

# Analysis Of Amazon Cell Phone Reviews Using Ibm Watson Services

## INTRODUCTION

### 1.1 Overview

This project aims at building a model to predict the helpfulness of the review and the rating based on the review text. The smart phone market has been growing increasingly, not only in the conventional sales but has also been penetrating in the online shop. But not all smart phones have good quality to support the needs of consumers and it is to be noticed by the consumer. Before consumers decide to buy a smart phone, they should know the details of the specifications and functions of the smart phone, it can be learned from results of a review of smart phone users.

### 1.2 Purpose

Mobile phones have revolutionized the way we purchase products online, making all the information available at our fingertips. Reviews and ratings submitted by consumers became an integral part of the customer's buying decision process. The review and rating platform provided by eCommerce players creates a transparent system for consumers to take decisions and feel confident about it.

However, it is difficult to read all the feedback for a particular item especially for the popular items with many comments. In this project, we will attempt to understand the factors that contribute to classifying reviews as positive or negative

We will be using Natural language processing to analyze the sentiment (positive or a negative) of the given review. A sample web application is integrated to the model built. My project aims at building a model to predict the helpfulness of the review and the rating based on the review text. Corpus-based and knowledge-based methods can be used to determine the semantic similarity of review text. I used Natural language processing to analyse the sentiment (positive or a negative) of the given review.

## 2 LITERATURE SURVEY

### 2.1 Existing problem

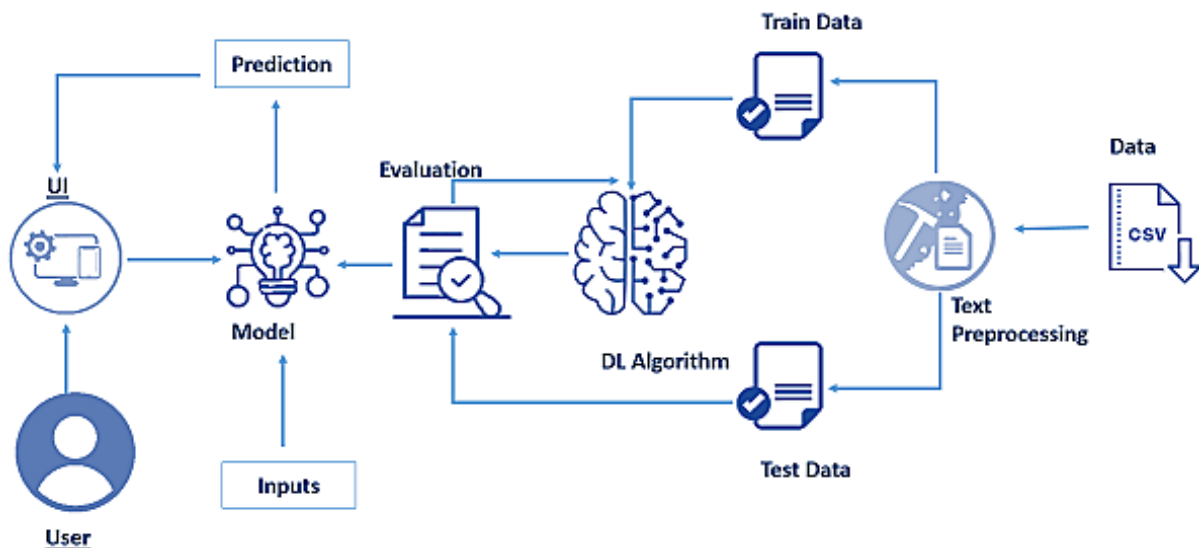
Amazon's product review platform shows the average length of the reviews comes close to 230 characters. Sentiment analysis shows that positive sentiment is prevalent among the reviews and in terms of emotions, 'trust', 'anticipation' and 'joy' have highest scores. So, I was assigned to create an analysis system capable of analyzing the reviews based on NLP.

### 2.2 Proposed solution

I started building a model to predict the helpfulness of the review and the rating based on the review text. Currently consumers who write reviews online are increasing. If the consumers read the whole review it can consume a lot of time. But if it is read without some evaluation it will be biased. Sentiment classification aims to overcome this problem by automatically classifying user review by positive or negative opinion.

## 3 THEORITICAL ANALYSIS

### 3.1 Block diagram



## **3.2 Hardware / Software designing**

### **SOFTWARE DESIGNING:**

For software we would need a compatible operating system for python, java script and HTML. Software needed are:

- 1) Tensorflow
- 2) Keras
- 3) Flask

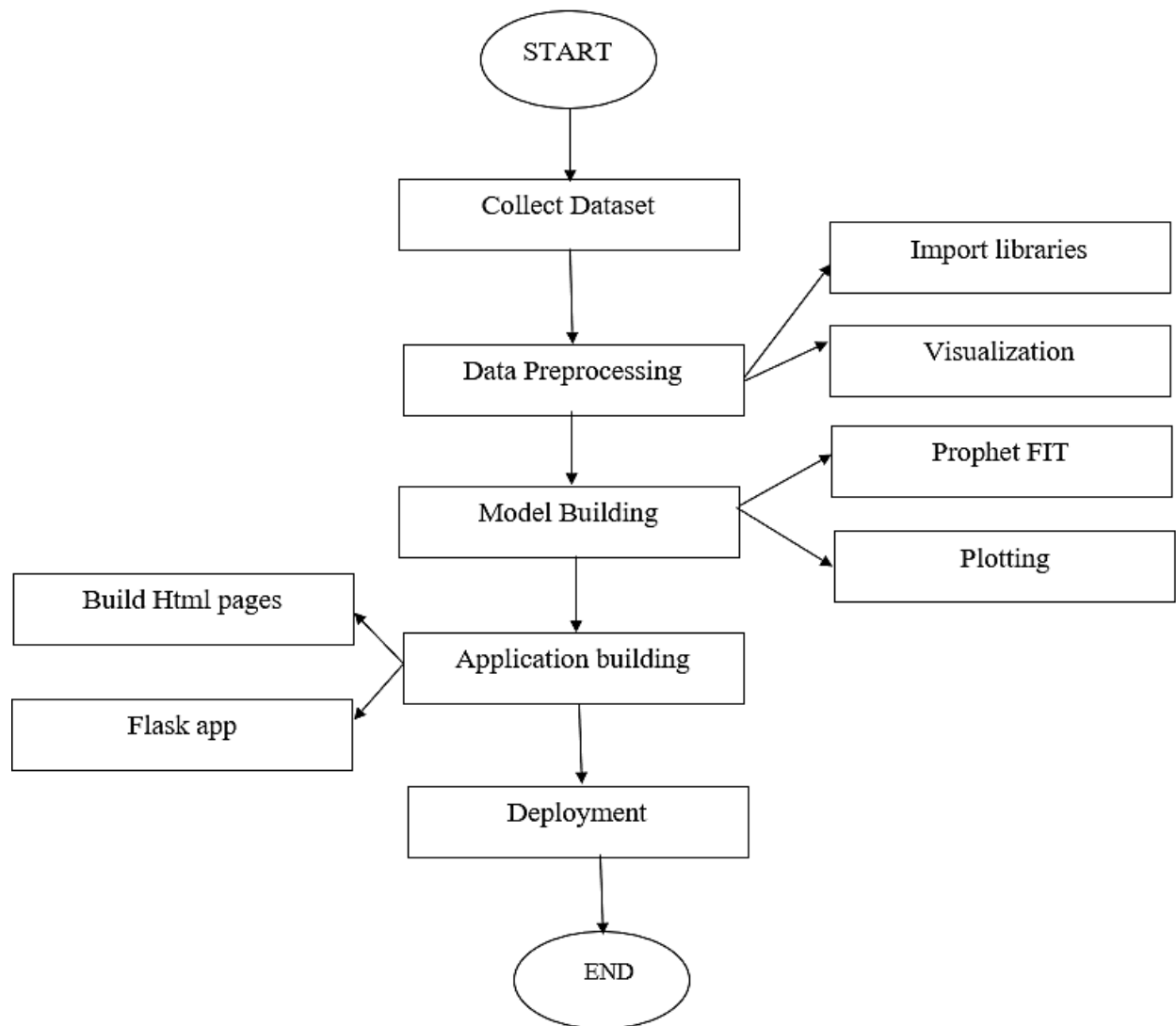
### **HARDWARE DESIGNING:**

- IBM Watson Studio

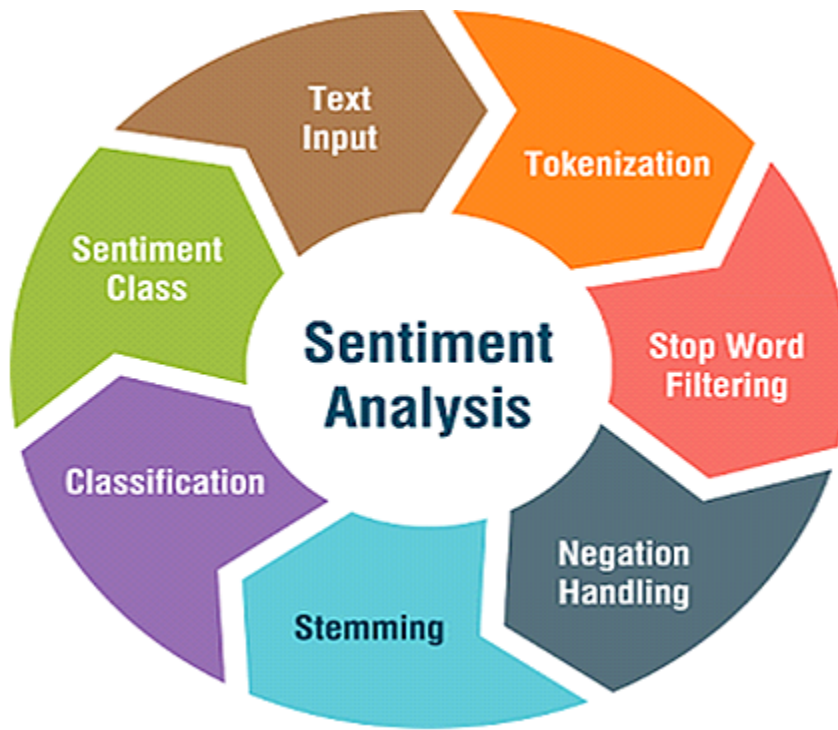
Watson Studio accelerates the machine and deep learning workflows required to infuse AI into your business to drive innovation. It provides a suite of tools for data scientists, application developers and subject matter experts, allowing them to collaboratively connect to data, wrangle that data and use it to build, train and deploy models at scale. Successful AI projects require a combination of algorithms + data + team, and a very powerful compute infrastructure.

- IBM Watson Machine Learning
- IBM Cloud Object Storage

## **4 FLOWCHART**



## 5 EXPERIMENTAL INVESTIGATION



**1) Data Collection:** Data for this experiment was collected from online sources. It is a dataset of size 24KB.

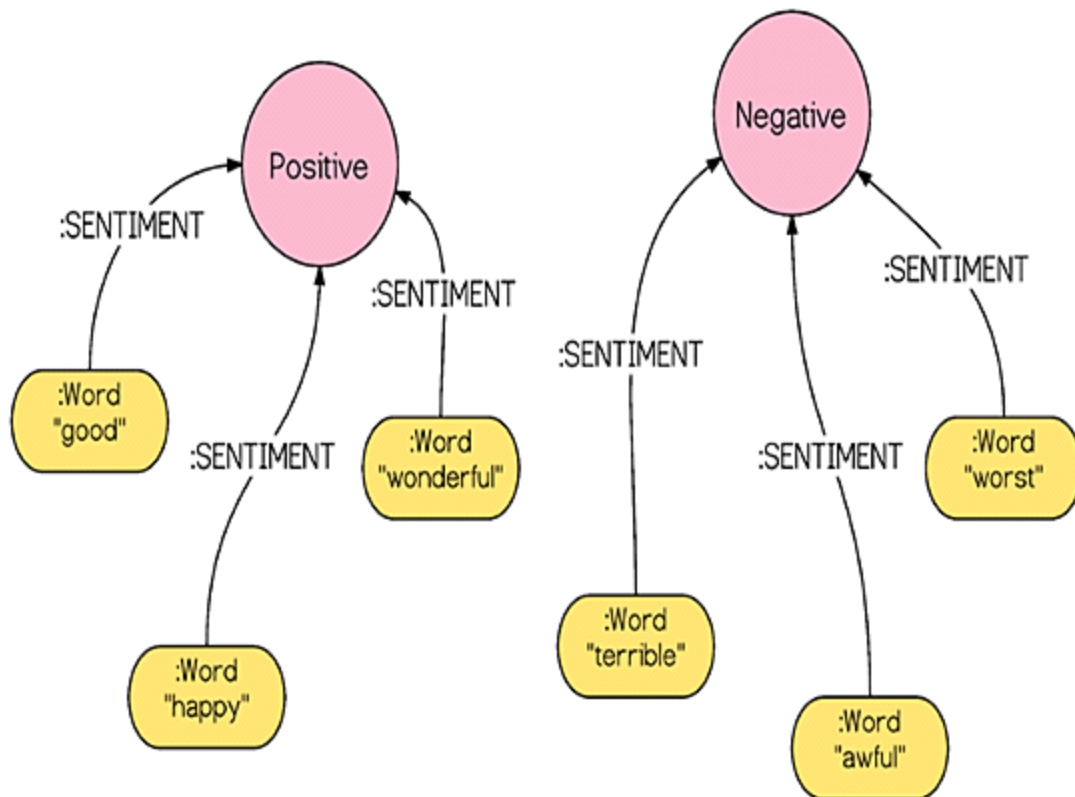
**2) Initial Processing Data:** I imported the dataset and using libraries like pandas and numpy & removed the punctuations and numbers. Also I removed the unwanted like name, date etc. I removed the null values and regular expressions. I converted each word into lower case of its own and applied stemming to remove the stop words.

**3) Model making:** I initialized the model and added input, hidden and output layer. I configured the learning process, trained and tested the model, and optimized the model. I predicted the model by giving inputs and finally saved it.

**4) Preparing HTML file:** Finally I prepared the HTML file and ran my model. This model will predict positive and negative reviews from the input given.

## 6 ACCURACY ACHIEVED:

I got an accuracy of 0.975 which is a good measure for Natural language processing. The Model analysis and predicts the review whether it is a positive or a negative review.



## **7 ADVANTAGES AND DISADVANTAGES:**

### **Advantages:**

- By using sentiment analysis, you gauge how customers feel about different areas of your business without having to read thousands of customer comments at once.
- Increases confidence in new customers and will save their time.
- Brings credibility to products and the company
- Knowing which product works best.

### **Disadvantages:**

- One negative review of a product or business can skew a potential customer's view of them.
- We need to keep reviews current and up to date. Otherwise they will seem out of date and irrelevant.
- Disgruntled customers have the freedom to say whatever they like. This could lead to malicious or damaging information being posted.
- Lack of touch or feel of products during online shopping is a drawback.

## **8.APPLICATIONS:**

63% of customers are more likely to purchase from a site which has user reviews. Also to make them user friendly and easy for customers, my project analyses whether the review is a positive or a negative review and allows the customer to make good decisions.

## 9.CONCLUSION

A model which can predict the amazon mobile phone reviews whether it is a positive or negative review using NLP was created.

## 10.BIBLIOGRAPHY:

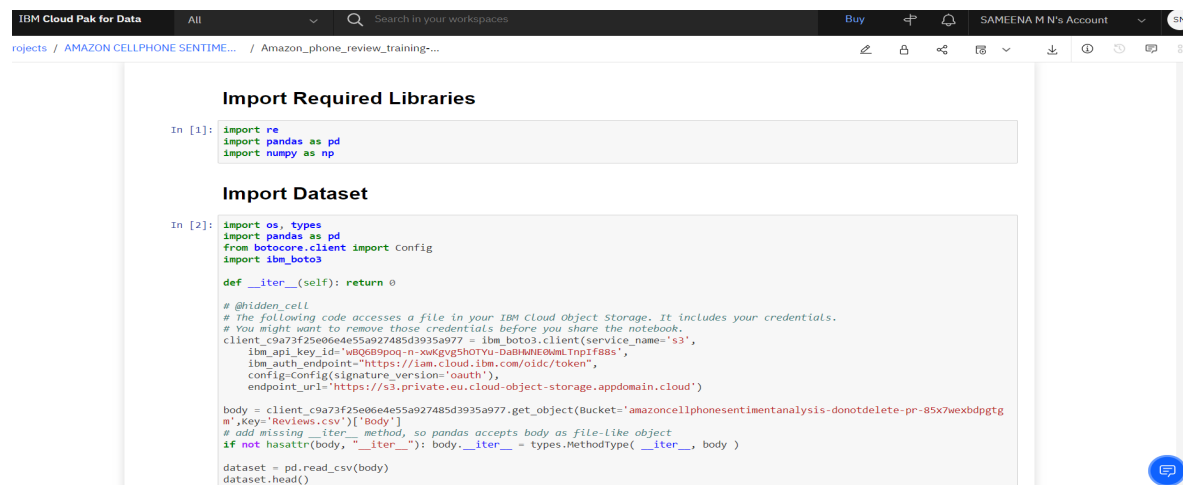
[www.wikipedia.org](http://www.wikipedia.org)

[www.google.com](http://www.google.com)

[www.github.org](http://www.github.org)

<https://github.com/grikomsn/amazon-cell-phones-reviews>

## 11.APPENDIX:



```
IBM Cloud Pak for Data | All | Search in your workspaces | Buy | SAMEENA M N's Account | en
rojects / AMAZON CELLPHONE SENTIMENT... / Amazon_phone_review_training...

Import Required Libraries

In [1]: import re
import pandas as pd
import numpy as np

Import Dataset

In [2]: import os, types
import pandas as pd
from botocore.client import Config
import boto3

def __iter__(self): return 0

# @hidden_cell
# The following code accesses a file in your IBM Cloud Object Storage. It includes your credentials.
# You might want to remove those credentials before you share the notebook.
client_c9a73f25e06e4e5a927485d3935a977 = boto3.client(service_name='s3',
    iam_api_key_id='w0Q6BSpq-n-xwkgvgSHOTVU-DaB4MNE0mLTnpif8Bs',
    iam_auth_endpoint='https://iam.cloud.ibm.com/oidc/token',
    config=Config(signature_version='oauth'),
    endpoint_url='https://s3.private.eu.cloud-object-storage.appdomain.cloud')

body = client_c9a73f25e06e4e5a927485d3935a977.get_object(Bucket='amazoncellphonesentimentanalysis-donotdelete-pr-85x7wexbdpgtg
m',key='Reviews.csv')['Body']
# add missing __iter__ method, so pandas accepts body as file-like object
if not hasattr(body, "__iter__"): body.__iter__ = types.MethodType( __iter__, body )

dataset = pd.read_csv(body)
dataset.head()
```



```
out[2]:
```

	asin	name	rating	date	verified	title	body	helpfulVotes
0	B0000SX2UC	Janet	3	October 11, 2005	False	Def not best, but not worst	I had the Samsung A600 for awhile which is abs...	1.0
1	B0000SX2UC	Luke Wyatt	1	January 7, 2004	False	Text Messaging Doesn't Work	Due to a software issue between Nokia and Spri...	17.0
2	B0000SX2UC	Brooke	5	December 30, 2003	False	Love This Phone	This is a great, reliable phone. I also purcha...	5.0
3	B0000SX2UC	amy m. teague	3	March 18, 2004	False	Love the Phone, BUT...!	I love the phone and all, because I really did...	1.0
4	B0000SX2UC	tristazbimmer	4	August 28, 2005	False	Great phone service and options, lousy case!	The phone has been great for every purpose it ...	1.0

## To Check First 5 observations

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

```
out[3]:
```

	asin	name	rating	date	verified	title	body	helpfulVotes
0	B0000SX2UC	Janet	3	October 11, 2005	False	Def not best, but not worst	I had the Samsung A600 for awhile which is abs...	1.0
1	B0000SX2UC	Luke Wyatt	1	January 7, 2004	False	Text Messaging Doesn't Work	Due to a software issue between Nokia and Spri...	17.0
2	B0000SX2UC	Brooke	5	December 30, 2003	False	Love This Phone	This is a great, reliable phone. I also purcha...	5.0
3	B0000SX2UC	amy m. teague	3	March 18, 2004	False	Love the Phone, BUT...!	I love the phone and all, because I really did...	1.0

## To Check Last 5 observations

```
In [4]: dataset.tail()
```

```
out[4]:
```

	asin	name	rating	date	verified	title	body	helpfulVotes
67981	B081H6STQQ	jande	5	August 16, 2019	False	Awesome Phone, but finger scanner is a big mis...	I love the camera on this phone. The screen is...	1.0
67982	B081H6STQQ	2cool4u	5	September 14, 2019	False	Simply Amazing!	I've been an Xperia user for several years and...	1.0
67983	B081H6STQQ	simon	5	July 14, 2019	False	great phon3, but many bugs need to fix. still ...	buy one more for my cousin	NaN
67984	B081TJFVCJ	Tobiasz Jedrysiak	5	December 24, 2019	True	Phone is like new	Product looks and works like new. Very much re...	NaN
67985	B0825BB7SG	Owen Gonzalez	5	December 11, 2019	False	Outstanding phone for the price	I love the size and style of this phone. It is...	NaN

## Checking For Null Values

```
In [5]: dataset.isnull().any()
```

```
Out[5]: asin          False
name          True
rating        False
date          False
verified       False
title         True
body          True
helpfulVotes  True
```

## Fill null values

```
In [6]: dataset['body'] = dataset['body'].fillna('').apply(str)
dataset['name'] = dataset['name'].fillna('').apply(str)
dataset['title'] = dataset['title'].fillna('').apply(str)
dataset['helpfulVotes'] = dataset['helpfulVotes'].fillna('').apply(str)
```

## Drop Null values

dataset.dropna(inplace=True)

```
In [7]: dataset.isnull().sum()
```

```
Out[7]: asin      0
name      0
rating    0
date      0
verified  0
title     0
body      0
helpfulVotes  0
dtype: int64
```

## Dropping columns

```
In [8]: dataset=dataset.drop(columns=['asin','name','helpfulVotes','date'],axis=1)
```

```
In [9]: dataset.isnull().sum()
```

```
Out[9]: rating    0
verified    0
title       0
body        0
```

```
In [10]: dataset.head()
```

```
Out[10]:
```

	rating	verified	title	body
0	3	False	Def not best, but not worst	I had the Samsung A600 for awhile which is abs...
1	1	False	Text Messaging Doesn't Work	Due to a software issue between Nokia and Spr...
2	5	False	Love This Phone	This is a great, reliable phone. I also purcha...
3	3	False	Love the Phone, BUT...!	I love the phone and all, because I really did...
4	4	False	Great phone service and options, lousy case!	The phone has been great for every purpose it ...

```
In [11]: a=dataset['rating'].tolist()
a
```

```
Out[11]: [3,
1,
5,
```

```
In [13]: dt=pd.DataFrame(d,columns=['emotion'])
dt
```

```
Out[13]:
```

	emotion
0	1
1	0
2	1
3	1
4	1
...	...
67981	1
67982	1
67983	1
67984	1
67985	1

67986 rows x 1 columns

```
In [14]: data1=pd.concat([dataset,dt],axis=1)
data1.head()
```

```
Out[14]:
```

	rating	verified	title	body	emotion
--	--------	----------	-------	------	---------

### Drop verified column from data1

```
In [15]: data1.drop(['verified'],axis=1,inplace=True)
```

### Join both title and body

```
In [16]: data1['Review'] = data1[['title', 'body']].agg(' '.join, axis=1)
```

```
In [17]: data1.head()
```

Out[17]:

	rating	title	body	emotion	Review
0	3	Def not best, but not worst	I had the Samsung A600 for awhile which is abs...	1	Def not best, but not worst I had the Samsung ...
1	1	Text Messaging Doesn't Work	Due to a software issue between Nokla and Spri...	0	Text Messaging Doesn't Work Due to a software ...
2	5	Love This Phone	This is a great, reliable phone. I also purcha...	1	Love This Phone This is a great, reliable phon...
3	3	Love the Phone, BUT...!	I love the phone and all, because I really did...	1	Love the Phone, BUT...! I love the phone and a...
4	4	Great phone service and options, lousy case!	The phone has been great for every purpose it ...	1	Great phone service and options, lousy case! T...

### Dropping Columns

```
In [18]: data1.drop(['title','body','rating'],axis=1,inplace=True)
```

```
In [19]: data1.head()
```

Out[19]:

	emotion	Review
0	1	Def not best, but not worst I had the Samsung ...
1	0	Text Messaging Doesn't Work Due to a software ...
2	1	Love This Phone This is a great, reliable phon...
3	1	Love the Phone, BUT...! I love the phone and a...
4	1	Great phone service and options, lousy case! T...

```
In [20]: len(list(data1['emotion']))
```

Out[20]: 67986

split the data into x(independent variable)

```
In [21]: x=data1.iloc[:,1].values
```

```
In [22]: len(x)
```

Out[22]: 67986

```
In [23]: data1.head()
```

Out[23]:

	emotion	Review
--	---------	--------

### Check the data1 shape

```
In [24]: data1.shape
```

Out[24]: (67986, 2)

```
In [25]: #import natural language toolkit
import nltk
#import stopwords library to remove stopwords
from nltk.corpus import stopwords
#library used for stem the words
from nltk.stem.porter import PorterStemmer
#create an object for stemming
ps = PorterStemmer()
#library used for stem the words
from nltk.stem import WordNetLemmatizer
#create an object for wordnet Lemmatizer
wordnet=WordNetLemmatizer()
```

```
In [26]: import re
```

```
In [27]: nltk.download('wordnet')
nltk.download('stopwords')
```

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

Out[27]: True

```
In [28]: import nltk
```

```
nltk.download("omw-1.4")
```

```
[nltk_data] Downloading package omw-1.4 to /home/wsuser/nltk_data...
```

```
In [29]: # Initialize empty array to append clean text
corpus=[]
# no of rows to clean
for i in range(len(x)):
    #replacing punctuations and numbers using re library
    temp=re.sub('[^a-zA-Z]', ' ',x[i])

    # convert all text to lower cases
    temp=temp.lower()
    # split to array(default delimiter is " ")
    temp=temp.split()

    # creating WordNetLemmatizer object to take main Lemma of each word
    wordnet = WordNetLemmatizer()
    #loop for lemmatization each word in string array at ith row
    temp=[wordnet.lemmatize(word) for word in temp if not word in set(
        stopwords.words('english'))]

    #rejoin all string array elements to create back into a string
    temp=' '.join(temp)
    #append each string to create array of clean text
    corpus.append(temp)
```

Before Stopwords: "This is a sample sentence, showing off the stop words filtration."

After removing stopwords: "This sample sentence, showing stop words filtration."

```
In [30]: corpus
Out[30]: ['def best worst samsune awhile absolute doo doo read review detect rage stupid thing finally died used nokia phone bought eara']
In [31]: #creating bag of word model
from sklearn.feature_extraction.text import CountVectorizer
#To extract max 2000 feature, "max_features" is attribute to
#experiment with to get better results
cv=CountVectorizer(max_features= 2000)
#z contains vectorized data (independent variable)
z=cv.fit_transform(corpus).toarray()
```

Save the Bag of word model

```
In [32]: import pickle
pickle.dump(cv,open('count_vec.pkl','wb'))
```

```
In [33]: type(z)
```

```
Out[33]: numpy.ndarray
```

```
In [34]: z
```

```
Out[34]: array([[0, 0, 1, ..., 0, 0, 0],
                [0, 0, 0, ..., 0, 0, 0],
                [0, 0, 0, ..., 0, 0, 0],
                ...,
                [0, 0, 0, ..., 0, 0, 0],
                [0, 0, 0, ..., 0, 0, 0],
                [0, 0, 0, ..., 0, 0, 0]])
```

Creating dependent variable

```
In [35]: # y contains answers if review is positive or negative
y=data1.iloc[:,0].values
y
```

```
Out[35]: array([1, 0, 1, ..., 1, 1, 1])
```

```
Out[36]: (67986, 2000)
```

```
In [37]: from sklearn.model_selection import train_test_split
x_train,x_test,y_train,y_test=train_test_split(z,y,test_size=0.3)
```

```
In [38]: x_train.shape
```

```
Out[38]: (47590, 2000)
```

```
In [39]: #import library which uses tensorflow as backend
#import keras
#sequential is to initialise the model
from tensorflow.keras.models import Sequential
#Dense is used to add layers
from tensorflow.keras.layers import Dense
```

```
In [40]: model=Sequential()
```

Adding Output Layer

```
In [41]: model.add(Dense(2000,activation="relu"))
model.add(Dense(1000,activation="relu"))
model.add(Dense(1,activation="sigmoid"))
```

compile the model

```
In [42]: model.compile(optimizer="adam",loss="binary_crossentropy",metrics=["accuracy"])
```

Training the Model

```
In [43]: model.fit(x_train,y_train,batch_size=32,epochs=10)
```

```
In [44]: #this will save your model weights
#and h5 is the extension for keras
model.save('phone_ibm.h5')
```

```
In [45]: !tar -zcvf amazon_review.zip phone_ibm.h5

phone_ibm.h5
```

```
In [46]: ls -l

amazon_review.zip
count_vec.pkl
phone_ibm.h5
```

```
In [47]: !pip install watson-machine-learning-client --upgrade

Requirement already satisfied: watson-machine-learning-client in /opt/conda/envs/Python-3.8-main/lib/python3.8/site-packages (1.0.391)
Requirement already satisfied: pandas in /opt/conda/envs/Python-3.8-main/lib/python3.8/site-packages (from watson-machine-learn
```

```
In [48]: from ibm_watson_machine_learning import APIClient
wml_credentials={
    "url":"https://eu-gb.ml.cloud.ibm.com",
    "apikey":"uWQwOGfEbmfU2M7_DrUhcAm3jGj1Cyjw_QYwFK5Tw4h8"
}
client = APIClient(wml_credentials)
```

```
In [49]: client = APIClient(wml_credentials)
```

```
In [50]: def guid_from_space_name(client,space_name):
space=client.spaces.get_details()
return(next(item for item in space['resources'] if item['entity']['name']==space_name)['metadata']['id'])
```

```
In [51]: space_uid=guid_from_space_name(client,'amazon_deploy')
print("Space UID="+space_uid)

Space UID=7d89cbf8-de4f-47b7-b85a-3f4bedb3cf9e
```

```
In [52]: client.set.default_space(space_uid)
```

```
Out[52]: 'SUCCESS'
```

```
In [53]: client.software_specifications.list()
```

```
In [54]: software_spec_uid=client.software_specifications.get_uid_by_name("tensorflow_2.4-py3.7")
software_spec_uid
```

```
Out[54]: '65e171d7-72d1-55d9-8ebb-f813d620c9bb'
```

```
In [55]: model_details = client.repository.store_model(model='amazon_review.zip',meta_props={
    client.repository.ModelMetaNames.NAME:"Amazonreviewmodel",
    client.repository.ModelMetaNames.TYPE:"tensorflow_2.4",
    client.repository.ModelMetaNames.SOFTWARE_SPEC_UID:software_spec_uid
})
model_id = client.repository.get_model_id(model_details)
```

Note: Warnings!! : Software specification tensorflow\_2.4-py3.7 specified for the wml\_model is deprecated and will be removed in the future. We recommend you use tensorflow\_rt22.1-py3.9 instead. For details see Supported Frameworks [https://dataplatform.cloud.ibm.com/docs/content/wsj/analyze-data/pm\\_service\\_supported\\_frameworks.html](https://dataplatform.cloud.ibm.com/docs/content/wsj/analyze-data/pm_service_supported_frameworks.html)

```
In [56]: model_id
```

```
Out[56]: '35cdd897-e8cc-45f5-8371-3b35d916b829'
```

```
temp.py X app.py X
1
2 from flask import render_template, Flask, request, url_for
3 from tensorflow.keras.models import load_model
4 import pickle
5 import tensorflow as tf
6 #graph = tf.get_default_graph()
7 with open(r'count_vec.pkl', 'rb') as file:
8     cv=pickle.load(file)
9 cla = load_model('amazon_review.h5')
10 #cla.compile(optimizer='adam', loss='binary_crossentropy')
11 app = Flask(__name__)
12 @app.route('/')
13 def index():
14     return render_template('index.html')
15 @app.route('/tpredict')
16 @app.route('/', methods = ['GET', 'POST'])
17 def page2():
18     if request.method == 'GET':
19         img_url = url_for('static', filename = 'style/3.jpg')
20         return render_template('index.html', url=img_url)
21     if request.method == 'POST':
22         topic = request.form['tweet']
23         print("Hey " + topic)
24         topic=cv.transform([topic])
25         print("\n"+str(topic.shape)+"\n")
26
27         y_pred = cla.predict(topic)
28         print("pred is "+str(y_pred))
29         if(y_pred > 0.5):
30             img_url = url_for('static', filename = 'style/1.jpg')
31             topic = "Positive Tweet"
32         else:
33             img_url = url_for('static', filename = 'style/2.jpg')
34             topic = "Negative Tweet"
35
36         return render_template('index.html', ypred = topic)
37
38
39
40
```

## Amazon Cellphone Sentiment Analysis

Type your Review here

VIDEO LINK -

<https://drive.google.com/file/d/1znyh9iHTD0q63NITVlgJ4k3jPe1Qg9NO/view?usp=sharing>

PROJECT LINK

<https://drive.google.com/drive/folders/1pw0DywbFDJWvj3aQnqCrwnHRPFuuM-pV?usp=sharing>