DLNLP Project - Amazon Food Reviews Dataset

Group 14 Anisha Siwas 025007

Sarthak Jain 025029

Tanya Goel 025034

Chirayu Jain 025049

In [1]:
```
import pandas as pd
import numpy as np
import re
import nltk
from nltk.tokenize import word_tokenize, sent_tokenize
from nltk.corpus import stopwords
from nltk.stem.porter import PorterStemmer
from nltk.stem import WordNetLemmatizer
import matplotlib.pyplot as plt
import seaborn as sns
from string import punctuation
from sklearn import svm


from nltk import ngrams
from itertools import chain
from wordcloud import WordCloud
```

In [2]:
```
#read the data
df = pd.read_csv("C:/Users/chira/Desktop/food_subset.csv")
```

In [3]:
```
df.head()
```

Out[3]:

	Id	ProductId	UserId	ProfileName	HelpfulnessNumerator	HelpfulnessDenominator	Score	Time	Summary	
0	1	B001EO5QW8	AOVROBZ8BNTP7	S. Potter	19	19	4	1163376000	Best of the Instant Oatmeals	McCa In Oat is gr you
1	2	B001EO5QW8	A3PMM0NFVEJGK9	Megan "Bad at Nicknames"	13	13	4	1166313600	Good Instant	This in oat fror bes
2	3	B003ZFRKGO	A2VOZX7YBT0D6D	Johnnycakes "Johnnycakes"	15	15	5	1325635200	Forget Molecular Gastronomy - this stuff rocke...	I I pro title Mole Gas
3	4	B000ITVLE2	A3NID9D9WMIV01	Louie Arrighi "Lou da Joo"	17	19	5	1260057600	tastes very fresh	expir date mo fror d
4	5	B001UJEN6C	A1XM65S80UQ2MD	Joseph Kagan	13	13	5	1276214400	Great Natural Energy	This fant pro w w

In [4]:
```
df['Text'].values[0]
```

Out[4]:
```
"McCann's Instant Oatmeal is great if you must have your oatmeal but can only scrape together two or three minu
tes to prepare it. There is no escaping the fact, however, that even the best instant oatmeal is nowhere near a
s good as even a store brand of oatmeal requiring stovetop preparation.  Still, the McCann's is as good as it g
ets for instant oatmeal. It's even better than the organic, all-natural brands I have tried.  All the varieties
in the McCann's variety pack taste good.  It can be prepared in the microwave or by adding boiling water so it
is convenient in the extreme when time is an issue.<br /><br />McCann's use of actual cane sugar instead of hig
h fructose corn syrup helped me decide to buy this product.  Real sugar tastes better and is not as harmful as
the other stuff. One thing I do not like, though, is McCann's use of thickeners.  Oats plus water plus heat sho
uld make a creamy, tasty oatmeal without the need for guar gum. But this is a convenience product.  Maybe the g
uar gum is why, after sitting in the bowl a while, the instant McCann's becomes too thick and gluey."
```

In [5]:
```
df.shape
```

Out[5]:
```
(17244, 10)
```

In [6]:
```
df= df.head(17244)
print(df.shape)
```

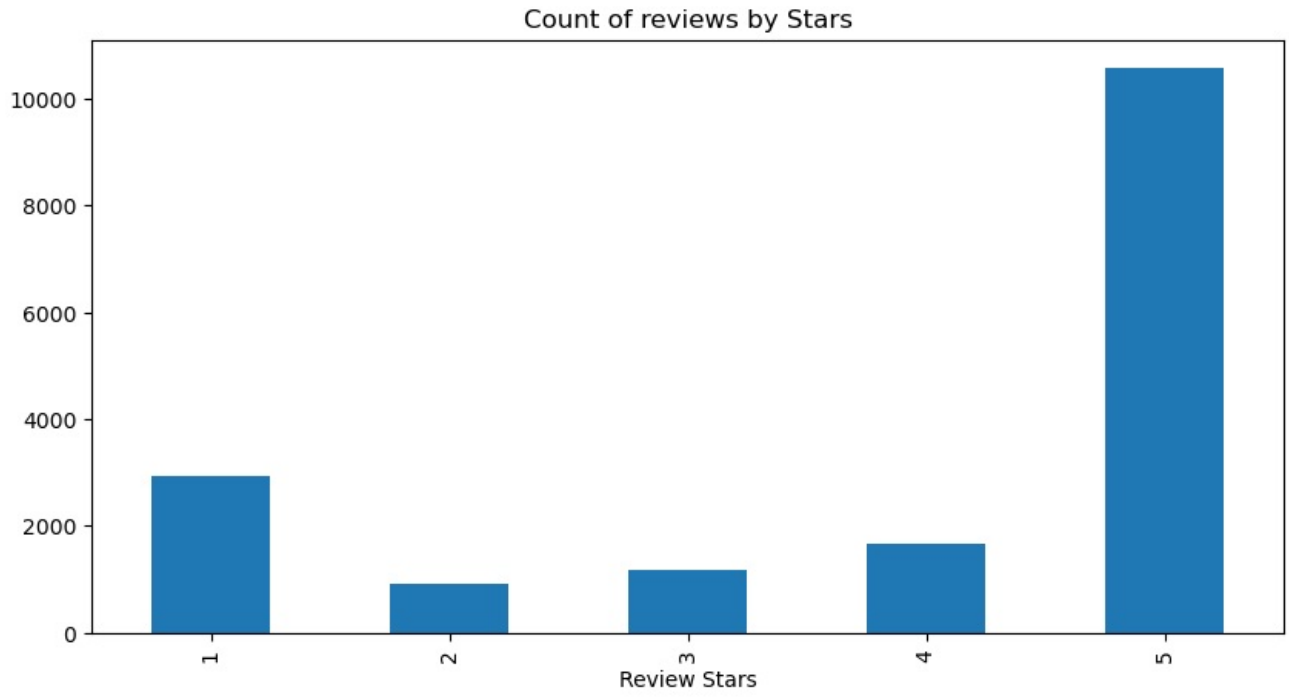
```
(17244, 10)
```

In [7]:
```
df['Score'].value_counts()
```

Out[7]:
```
5    10562
1     2929
4     1664
3     1178
2      911
Name: Score, dtype: int64
```

In [8]:
```
ax= df['Score'].value_counts().sort_index().plot(kind='bar',title ='Count of reviews by Stars',figsize =(10,5))
ax.set_xlabel('Review Stars')
```

Out[8]:
```
Text(0.5, 0, 'Review Stars')
```



In [ ]:
```
```

Most of the review are 5 stars but less 1 stars so we can say that most customers are tend to positive review

In [9]:
```
text = df['Text'][45]
print(text)
```

```
I received this mix along with a waffle maker as a gift.  It's so good that I keep buying this same brand mix w
hen I run out.<br /><br />I cut the Farmhouse Waffles recipe in half (so that's 1 cup Farmhouse Pancake and Waf
fle Mix; 1 egg; 5/8 cup water; 2 tablespoon melted butter) and it's more than enough batter for 2 waffles in a
full-size Belgian waffle maker.<br /><br />Easy to make and very tasty.
```

In [10]:
```
Amazon_Review =df
```

In [11]:
```
Amazon_Review.dtypes
```

Out[11]:
```
Id                       int64
ProductId               object
UserId                  object
ProfileName             object
HelpfulnessNumerator     int64
HelpfulnessDenominator   int64
Score                    int64
Time                     int64
Summary                 object
Text                    object
dtype: object
```

In [12]:
```
Amazon_Review.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 17244 entries, 0 to 17243
Data columns (total 10 columns):
 #   Column                  Non-Null Count  Dtype
---  ------                  --------------  -----
 0   Id                      17244 non-null  int64
 1   ProductId               17244 non-null  object
 2   UserId                  17244 non-null  object
 3   ProfileName             17243 non-null  object
 4   HelpfulnessNumerator    17244 non-null  int64
 5   HelpfulnessDenominator  17244 non-null  int64
 6   Score                   17244 non-null  int64
 7   Time                    17244 non-null  int64
 8   Summary                 17244 non-null  object
 9   Text                    17244 non-null  object
dtypes: int64(5), object(5)
memory usage: 1.3+ MB
```

In [13]:
```
##Removing the Duplicates if any
Amazon_Review.duplicated().sum()
Amazon_Review.drop_duplicates(inplace=True)
```

In [14]:
```
#Check for the null values in each column
Amazon_Review.isnull().sum()
```

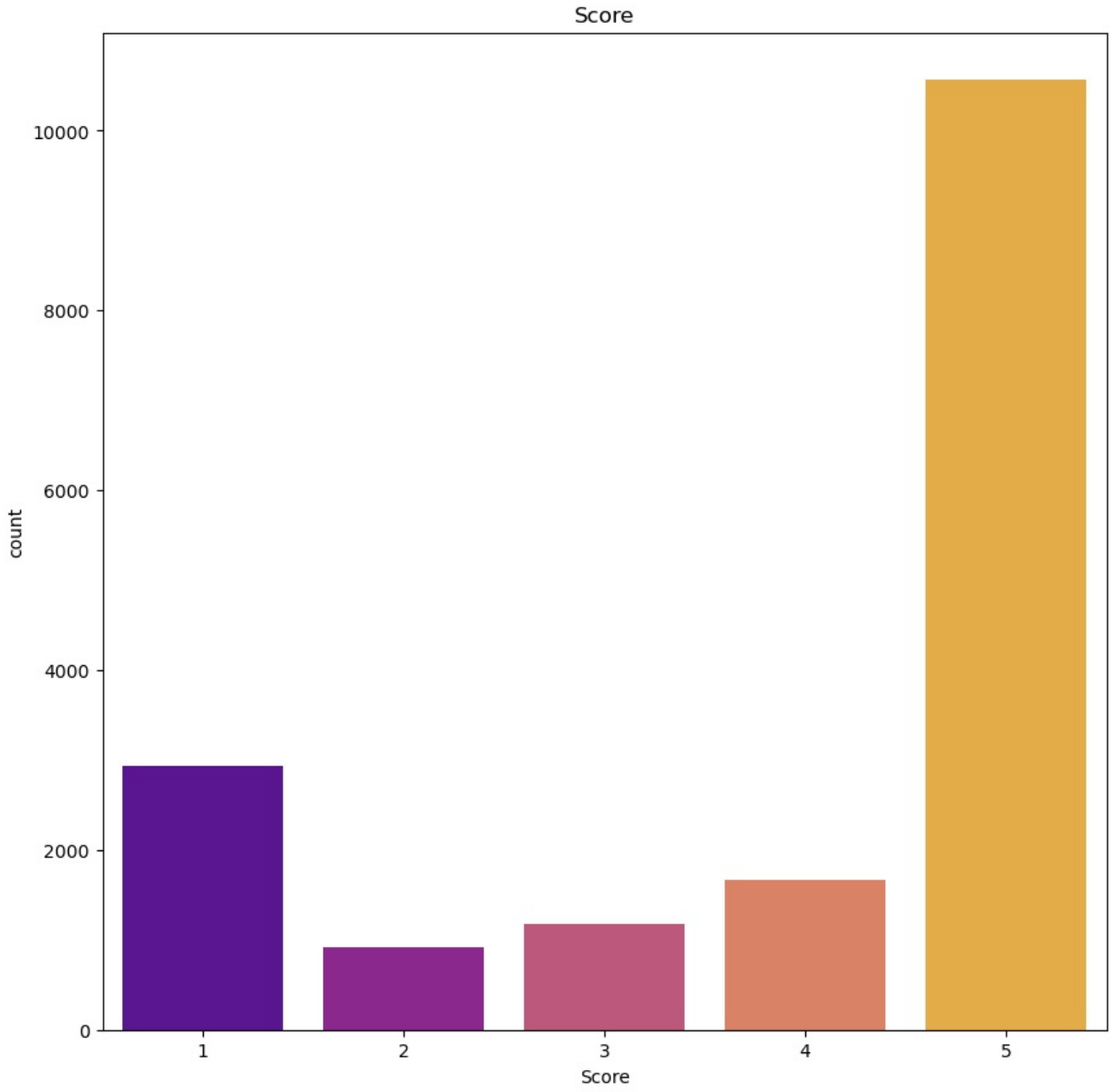
Out[14]:
```
Id                      0
ProductId               0
UserId                  0
ProfileName             1
HelpfulnessNumerator    0
HelpfulnessDenominator  0
Score                   0
Time                    0
Summary                 0
Text                    0
dtype: int64
```

In [15]:
```
##Remove the NaN values from the dataset
Amazon_Review.isnull().sum()
Amazon_Review.dropna(how='any',inplace=True)
```

In [16]:
```
import seaborn as sns
sns.countplot(Amazon_Review['Score'], palette="plasma")
fig = plt.gcf()
fig.set_size_inches(10,10)
plt.title('Score')
```

```
C:\Users\chira\anaconda3\lib\site-packages\seaborn\_decorators.py:36: FutureWarning: Pass the following variable as a keyword arg: x. From version 0.12, the only valid positional argument will be `data`, and passing other arguments without an explicit keyword will result in an error or misinterpretation.
  warnings.warn(
```

Out[16]:
```
Text(0.5, 1.0, 'Score')
```



In [17]:
```
Amazon_Review = pd.DataFrame(Amazon_Review , columns=['UserId','Score', 'Text'])
print(Amazon_Review.shape)
Amazon_Review.head()
```

```
(17243, 3)
```

Out[17]:

	UserId	Score	Text
0	AOVROBZ8BNTP7	4	McCann's Instant Oatmeal is great if you must ...
1	A3PMM0NFVEJGK9	4	This is a good instant oatmeal from the best o...
2	A2VOZX7YBT0D6D	5	I know the product title says Molecular Gastro...
3	A3NID9D9WMIV01	5	The expiration date is 21 months from the day ...
4	A1XM65S80UQ2MD	5	This is a fantastic product, and I wish it was...

In [23]:
```
import re

import nltk
nltk.download('punkt')

from nltk.tokenize import word_tokenize

nltk.download('stopwords')
from nltk.corpus import stopwords
stop_words = set(stopwords.words("english"))

#stop_words.extend(['crypto','even','early'])

import nltk
nltk.download('wordnet')
nltk.download('omw-1.4')
from nltk.stem import WordNetLemmatizer
```

```
lemmatizer = WordNetLemmatizer()
```

```
[nltk_data] Downloading package punkt to
[nltk_data]     C:\Users\chira\AppData\Roaming\nltk_data...
[nltk_data]   Package punkt is already up-to-date!
[nltk_data] Downloading package stopwords to
[nltk_data]     C:\Users\chira\AppData\Roaming\nltk_data...
[nltk_data]   Package stopwords is already up-to-date!
[nltk_data] Downloading package wordnet to
[nltk_data]     C:\Users\chira\AppData\Roaming\nltk_data...
[nltk_data]   Package wordnet is already up-to-date!
[nltk_data] Downloading package omw-1.4 to
[nltk_data]     C:\Users\chira\AppData\Roaming\nltk_data...
[nltk_data]   Package omw-1.4 is already up-to-date!
```

In [24]:
```
!pip install demoji
```

```
Requirement already satisfied: demoji in c:\users\chira\anaconda3\lib\site-packages (1.1.0)
```

In [25]:
```
import demoji
def handle_emoji(string):
    #x = string.to_string(header=False, index=False)
    emojis = demoji.findall(string)
    #print(emojis)
    for emoji in emojis:
        string = string.replace(emoji, " " + emojis[emoji].split(":")[0])

    return string
```

In [21]:
```
def text_cleaner(review):
  # removing the not required texts
  cleaned_review = re.sub(re.compile('<.*?>'), '', review) #removing HTML tags
  cleaned_review = re.sub('[^A-Za-z]+', ' ', cleaned_review) #taking only words
  cleaned_review = handle_emoji(cleaned_review)
  cleaned_review = re.sub(r"http\S+", "", cleaned_review)
  cleaned_review = cleaned_review.lower()

  tokens = nltk.word_tokenize(cleaned_review)

  filtered_review = [word for word in tokens if word not in stop_words] # removing stop words

  lemm_review = [lemmatizer.lemmatize(word) for word in filtered_review]
  review = " ".join(lemm_review)
  return(review)
```

In [26]:
```
cleanText=[]

for t in Amazon_Review['Text']:
 cleanText.append(text_cleaner(t))

Amazon_Review["cleanText"] = cleanText
Amazon_Review.head()
```

Out[26]:

	UserId	Score	Text	cleanText
0	AOVROBZ8BNTP7	4	McCann's Instant Oatmeal is great if you must ...	mccann instant oatmeal great must oatmeal scra...
1	A3PMM0NFVEJGK9	4	This is a good instant oatmeal from the best o...	good instant oatmeal best oatmeal brand us can...
2	A2VOZX7YBT0D6D	5	I know the product title says Molecular Gastro...	know product title say molecular gastronomy le...
3	A3NID9D9WMIV01	5	The expiration date is 21 months from the day ...	expiration date month day bought product tuna ...
4	A1XM65S80UQ2MD	5	This is a fantastic product, and I wish it was...	fantastic product wish readily available store...

In [27]:
```
#WORD CLOUD
```

In [28]:
```
# importing all necessary modules
from wordcloud import WordCloud, STOPWORDS
import matplotlib.pyplot as plt
```

In [29]:
```
wordcloud = WordCloud(
        background_color = 'white',
        max_words = 200,
        max_font_size = 40,
        scale = 3,
        random_state = 42,
        stopwords= stop_words
    ).generate(str(cleanText))

# plot the WordCloud image
plt.figure(figsize = (8, 8), facecolor = None)
plt.imshow(wordcloud)
plt.axis("off")
plt.tight_layout(pad = 0)

plt.show()
```

In [30]:
```
#After Word Cloud
new_stopwords = ['also','get','one','use','since','got','even','u','say','k','still']
```

In [31]:
```
stpwrd = nltk.corpus.stopwords.words('english')
stpwrd.extend(new_stopwords)
```

In [ ]:
```
new clean review
```

In [32]:
```
wordcloud = WordCloud(
        background_color = 'white',
        max_words = 200,
        max_font_size = 40,
        scale = 3,
        random_state = 42,
        stopwords= stpwrd
    ).generate(str(cleanText))

# plot the WordCloud image
plt.figure(figsize = (8, 8), facecolor = None)
plt.imshow(wordcloud)
plt.axis("off")
plt.tight_layout(pad = 0)

plt.show()
```

In [38]:
```
cleanText
```

Out[38]:
```
['mccann instant oatmeal great must oatmeal scrape together two three minute prepare escaping fact however even best instant oatmeal nowhere near good even store brand oatmeal requiring stovetop preparation still mccann good get instant oatmeal even better organic natural brand tried variety mccann variety pack taste good prepared microwave adding boiling water convenient extreme time issue mccann use actual cane sugar instead high fructose corn syrup helped decide buy product real sugar taste better harmful stuff one thing like though mccann use thickener oat plus water plus heat make creamy tasty oatmeal without need guar gum convenience product maybe guar
```

```
 'bought case coconut milk based lack bpa packaging happy product smell strange quite watery gave wife upset st
omach two occasion tried can came sri lanka dented',
 'dairy free family use coconut milk substitute pretty familiar coconut milk purchased several can native fores
t retail store liked use coffee creamer however stuff delivered amazon completely different can bought store mi
lk thin mind incredibly oily shook can thoroughly opened see oily floating top milk separate fridge coconut oil
forming thick layer top odd flavor milk bummed go can order thai kitchen',
 'reading every review checked see country origin listed description thailand tried one replacement case well h
orribly greasy product sri lanka amazon immediately reimbursed used product amazon can must thailand product dr
eamy really wanted decided call company description amazon false advertising woman spoke absolute brat borderli
ne rude excuse excuse maybe tired fielding complaint isnt job surprised great conversation learned much small c
ompany like like nutiva complained leaking coconut oil devleoped great relationship one representative sent rep
lacement even though told amazon already reembursed wanted let know problem biggest excuse among several side s
tepping maneuver sample sent match product speaks volume lack quality control company make wonder can amazon po
or consumer suffer edward son faulty business product sri lankan used another month seriously small company poo
r customer service doomed fail matter great product lousy product like unwilling try thing stand behind product
',
 'ordering native forest coconut milk amazon quite frequently delectable coconut milk could find anywhere five
star review spot loved sudden started receiving completely different product label seemed like watered coconut
milk coconut oil added result liquid oil solid translucent like coconut oil throughout matter much shook heated
unopened hot water shaking disgusting unusable oily slop refrigerated open oil would rise top form solid top la
yer never seen brand coconut milk can say product sri lanka spite amazon listing stating thailand suspect reaso
n poor quality would bet can label stating thailand real native forest coconut milk sri lanka can sort evil dop
pelg auml nger update due new review decided give product try look like amazon native forest addressed issue in
ferior product sri lanka recieved several pack thailand super creamy delicious thank goodness love product also
wanted thank amazon always refunded money complained sri lankan slop',
 'use canned clam longer live northeast use linguine clam sauce found crown prince chopped clam like minced cho
pped also packed water rather clam juice add bottled clam juice make better overall adequate returned snow chop
ped clam packed clam juice',
 'looking supplement breastmilk horrible time finding something worked tried least different formula including
new alimentum formula son could tolerate physically still tasted good buy formula cheaper walmart',
 'reluctant buy product bad review thought maybe good review put family team concocted product day co worker ga
ve chocolate brownie made erythritol enjoyed waited weird aftertaste waited laxative effect neither grocery sto
re today picked sugar free oreo saw bottle sugar free hershey chocolate syrup instantly thought back review low
carber use stevia soy milk hemp milk tea coffee think stevia terrible chocolate chocolate product usually made
maltitol sucralose mix enjoy product like would buy explosive diarrhea upset stomach related episode people mal
titol maybe consume lot maybe special tried recently chocolate brownie thought heck mixed three teaspoon syrup
soy milk absolutely thought great delicious naysayer exaggerating flavor sweet expected okay since commercial p
roduct way sweet like sugar free chocolate market thought genius sweet enough go low carb everything start tast
e sweet obnoxious awhile notice slight bitter aftertaste attributable natural chocolate bitterness fleeting unp
leasant mind lingering taste mouth slightly sweet chocolate chemical taste aftertaste taste natural though orga
nic type buy ready side five star raters wanted sure took teaspoon chocolate syrup ate straight sudden one star
raters right sour people right taste terrible bottle say lowcarber diabetic looking chocolate treat form drink
like chocolate milk example though milk carbs would want drink milk add unsweetened soy milk hemp milk mean get
great whatever tempted eat pour throat otherwise think gone bad throw treasure close world sugar free chocolate
syrup forever sure hershey trying make money increasingly hard niche market cater least know need trying make h
appy thank hershey give product chance come organic carb free non diarrhea inducing syrup pleasant aftertaste m
outh feel compromise deal hope try see value product like',
 'tried bottle new improved strong chemical sour taste little chocolate flavor bad bottle throw away',
 ...]
```

In [44]:
```
 example = cleanText[15]
print(example)
```

```
u made version cadbury chocolate distinct taste uk version chocolate eating since childhood like new taste frie
nd india buy chocolate either made uk india otherwise disappointed
```

In [46]:
```
tokens = nltk.word_tokenize(example)
tokens[:12]
```

Out[46]:
```
['u',
 'made',
 'version',
 'cadbury',
 'chocolate',
 'distinct',
 'taste',
 'uk',
 'version',
 'chocolate',
 'eating',
 'since']
```

In [60]:
```
nltk.download('averaged_perceptron_tagger')
 nltk.pos_tag(tokens)
```

```
[nltk_data] Downloading package averaged_perceptron_tagger to
[nltk_data]     C:\Users\chira\AppData\Roaming\nltk_data...
[nltk_data]   Unzipping taggers\averaged_perceptron_tagger.zip.
```

Out[60]:
```
[('u', 'JJ'),
 ('made', 'VBN'),
 ('version', 'NN'),
 ('cadbury', 'NN'),
 ('chocolate', 'NN'),
 ('distinct', 'JJ'),
 ('taste', 'NN'),
 ('uk', 'JJ'),
 ('version', 'NN'),
 ('chocolate', 'NN'),
 ('eating', 'VBG'),

 ('since', 'IN'),
 ('childhood', 'NN'),
 ('like', 'IN'),
 ('new', 'JJ'),
 ('taste', 'NN'),
 ('friend', 'NN'),
 ('india', 'NN'),
 ('buy', 'VB'),
 ('chocolate', 'NN'),
 ('either', 'CC'),
 ('made', 'VBD'),
 ('uk', 'JJ'),
 ('india', 'JJ'),
 ('otherwise', 'RB'),
 ('disappointed', 'JJ')]
```

In [57]:
```
from nltk.sentiment import SentimentIntensityAnalyzer
from tqdm.notebook import tqdm
import nltk
nltk.download('vader_lexicon')
```

```
[nltk_data] Downloading package vader_lexicon to
[nltk_data]     C:\Users\chira\AppData\Roaming\nltk_data...
```

Out[57]:
```
True
```

In [58]:
```
sia = SentimentIntensityAnalyzer()
```

In [61]:
```
sia.polarity_scores('I am so happy!')
```

Out[61]:
```
{'neg': 0.0, 'neu': 0.318, 'pos': 0.682, 'compound': 0.6468}
```

In [62]:
```
sia.polarity_scores('This is the worst thing ever.')
```

Out[62]:
```
{'neg': 0.451, 'neu': 0.549, 'pos': 0.0, 'compound': -0.6249}
```

In [64]:
```
analyzer = SentimentIntensityAnalyzer()

Amazon_Review['compound'] = [analyzer.polarity_scores(x)['compound'] for x in
Amazon_Review['cleanText']]

Amazon_Review['neg'] = [analyzer.polarity_scores(x)['neg'] for x in
Amazon_Review['cleanText']]

Amazon_Review['neu'] = [analyzer.polarity_scores(x)['neu'] for x in
Amazon_Review['cleanText']]

Amazon_Review['pos'] = [analyzer.polarity_scores(x)['pos'] for x in
Amazon_Review['cleanText']]
```

In [70]:
```
Amazon_Review.head()
```

Out[70]:

	UserId	Score	Text	cleanText	compound	neg	neu	pos
0	AOVROBZ8BNTP7	4	McCann's Instant Oatmeal is great if you must ...	mccann instant oatmeal great must oatmeal scra...	0.9323	0.057	0.767	0.176
1	A3PMM0NFVEJGK9	4	This is a good instant oatmeal from the best o...	good instant oatmeal best oatmeal brand us can...	0.9781	0.032	0.557	0.411
2	A2VOZX7YBT0D6D	5	I know the product title says Molecular Gastro...	know product title say molecular gastronomy le...	0.9859	0.097	0.678	0.225
3	A3NID9D9WMIV01	5	The expiration date is 21 months from the day ...	expiration date month day bought product tuna ...	0.9062	0.000	0.725	0.275
4	A1XM65S80UQ2MD	5	This is a fantastic product, and I wish it was...	fantastic product wish readily available store...	0.9565	0.066	0.661	0.273

In [71]:
```
# Run the polarity score on the entire dataset
res = {}
for i, row in tqdm(df.iterrows(), total=len(df)):
    text = row['Text']
    myid = row['Id']
    res[myid] = sia.polarity_scores(text)
```

```
  0%|          | 0/17243 [00:00<?, ?it/s]
```

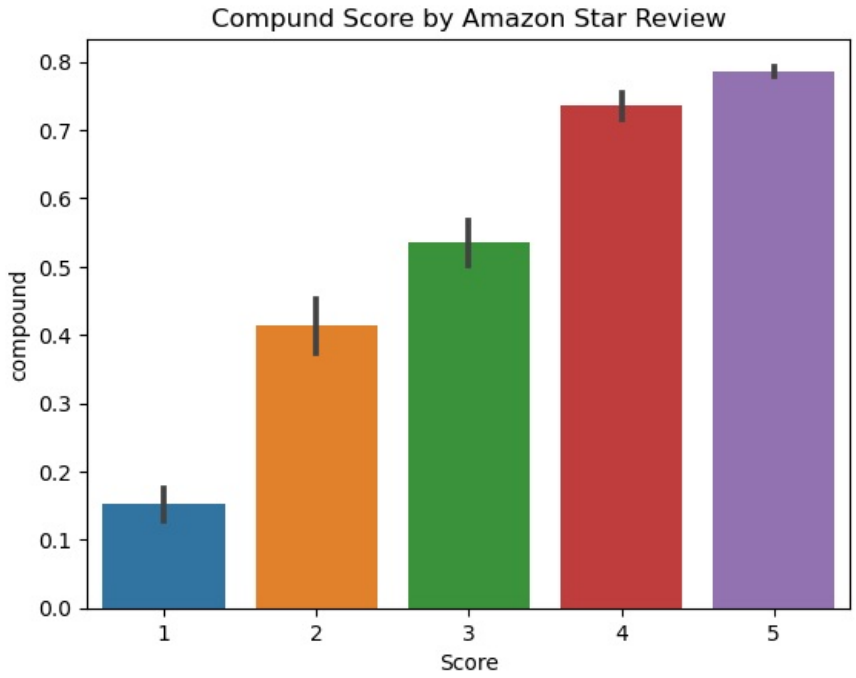
In [72]:
```
vaders = pd.DataFrame(res).T
vaders = vaders.reset_index().rename(columns={'index': 'Id'})
vaders = vaders.merge(df, how='left')
```

In [73]:
```
# Now we have sentiment score and metadata
vaders.head()
```

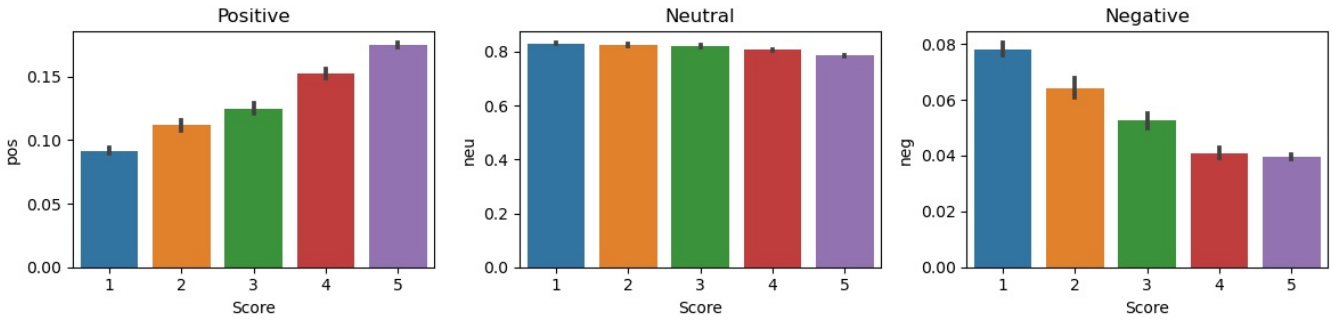
Out[73]:

	Id	neg	neu	pos	compound	ProductId	UserId	ProfileName	HelpfulnessNumerator	HelpfulnessDenominator	Score
0	1	0.069	0.839	0.092	0.7103	B001EO5QW8	AOVROBZ8BNTP7	S. Potter	19	19	4
1	2	0.024	0.720	0.256	0.9779	B001EO5QW8	A3PMM0NFVEJGK9	Megan "Bad at Nicknames"	13	13	4
2	3	0.040	0.794	0.165	0.9957	B003ZFRKGO	A2VOZX7YBT0D6D	Johnnycakes "Johnnycakes"	15	15	5
3	4	0.070	0.885	0.045	-0.4721	B000ITVLE2	A3NID9D9WMIV01	Louie Arrighi "Lou da Joo"	17	19	5
4	5	0.035	0.801	0.163	0.9676	B001UJEN6C	A1XM65S80UQ2MD	Joseph Kagan	13	13	5

In [74]:
```
ax = sns.barplot(data=vaders, x='Score', y='compound')
ax.set_title('Compund Score by Amazon Star Review')
plt.show()
```



In [75]:
```
fig, axs = plt.subplots(1, 3, figsize=(12, 3))
sns.barplot(data=vaders, x='Score', y='pos', ax=axs[0])
sns.barplot(data=vaders, x='Score', y='neu', ax=axs[1])
sns.barplot(data=vaders, x='Score', y='neg', ax=axs[2])
axs[0].set_title('Positive')
axs[1].set_title('Neutral')
axs[2].set_title('Negative')
plt.tight_layout()
plt.show()
```



Textblob

In [76]:
```
from textblob import TextBlob
```

In [78]:
```
text_review = pd.DataFrame(Amazon_Review, columns=['Score', 'Text','cleanText'])
```

In [79]:
```
def sentiment_analysis(rating_tb):
 def getSubjectivity(text):
   return TextBlob(text).sentiment.subjectivity

 #Create a function to get the polarity
 def getPolarity(text):
   return TextBlob(text).sentiment.polarity

 #Create two new columns ‘Subjectivity’ & ‘Polarity’
 text_review['TextBlob_Subjectivity'] = text_review['cleanText'].apply(getSubjectivity)
 text_review ['TextBlob_Polarity'] =text_review['cleanText'].apply(getPolarity)
 def getAnalysis(score):
  if score < 0:
    return 'Negative'
  elif score == 0:
    return 'Neutral'
  else:
    return 'Positive'
 text_review['TextBlob_Analysis'] = rating_tb['TextBlob_Polarity'].apply(getAnalysis)
 return text_review
```

In [80]:
```
sentiment_analysis(text_review )
```

Out[80]:

	Score	Text	cleanText	TextBlob_Subjectivity	TextBlob_Polarity	TextBlob_Analysis
0	4	McCann's Instant Oatmeal is great if you must ...	mccann instant oatmeal great must oatmeal scra...	0.538509	0.243947	Positive
1	4	This is a good instant oatmeal from the best o...	good instant oatmeal best oatmeal brand us can...	0.494466	0.396667	Positive
2	5	I know the product title says Molecular Gastro...	know product title say molecular gastronomy le...	0.574573	0.089922	Positive
3	5	The expiration date is 21 months from the day ...	expiration date month day bought product tuna ...	0.580000	0.540000	Positive
4	5	This is a fantastic product, and I wish it was...	fantastic product wish readily available store...	0.485952	0.071786	Positive
...	...	...	...	...	...	...
17239	5	My sister-in-law is a connaisseur of salted ca...	sister law connaisseur salted caramel given gi...	0.600000	0.000000	Neutral
17240	5	My review covers the Stevia In The Raw product...	review cover stevia raw product packet form tr...	0.529011	0.150762	Positive
17241	1	First of all, I have no ties with Truvia. In f...	first tie truvia fact decided replace truvia r...	0.505875	0.039013	Positive
17242	5	I'm not 100% certain this is the same product ...	certain product sold place could find review w...	0.566010	0.017863	Positive
17243	5	In the past, I would have to buy a large quant...	past would buy large quantity baker ammonia wo...	0.458333	0.164683	Positive

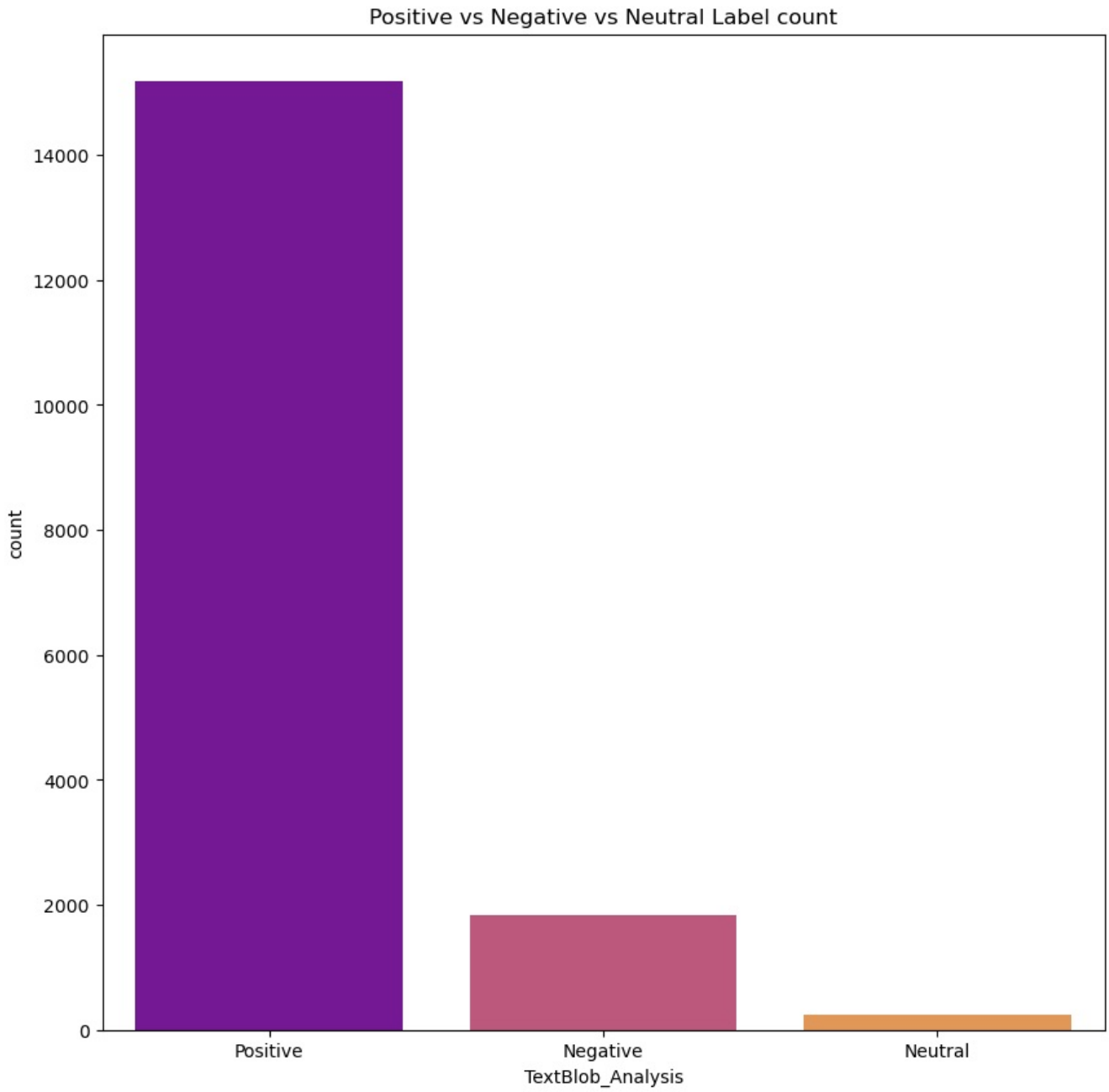
17243 rows × 6 columns

In [82]:
```
import seaborn as sns
sns.countplot(text_review['TextBlob_Analysis'], palette="plasma")
fig = plt.gcf()
fig.set_size_inches(10,10)
plt.title('Positive vs Negative vs Neutral Label count')
```

```
C:\Users\chira\anaconda3\lib\site-packages\seaborn\_decorators.py:36: FutureWarning: Pass the following variable as a keyword arg: x. From version 0.12, the only valid positional argument will be `data`, and passing other arguments without an explicit keyword will result in an error or misinterpretation.
  warnings.warn(
```

Out[82]:

```
Text(0.5, 1.0, 'Positive vs Negative vs Neutral Label count')
```



In [ ]: