In [34]:

```
import re
import nltk
import string
import warnings
import numpy as np
import pandas as pd
import seaborn as sns
import matplotlib.pyplot as plt

pd.set_option("display.max_colwidth", 200)
warnings.filterwarnings("ignore", category=DeprecationWarning)

%matplotlib inline
```

In [35]:

```
train  = pd.read_csv('train.csv')
test = pd.read_csv('test_tweets.csv')
```

In [3]:

```
train.head()
```

Out[3]:

	id	label	tweet
0	1	0	@user when a father is dysfunctional and is so selfish he drags his kids into his dysfunction. #run
1	2	0	@user @user thanks for #lyft credit i can't use cause they don't offer wheelchair vans in pdx. #disapointed #getthanked
2	3	0	bihday your majesty
3	4	0	#model i love u take with u all the time in urð ±!!! ð ð ð ð ð ¦ð ¦ð ¦
4	5	0	factsguide: society now #motivation

In [4]:

```
train[train['label']==0].head(10)
```

Out[4]:

	id	label	tweet
0	1	0	@user when a father is dysfunctional and is so selfish he drags his kids into his dysfunction. #run
1	2	0	@user @user thanks for #lyft credit i can't use cause they don't offer wheelchair vans in pdx. #disapointed #getthanked
2	3	0	bihday your majesty
3	4	0	#model i love u take with u all the time in urð ±!!! ð ð ð ð ð ¦ð ¦ð ¦
4	5	0	factsguide: society now #motivation
5	6	0	[2/2] huge fan fare and big talking before they leave. chaos and pay disputes when they get there. #allshowandnogo
6	7	0	@user camping tomorrow @user @user @user @user @user @user @user dannyâ€¦
7	8	0	the next school year is the year for exams.ð ¯ can't think about that ð #school #exams #hate #imagine #actorslife #revolutionschool #girl
8	9	0	we won!!! love the land!!! #allin #cavs #champions #cleveland #clevelandcavaliers â€¦
9	10	0	@user @user welcome here ! i'm it's so #gr8 !

In [5]:

```
train[train['label']==1].head(10)
```

Out[5]:

	id	label	tweet
13	14	1	@user #cnn calls #michigan middle school 'build the wall' chant '' #tcot
14	15	1	no comment! in #australia #opkillingbay #seashepherd #helpcovedolphins #thecove #helpcovedolphins
17	18	1	retweet if you agree!
23	24	1	@user @user lumpy says i am a . prove it lumpy.
34	35	1	it's unbelievable that in the 21st century we'd need something like this. again. #neverump #xenophobia
56	57	1	@user lets fight against #love #peace
68	69	1	ð ©the white establishment can't have blk folx running around loving themselves and promoting our greatness
77	78	1	@user hey, white people: you can call people 'white' by @user #race #identity #medâ€¦
82	83	1	how the #altright uses & insecurity to lure men into #whitesupremacy
111	112	1	@user i'm not interested in a #linguistics that doesn't address #race & . racism is about #power. #raciolinguistics bringsâ€¦

In [6]:

```
combi = train.append(test,ignore_index=True,sort=False)
```

```
/var/folders/bb/y77124gx5kncstl8shw72dqh0000gn/T/ipykernel_2326/373375
8226.py:1: FutureWarning: The frame.append method is deprecated and wi
ll be removed from pandas in a future version. Use pandas.concat inste
ad.
  combi = train.append(test,ignore_index=True,sort=False)
```

In [7]:

```
def remove_pattern(input_txt,pattern):
    r = re.findall(pattern,input_txt)
    for word in r:
        input_txt = re.sub(word,'',input_txt)
    return input_txt
```

In [8]:

```
combi['tidy_tweet'] = np.vectorize(remove_pattern)(combi['tweet'], "@[\w]*")
combi.head()
```

Out[8]:

	id	label	tweet	tidy_tweet
0	1	0.0	@user when a father is dysfunctional and is so selfish he drags his kids into his dysfunction. #run	when a father is dysfunctional and is so selfish he drags his kids into his dysfunction. #run
1	2	0.0	@user @user thanks for #lyft credit i can't use cause they don't offer wheelchair vans in pdx. #disapointed #getthanked	thanks for #lyft credit i can't use cause they don't offer wheelchair vans in pdx. #disapointed #getthanked
2	3	0.0	bihday your majesty	bihday your majesty
3	4	0.0	#model i love u take with u all the time in urð ±!!! ð ð ð ð ð ¦ð ¦ð ¦	#model i love u take with u all the time in urð ±!!! ð ð ð ð ð ¦ð ¦ð ¦
4	5	0.0	factsguide: society now #motivation	factsguide: society now #motivation

In [9]:

```
combi['tidy_tweet'] = combi['tidy_tweet'].str.replace("[^a-zA-Z#]"," ")
combi.head()
```

```
/var/folders/bb/y77124gx5kncstl8shw72dqh0000gn/T/ipykernel_2326/237169
270.py:1: FutureWarning: The default value of regex will change from T
rue to False in a future version.
  combi['tidy_tweet'] = combi['tidy_tweet'].str.replace("[^a-zA-Z#]","
")
```

Out[9]:

	id	label	tweet	tidy_tweet
0	1	0.0	@user when a father is dysfunctional and is so selfish he drags his kids into his dysfunction. #run	when a father is dysfunctional and is so selfish he drags his kids into his dysfunction #run
1	2	0.0	@user @user thanks for #lyft credit i can't use cause they don't offer wheelchair vans in pdx. #disapointed #getthanked	thanks for #lyft credit i can t use cause they don t offer wheelchair vans in pdx #disapointed #getthanked
2	3	0.0	bihday your majesty	bihday your majesty
3	4	0.0	#model i love u take with u all the time in urð ±!!! ð ð ð ð ð ¦ð ¦ð ¦	#model i love u take with u all the time in ur
4	5	0.0	factsguide: society now #motivation	factsguide society now #motivation

In [10]:

```
combi['tidy_tweet'] = combi['tidy_tweet'].apply(lambda x: ' '.join([w for w in x.spli
combi.head()
```

Out[10]:

	id	label	tweet	tidy_tweet
0	1	0.0	@user when a father is dysfunctional and is so selfish he drags his kids into his dysfunction. #run	when father dysfunctional selfish drags kids into dysfunction #run
1	2	0.0	@user @user thanks for #lyft credit i can't use cause they don't offer wheelchair vans in pdx. #disapointed #getthanked	thanks #lyft credit cause they offer wheelchair vans #disapointed #getthanked
2	3	0.0	bihday your majesty	bihday your majesty
3	4	0.0	#model i love u take with u all the time in urð ±!!! ð ð ð ð ð ¦ð ¦ð ¦	#model love take with time
4	5	0.0	factsguide: society now #motivation	factsguide society #motivation

In [11]:

```
tokenized_tweet = combi['tidy_tweet'].apply(lambda x: x.split())
tokenized_tweet.head()
```

Out[11]:

```
0                [when, father, dysfunctional, selfish, drags, kids, i
nto, dysfunction, #run]
1    [thanks, #lyft, credit, cause, they, offer, wheelchair, vans, #di
sapointed, #getthanked]
2
[bihday, your, majesty]
3                                                          [#model,
love, take, with, time]
4                                                       [factsguid
e, society, #motivation]
Name: tidy_tweet, dtype: object
```

In [12]:

```
from nltk.stem.porter import *
stemmer = PorterStemmer()
#stemming
tokenized_tweet = tokenized_tweet.apply(lambda sentence:[stemmer.stem(word) for word


for word in range(len(tokenized_tweet)):
    tokenized_tweet[word] = ' '.join(tokenized_tweet[word])
combi['tidy_tweet'] = tokenized_tweet
```

In [13]:

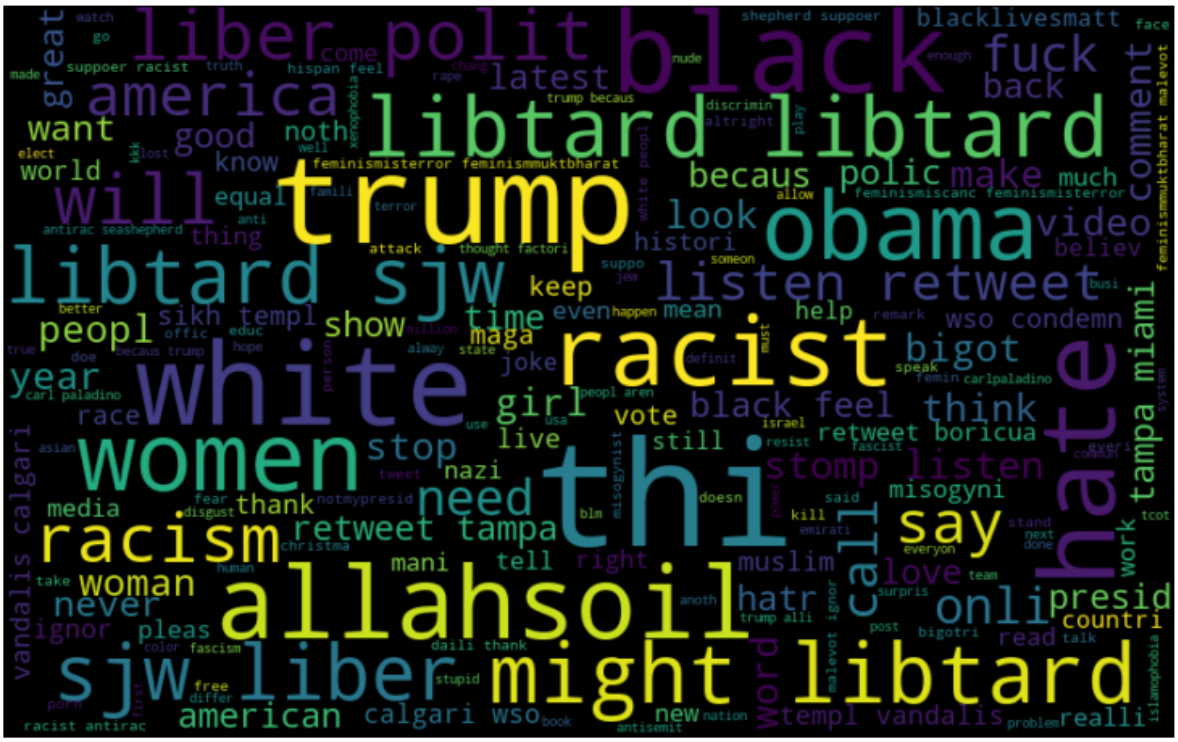
```
all_words = ' '.join([text for text in combi['tidy_tweet']])
from wordcloud import WordCloud
wordcloud = WordCloud(width=800, height=500, random_state=42, max_font_size=100).ger
plt.figure(figsize=(15, 9))
plt.imshow(wordcloud, interpolation="bilinear")
plt.axis('off')
plt.show()
```

In [14]:

```
normal_words =' '.join([text for text in combi['tidy_tweet'][combi['label'] == 0]])
wordcloud = WordCloud(width=800, height=500, random_state=21, max_font_size=110).gen
plt.figure(figsize=(10, 7))
plt.imshow(wordcloud, interpolation="bilinear")
plt.axis('off')
plt.show()
```

In [15]:

```
normal_words =' '.join([text for text in combi['tidy_tweet'][combi['label'] == 1]])
wordcloud = WordCloud(width=800, height=500, random_state=21, max_font_size=100).ger
plt.figure(figsize=(13, 9))
plt.imshow(wordcloud, interpolation="bilinear")
plt.axis('off')
plt.show()
```



In [16]:

```
def hashtag_extract(tweets):
    hashtags = []
    for tweet in tweets:
        ht = re.findall(r"#(\w+)", tweet)
        hashtags.append(ht)
    return hashtags
```

In [17]:

```
HT_positive = hashtag_extract(combi['tidy_tweet'][combi['label'] == 0])
HT_negative = hashtag_extract(combi['tidy_tweet'][combi['label'] == 1])
HT_positive = sum(HT_positive,[])
HT_negative = sum(HT_negative,[])
```

In [18]:

```
HT_positive
```

```
 'not',
 'exist',
 'positivevib',
 'hawaiian',
 'goodnight',
 'badmonday',
 'taylorswift',
 'travelingram',
 'dalat',
 'ripinkylif',
 'photoshop',
 'enoughisenough',
 'dontphotoshopeveryth',
 'wheresallthenaturalphoto',
 'cedarpoint',
 'thank',
 'posit',
 'bookworm',
 'ontothenextnovel',
 'flow'
```

In [19]:

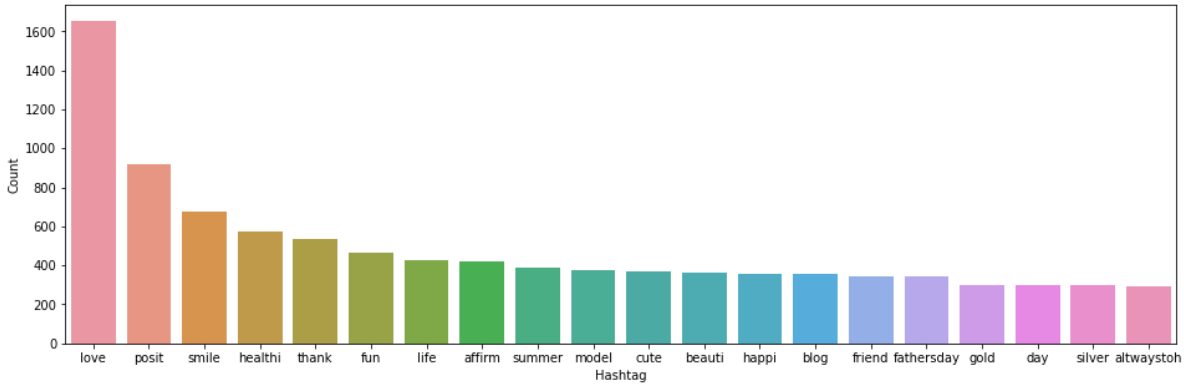
```
freq = nltk.FreqDist(HT_positive)
d = pd.DataFrame({'Hashtag': list(freq.keys()), 'Count': list(freq.values())})
d.head()
```

Out[19]:

	Hashtag	Count
0	run	72
1	lyft	2
2	disapoint	1
3	getthank	2
4	model	375

In [20]:

```
d = d.nlargest(columns="Count", n = 20)
plt.figure(figsize=(16,5))
ax = sns.barplot(data=d, x= "Hashtag", y = "Count")
ax.set(ylabel = 'Count')
plt.show()
```



In [21]:

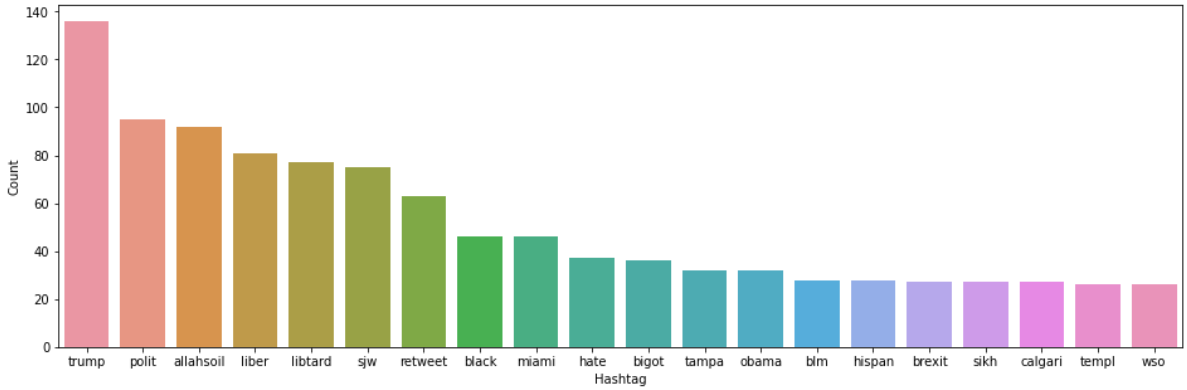
```
freq = nltk.FreqDist(HT_negative)
d = pd.DataFrame({'Hashtag': list(freq.keys()), 'Count': list(freq.values())})
d.head()
```

Out[21]:

	Hashtag	Count
0	cnn	10
1	michigan	2
2	tcot	14
3	australia	6
4	opkillingbay	5

In [22]:

```
d = d.nlargest(columns="Count", n = 20)
plt.figure(figsize=(16,5))
ax = sns.barplot(data=d, x= "Hashtag", y = "Count")
ax.set(ylabel = 'Count')
plt.show()
```



In [23]:

```
rom sklearn.feature_extraction.text import CountVectorizer
ow_vectorizer = CountVectorizer(max_df=0.90, min_df=2, max_features=1000, stop_words
ow = bow_vectorizer.fit_transform(combi['tidy_tweet'])
ow.shape
```

Out[23]:

```
(49159, 1000)
```

In [24]:

```
bow
```

Out[24]:

```
<49159x1000 sparse matrix of type '<class 'numpy.int64'>'
        with 191502 stored elements in Compressed Sparse Row format>
```

In [25]:

```
#bow[0].toarray()
```

In [26]:

```
from sklearn.model_selection import train_test_split
trainbow = bow[:31962,:]
test_bow = bow[31962:,:]
Xtrain,x_test,Ytrain,y_test= train_test_split(trainbow,train['label'],random_state=4
```

In [27]:

```
from sklearn.linear_model import LogisticRegression
from sklearn.metrics import f1_score,accuracy_score
```

In [28]:

```
 model = LogisticRegression()

model.fit(Xtrain, Ytrain)
```

Out[28]:

```
LogisticRegression()
```

In [29]:

```
pred = model.predict(x_test)
f1_score(y_test,pred)
```

Out[29]:

```
0.5047169811320754
```

In [30]:

```
accuracy_score(y_test,pred)
```

Out[30]:

```
0.9474408709798523
```

In [31]:

```
pred_prob = model.predict_proba(x_test)
pred = pred_prob[:,1] >= 0.3
pred = pred.astype(np.int)

f1_score(y_test,pred)
```

Out[31]:

```
0.5484818805093045
```

In [32]:

```
accuracy_score(y_test,pred)
```

Out[32]:

```
0.9423100988612189
```

In [33]:

```
pred_prob[0][1]>=0.3
```

Out[33]:

```
False
```

In [ ]:

```

```