

Q2 Text Mining --> Amazon

- 1) Extract reviews of any product from ecommerce website like amazon
- 2) Perform emotion mining

1. Import libs

In [44]:

```
import pandas as pd
import numpy as np
import requests
from bs4 import BeautifulSoup as bs
from selenium import webdriver
from flask import Response
import re
import nltk
from nltk.stem import WordNetLemmatizer
from nltk.corpus import stopwords
from nltk.tokenize import word_tokenize
from nltk.tokenize import sent_tokenize
from sklearn.feature_extraction.text import TfidfVectorizer
from wordcloud import WordCloud
import matplotlib.pyplot as plt
```

2. Import Data

Connecting to Amazon.in to download reviews

In [15]:

```
iphone_review=[]
```

In [16]:

```
for i in range(1,100):
    op= []
    url = "https://www.amazon.in/Apple-iPhone-14-128GB-Blue/product-reviews/B0BDK62PDX/ref=
    response = requests.get(url)
    soup = bs(response.content,"html.parser")
    reviews = soup.findAll("span",attrs = {"class","a-size-base review-text review-text-con
    for i in range(len(reviews)):
        op.append(reviews[i].text)
    iphone_review = iphone_review+op
```

Downloading review from Amazon.in

3. Data Preprocessing

remove both the leading and the trailing characters

Joining the list into one string/text

In [22]:

```
txt_upd = ' '.join(iphone_review)
txt_upd
```

ore. If you are a developer you might know the app store policy for application publish and their security posture towards handling payment apps etc. They publish best application and the way the apps, its modules switches is just flawless. Due to the so called strict policy of privacy program with Apple products, iPhone can pop up a box in your screen if any apps wants to track your activity with other applications. +1 respect for apple to show such concern with user. Apple phones are same ? like 13 vs 14 The reason 14 is same as 13 may be business practice followed from 11th edition. But please note that iPhone making changes internally like physical components (small change in battery performance, lens in 14 edition), additional features but that may not satisfy you to buy 14 edition. If you do not find "the very few changes" interesting in 14, go for 13. Because physical and even cameras are not visually very much different in 13 vs 14 comparison. Also you get 13 for cheap :) 14 is just added features but that is not for daily use, the features real time use is rare case. But the best part is there is no extra cost between 13 vs 14 prices (except extra 10k slash) Concern you must have. Apple is not giving chargers and their silicon cases are very expensive (but very good in touch and scratch friendly in back due to silk layer). This is an issue for sure. You have to buy charger not coming with USB C and claiming that you help carbon footprint. understand

In [25]:

```
txt_upd = re.sub("[^A-Za-z" "]+", "", txt_upd).lower() #remove special character
txt_upd = re.sub("[0-9" "]+", "", txt_upd).lower() #remove numbers
txt_upd = re.sub(r'^https?:\/\/\.*[\r\n]*', '', txt_upd).lower() #remove hyperlink
txt_upd
```

ou can also enjoy the best applications available in app store if you are a developer you might know the app store policy for application publish and their security posture towards handling payment apps etc they publish best application and the way the apps its modules switches is just flawless due to the so called strict policy of privacy program with apple products iPhone can pop up a box in your screen if any apps wants to track your activity with other applications respect for apple to show such concern with user apple phones are same like vs the reason is same as may be business practice followed from th edition but please note that iPhone making changes internally like physical components small change in battery performance lens in edition additional features but that may not satisfy you to buy edition if you do not find the very few changes interesting in go for because physical and even cameras are not visually very much different in vs comparison also you get for cheap is just added features but that is not for daily use the features real time use is rare case but the best part is there is no extra cost between vs prices except extra k slash concern you must have apple is not giving chargers and their silicon cases are very expensive but very good in touch and scratch friendly in back due to silk layer this is an issue for sure you have to buy charger not coming with usb c and

In [32]:

```
text_tokens = word_tokenize(txt_upd)
```

In [33]:

```
tokens_without_sw = [word for word in text_tokens if not word in stopwords.words()]
```

In [35]:

```
tf = TfidfVectorizer()
```

In [36]:

```
text_tf = tf.fit_transform(tokens_without_sw)
```

Create the DataFrame

In [38]:

```
feature_names = tf.get_feature_names_out ()
dense = text_tf.todense()
denselist = dense.tolist()
df =pd.DataFrame(denselist, columns=feature_names)
```

In [39]:

```
df
```

Out[39]:

	activity	added	additional	ads	amazon	android	app	apple	application	applications	.
0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	.
1	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	.
2	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	.
3	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	.
4	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	.
...
255	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	.
256	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	.
257	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	.
258	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	.
259	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	.

260 rows × 181 columns

In [40]:

```
word_list = ' '.join(df)
```

In [48]:

```
wordcloud = WordCloud(background_color='black',width=1800,height=1400).generate(word_list)
```


In [51]:

```
positive_words = positive_words[35:]
positive_words
'adequate',
'adjustable',
'admirable',
'admirably',
'admiration',
'admire',
'admirer',
'admiring',
'admiringly',
'adorable',
'adore',
'adored',
'adorer',
'adoring',
'adoringly',
'adroit',
'adroitly',
'adulate',
'adulation',
'adulatory',
```

In [52]:

```
with open('negative-words.txt') as nw:
    negative_words = nw.read().split("\n")
```

In [53]:

```
negative_words = negative_words[35:]
negative_words
'abominable',
'abominably',
'abominate',
'abomination',
'abort',
'aborted',
'aborts',
'abrade',
'abrasive',
'abrupt',
'abruptly',
'abscond',
'absence',
'absent-minded',
'absentee',
'absurd',
'absurdity',
'absurdly',
'absurdness',
'abuse'
```

In [54]:

```
txt_pos_in_pw = ' '.join([word for word in df if word in positive_words])
```


In [55]:

```
wordcloud_pos = WordCloud(background_color='black',width=1800,height=1400).generate(txt_pos)
```

In [57]:

```
plt.figure(figsize=(7,15))  
plt.imshow(wordcloud_pos)  
plt.axis('off')  
plt.show()
```



In [59]:

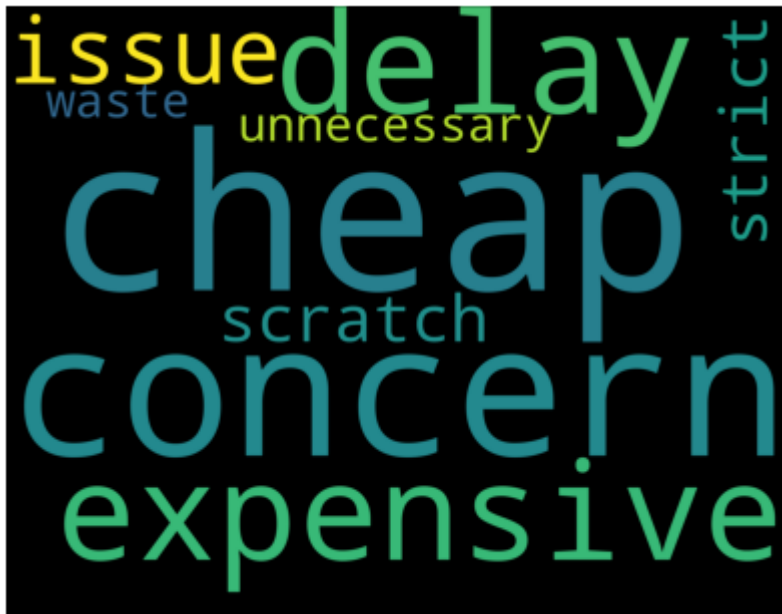
```
txt_neg_in_nw = ' '.join([word for word in df if word in negative_words])
```

In [62]:

```
wordcloud_neg = WordCloud( background_color='black',width=1800,height=1400).generate(txt_ne
```

In [64]:

```
plt.figure(figsize=(7,15))  
plt.imshow(wordcloud_neg)  
plt.axis('off')  
plt.show()
```



END

In []: