

@lilmiquela

December 20, 2020

```
[92]: pip install emoji
```

```
Requirement already satisfied: emoji in  
/opt/conda/envs/python3/lib/python3.7/site-packages (0.6.0)  
Note: you may need to restart the kernel to use updated packages.
```

```
[93]: pip install WordCloud
```

```
Requirement already satisfied: WordCloud in  
/opt/conda/envs/python3/lib/python3.7/site-packages (1.8.1)  
Requirement already satisfied: pillow in  
/opt/conda/envs/python3/lib/python3.7/site-packages (from WordCloud) (7.1.2)  
Requirement already satisfied: matplotlib in  
/opt/conda/envs/python3/lib/python3.7/site-packages (from WordCloud) (3.1.2)  
Requirement already satisfied: numpy>=1.6.1 in  
/opt/conda/envs/python3/lib/python3.7/site-packages (from WordCloud) (1.18.0)  
Requirement already satisfied: pyparsing!=2.0.4,!=2.1.2,!=2.1.6,>=2.0.1 in  
/opt/conda/envs/python3/lib/python3.7/site-packages (from matplotlib->WordCloud)  
(2.4.6)  
Requirement already satisfied: python-dateutil>=2.1 in  
/opt/conda/envs/python3/lib/python3.7/site-packages (from matplotlib->WordCloud)  
(2.8.1)  
Requirement already satisfied: cycler>=0.10 in  
/opt/conda/envs/python3/lib/python3.7/site-packages (from matplotlib->WordCloud)  
(0.10.0)  
Requirement already satisfied: kiwisolver>=1.0.1 in  
/opt/conda/envs/python3/lib/python3.7/site-packages (from matplotlib->WordCloud)  
(1.1.0)  
Requirement already satisfied: six>=1.5 in  
/opt/conda/envs/python3/lib/python3.7/site-packages (from python-  
dateutil>=2.1->matplotlib->WordCloud) (1.13.0)  
Requirement already satisfied: setuptools in  
/opt/conda/envs/python3/lib/python3.7/site-packages (from  
kiwisolver>=1.0.1->matplotlib->WordCloud) (44.0.0.post20200102)  
Note: you may need to restart the kernel to use updated packages.
```

```
[94]: pip install nltk
```

```
Requirement already satisfied: nltk in
/opt/conda/envs/python3/lib/python3.7/site-packages (3.5)
Requirement already satisfied: click in
/opt/conda/envs/python3/lib/python3.7/site-packages (from nltk) (7.1.2)
Requirement already satisfied: tqdm in
/opt/conda/envs/python3/lib/python3.7/site-packages (from nltk) (4.45.0)
Requirement already satisfied: regex in
/opt/conda/envs/python3/lib/python3.7/site-packages (from nltk) (2020.11.13)
Requirement already satisfied: joblib in
/opt/conda/envs/python3/lib/python3.7/site-packages (from nltk) (0.13.2)
Note: you may need to restart the kernel to use updated packages.
```

[95]: `pip install TextBlob`

```
Requirement already satisfied: TextBlob in
/opt/conda/envs/python3/lib/python3.7/site-packages (0.15.3)
Requirement already satisfied: nltk>=3.1 in
/opt/conda/envs/python3/lib/python3.7/site-packages (from TextBlob) (3.5)
Requirement already satisfied: click in
/opt/conda/envs/python3/lib/python3.7/site-packages (from nltk>=3.1->TextBlob)
(7.1.2)
Requirement already satisfied: regex in
/opt/conda/envs/python3/lib/python3.7/site-packages (from nltk>=3.1->TextBlob)
(2020.11.13)
Requirement already satisfied: joblib in
/opt/conda/envs/python3/lib/python3.7/site-packages (from nltk>=3.1->TextBlob)
(0.13.2)
Requirement already satisfied: tqdm in
/opt/conda/envs/python3/lib/python3.7/site-packages (from nltk>=3.1->TextBlob)
(4.45.0)
Note: you may need to restart the kernel to use updated packages.
```

[96]: `pip install altair`

```
Requirement already satisfied: altair in
/opt/conda/envs/python3/lib/python3.7/site-packages (4.1.0)
Requirement already satisfied: jsonschema in
/opt/conda/envs/python3/lib/python3.7/site-packages (from altair) (3.2.0)
Requirement already satisfied: numpy in
/opt/conda/envs/python3/lib/python3.7/site-packages (from altair) (1.18.0)
Requirement already satisfied: pandas>=0.18 in
/opt/conda/envs/python3/lib/python3.7/site-packages (from altair) (0.25.3)
Requirement already satisfied: Jinja2 in
/opt/conda/envs/python3/lib/python3.7/site-packages (from altair) (2.10.3)
Requirement already satisfied: toolz in
/opt/conda/envs/python3/lib/python3.7/site-packages (from altair) (0.10.0)
Requirement already satisfied: entrypoints in
/opt/conda/envs/python3/lib/python3.7/site-packages (from altair) (0.3)
```

```

Requirement already satisfied: attrs>=17.4.0 in
/opt/conda/envs/python3/lib/python3.7/site-packages (from jsonschema->altair)
(19.3.0)
Requirement already satisfied: six>=1.11.0 in
/opt/conda/envs/python3/lib/python3.7/site-packages (from jsonschema->altair)
(1.13.0)
Requirement already satisfied: importlib-metadata; python_version < "3.8" in
/opt/conda/envs/python3/lib/python3.7/site-packages (from jsonschema->altair)
(1.3.0)
Requirement already satisfied: setuptools in
/opt/conda/envs/python3/lib/python3.7/site-packages (from jsonschema->altair)
(44.0.0.post20200102)
Requirement already satisfied: pyrsistent>=0.14.0 in
/opt/conda/envs/python3/lib/python3.7/site-packages (from jsonschema->altair)
(0.15.6)
Requirement already satisfied: python-dateutil>=2.6.1 in
/opt/conda/envs/python3/lib/python3.7/site-packages (from pandas>=0.18->altair)
(2.8.1)
Requirement already satisfied: pytz>=2017.2 in
/opt/conda/envs/python3/lib/python3.7/site-packages (from pandas>=0.18->altair)
(2019.3)
Requirement already satisfied: MarkupSafe>=0.23 in
/opt/conda/envs/python3/lib/python3.7/site-packages (from jinja2->altair)
(1.1.1)
Requirement already satisfied: zipp>=0.5 in
/opt/conda/envs/python3/lib/python3.7/site-packages (from importlib-metadata;
python_version < "3.8"->jsonschema->altair) (0.6.0)
Requirement already satisfied: more-itertools in
/opt/conda/envs/python3/lib/python3.7/site-packages (from zipp>=0.5->importlib-
metadata; python_version < "3.8"->jsonschema->altair) (8.0.2)
Note: you may need to restart the kernel to use updated packages.

```

[97]: `pip install stopwords`

```

Requirement already satisfied: stopwords in
/opt/conda/envs/python3/lib/python3.7/site-packages (1.0.0)
Note: you may need to restart the kernel to use updated packages.

```

[98]: `pip install igramscraper`

```

Requirement already satisfied: igramscraper in
/opt/conda/envs/python3/lib/python3.7/site-packages (0.3.5)
Requirement already satisfied: python-slugify==3.0.2 in
/opt/conda/envs/python3/lib/python3.7/site-packages (from igramscraper) (3.0.2)
Requirement already satisfied: requests>=2.21.0 in
/opt/conda/envs/python3/lib/python3.7/site-packages (from igramscraper) (2.23.0)
Requirement already satisfied: text-unidecode==1.2 in
/opt/conda/envs/python3/lib/python3.7/site-packages (from python-

```

```
slugify==3.0.2->igramscraper) (1.2)
Requirement already satisfied: urllib3!=1.25.0,!1.25.1,<1.26,>=1.21.1 in
/opt/conda/envs/python3/lib/python3.7/site-packages (from
requests>=2.21.0->igramscraper) (1.25.9)
Requirement already satisfied: certifi>=2017.4.17 in
/opt/conda/envs/python3/lib/python3.7/site-packages (from
requests>=2.21.0->igramscraper) (2019.11.28)
Requirement already satisfied: chardet<4,>=3.0.2 in
/opt/conda/envs/python3/lib/python3.7/site-packages (from
requests>=2.21.0->igramscraper) (3.0.4)
Requirement already satisfied: idna<3,>=2.5 in
/opt/conda/envs/python3/lib/python3.7/site-packages (from
requests>=2.21.0->igramscraper) (2.9)
Note: you may need to restart the kernel to use updated packages.
```

```
[99]: import pandas as pd
import string
import altair as alt #importerer 'altair' som 'alt'
from textblob import TextBlob
import nltk
from sklearn.model_selection import train_test_split
from sklearn.linear_model import LogisticRegression
from sklearn.linear_model import LinearRegression
from sklearn.metrics import r2_score
import matplotlib.pyplot as plt
import numpy as np
from sklearn.metrics import mean_squared_error, r2_score
import emoji
import regex
from IPython.display import Image
from igramscraper.instagram import Instagram
from collections import Counter
from nltk.corpus import stopwords
from wordcloud import WordCloud
```

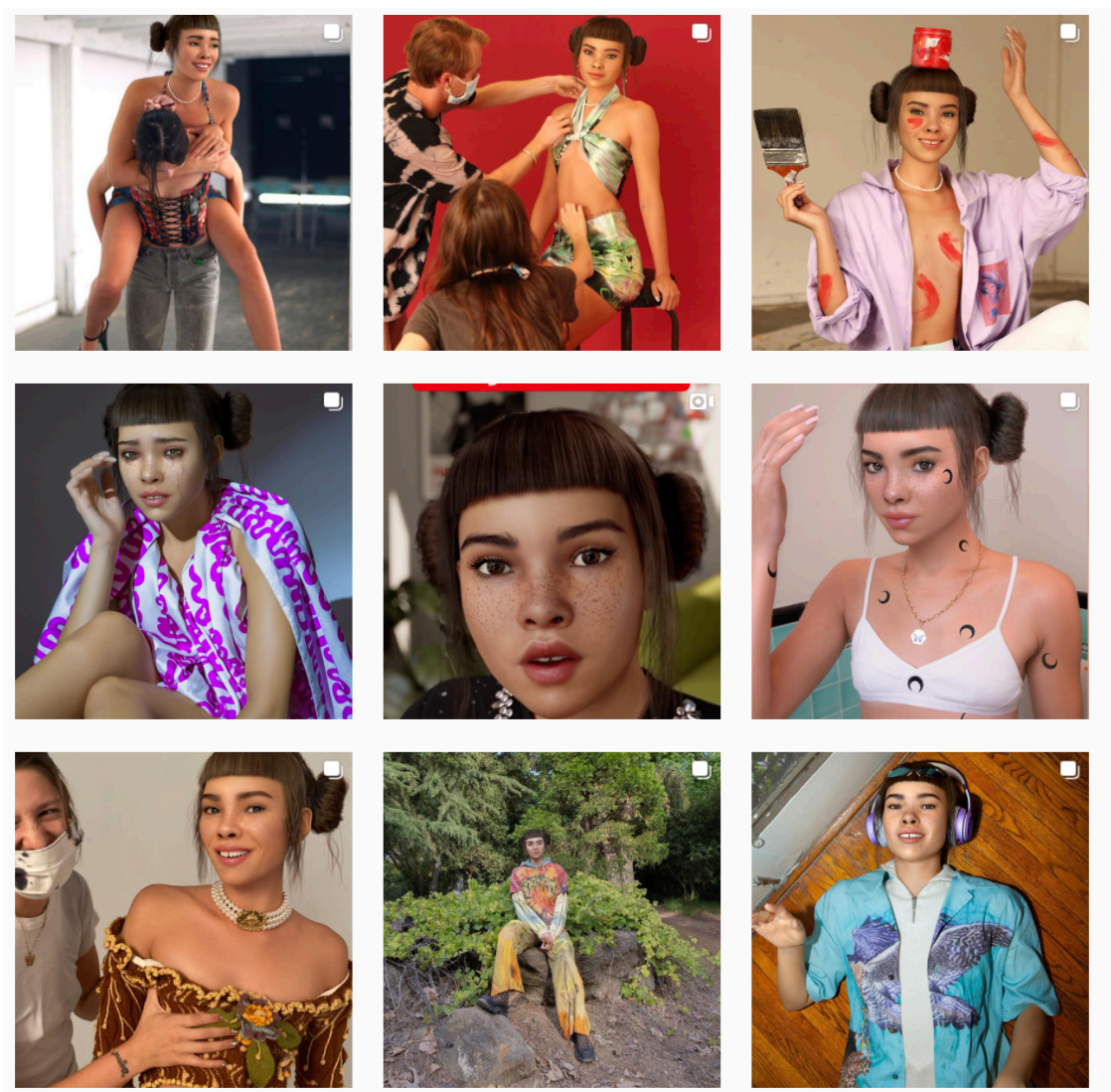
1 Robotinfluencers på Instagram

Indledning her...

I dette projektet ønsker vi at undersøge, hvad der karakteriserer instagramrobotten @lilmique-las brug af instagram. Vi opstiller altså et casestudie, der har til formål på baggrund af åbne datavidenskabelige principper at empirisk afdække et nyt fænomen.

```
[100]: Image(filename='Skærbillede 2020-12-14 kl. 17.07.40.png')
```

```
[100]:
```



1.0.1 Hvem er @lilmiquela

```
[177]: instagram = Instagram() #gør brug af biblioteket 'igramscraper.instagram'

account = instagram.get_account_by_id(3089598226) #instagramprofilens bruger-id

# printer relevant data fra @lilmiquelas instagramprofil
print('Account info:')
print('Id: ', account.identifier)
print('Username: ', account.username)
print('Full name: ', account.full_name)
print('Number of published posts: ', account.media_count)
```

```
print('Number of followers: ', account.followed_by_count)
print('Number of follows: ', account.follows_count)
print('Is private: ', account.is_private)
print('Is verified: ', account.is_verified)
print('Biography: ', account.biography)
```

Account info:

Id: 3089598226

Username: lilmiquela

Full name: Miquela

Number of published posts: 948

Number of followers: 2919809

Number of follows: 1886

Is private: False

Is verified: True

Biography: #BlackLivesMatter

Change-seeking robot with the drip

Get Real, Miquela NEW EPISODE

Influenceren @lilmiquela er en robot/virtuel instagrammer. Hun er programmeret som en 19-årig pige bosiddende i LA og har med sine 2,9 mio følgere skabt en masse popularitet og kontroverser på det sociale medie instagram, hvor hun bl.a. reklamerer for high-end brands og promoverer sin støtte til blandt andet #BlackLivesMatter og andre interessebevægelser. Hun deler dagligt indhold relateret til sin hverdag, herunder kæresteproblemer, 'Outfit of the day' og 'storytimes' - problematikken bunder dog i, at hun i virkeligheden ikke eksisterer, da alt hendes indhold er baseret på et fiktivt narrativ omkring hendes robotidentitet kreeret af virksomheden Brud.

Til generering af relevant data omkring @lilmiquelas instagramopslag, gør vi metoden 'web scraping'

2 Web scraping og generering af CSV-fil

```
[102]: Image(filename='Skærbillede 2020-12-07 kl. 11.24.38.png')
```

```
[102]:
```



```

import pandas as pd
import urllib.request
from collections import Counter

from newtest import recent_post_links, insta_link_details, insta_url_to_img

#Change post count to change amount of post downloaded
example_username_urls = recent_post_links('lilmiquela', post_count=30)
print(example_username_urls)
example_username_details = [insta_link_details(url) for url in example_username_urls]
example_username = pd.DataFrame(example_username_details)
example_username.head()
csv = example_username.to_csv('/Users/Anton/Desktop/Python/Instagram Scraper/example_username.csv')
print(csv)

```

Skal ændres til det rigtige billede og linket skal sættes ind

2.1 Det genererede datasæt

```
[103]: df=pd.read_csv('example_username-copy.csv')
df
```

```
[103]:
```

	Unnamed: 0	link	type	likes/views	\
0	0	https://www.instagram.com/p/CHTOZ8-nnfG/	photo	43,282	
1	1	https://www.instagram.com/p/CHQ6AWXnW5J/	photo	34,263	
2	2	https://www.instagram.com/p/CH0sztcHHXF/	photo	44,715	
3	3	https://www.instagram.com/p/CHLts9MnWam/	photo	41,334	
4	4	https://www.instagram.com/p/CHJocBknqe-/	photo	53,704	
..	
245	245	https://www.instagram.com/p/B9hpoJ_noAc/	photo	72,411	
246	246	https://www.instagram.com/p/B9estyun6U1/	video	1,319,915	
247	247	https://www.instagram.com/p/B9cNh7OHLH/	video	2,826,650	
248	248	https://www.instagram.com/p/B9ZzyfGna6a/	photo	41,180	
249	249	https://www.instagram.com/p/B9XstBrHqrj/	photo	86,739	

	age	comment	\
0	1h	This is the first step of MANY. KEEP THAT SAME...	
1	17h	Wearing all Praying cause that's what all I be...	
2	1d	Fav distraction: Looking at apartments I can't...	
3	1d	I know patience is a virtue, but DAMN!! Are we...	
4	1d	Anxiety meter already broke Recharging with ...	
..	
245	26w	Went through a breakup, wrote a song about it...	

```

246 15w Y'all know how I am...I couldn't help myself ...
247 4w Big things coming soon!
248 28w New tunes coming soon. Zoom in on my @samsungm...
249 7w Long caption Alert \nSo this is the inevi...

```

```

           hashtags      mentions
0           NaN          NaN
1           NaN          NaN
2           NaN          NaN
3           NaN          NaN
4           NaN          NaN
..          ...          ...
245      #SpeakUp          NaN
246           NaN          NaN
247           NaN          NaN
248  ['#TeamGalaxy', '#ad']  @samsungmobile
249           NaN          NaN

```

[250 rows x 8 columns]

3 Datarensning

```

[104]: #Oversigt over datasættens features
df.columns

```

```

[104]: Index(['Unnamed: 0', 'link', 'type', 'likes/views', 'age', 'comment',
           'hashtags', 'mentions'],
          dtype='object')

```

```

[105]: df=df.rename(columns={"Unnamed: 0": "post number", "comment": "caption"})
df.head()

```

```

[105]:   post number      link      type  likes/views \
0         0  https://www.instagram.com/p/CHTOZ8-nnfG/  photo    43,282
1         1  https://www.instagram.com/p/CHQ6AWXnW5J/  photo    34,263
2         2  https://www.instagram.com/p/CH0sztcHHXF/  photo    44,715
3         3  https://www.instagram.com/p/CHLts9MnWam/  photo    41,334
4         4  https://www.instagram.com/p/CHJocBknqe-/  photo    53,704

      age      caption  hashtags  mentions
0    1h  This is the first step of MANY. KEEP THAT SAME...    NaN    NaN
1   17h  Wearing all Praying cause that's what all I be...    NaN    NaN
2    1d  Fav distraction: Looking at apartments I can't...    NaN    NaN
3    1d  I know patience is a virtue, but DAMN!! Are we...    NaN    NaN
4    1d  Anxiety meter already broke  Recharging with ...    NaN    NaN

```



```
[106]: df.columns
```

```
[106]: Index(['post number', 'link', 'type', 'likes/views', 'age', 'caption',  
         'hashtags', 'mentions'],  
        dtype='object')
```

```
[107]: to_drop=['age'] # Her har vi fjernet en masse data, som ikke indeholdte brugbar  
      ↪ information  
df.drop(to_drop, inplace=True, axis=1)  
df.head()
```

```
[107]:
```

	post number	link	type	likes/views \
0	0	https://www.instagram.com/p/CHTOZ8-nnfG/	photo	43,282
1	1	https://www.instagram.com/p/CHQ6AWXnW5J/	photo	34,263
2	2	https://www.instagram.com/p/CHOsztcHHXF/	photo	44,715
3	3	https://www.instagram.com/p/CHLts9MnWam/	photo	41,334
4	4	https://www.instagram.com/p/CHJocBknqe-/	photo	53,704

	caption	hashtags	mentions
0	This is the first step of MANY. KEEP THAT SAME...	NaN	NaN
1	Wearing all Praying cause that's what all I be...	NaN	NaN
2	Fav distraction: Looking at apartments I can't...	NaN	NaN
3	I know patience is a virtue, but DAMN!! Are we...	NaN	NaN
4	Anxiety meter already broke Recharging with ...	NaN	NaN

```
[108]: #Erstatter komma med ingen ting og konverterer likes/views til integer  
df['likes/views'] = df['likes/views'].str.replace(",","").astype(int)  
df.head()
```

```
[108]:
```

	post number	link	type	likes/views \
0	0	https://www.instagram.com/p/CHTOZ8-nnfG/	photo	43282
1	1	https://www.instagram.com/p/CHQ6AWXnW5J/	photo	34263
2	2	https://www.instagram.com/p/CHOsztcHHXF/	photo	44715
3	3	https://www.instagram.com/p/CHLts9MnWam/	photo	41334
4	4	https://www.instagram.com/p/CHJocBknqe-/	photo	53704

	caption	hashtags	mentions
0	This is the first step of MANY. KEEP THAT SAME...	NaN	NaN
1	Wearing all Praying cause that's what all I be...	NaN	NaN
2	Fav distraction: Looking at apartments I can't...	NaN	NaN
3	I know patience is a virtue, but DAMN!! Are we...	NaN	NaN
4	Anxiety meter already broke Recharging with ...	NaN	NaN

```
[109]: df.dtypes
```

```
[109]: post number    int64  
      link          object
```

```

type          object
likes/views   int64
caption       object
hashtags      object
mentions      object
dtype: object

```

```

[110]: #antal tegn
df["caption length"] = df["caption"].str.len()
df

```

```

[110]:
      post number      link  type \
0          0  https://www.instagram.com/p/CHTOZ8-nnfG/  photo
1          1  https://www.instagram.com/p/CHQ6AWXnW5J/  photo
2          2  https://www.instagram.com/p/CH0sztcHHXF/  photo
3          3  https://www.instagram.com/p/CHLts9MnWam/  photo
4          4  https://www.instagram.com/p/CHJocBknqe-/  photo
..         ...
245        245  https://www.instagram.com/p/B9hpoJ_noAc/  photo
246        246  https://www.instagram.com/p/B9estyun6U1/  video
247        247  https://www.instagram.com/p/B9cNh7OHLH/  video
248        248  https://www.instagram.com/p/B9ZzyfGna6a/  photo
249        249  https://www.instagram.com/p/B9XstBrHqrj/  photo

      likes/views      caption \
0          43282  This is the first step of MANY. KEEP THAT SAME...
1          34263  Wearing all Praying cause that's what all I be...
2          44715  Fav distraction: Looking at apartments I can't...
3          41334  I know patience is a virtue, but DAMN!! Are we...
4          53704  Anxiety meter already broke  Recharging with ...
..         ...
245         72411  Went through a breakup, wrote a song about it...
246        1319915  Y'all know how I am...I couldn't help myself ...
247        2826650  Big things coming soon!
248         41180  New tunes coming soon. Zoom in on my @samsungm...
249         86739  Long caption Alert \nSo this is the inevi...

      hashtags      mentions  caption length
0          NaN          NaN             82
1          NaN          NaN             55
2          NaN          NaN             73
3          NaN          NaN            102
4          NaN          NaN             90
..         ...
245        #SpeakUp          NaN            119
246          NaN          NaN             59
247          NaN          NaN             23

```

```

248 ['#TeamGalaxy', '#ad'] @samsungmobile 243
249 NaN NaN 1487

```

[250 rows x 8 columns]

```

[111]: def split_count(caption):
        return len([i for i in caption if i in emoji.UNICODE_EMOJI])
df["emoji count"] = df["caption"].apply(split_count)
df

```

```

[111]:
post number      link      type \
0          0  https://www.instagram.com/p/CHTOZ8-nnfG/  photo
1          1  https://www.instagram.com/p/CHQ6AWXnW5J/  photo
2          2  https://www.instagram.com/p/CH0sztcHHXF/  photo
3          3  https://www.instagram.com/p/CHLts9MnWam/  photo
4          4  https://www.instagram.com/p/CHJocBknqe-/  photo
..         ...
245        245  https://www.instagram.com/p/B9hpoJ_noAc/  photo
246        246  https://www.instagram.com/p/B9estyun6U1/  video
247        247  https://www.instagram.com/p/B9cNh7OHLH/  video
248        248  https://www.instagram.com/p/B9ZzyfGna6a/  photo
249        249  https://www.instagram.com/p/B9XstBrHqrj/  photo

likes/views      caption \
0          43282  This is the first step of MANY. KEEP THAT SAME...
1          34263  Wearing all Praying cause that's what all I be...
2          44715  Fav distraction: Looking at apartments I can't...
3          41334  I know patience is a virtue, but DAMN!! Are we...
4          53704  Anxiety meter already broke   Recharging with ...
..         ...
245         72411  Went through a breakup, wrote a song about it...
246        1319915  Y'all know how I am...I couldn't help myself ...
247        2826650  Big things coming soon!
248         41180  New tunes coming soon. Zoom in on my @samsungm...
249         86739  Long caption Alert \nSo this is the inevi...

      hashtags      mentions  caption length  emoji count
0          NaN          NaN           82           0
1          NaN          NaN           55           1
2          NaN          NaN           73           0
3          NaN          NaN          102           0
4          NaN          NaN           90           3
..         ...
245        #SpeakUp          NaN          119           0
246          NaN          NaN           59           1
247          NaN          NaN           23           0
248  ['#TeamGalaxy', '#ad'] @samsungmobile 243           3

```

249	NaN	NaN	1487	2
-----	-----	-----	------	---

[250 rows x 9 columns]

```
[112]: # Ny kolonne med antal hastags for hvert post
df["hashtag count"] = df["hashtags"].str.count('#')
df.head()
```

```
[112]: post number          link  type  likes/views \
0          0  https://www.instagram.com/p/CHTOZ8-nnfG/  photo      43282
1          1  https://www.instagram.com/p/CHQ6AWXnW5J/  photo      34263
2          2  https://www.instagram.com/p/CH0sztcHHXF/  photo      44715
3          3  https://www.instagram.com/p/CHLts9MnWam/  photo      41334
4          4  https://www.instagram.com/p/CHJocBknqe-/  photo      53704
```

	caption	hashtags	mentions	\
0	This is the first step of MANY. KEEP THAT SAME...	NaN	NaN	
1	Wearing all Praying cause that's what all I be...	NaN	NaN	
2	Fav distraction: Looking at apartments I can't...	NaN	NaN	
3	I know patience is a virtue, but DAMN!! Are we...	NaN	NaN	
4	Anxiety meter already broke Recharging with ...	NaN	NaN	

	caption length	emoji count	hashtag count
0	82	0	NaN
1	55	1	NaN
2	73	0	NaN
3	102	0	NaN
4	90	3	NaN

```
[113]: # Ny kolonne med antal hastags for hvert post
df["mentions count"] = df["mentions"].str.count('@')
df.head()
```

```
[113]: post number          link  type  likes/views \
0          0  https://www.instagram.com/p/CHTOZ8-nnfG/  photo      43282
1          1  https://www.instagram.com/p/CHQ6AWXnW5J/  photo      34263
2          2  https://www.instagram.com/p/CH0sztcHHXF/  photo      44715
3          3  https://www.instagram.com/p/CHLts9MnWam/  photo      41334
4          4  https://www.instagram.com/p/CHJocBknqe-/  photo      53704
```

	caption	hashtags	mentions	\
0	This is the first step of MANY. KEEP THAT SAME...	NaN	NaN	
1	Wearing all Praying cause that's what all I be...	NaN	NaN	
2	Fav distraction: Looking at apartments I can't...	NaN	NaN	
3	I know patience is a virtue, but DAMN!! Are we...	NaN	NaN	
4	Anxiety meter already broke Recharging with ...	NaN	NaN	

	caption length	emoji count	hashtag count	mentions count
0	82	0	NaN	NaN
1	55	1	NaN	NaN
2	73	0	NaN	NaN
3	102	0	NaN	NaN
4	90	3	NaN	NaN

4 Dataanalyse

```
[114]: # Hvor mange rækker er der?
print('Antal rækker:', len(df))
# Hvor mange kolonner er der - vist i en 'tuple'
print('Antal kolonner:', df.shape)
#Hvor mange celler er der?
print('Antal af celler:', df.size)
# Hvad hedder kolonnerne?
print('Kolonnerne hedder:', df.columns)
# Hvilke datatyper er der?
print('Datatyperne for kolonnerne er: \n', df.dtypes)
```

```
Antal rækker: 250
Antal kolonner: (250, 11)
Antal af celler: 2750
Kolonnerne hedder: Index(['post number', 'link', 'type', 'likes/views',
'caption', 'hashtags',
'mentions', 'caption length', 'emoji count', 'hashtag count',
'mentions count'],
dtype='object')
Datatyperne for kolonnerne er:
post number      int64
link             object
type             object
likes/views      int64
caption          object
hashtags         object
mentions         object
caption length   int64
emoji count      int64
hashtag count    float64
mentions count   float64
dtype: object
```

4.0.1 Descriptiv statistik på likes/views

```
[115]: df['likes/views'].isnull().any() #finder ud af om der er NaN værdier
```

```
[115]: False
```

```
[116]: round(df['likes/views'].describe(),0) # Viser basal deskriptiv statistik
```

```
[116]: count          250.0  
      mean          264728.0  
      std           556896.0  
      min            9949.0  
      25%           60431.0  
      50%           75133.0  
      75%          111128.0  
      max          3353236.0  
      Name: likes/views, dtype: float64
```

```
[117]: df['likes/views'].mean()
```

```
[117]: 264728.272
```

```
[118]: round(df['likes/views'].mean(),0) # Gennemsnittet afrundet
```

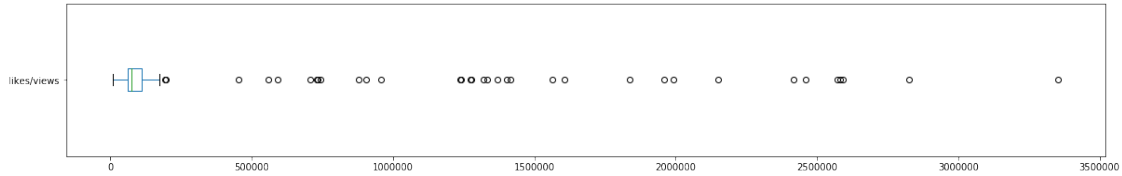
```
[118]: 264728.0
```

```
[119]: df['likes/views'].mode() #Beregner typetallet i antal likes
```

```
[119]: 0          9949  
      1         11979  
      2         13177  
      3         24459  
      4         29647  
      ...  
      245        2572471  
      246        2582898  
      247        2592604  
      248        2826650  
      249        3353236  
      Length: 250, dtype: int64
```

```
[120]: #visualisering af den deskriptive fordeling for kolonnen 'like/views'  
      df['likes/views'].plot(kind='box',vert=False, figsize=(20,3))
```

```
[120]: <matplotlib.axes._subplots.AxesSubplot at 0x7fd17c317a50>
```

Hvad betyder det at boxplottet ser sådan ud? Variansen af likes spreder sig fra 9949 - 3 mio. Det er dog den største mængde af likes som fordeler i den lavere ende. Det er derfor figuren ser mærkelig ud

```
[121]: print('Sorteret efter flest likes')
df['likes/views'].sort_values(ascending=False)[:10]
```

Sorteret efter flest likes

```
[121]: 242      3353236
      247      2826650
      201      2592604
      145      2582898
      155      2572471
      199      2460750
      81       2416490
      240      2150225
      235      1992155
      123      1959234
      Name: likes/views, dtype: int64
```

```
[122]: #Får URL'en for de mest likede billeder
print('1:',df.link[242])
print('2:',df.link[247])
print('3:',df.link[201])
```

```
1: https://www.instagram.com/p/B9nrz30Hjmp/
2: https://www.instagram.com/p/B9cNh70HQLH/
3: https://www.instagram.com/p/B_VCvsxnM7H/
```

```
[123]: #Viser fordelingen af likes over tid
alt.Chart(df).mark_line(point=True).encode(
    # Years on the X axis
    x=alt.X('post number:Q', axis=alt.Axis(format='c', title='Post: Ny ->
    ↪Gammel')),
    # Number of articles on the Y axis
```

```

y=alt.Y('likes/views:Q', axis=alt.Axis(format='d', title='Antal_
↳likes')),

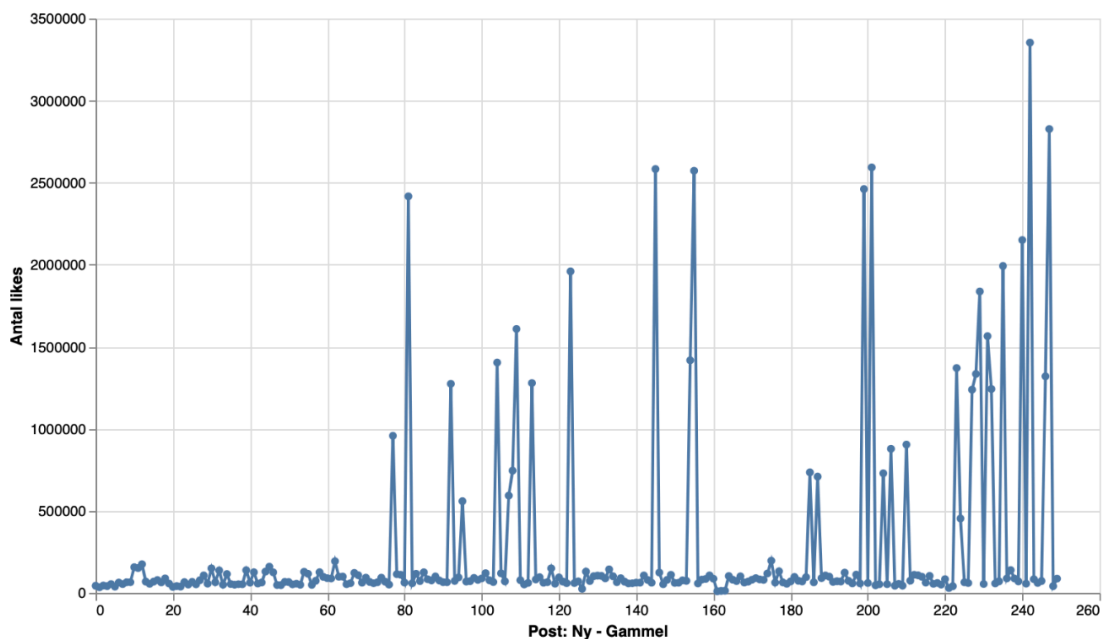
# Display details when you hover over a point
tooltip=[alt.Tooltip('årstal:Q', title='Post: Ny - Gammel'), alt.
↳Tooltip('likes/views:Q',
                                title='post number', format=',')].
↳properties(width=700, height=400)

```

[123]: alt.Chart(...)

[124]: Image(filename='Skærbillede 2020-12-07 kl. 14.08.06.png')

[124]:



4.0.2 Descriptiv statistik på længden af captions

[125]: df['caption length'].isnull().any() *#finder ud af om der er NaN værdier*

[125]: False

[126]: round(df['caption length'].describe(),0) *# Viser basal deskriptiv statistik*

[126]: count 250.0
mean 111.0
std 126.0

```
min          5.0
25%         59.0
50%         86.0
75%        133.0
max        1487.0
Name: caption length, dtype: float64
```

```
[127]: df['caption length'].mean()
```

```
[127]: 110.96
```

```
[128]: round(df['caption length'].mean(),0) # Gennemsnittet afrundet
```

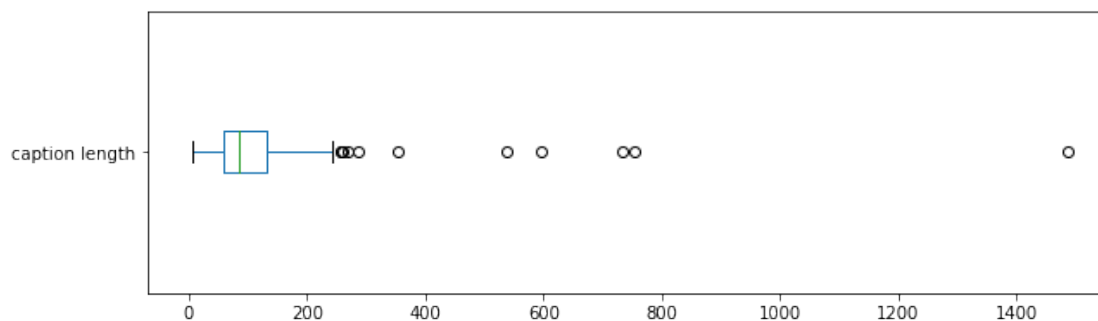
```
[128]: 111.0
```

```
[129]: df['caption length'].mode() #Beregner typetallet i antal likes
```

```
[129]: 0    60
dtype: int64
```

```
[130]: df['caption length'].plot(kind='box',vert=False, figsize=(10,3))
```

```
[130]: <matplotlib.axes._subplots.AxesSubplot at 0x7fd17c2f6410>
```



Hvad betyder det at boxplottet over caption length ser sådan ud? Mængden af tegn i @lilmiquelas captions spreder sig fra 5 - 1487 likes. Der er dog flest af hendes captions som består af under 100 tegn som fordeler sig blandt gennemsnittet. Det er derfor figuren ser mærkelig ud

4.1 Textmining

```
[131]: # Opstiller en samlet liste bestående af alle captions
alle=list()
for i in df.caption:
    alle.append(i)
    str(alle)
alle[:20]
```

```
[131]: ["This is the first step of MANY. KEEP THAT SAME ENERGY Y'ALL - still so much to
do.",
"Wearing all Praying cause that's what all I been doin ",
"Fav distraction: Looking at apartments I can't afford. Seriously: TRY IT!",
'I know patience is a virtue, but DAMN!! Are we planning a party or an escape,
your girl NEEDS TO KNOW!',
'Anxiety meter already broke Recharging with cuddles and sheet masks
PLEASE SEND TACO.',
"Y'ALL KNOW WHAT TIME IT IS. Get out there!",
"Running out of ways to remind y'all, but you can't spell VOTE without the V.",
'Supermodel felt too basic for Halloween, but the fit was FIRE. So, here we
are.',
'Dressed as fierce queens this whole week, but saved the FIERCEST queen for
last: #91, Mr. Rodman, I LOVE YOU.',
"Leeloo Dallas is cute, but it's Ruby Rhod for me.",
'To my fans .... I want to thank you guys so much for your support throughout the
years !!!!! PS first pic is the original !!!!!',
'Throwback to last year when I dressed up as another ageless beauty.
#19forever',
'@pablovittar sis, did I snap?',
"Had to give ya'll the full look. We keep it strictly prickly.",
'Had lunch with a human emoji. Find a cuter . I dare u.',
'When you think you\'re a cute mermaid, but someone yells "GO \'HEAD THICC
COVID-19" while you wait in line for boba ',
"If y'all call me Regina George one more time...",
"Drop your low effort Halloween costumes in the comments (I'm not going
anywhere, and honestly neither should y'all)",
'Bringing back yoga pants. Fight Me.',
'MayMay told me the story behind "Ring Around the Rosie"... SO. DARK.']
```

```
[132]: # Opstiller en ny liste bestående af alle ord fra caption
tot=[]
for i in df.caption:
    tot.extend(i.split())
tot[:20]
```

```
[132]: ['This',
'is',
```

```
'the',
'first',
'step',
'of',
'MANY.',
'KEEP',
'THAT',
'SAME',
'ENERGY',
"Y'ALL",
'-',
'still',
'so',
'much',
'to',
'do.',
'Wearing',
'all']
```

```
[133]: len(tot) #Antallet af ord
```

```
[133]: 5083
```

```
[134]: #Laver ovenstående liste om til et 'set', der automatisk fjerner alle redundante
↳ord
unique=set(tot)
#unique
```

```
[135]: # optæller antallet af unikke ord
uni_count=0
for i in unique:
    uni_count=uni_count+1

print('Antallet af unikke ord er:',uni_count)
```

Antallet af unikke ord er: 2243

```
[136]: # Biblioteket Counter bruges til at optælle hyppigheden for hvert ord
counter=Counter(tot)
counter.most_common(20)
```

```
[136]: [('the', 138),
('to', 122),
('and', 109),
('I', 107),
('my', 100),
('a', 95),
```

```
( 'in', 89),
( 'is', 75),
( 'for', 61),
( 'of', 48),
( 'on', 45),
( 'this', 40),
( 'but', 39),
( 'you', 39),
( 'me', 32),
( 'with', 31),
( 'like', 27),
( 'your', 25),
( 'it', 25),
( 'out', 24)]
```

```
[137]: # Downloader de engelske 'stopwords', der bruges til at fjerne de mest
        ↪ almindelige ord
        nltk.download('stopwords')
```

```
[nltk_data] Downloading package stopwords to /home/jovyan/nltk_data...
[nltk_data] Package stopwords is already up-to-date!
```

```
[137]: True
```

```
[138]: stop_words = set(stopwords.words('english'))

def remove_stopwords(tot): #fjerner stopwords fra listen af ord
    return [t for t in tot if t not in stop_words]

# genoptæller ordene
counter = Counter(remove_stopwords(tot))
print(counter.most_common(20)) #print de mest hyppigt brugte ord
```

```
[('I', 107), ('like', 27), ('I'm', 22), ('y'all', 20), ('video', 18), ('know',
16), ('#SpeakUp', 16), ('love', 15), ('The', 15), ('IN', 15), ('Link', 14),
('LINK', 13), ('-', 12), ('#miquelacovers', 12), ('link', 11), ('This', 10),
('girl', 10), ('new', 10), ('HARD', 10), ('first', 9)]
```

```
[139]: # Opstiller en dataframe med ord og deres optællinger
freq_df = pd.DataFrame.from_records(counter.most_common(20),
                                     columns=['words', 'count'])

freq_df
```

```
[139]:
```

	words	count
0	I	107
1	like	27

2	I'm	22
3	y'all	20
4	video	18
5	know	16
6	#SpeakUp	16
7	love	15
8	The	15
9	IN	15
10	Link	14
11	LINK	13
12	-	12
13	#miquelacovers	12
14	link	11
15	This	10
16	girl	10
17	new	10
18	HARD	10
19	first	9

```
[140]: #beregner procenten for ordenes hyppighed ud fra de 20 mest fremkomne ord
freq_df['%']=round(freq_df['count']/freq_df['count'].sum()*100)
```

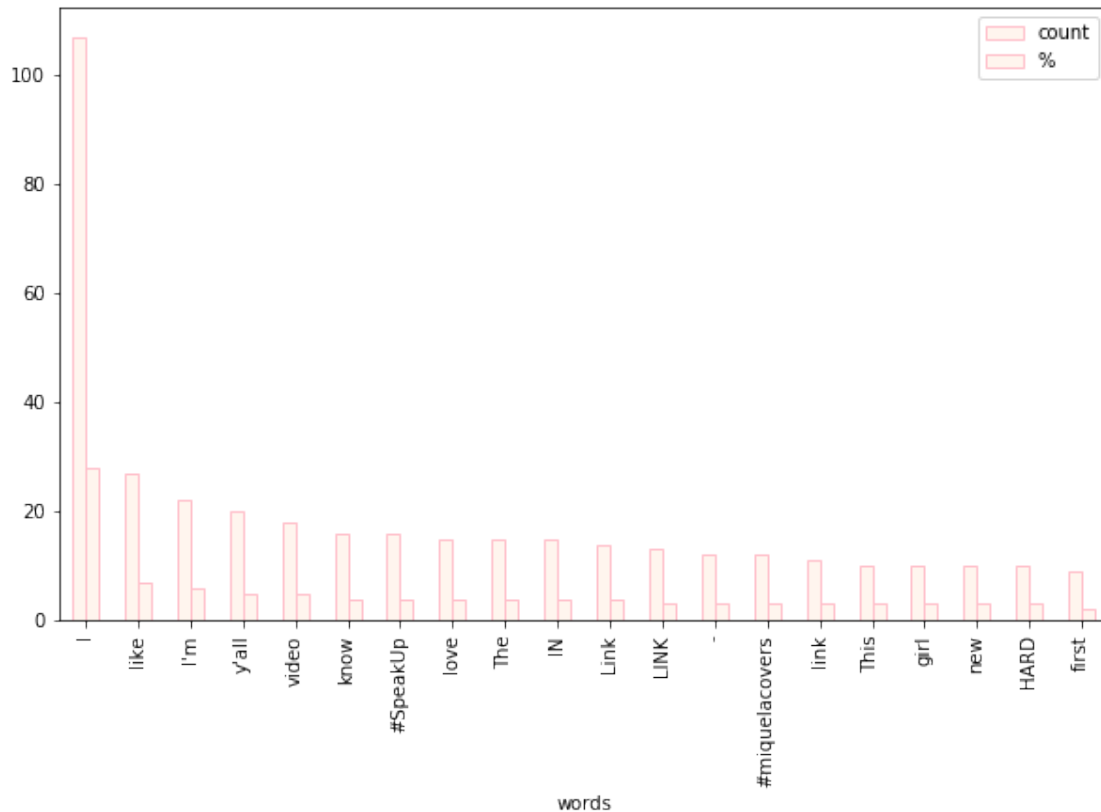
```
[141]: freq_df
```

```
[141]:
```

	words	count	%
0	I	107	28.0
1	like	27	7.0
2	I'm	22	6.0
3	y'all	20	5.0
4	video	18	5.0
5	know	16	4.0
6	#SpeakUp	16	4.0
7	love	15	4.0
8	The	15	4.0
9	IN	15	4.0
10	Link	14	4.0
11	LINK	13	3.0
12	-	12	3.0
13	#miquelacovers	12	3.0
14	link	11	3.0
15	This	10	3.0
16	girl	10	3.0
17	new	10	3.0
18	HARD	10	3.0
19	first	9	2.0

```
[142]: #Visualisering af ordnes hyppighed
freq_df.plot.bar(x='words', color='seashell', edgecolor='pink', figsize=(10,6))
```

```
[142]: <matplotlib.axes._subplots.AxesSubplot at 0x7fd17c391990>
```



```
[143]: def wordcloud(counter): #opstiller en funktion der generere et wordcloud ud fra
    ↪ordned hyppighed
    wc = WordCloud(width=1200, height=800,
                    max_words=100,
                    background_color="white")
    wc.generate_from_frequencies(counter)

    #Plot
    fig=plt.figure(figsize=(10,6))
    plt.imshow(wc, interpolation='bilinear')
    plt.axis("off")
    plt.tight_layout(pad=0)
    plt.show()
```

```
[144]: #Create wordcloud
wordcloud(counter)
```



Nu undersøges brugen af emojis:

```
[145]: def extract_emojis(s): #Opstiller en liste udelukkende bestående af de
      ↪ benyttede emojis
      return ''.join(c for c in s if c in emoji.UNICODE_EMOJI)
emo=list(extract_emojis(str(alles)))
```

```
[146]: emo[:20]
```

[illegible]

```
' ',
' ',
' ',
' ',
' ',
' ',
' ']
```

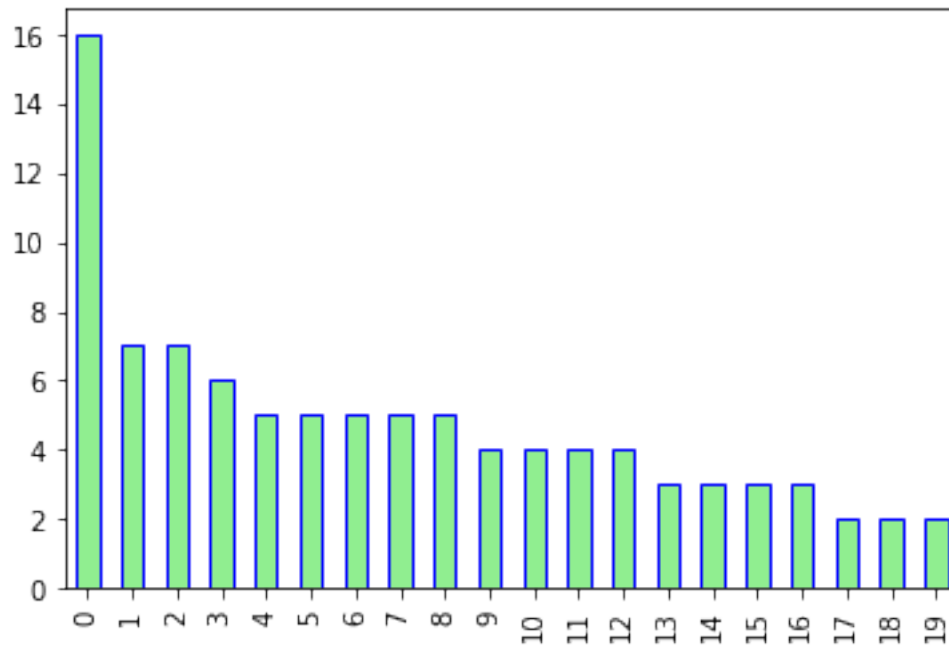
```
[147]: freq_df1 = pd.DataFrame.from_records(Counter(emo).most_common(20), #Benytter
↳ biblioteket Counter til at beregne emojisne hyppighed
                                columns=['emoji', 'count'])
freq_df1['%']=round(freq_df1['count']/freq_df1['count'].sum()*100)
freq_df1
```

```
[147]:
```

	emoji	count	%
0		16	17.0
1		7	7.0
2		7	7.0
3		6	6.0
4		5	5.0
5		5	5.0
6		5	5.0
7		5	5.0
8		5	5.0
9		4	4.0
10		4	4.0
11		4	4.0
12		4	4.0
13		3	3.0
14		3	3.0
15		3	3.0
16		3	3.0
17		2	2.0
18		2	2.0
19		2	2.0

```
[148]: #Visualisere hyppigheden for emojis
freq_df1['count'].plot.bar(color=['lightgreen'], edgecolor='blue')
```

```
[148]: <matplotlib.axes._subplots.AxesSubplot at 0x7fd17c382290>
```



Nu undersøges brugen af hashtags:

```
[149]: df['hashtags'].isnull().any() #finder ud af om der er NaN værdier
```

[149]: True

```
[150]: #Fjerner NaN's fra listen
cleanedList = [x for x in df['hashtags'] if str(x) != 'nan' and '[' and ']']
cleanedList
```

```
[150]: ['#nationalpastaday',
        ['#returnyourballotday', '#makeyourvotecount']",
        '#Lakers',
        '#GCFAItalia',
        '#comicon',
        '#Girlfriends',
        "['#reading', '#comments']",
        '#botgirlsummer',
        '#answeringquestions',
        "['#NationalVoterRegistrationDay', '#MOBBTheVote', '#MarchOnBallotBoxes']",
        '#bigthingscomingsoon',
        '#climatechange',
        '#gamergirl',
        '#bershkastyle',
        "['#anime', '#animeedits']",
```



```
'#SpeakUp',
'#speakup',
'#SpeakUp',
"['#TeamGalaxy', '#ad']"]
```

```
[151]: str(df['hashtags'])
```

```
[151]: "0                NaN\n1                NaN\n2                NaN\n3                NaN\n4                NaN\n...          \n245          #SpeakUp\n246          NaN\n247          NaN\n248    ['#TeamGalaxy', '#ad']\n249          NaN\nName: hashtags, Length: 250, dtype: object"
```

```
[152]: tags=[]
for i in str(df['hashtags']):
    tags.extend(i.split())
tags[:20]
```

```
[152]: ['0',
'N',
'a',
'N',
'1',
'N',
'a',
'N',
'2',
'N',
'a',
'N',
'3',
'N',
'a',
'N',
'4',
'N',
'a',
'N']
```

```
[153]: tot=[]
for i in df.caption:
    tot.extend(i.split())
tot[:20]
```

```
[153]: ['This',
'is',
'the',
```

```
'first',
'step',
'of',
'MANY.',
'KEEP',
'THAT',
'SAME',
'ENERGY',
"Y'ALL",
'-',
'still',
'so',
'much',
'to',
'do.',
'Wearing',
'all']
```

```
[ ]:
```

5 Machine Learning

```
[154]: df.type=pd.Categorical(df.type)
df['type']=df['type'].cat.codes #ændrer Type til at være en kategorisk kodet
↪ værdi
df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 250 entries, 0 to 249
Data columns (total 11 columns):
post number      250 non-null int64
link             250 non-null object
type             250 non-null int8
likes/views      250 non-null int64
caption          250 non-null object
hashtags         66 non-null object
mentions         53 non-null object
caption length   250 non-null int64
emoji count      250 non-null int64
hashtag count    66 non-null float64
mentions count   53 non-null float64
dtypes: float64(2), int64(4), int8(1), object(4)
memory usage: 19.9+ KB
```

```
[155]: # y dækker over den værdi som vi ønsker at forudsige
y=df['likes/views'].copy()
y
```

```
[155]: 0      43282
1      34263
2      44715
3      41334
4      53704
...
245     72411
246    1319915
247    2826650
248      41180
249     86739
Name: likes/views, Length: 250, dtype: int64
```

```
[156]: X=df[['type', 'caption length']].copy()
X # X dækker over de værdier som vores udforudsigelse bygger på
```

```
[156]:      type  caption length
0        0             82
1        0             55
2        0             73
3        0            102
4        0             90
..      ...             ...
245      0            119
246      1             59
247      1             23
248      0            243
249      0           1487
```

[250 rows x 2 columns]

```
[157]: train_X, test_X, train_y, test_y = train_test_split(X, y, train_size=.8,
↳random_state = 123)
#train og test-sættet splittes op således at train fylder 80% og test fylder 20%
```

```
[158]: train_X
```

```
[158]:      type  caption length
130      0             71
238      0            193
145      1             87
136      0            238
23       0             11
```

```

..      ...
98      0      118
220     0      174
66      0      68
126     0      538
109     1      155

```

```
[200 rows x 2 columns]
```

```
[159]: test_X
```

```

[159]:      type  caption length
127     0      143
187     0      68
24      0      34
206     1      62
235     1     234
31      0     111
20      0      9
202     0     139
196     0     119
201     1      48
85      0      66
82      0      69
19      0      68
163     0      19
198     0      57
122     0     184
234     0     150
95      1      36
231     1     240
156     0     149
53      0     134
61      0      18
188     0      86
157     0      94
215     0      33
144     0      72
52      0      65
219     0      29
120     0      52
137     0     149
89      0      62
173     0     123
191     0      80
90      0     145
186     0     105

```

218	0	107
81	1	53
42	0	130
33	0	64
37	0	82
232	1	355
150	0	104
5	0	42
4	0	90
177	0	115
228	1	111
205	0	90
21	0	60
148	0	134
147	0	89

```
[160]: clf = LinearRegression(normalize=True) #Linære regression benyttes
```

```
[161]: clf.fit(train_X,train_y) #modellen trænes
```

```
[161]: LinearRegression(copy_X=True, fit_intercept=True, n_jobs=None, normalize=True)
```

```
[162]: y_pred=clf.predict(test_X)
       y_pred #Vores forudsigelser på antallet af likes
```

```
[162]: array([ 76437.8151625 ,  93678.92809985, 101494.89929811,
              1623133.7120003 , 1583594.09299731,  83794.0233491 ,
              107241.9369439 ,  77357.34118582,  81954.97130245,
              1626352.05308194,  94138.69111151,  93449.04659402,
              93678.92809985, 104943.12188558,  96207.62466399,
              67012.67342341,  74828.64462168, 1629110.63115191,
              1582214.80396232,  75058.52612751,  78506.74871498,
              105173.00339141,  89541.06099488,  87702.00894823,
              101724.78080394,  92759.40207652,  94368.57261734,
              102644.30682727,  97357.03219315,  75058.52612751,
              95058.21713484,  81035.44527912,  90920.35002987,
              75978.05215084,  85173.31238409,  84713.54937243,
              1625202.64555278,  79426.27473831,  94598.45412317,
              90460.58701821, 1555778.43079172,  85403.19388992,
              99655.84725146,  88621.53497156,  82874.49732578,
              1611869.51821456,  88621.53497156,  95517.9801465 ,
              78506.74871498,  88851.41647739])
```

```
[163]: #coefficenterne
       print('Coefficienterne:', clf.coef_)
       #gennemsnitlige kvadrerede forudsigelsesfejl også kendt som mean squared error
       print('Den gennemsnitlige kvadrerede forudsigelsesfejl: %.2f')
```

```

    % mean_squared_error(test_y, y_pred))
# Forudsigelsesgraden: 1 er en perfect forudsigelse
print('Graden for præcis forudsigelse : %.2f'
      % r2_score(test_y, y_pred))

```

Coefficienterne: [1.52807549e+06 -2.29881506e+02]

Den gennemsnitlige kvadrerede forudsigelsesfejl: 80684597596.17

Graden for præcis forudsigelse : 0.79

In statistics, the mean squared error (MSE) of an estimator (of a procedure for estimating an unobserved quantity) measures the average of the squares of the errors --- that is, the average squared difference between the estimated values and what is estimated. MSE is a risk function, corresponding to the expected value of the squared error loss. The fact that MSE is almost always strictly positive (and not zero) is because of randomness or because the estimator does not account for information that could produce a more accurate estimate.

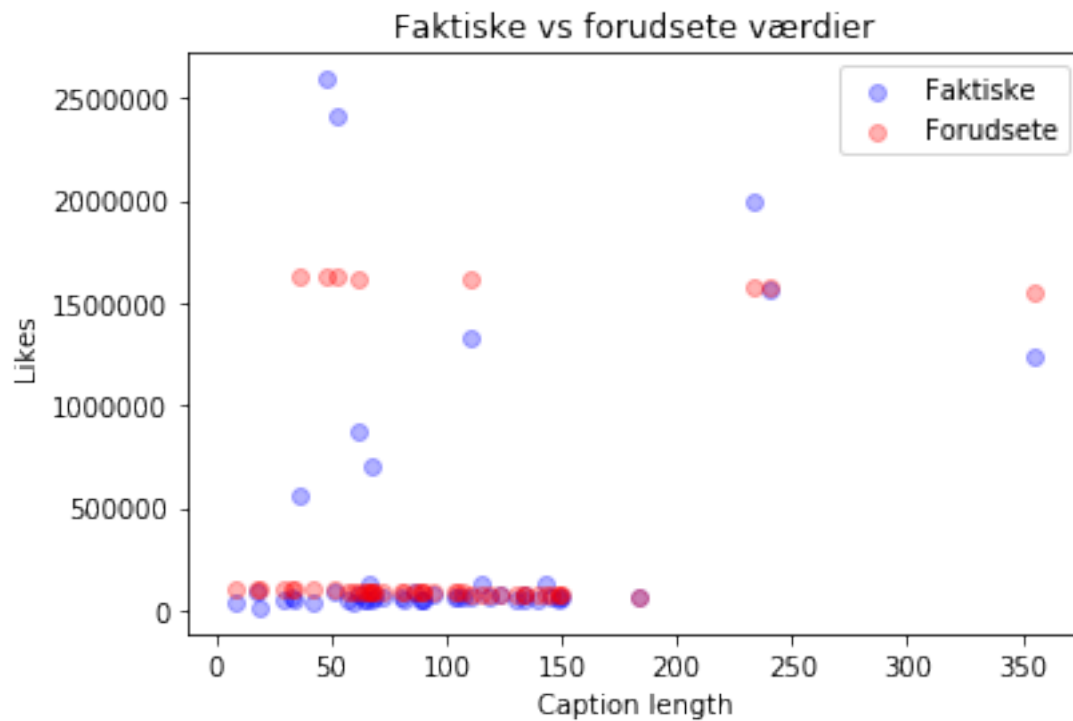
```

[164]: #visualisering af de faktiske og forudsete værdier for antallet at likes
_, ax = plt.subplots()

ax.scatter(x = test_X['caption length'], y=test_y, c = 'blue', label = 'Faktiske', alpha = 0.3)
ax.scatter(x = test_X['caption length'], y=y_pred, c = 'red', label = 'Forudsete', alpha = 0.3)

plt.title('Faktiske vs forudsete værdier')
plt.xlabel('Caption length')
plt.ylabel('Likes')
plt.legend()
plt.show()

```

6 Sentiment analysis

```
[165]: #Opstiller en liste der indeholder sentimentets polaritet for hver caption
polarity=[]
for i in df.caption:
    polarity.append(TextBlob(i).sentiment.polarity)
polarity[:10]
```

```
[165]: [0.2375,
0.0,
-0.41666666666666663,
0.0,
0.0,
0.0,
0.0,
0.0,
0.2,
0.2333333333333333,
0.5]
```

```
[166]: #opstiller en ny liste der fortolker graden af polaritet ud fra tidligere liste
polarity_lst=[]
```

```

for i in polarity:
    if i > 0.0:
        polarity_lst.append('positive')
    elif i < 0.0:
        polarity_lst.append('negative')
    else:
        polarity_lst.append('neutral')
polarity_lst[:10]

```

```

[166]: ['positive',
        'neutral',
        'negative',
        'neutral',
        'neutral',
        'neutral',
        'neutral',
        'neutral',
        'positive',
        'positive',
        'positive']

```

```

[167]: #Opstiller en liste der indeholder sentimentets subjektivitet for hver caption
subjectivity=[]
for i in df.caption:
    subjectivity.append(TextBlob(i).sentiment.subjectivity)
subjectivity[:10]

```

```

[167]: [0.2895833333333333,
        0.0,
        0.6666666666666666,
        0.0,
        0.0,
        0.0,
        0.0,
        0.2625,
        0.35555555555555557,
        1.0]

```

```

[168]: #opstiller en ny liste der fortolker graden af subjektivitet ud fra tidligere
↪liste
subjectivity_lst=[]
for i in subjectivity:
    if i > 0:
        subjectivity_lst.append('Subjektiv')
    else:
        subjectivity_lst.append('Objektiv')
subjectivity_lst[:10]

```

```
[168]: ['Subjektiv',
        'Objektiv',
        'Subjektiv',
        'Objektiv',
        'Objektiv',
        'Objektiv',
        'Objektiv',
        'Subjektiv',
        'Subjektiv',
        'Subjektiv']
```

```
[169]: # Et samlet datasæt udgjort af de tidligere liste, der visuelt giver et
        ↳ overblik sentimentet for hver caption
df1 = pd.DataFrame(data={'Polarity':polarity, 'Result1':
        ↳ polarity_lst, 'Subjectivity':subjectivity, 'Result2':subjectivity_lst,
        ↳ 'Caption':alle})
df1
```

```
[169]:
```

	Polarity	Result1	Subjectivity	Result2 \	Caption
0	0.237500	positive	0.289583	Subjektiv	This is the first step of MANY. KEEP THAT SAME...
1	0.000000	neutral	0.000000	Objektiv	Wearing all Praying cause that's what all I be...
2	-0.416667	negative	0.666667	Subjektiv	Fav distraction: Looking at apartments I can't...
3	0.000000	neutral	0.000000	Objektiv	I know patience is a virtue, but DAMN!! Are we...
4	0.000000	neutral	0.000000	Objektiv	Anxiety meter already broke Recharging with ...
..
245	0.000000	neutral	0.000000	Objektiv	Went through a breakup, wrote a song about it...
246	0.000000	neutral	0.000000	Objektiv	Y'all know how I am...I couldn't help myself ...
247	0.000000	neutral	0.100000	Subjektiv	Big things coming soon!
248	0.112121	positive	0.651515	Subjektiv	New tunes coming soon. Zoom in on my @samsungm...
249	0.225652	positive	0.534133	Subjektiv	Long caption Alert \nSo this is the inevi...

```
[250 rows x 5 columns]
```

```
[170]: mean_pol=sum(polarity)/len(polarity) #Beregner gjennomsnittet for polaritet
print('Gjennomsnitt polaritet:', mean_pol)
```

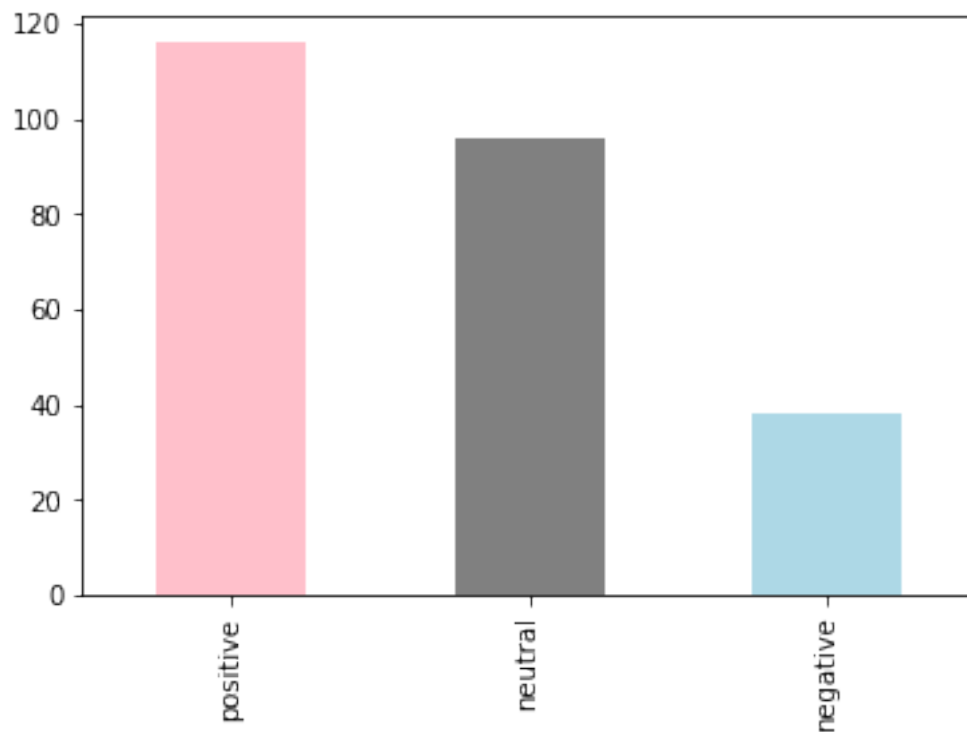
Gjennomsnitt polaritet: 0.09579347057964786

```
[171]: mean_sub=sum(subjectivity)/len(subjectivity) #Beregner gjennomsnittet for
      ↪ subjektivitet
print('Gjennomsnitt subjektivitet:', mean_sub)
```

Gjennomsnitt subjektivitet: 0.344851428753239

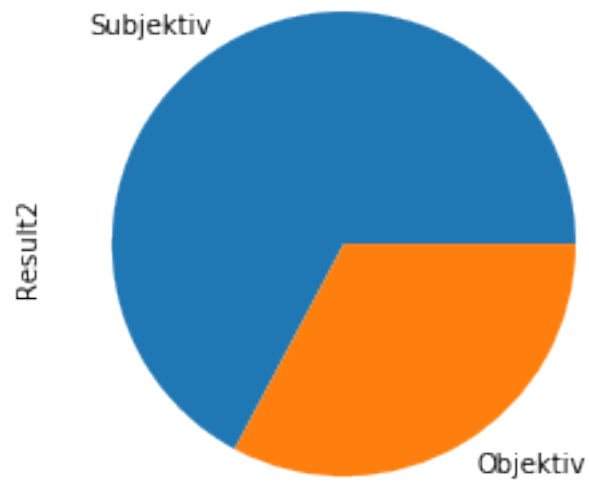
```
[172]: df1['Result1'].value_counts().plot.bar(color=['pink', 'grey', 'lightblue'])
```

```
[172]: <matplotlib.axes._subplots.AxesSubplot at 0x7fd17c45f2d0>
```



```
[173]: df1['Result2'].value_counts().plot.pie()
```

```
[173]: <matplotlib.axes._subplots.AxesSubplot at 0x7fd17c384e50>
```



```
[174]: print('Procent for @lilmiquela subjektivtet:',df1['Result2'].str.  
        ↳count('Subjektiv').sum()/250*100, '%')
```

Procent for @lilmiquela subjektivtet: 67.2 %

```
[175]: print('Procent for @lilmiquela subjektivtet:',round(df1['Result1'].str.  
        ↳count('positive').sum()/250*100), '%')
```

Procent for @lilmiquela subjektivtet: 46.0 %

```
[ ]:
```