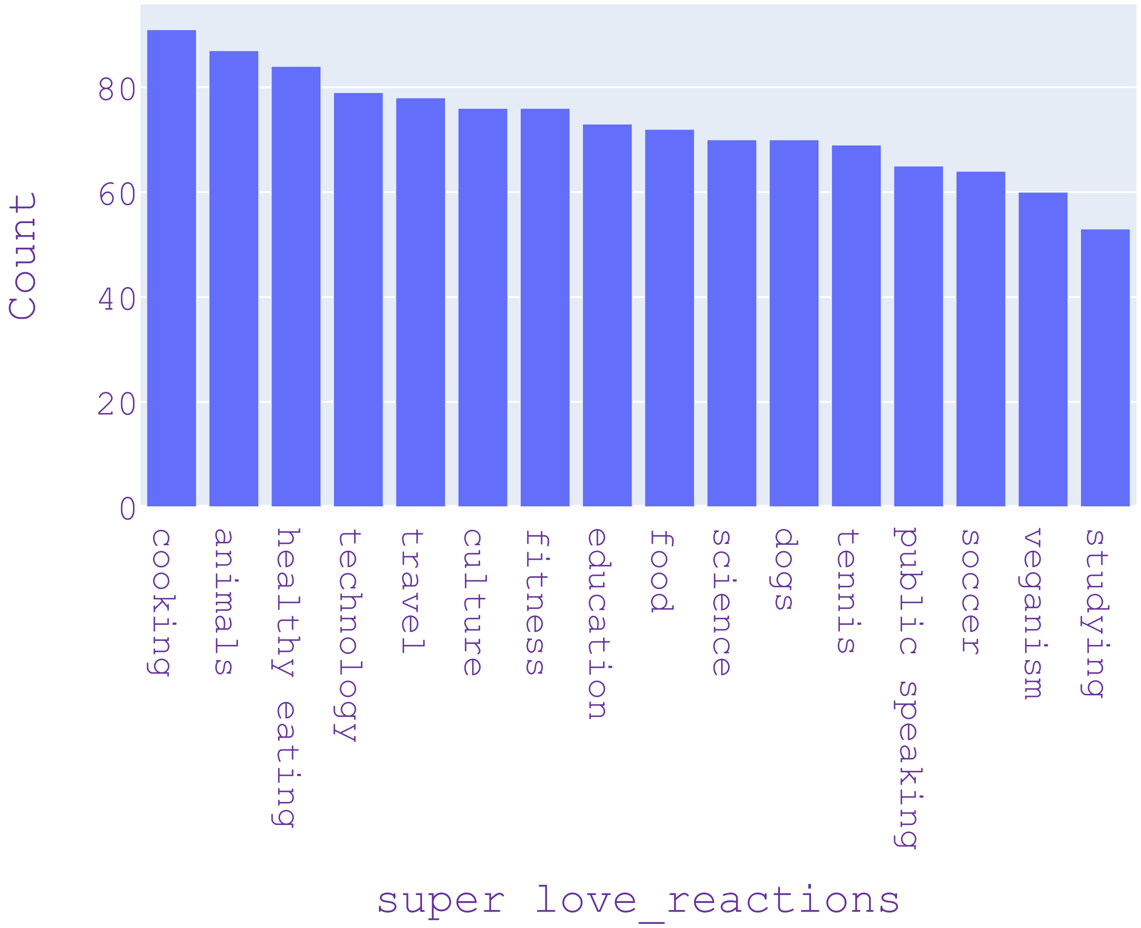


Count of peeking_reactions (12)

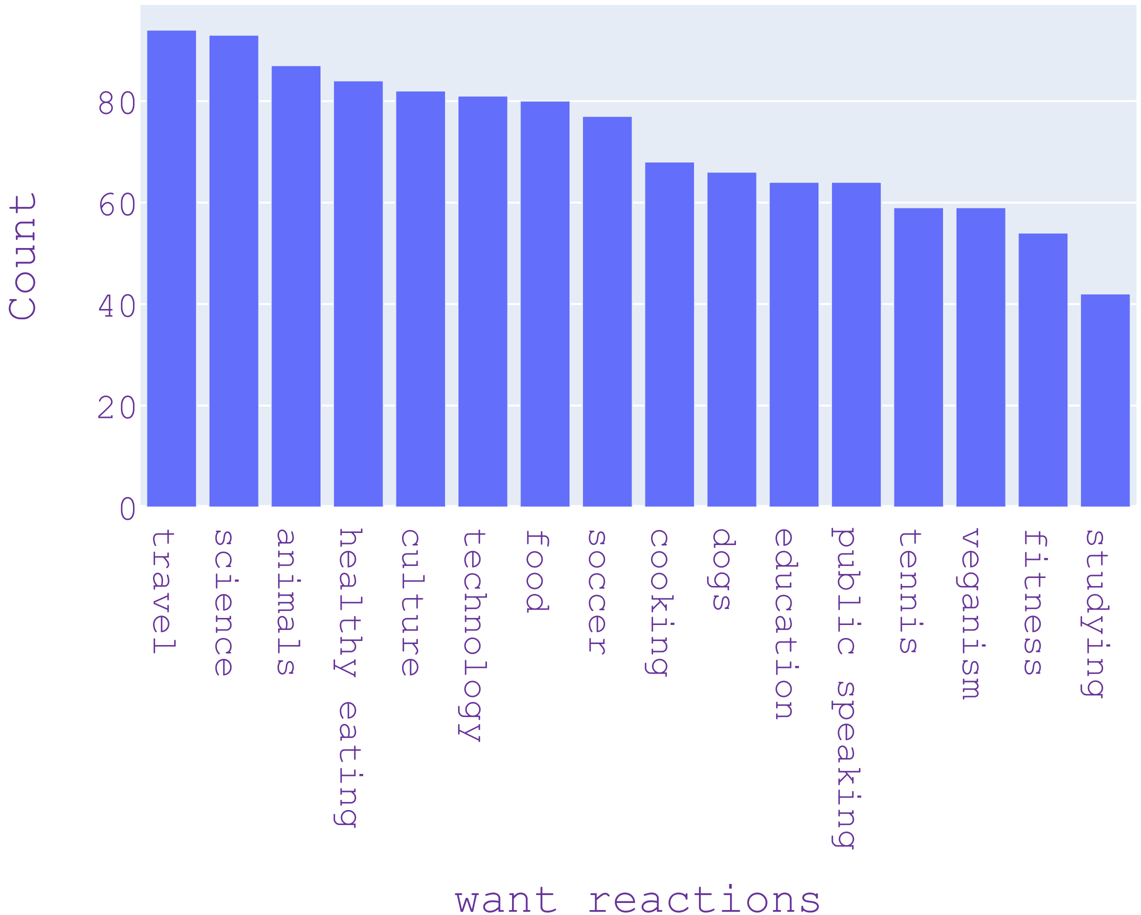


Count of scared_reactions (13)



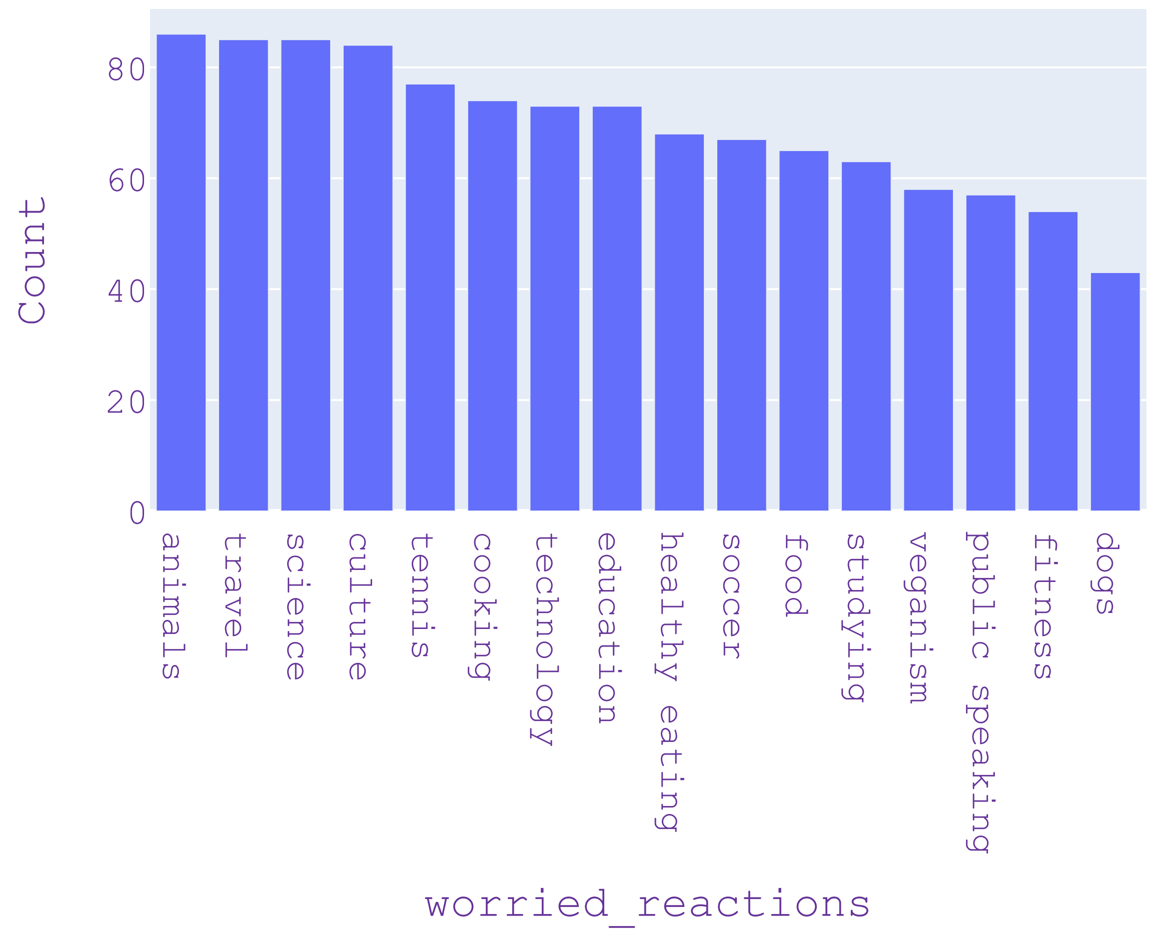


Count of super love_reactions (14)

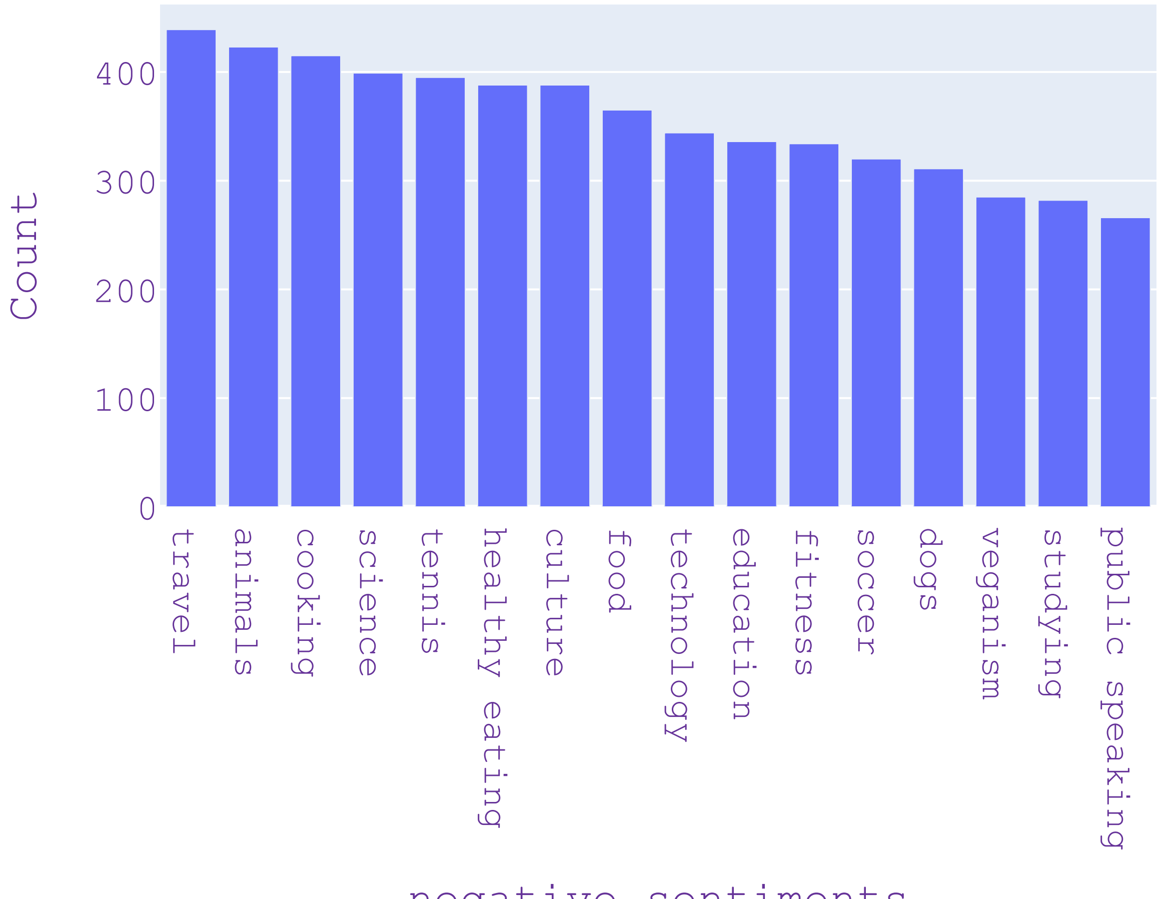


Count of want_reactions (15)

Count of worried_reactions (16)

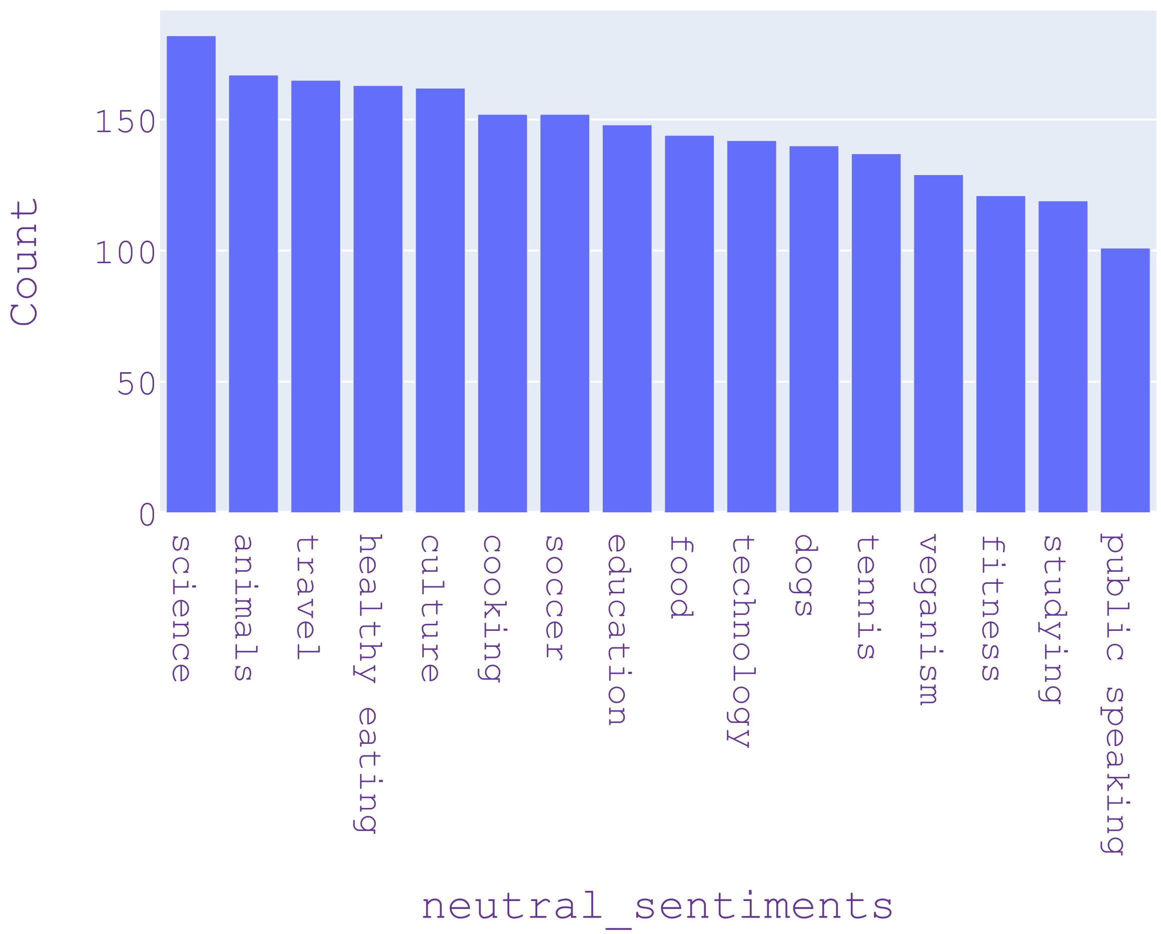


Count of negative_sentiments (17)

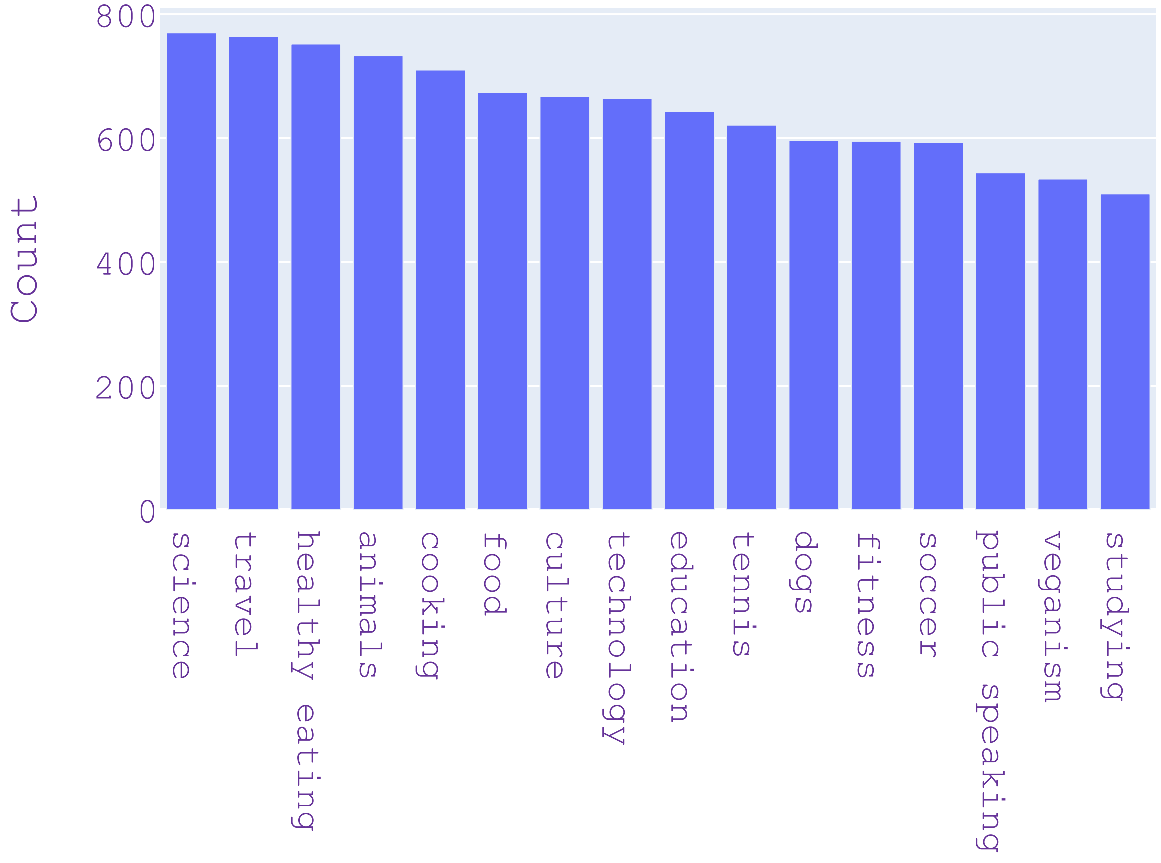


negative_sentiments

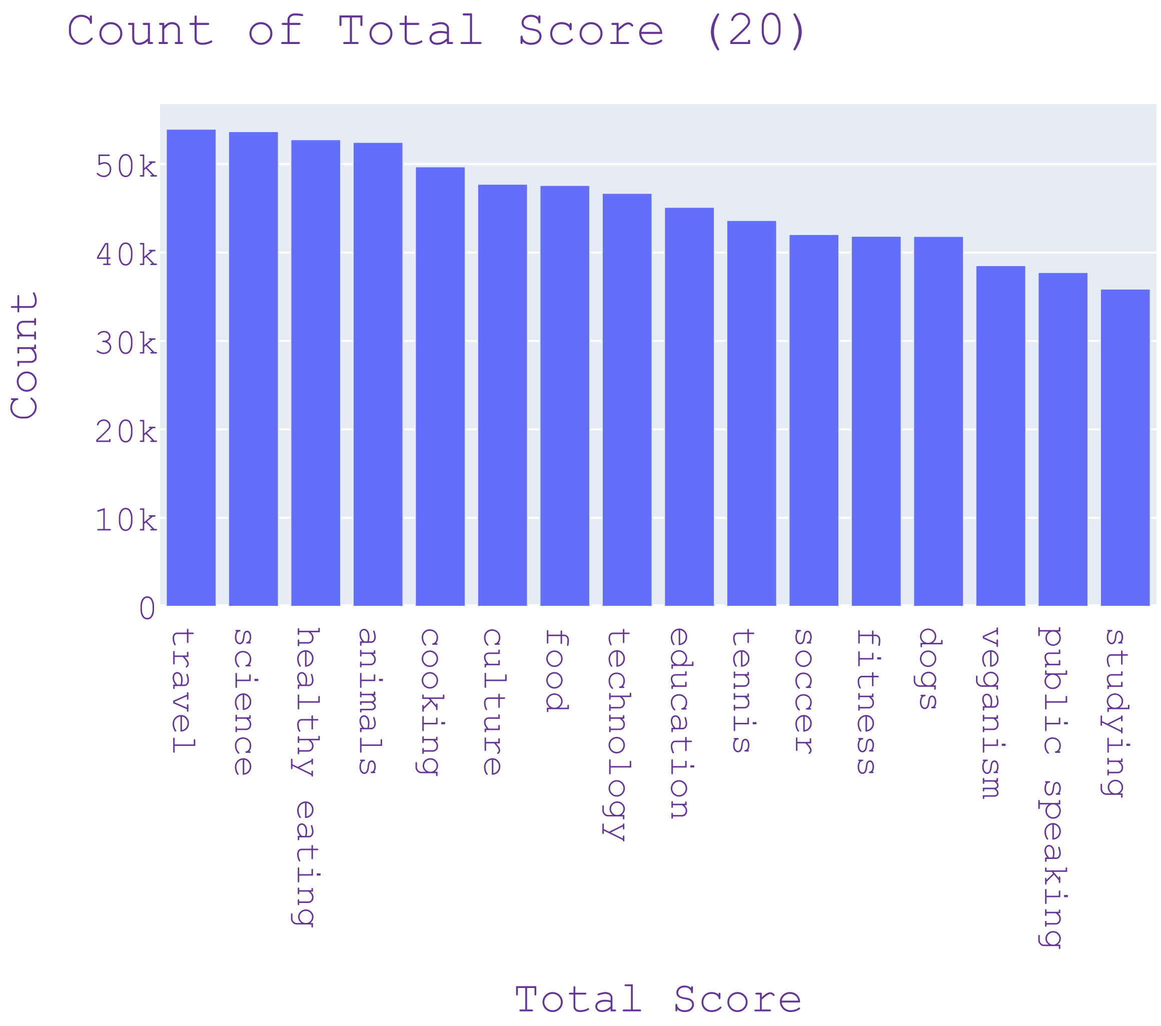
Count of neutral_sentiments (18)



Count of positive_sentiments (19)

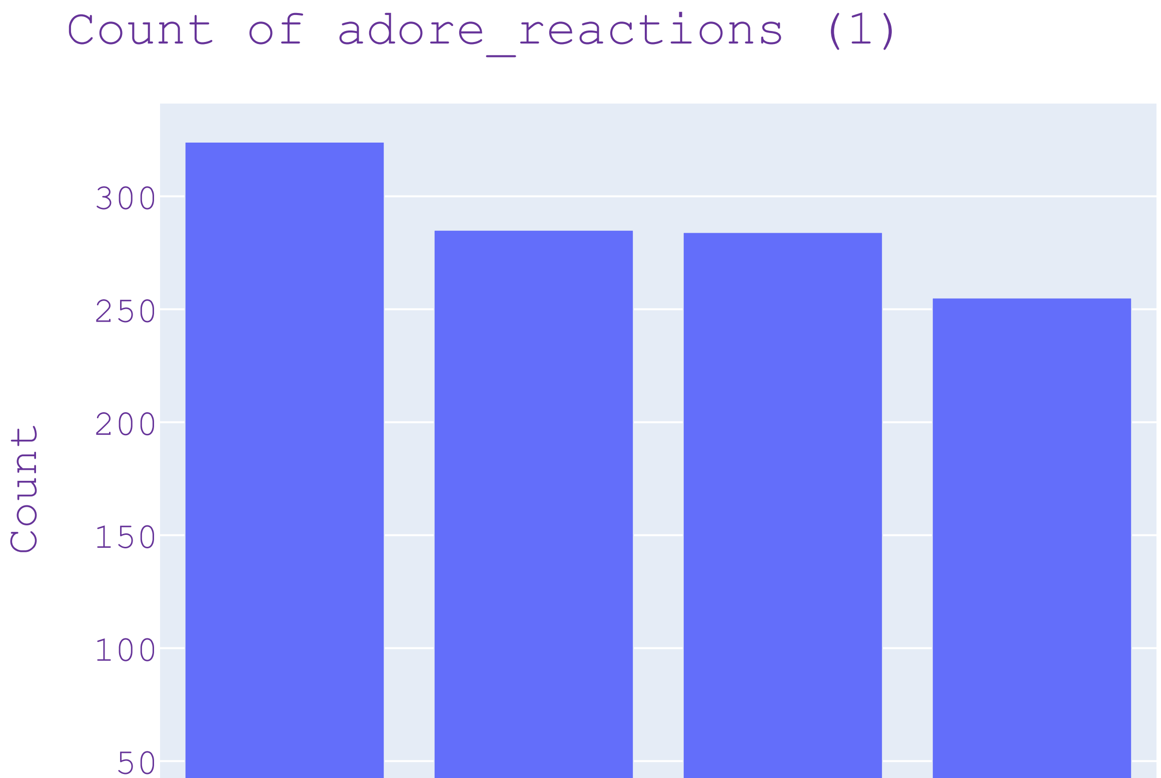


positive_sentiments

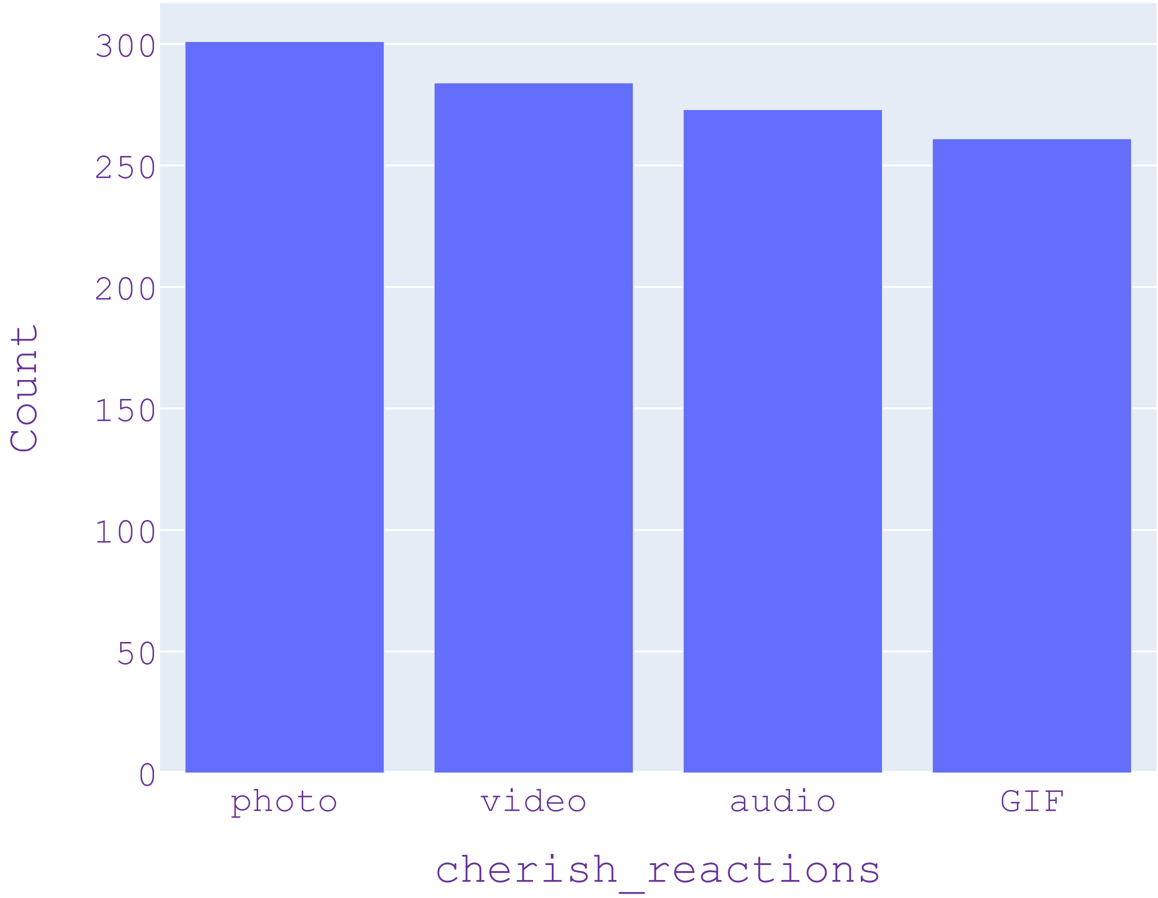


Displaying above data visually for Content Type

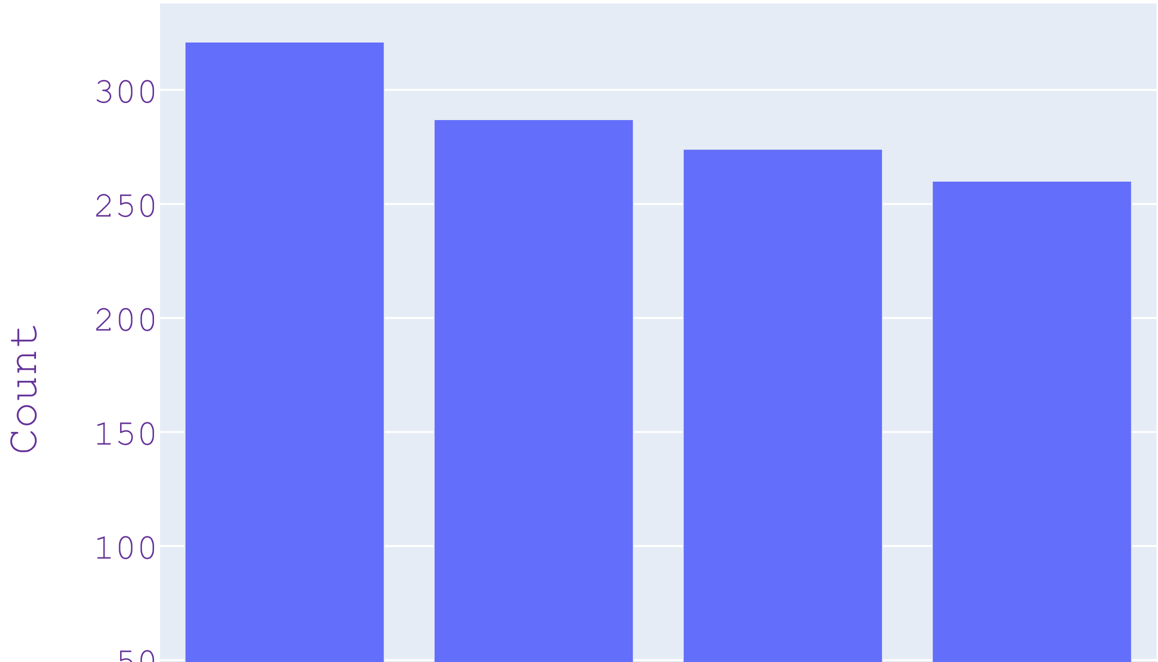
```
In [35]: reaction_count(content_reactions_types_df_cont_type,'Content_Type')
```



Count of cherish_reactions (2)

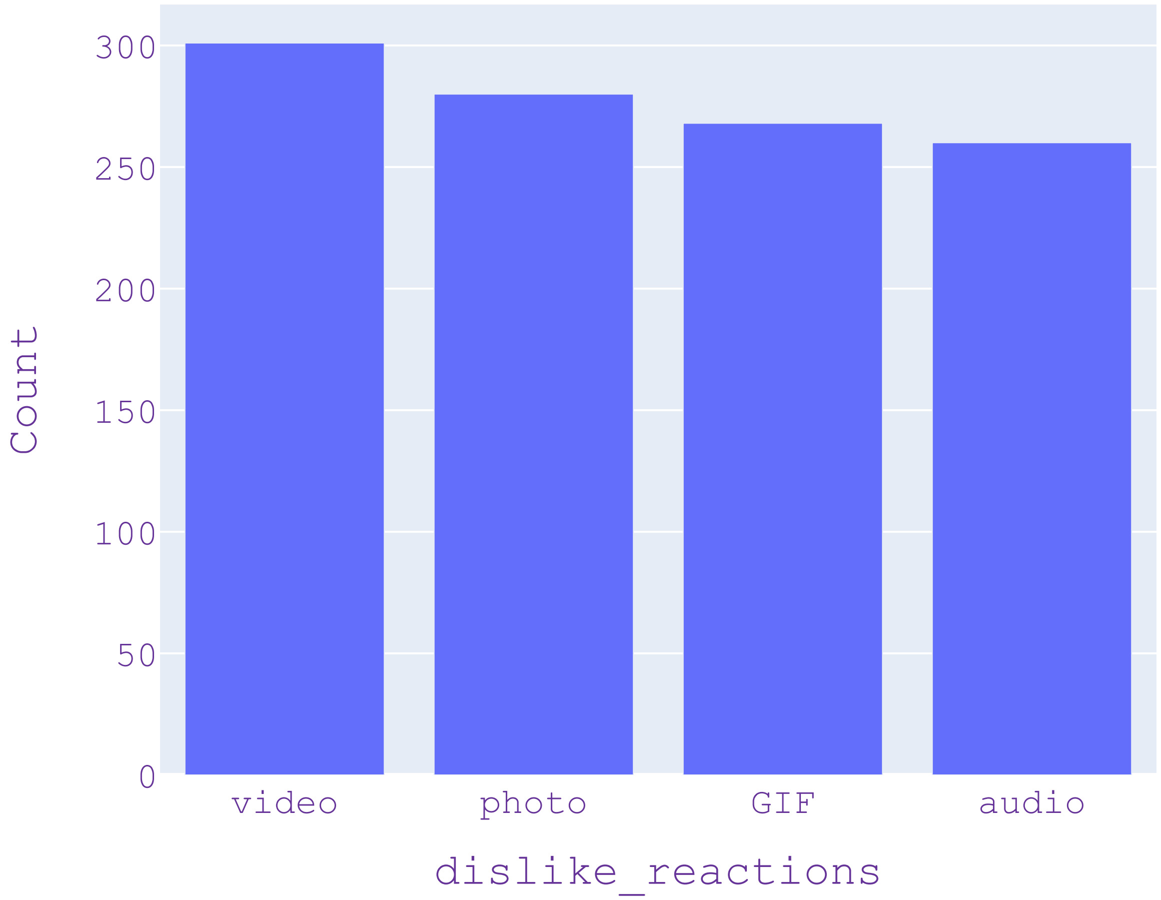


Count of disgust_reactions (3)

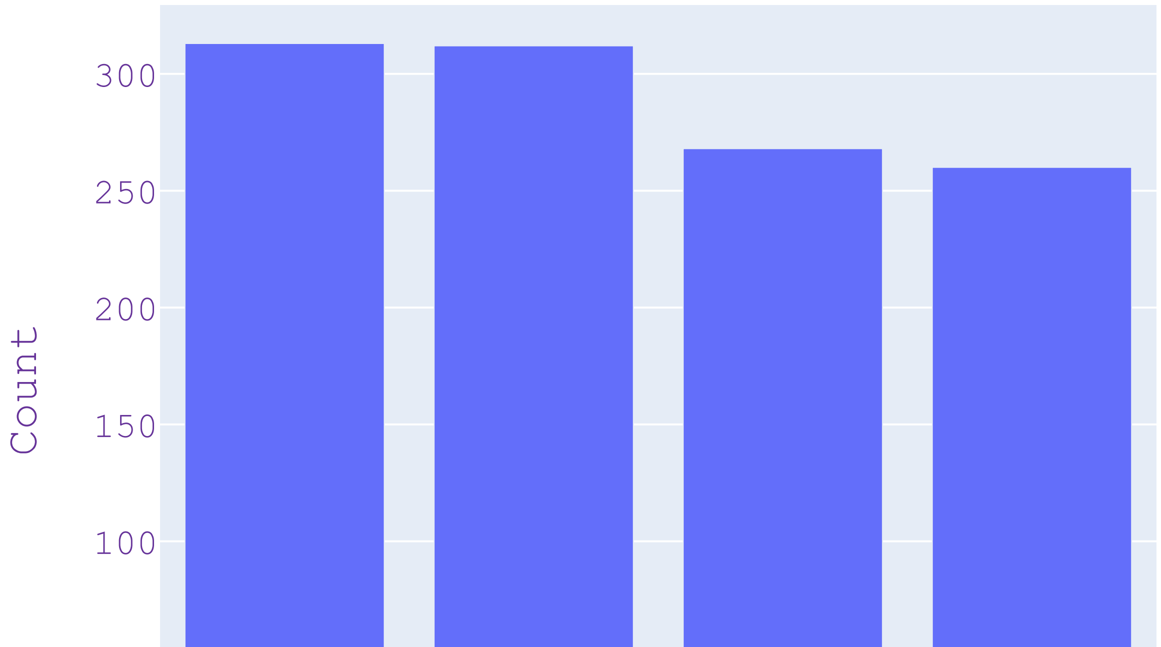


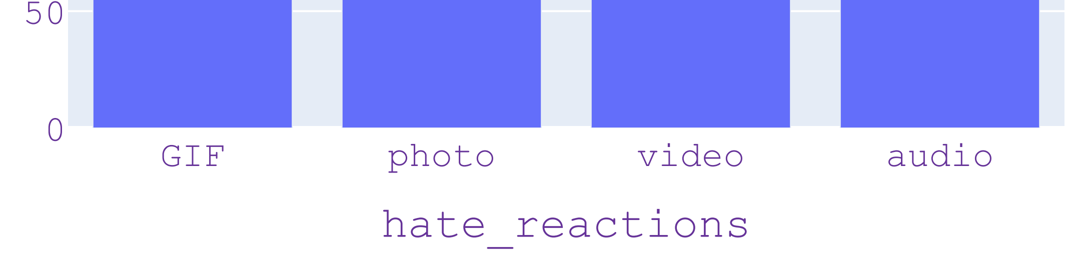
0
photo
video
GIF
audio
disgust_reactions

Count of dislike_reactions (4)

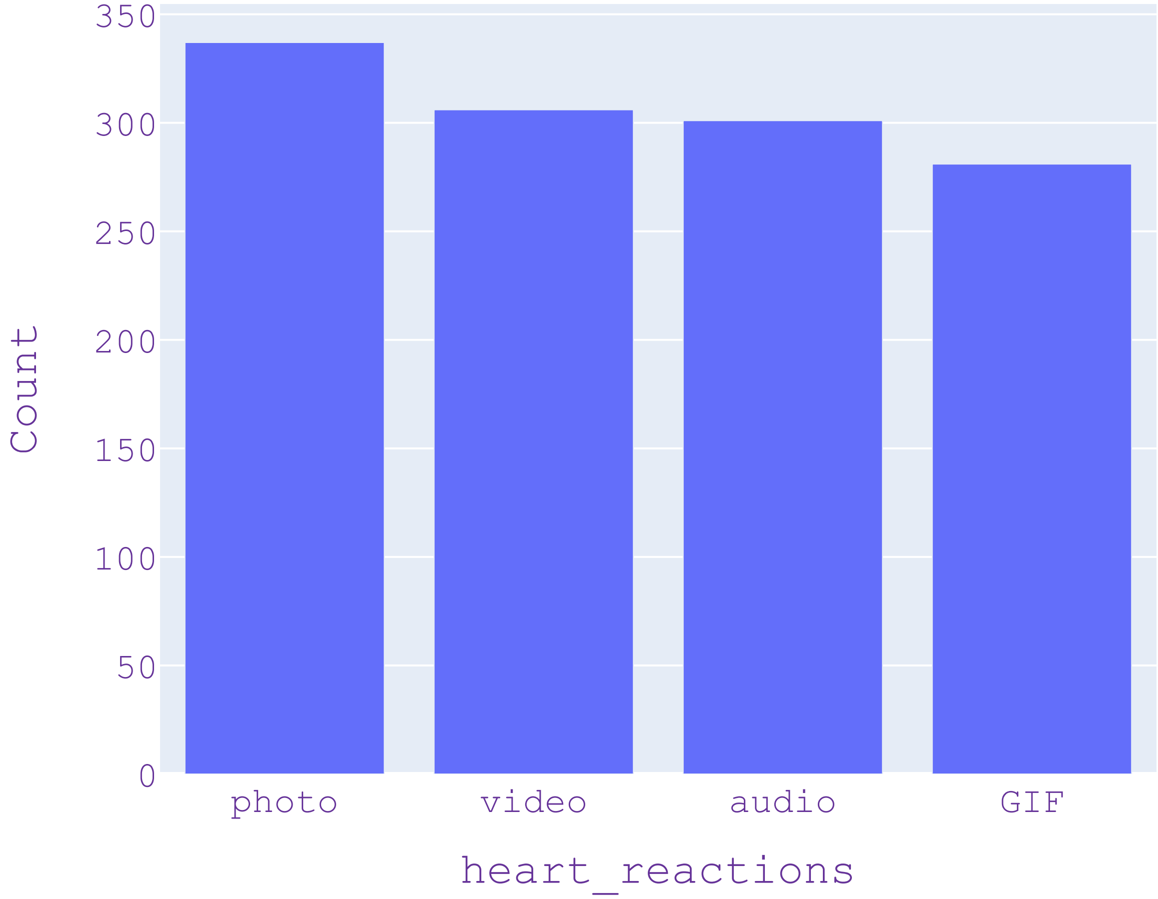


Count of hate_reactions (5)

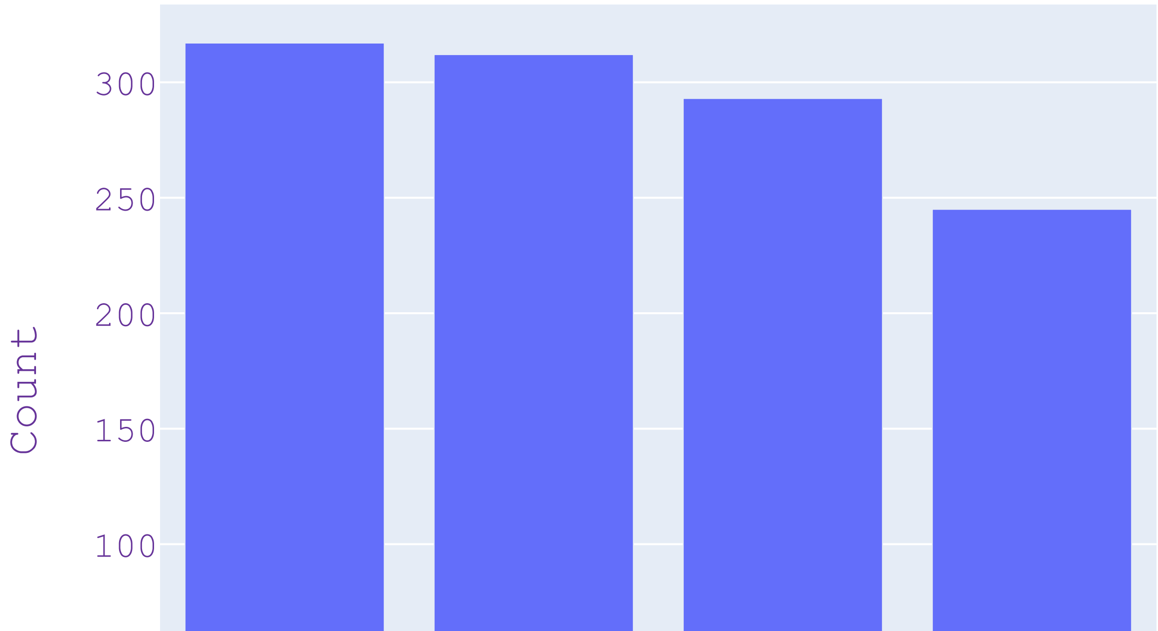


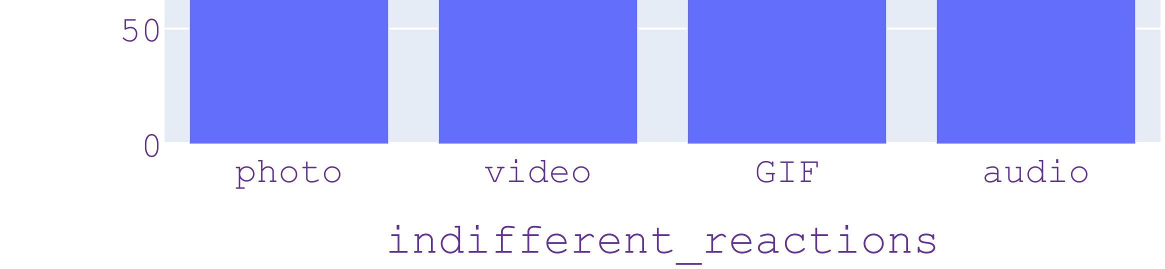


Count of heart_reactions (6)

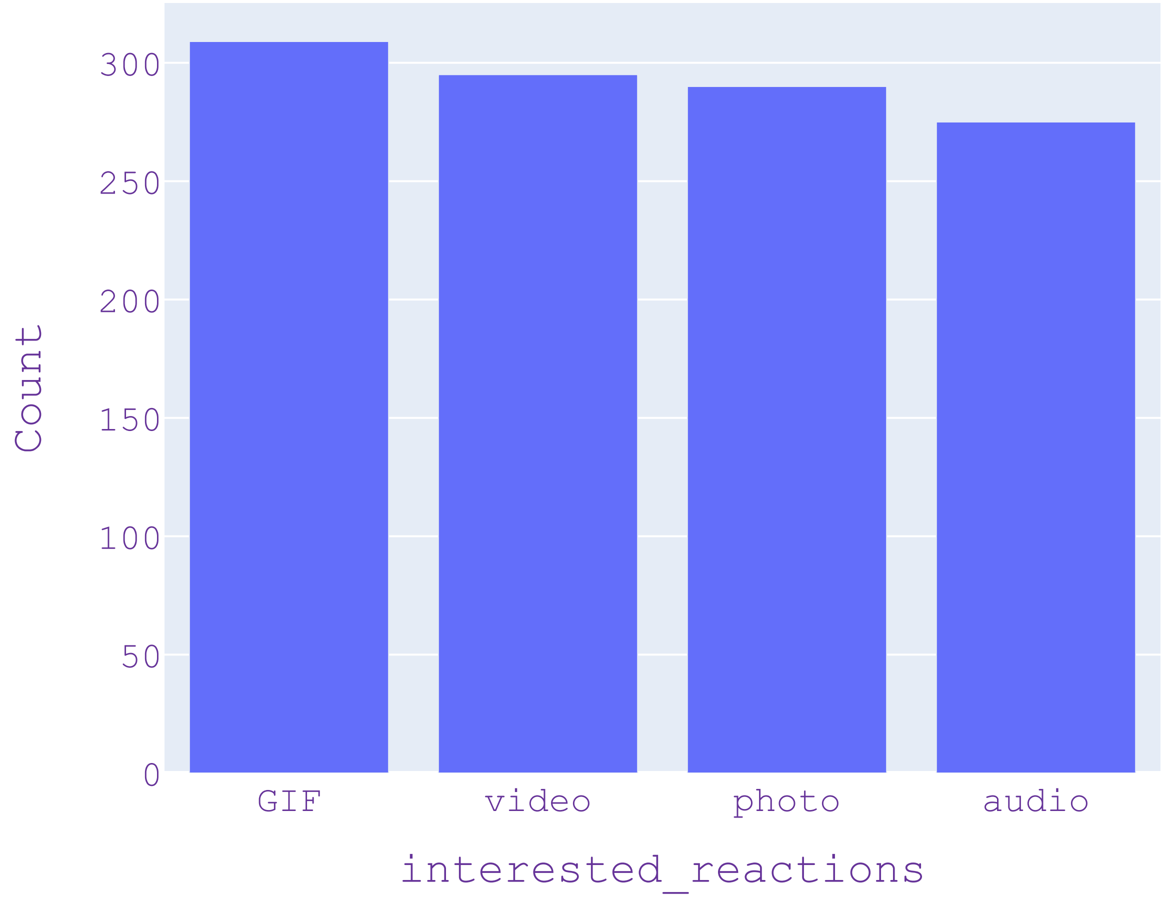


Count of indifferent_reactions (7)

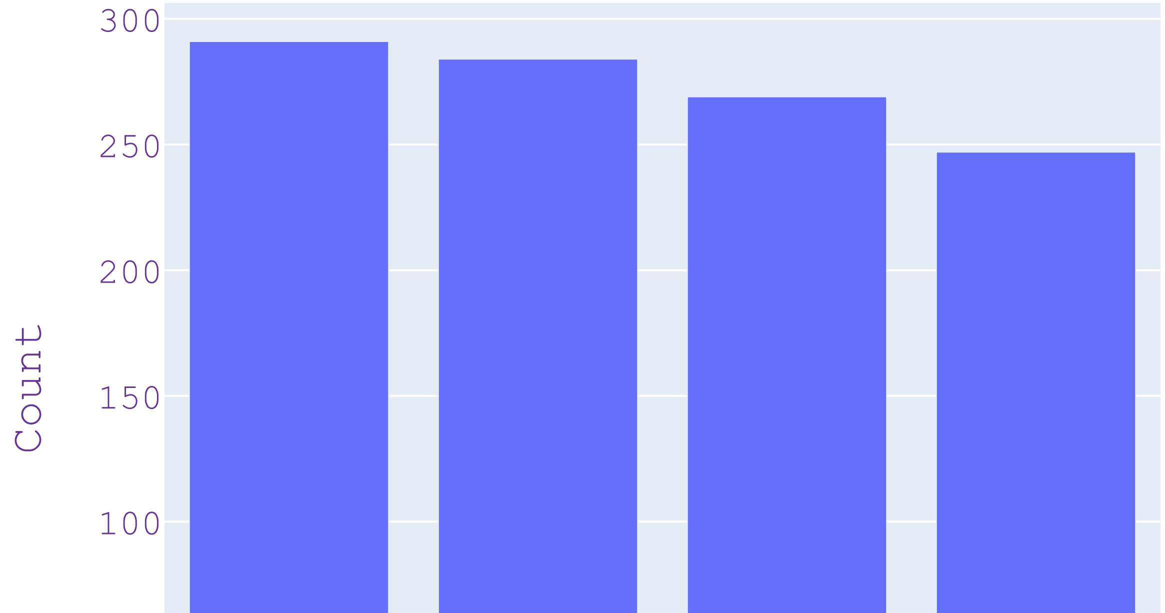


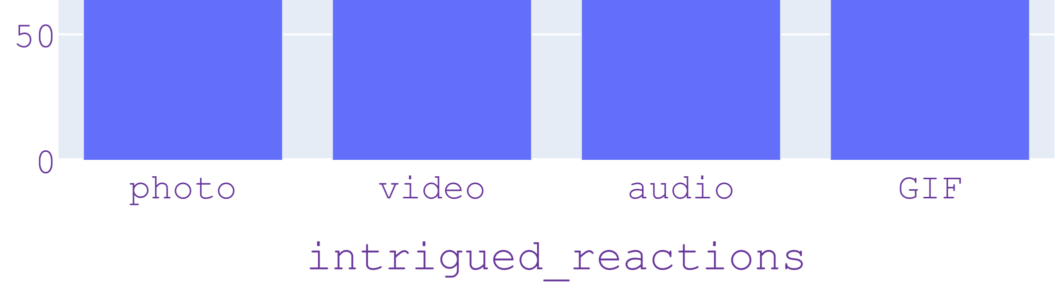


Count of interested_reactions (8)

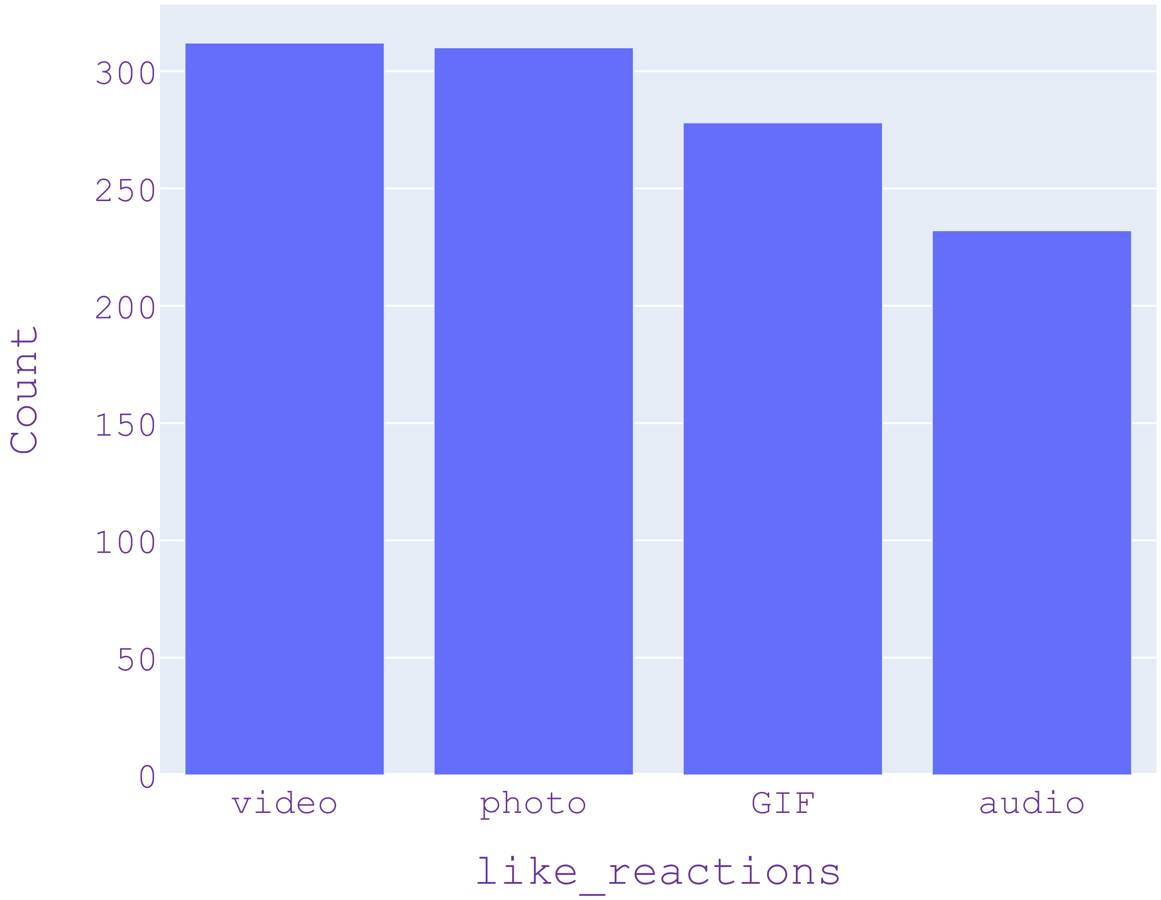


Count of intrigued_reactions (9)

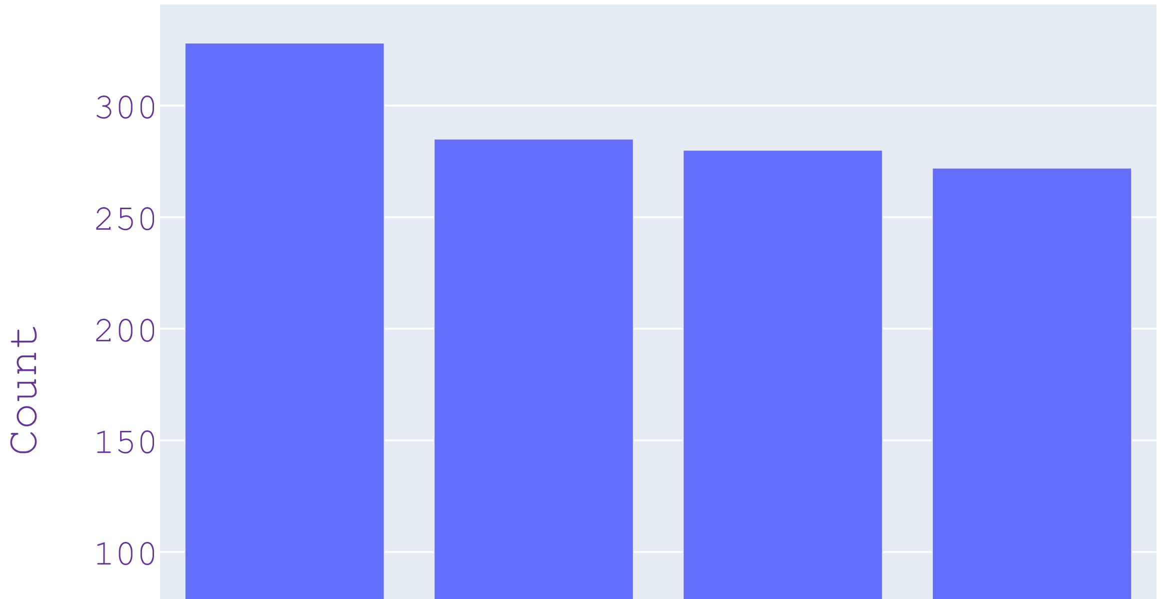


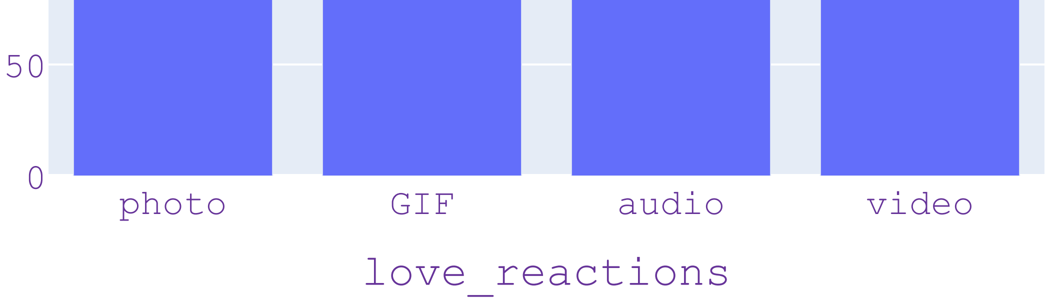


Count of like_reactions (10)

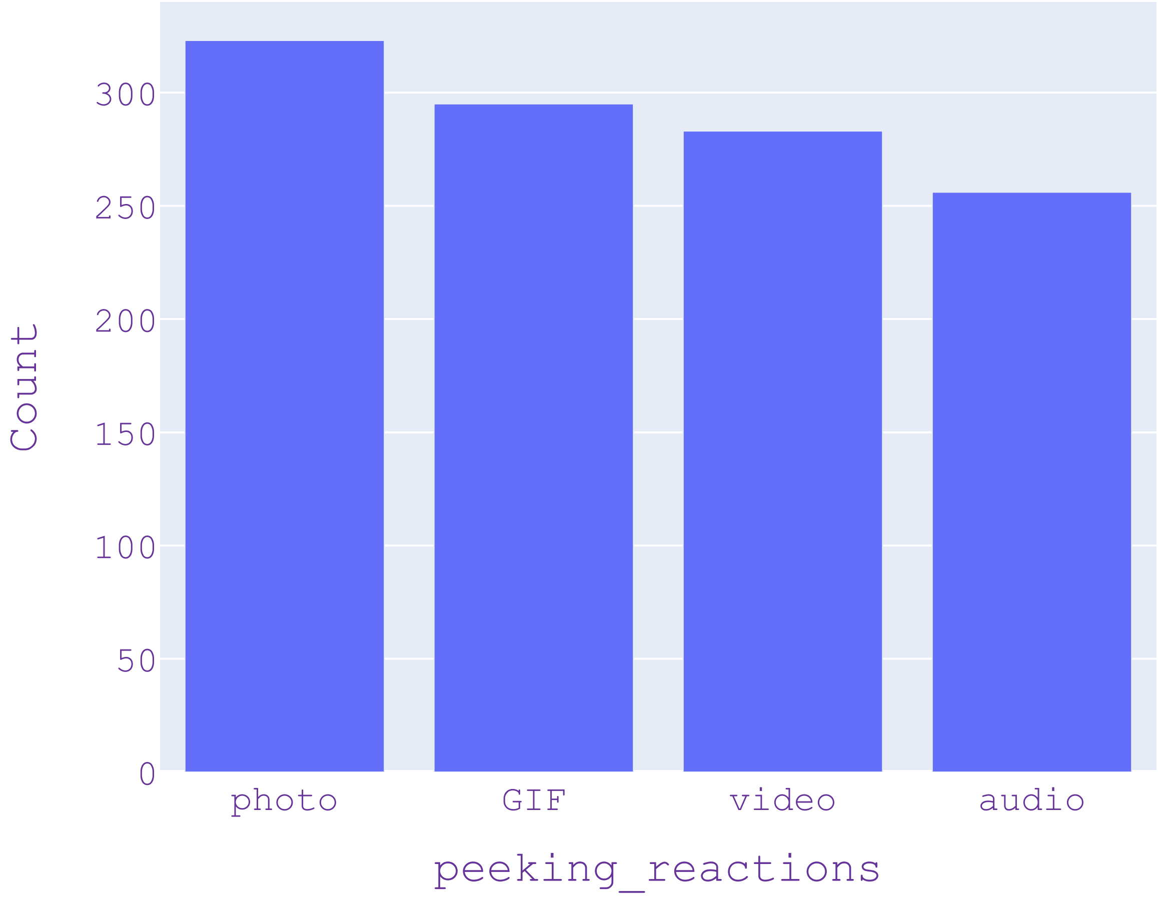


Count of love_reactions (11)

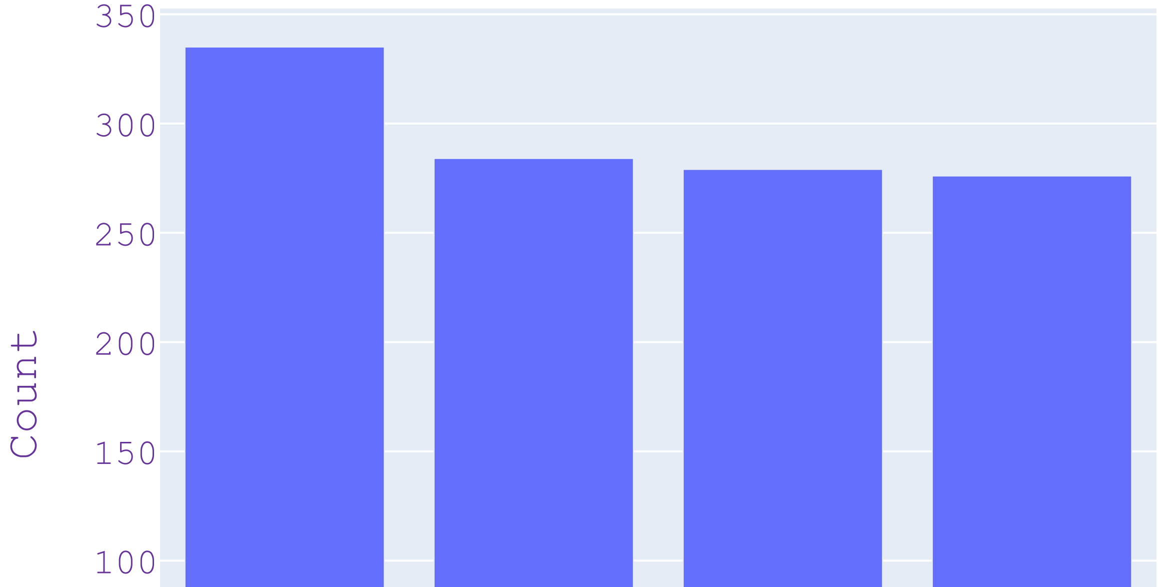


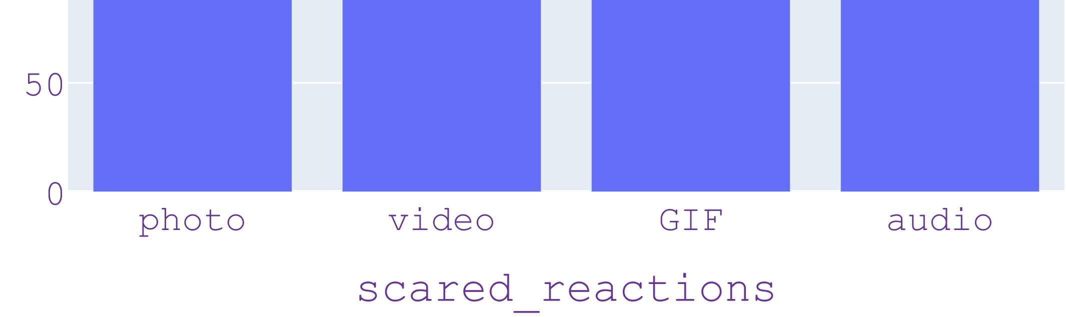


Count of peeking_reactions (12)

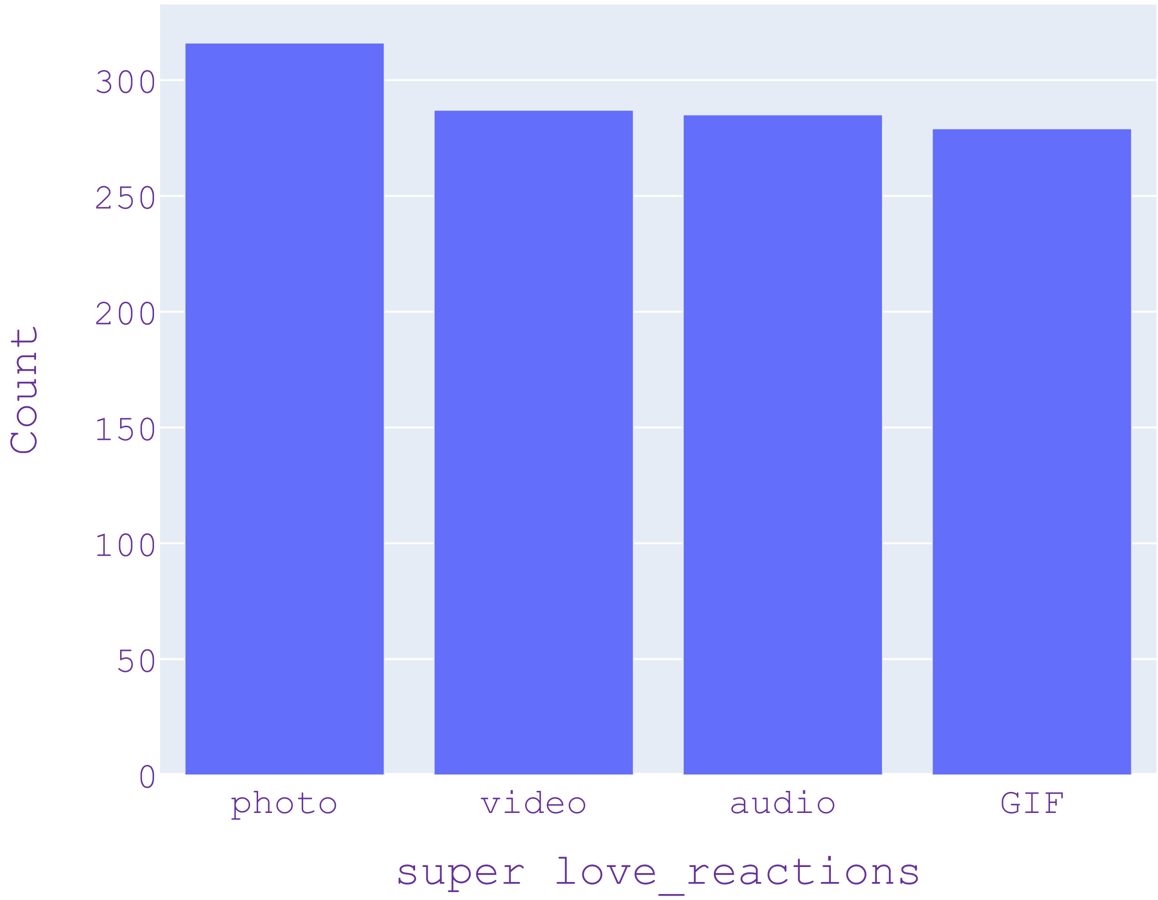


Count of scared_reactions (13)

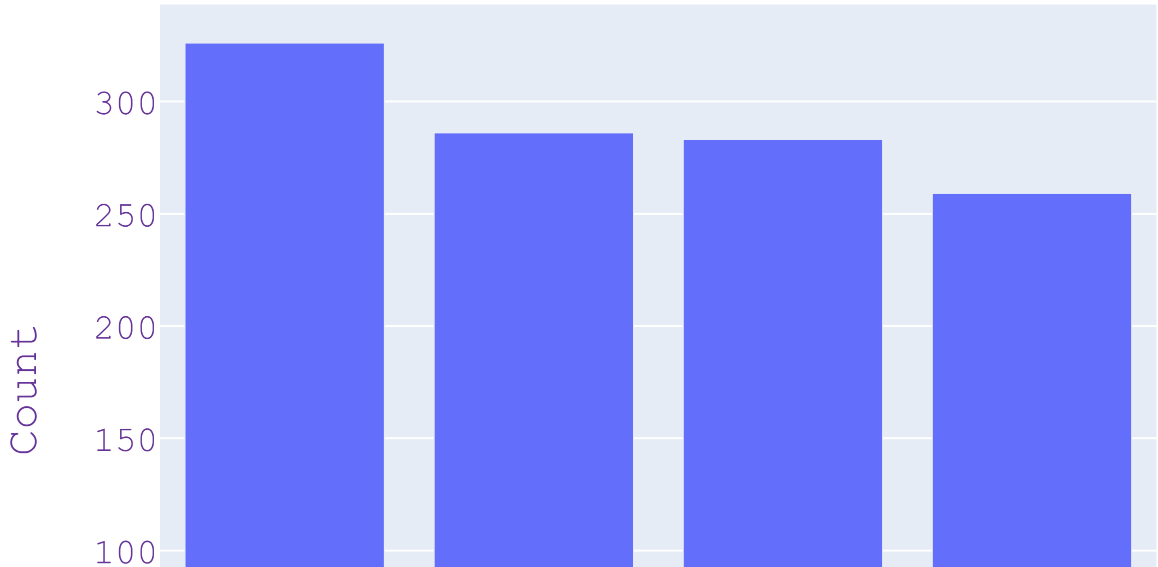




Count of super love_reactions (14)

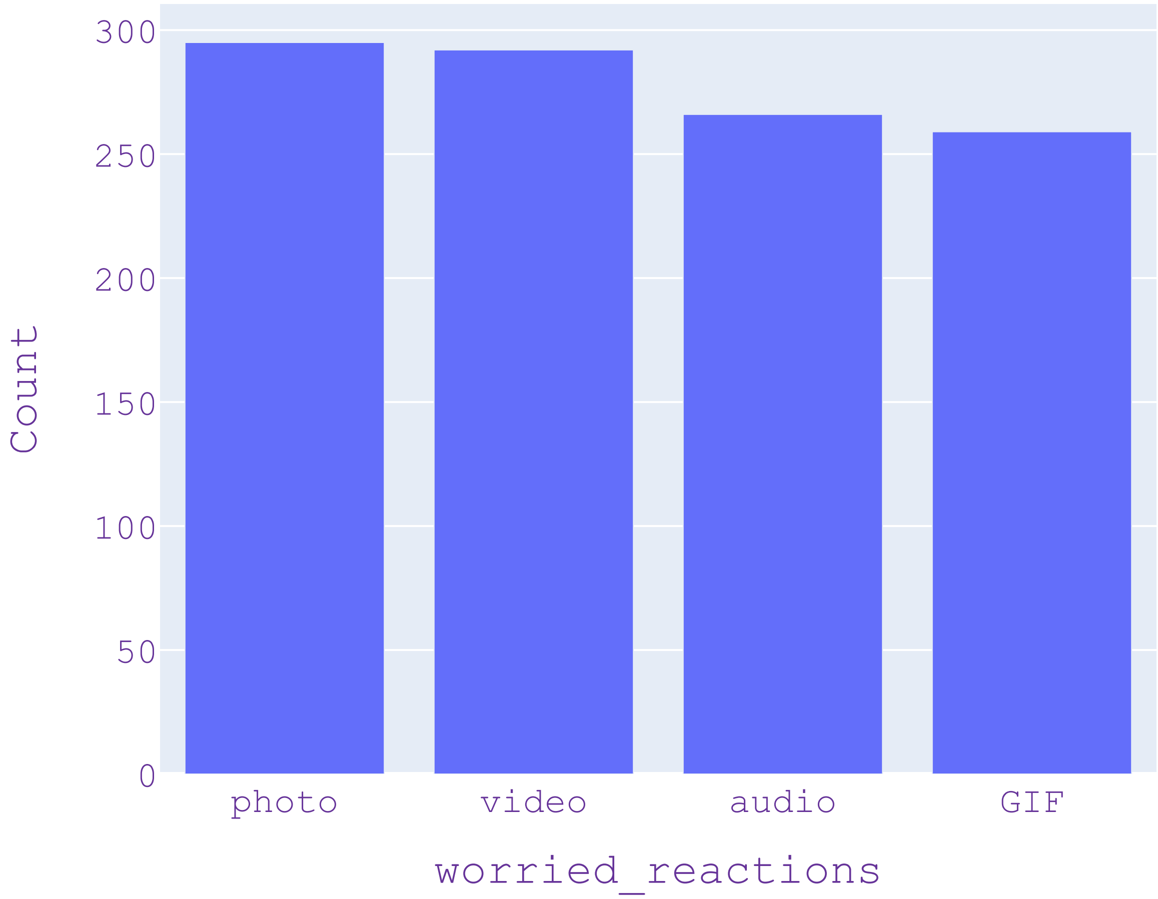


Count of want_reactions (15)

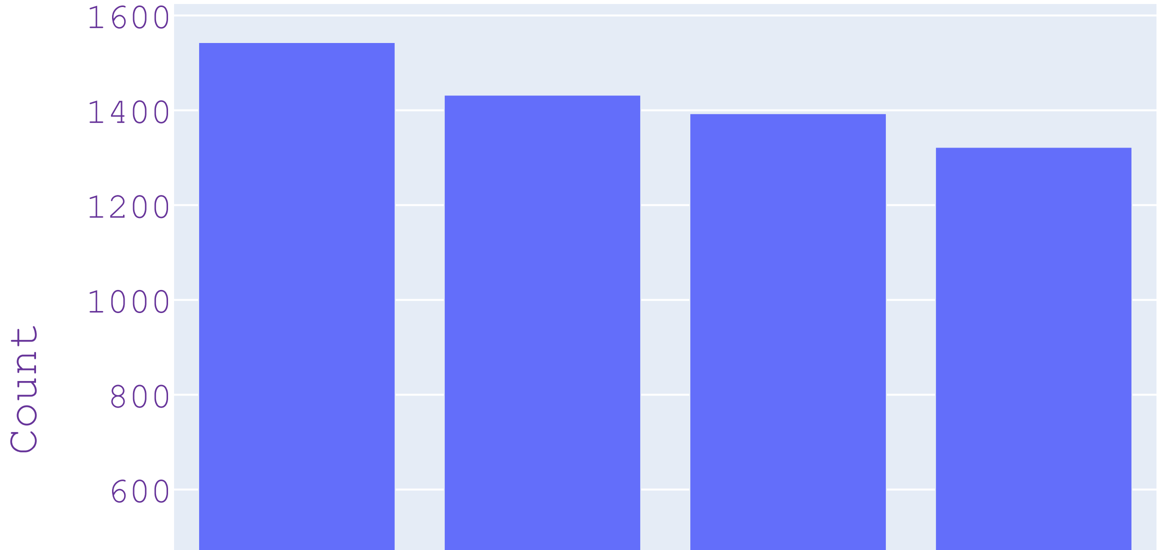


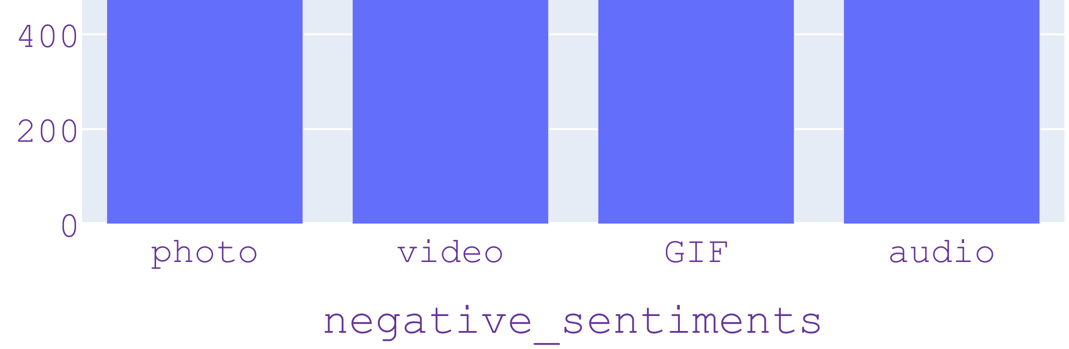
50
0
photo
video
audio
GIF
want_reactions

Count of worried_reactions (16)

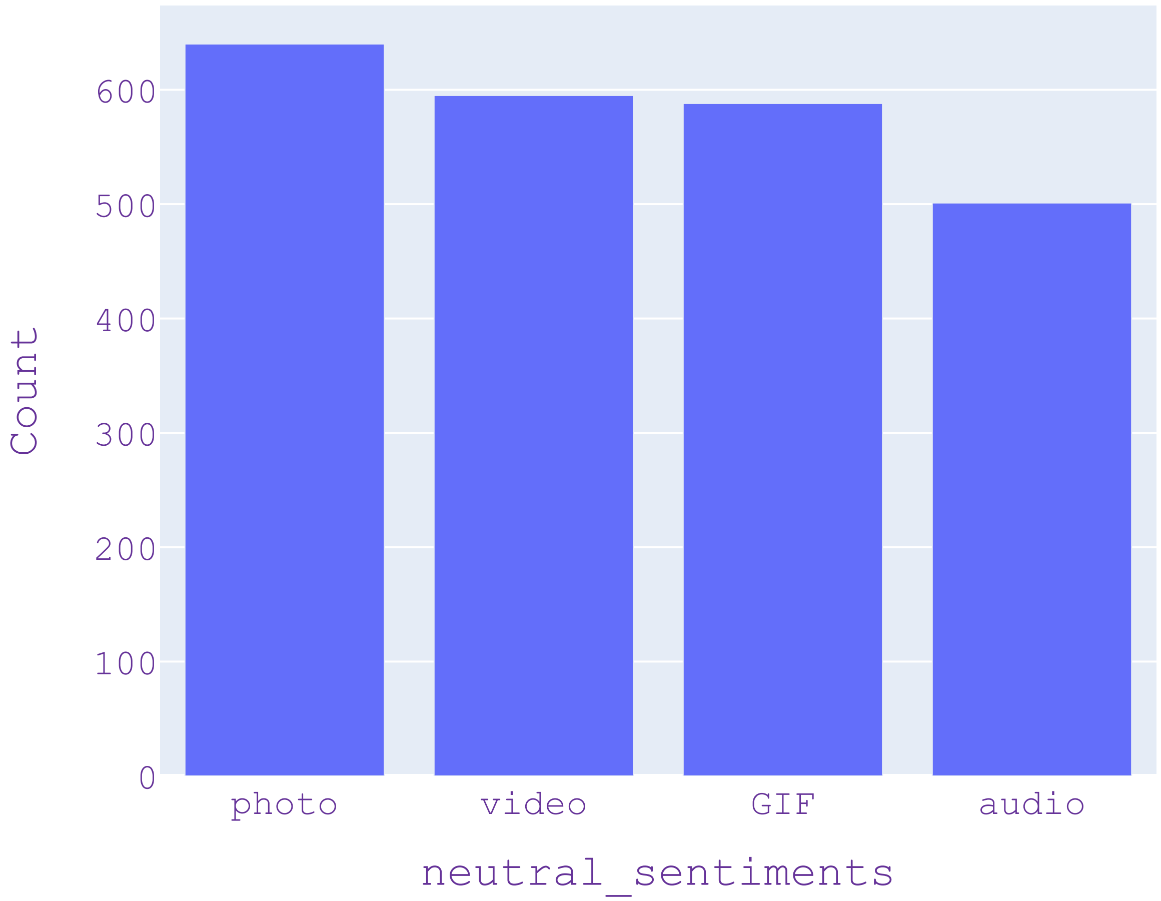


Count of negative_sentiments (17)

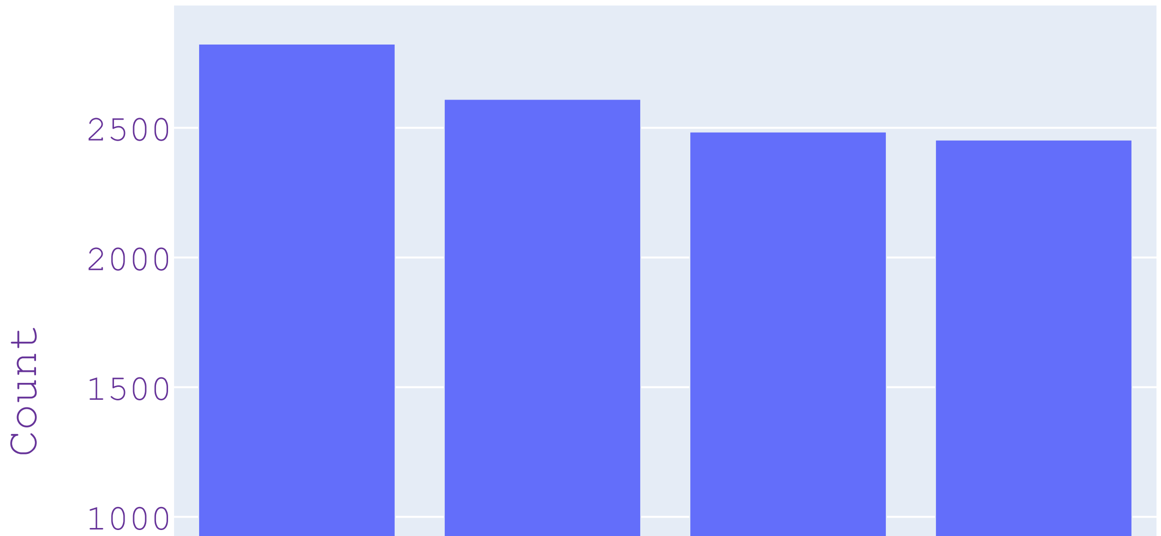


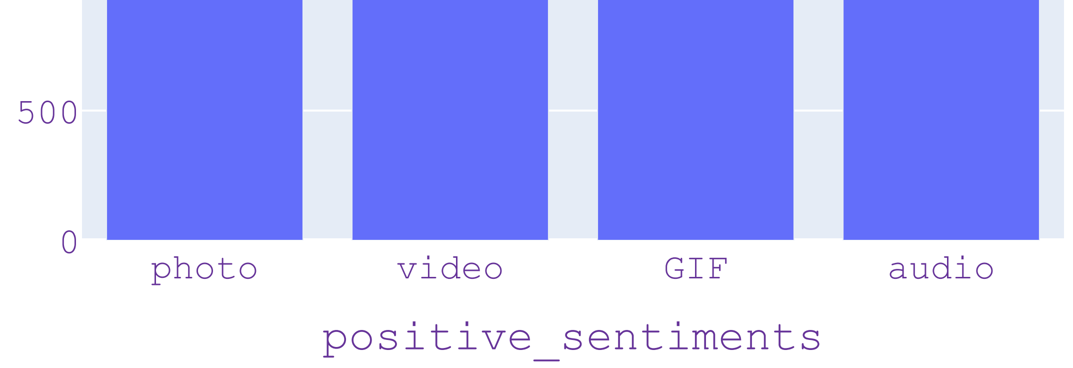


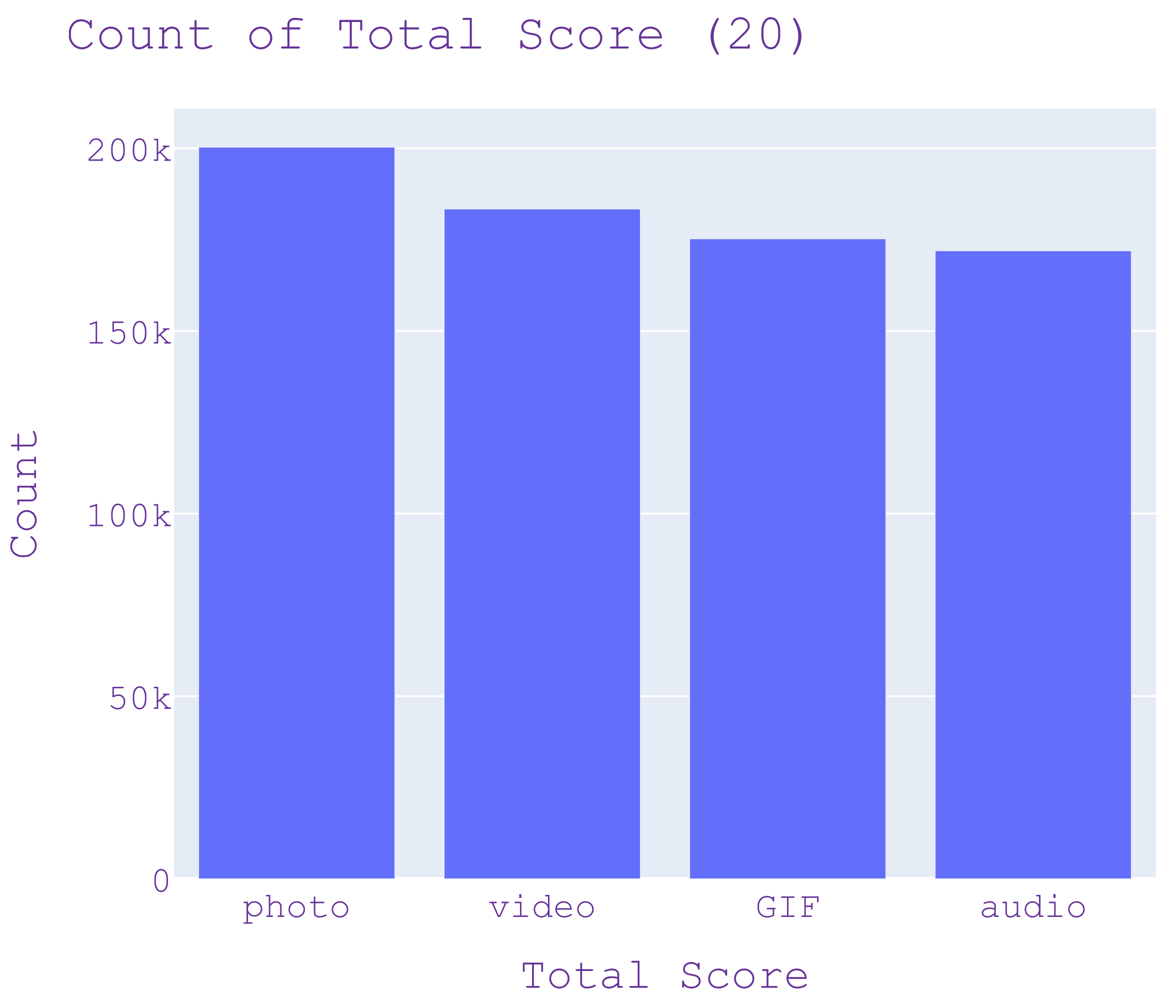
Count of neutral_sentiments (18)



Count of positive_sentiments (19)







Grouping and getting Sum by Category, Content_Type reactions and score

In [36]:

```
content_reactions_types_df_cat_cont_type=gg_reactions_types_df.groupby(['Category','Cont
content_reactions_types_df_cat_cont_type
```

Out[36]:

	Category	Content_Type	adore_reactions	cherish_reactions	disgust_reactions	dislike_reactions	hate
2	animals	photo	25	29	28	12	
1	animals	audio	28	33	26	30	
29	healthy eating	audio	17	35	29	22	
39	science	video	30	20	30	30	
59	travel	video	21	29	27	27	

...	...	...	...	...	...	...
41	soccer	audio	15	10	8	6
12	dogs	GIF	6	10	5	7
45	studying	audio	8	7	5	7
63	veganism	video	2	9	4	6
32	public speaking	GIF	8	6	2	6

64 rows × 22 columns

In [37]:

```
content_reactions_types_df_cat_cont_type[["Category","Content_Type","Total Score"]][:5]
```

Out[37]:

	Category	Content_Type	Total Score
2	animals	photo	17832.0
1	animals	audio	17513.0
29	healthy eating	audio	16772.0
39	science	video	16771.0
59	travel	video	16663.0