4/18/2021

Term 2 – Text Analytics Assignment Report

AMPBA Winters Batch 15

Deep Kamal Singh
PGID-12020053

Table of Contents

Assignment Instructions

Deliverables:

A report **(A pdf file)** & 7 Python **(.py)** code files(one for extracting the data from JSON, one for each question). Also, submit a short README file which details the steps to run the pieces of code. Include output for each question in the report.

General Instructions:

1. This is an individual assignment. The **maximum marks** are **40**.
2. **Do NOT** submit **.zip** files otherwise the submission will not be considered.
3. Please **do not** submit **.ipynb** files. Submit only **.py** files
4. Any late submission will attract a penalty as mentioned in the course outline.
5. The honor code for this submission is **2N-b**.
6. Use **Spacy (v3.0.5)** with **en_core_web_sm** for all NLP modules wherever applicable.
7. Please use a **random seed(random_state) of 50** wherever needed.

Assignment Questions

You are given the reviews dataset. These are 194439 amazon reviews for cell phones and accessories taken from https://jmcauley.ucsd.edu/data/amazon/

1. Extract the reviewText and overall fields from this file. These are the only two fields we will work with. **Submit q0.py**

2. Take the first 10000 review texts. Perform only these steps as part of pre-processing: lowercasing and removing punctuation. Compute IDF of all words in these reviews. Report the top 20 words and bottom 20 words, based on IDF, with their IDF scores. **Submit q1.py**. **[6 marks]**

3. Take the first 10 review texts. Perform sentence detection using Spacy. Each line should have review ID (i.e., line number from the file) and the sentence itself. **Submit q2.py**. **[6 marks]**

4. Take the first 10 reviews texts. Perform word tokenization, lemmatization, part-of-speech tagging. Use Spacy. Each line should have review ID (i.e., line number from the file), token (i.e. word), lemma, and POS tag. **Submit q3.py**. **[6 marks]**

5. Take the first 1000 review texts. Using gensim, create an LDA model with 10 topics. Report the top 50 words with probs for each of the ten topics. Each line has topic number, word, prob in that topic. **Submit q4.py**. **[6 marks]**

6. Use the entire dataset. Take the first 80% dataset for train and remaining 20% for test. On the train set, obtain TFIDF features (with 50K vocabulary) and learn a multinomial Naïve Bayes model. Report the accuracy on the test set for this five-class classification problem. Accuracy should be reported as class-wise precision, recall and F1. **Submit q5.py**. **[10 marks]**

7. Take the first 1000 "rating-1.0" reviews. Summarize them to 1% (in terms of words) using gensim and send across your summary. Also, take the first 1000 "rating-5.0" reviews. Summarize them to approximately 300 words using gensim and send across your summary. **Submit q6.py**. **[6 marks]**

Deadline: 18th April 2021, 23:55 hrs

Solutions

Q1.

Extract the *reviewText* and overall fields from this file. These are the only two fields we will work with.

Solution of Q1

Code file: q0.py

Libraries:

- Pandas (https://pandas.pydata.org/)

Code notes

After reviewing the json file, we see it contains JSON lines, and it's size is roughly 142MB, and contains data in JSON lines format, thus Pandas library is chosen to directly load the JSON into Pandas DF, this same class will be imported by all subsequent code files for data load

Code

```
import pandas as pd

class ReviewDataLoader:
    review_data = pd.read_json('Cell_Phones_and_Accessories_5.json', lines=True)
    # This method will be used by external code to access the loaded data set
    def get_review_data(self):
        return self.review_data

# this condition checks if the file is imported or directly executed
# Only in case of direct execution it runs below code
if __name__ == '__main__':
    # Fetching only 2 fields  - reviewText and overall, and saving it in a local variable
    d = ReviewDataLoader().get_review_data()[['reviewText', 'overall']]
    # printing summary of records
    print("Printing name of columns\n", d.columns, "\nPrinting info of loaded dataframe\n",
d.info(),
          "\nPrinting summary of loaded DF\n", d.describe())
    # Displaying top 10 records
    print("\nJust displaying top 10 records   >>>>\n", d.head(10))
```

Execution output of q0.py

Although no output is expected, but for purpose of understanding the data, some basic printing is added.

Q2.

Take the first 10000 review texts. Perform only these steps as part of pre-processing: lowercasing and removing punctuation. Compute IDF of all words in these reviews. Report the top 20 words and bottom 20 words, based on IDF, with their IDF scores.

Solution of Q2

Code file: q1.py

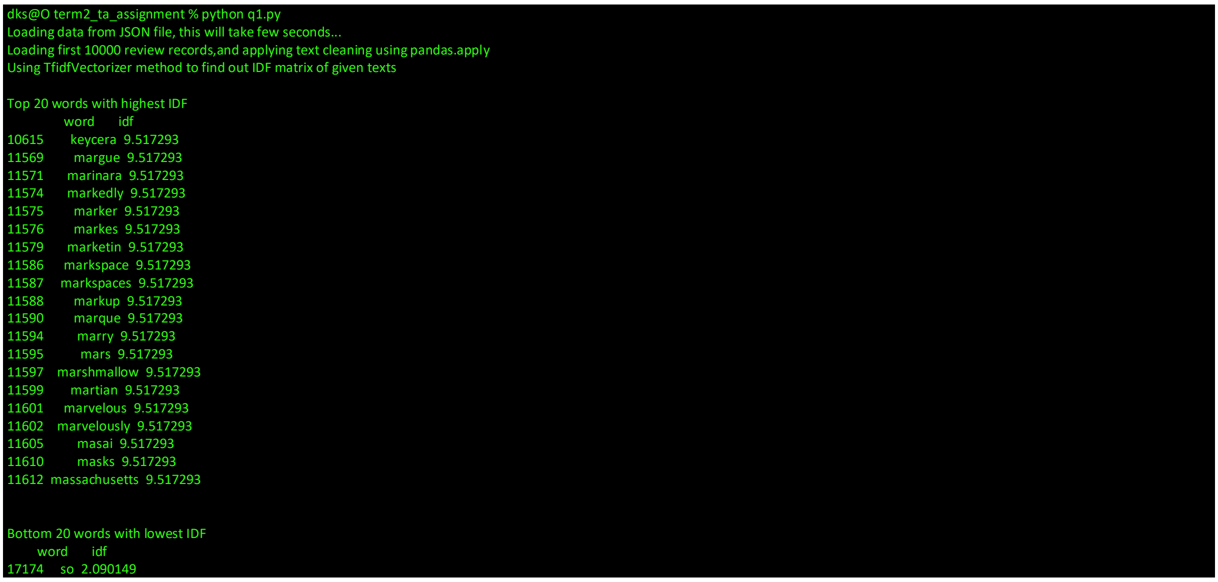
Code notes

IDF calculation was attempted via TextBlob method, and a revised method was also written, however fastest is sklearn library's TFIDFVectorizer, its faster by many times – this it is used

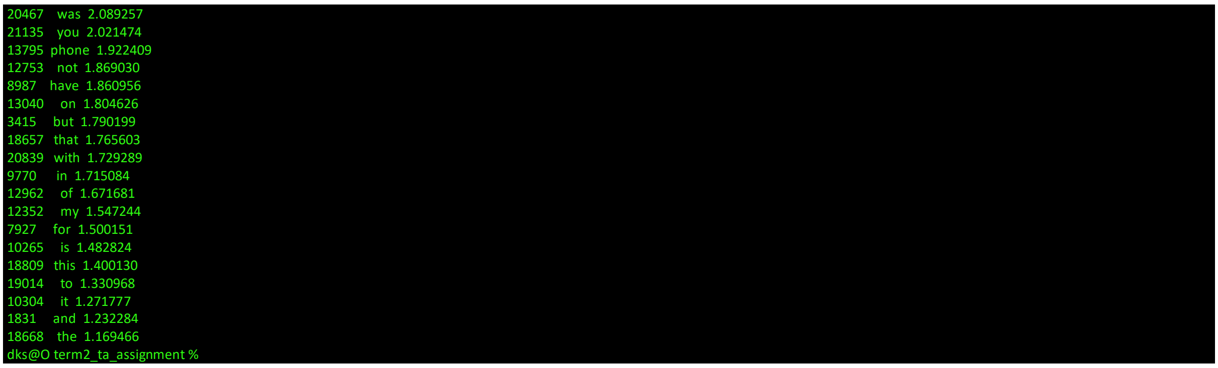
Code Snippet

```
# we are using sklearn.feature_extraction.text.TfidfVectorizer from sklearn library for IDF
extractions
def TfidfVectorizer_method(first_10000_review_texts):
    print("Using TfidfVectorizer method to find out IDF matrix of given texts")

    # using ngram_range for single word, use_idf=True is flag to keep idf_ matrix available,
    # we will access idf_ for our purpose
    tfidf_vectorizer = TfidfVectorizer(ngram_range=(1, 1), use_idf=True)
    # fitting first_10000_review_texts in tfidf_vectorizer - this is where we are preparing our
model
    tfidf_vectorizer.fit_transform(first_10000_review_texts)
    # since idf_ and feature_names (word) come in separate list, we will keep them in pandas DF for
sorting and printing
    word_idf = pd.DataFrame(columns=['word', 'idf'])
    # tfidf_vectorizer.get_feature_names() gives list of all words
    word_idf['word'] = tfidf_vectorizer.get_feature_names()
    # tfidf_vectorizer.idf_ contains IDF of each word indexed by the sequence of
tfidf_vectorizer.get_feature_names()
    word_idf['idf'] = tfidf_vectorizer.idf_
    # pandas sort_values are quick and easy way to sort out a DF by a field, using idf_ in this case
    word_idf = word_idf.sort_values(by='idf', ascending=False)
    print("\nTop 20 words with highest IDF\n", word_idf.head(20))
    print("\n\nBottom 20 words with lowest IDF\n", word_idf.tail(20))
```

Execution output of q1.py



```
dks@O term2_ta_assignment % python q1.py
Loading data from JSON file, this will take few seconds...
Loading first 10000 review records,and applying text cleaning using pandas.apply
Using TfidfVectorizer method to find out IDF matrix of given texts

Top 20 words with highest IDF
             word       idf
10615      keycera  9.517293
11569       margue  9.517293
11571     marinara  9.517293
11574     markedly  9.517293
11575       marker  9.517293
11576       markes  9.517293
11579     marketin  9.517293
11586    markspace  9.517293
11587   markspaces  9.517293
11588       markup  9.517293
11590       marque  9.517293
11594        marry  9.517293
11595         mars  9.517293
11597   marshmallow  9.517293
11599      martian  9.517293
11601    marvelous  9.517293
11602  marvelously  9.517293
11605        masai  9.517293
11610        masks  9.517293
11612 massachusetts  9.517293


Bottom 20 words with lowest IDF
      word       idf
17174    so  2.090149
```



```
20467   was  2.089257
21135   you  2.021474
13795 phone  1.922409
12753   not  1.869030
8987   have  1.860956
13040    on  1.804626
3415    but  1.790199
18657  that  1.765603
20839  with  1.729289
9770     in  1.715084
12962    of  1.671681
12352    my  1.547244
7927    for  1.500151
10265    is  1.482824
18809  this  1.400130
19014    to  1.330968
10304    it  1.271777
1831    and  1.232284
18668   the  1.169466
dks@O term2_ta_assignment %
```

Q3.

Take the first 10 review texts. Perform sentence detection using Spacy. Each line should have review ID (i.e., line number from the file) and the sentence itself.

Solution of Q3

Code file: q2.py

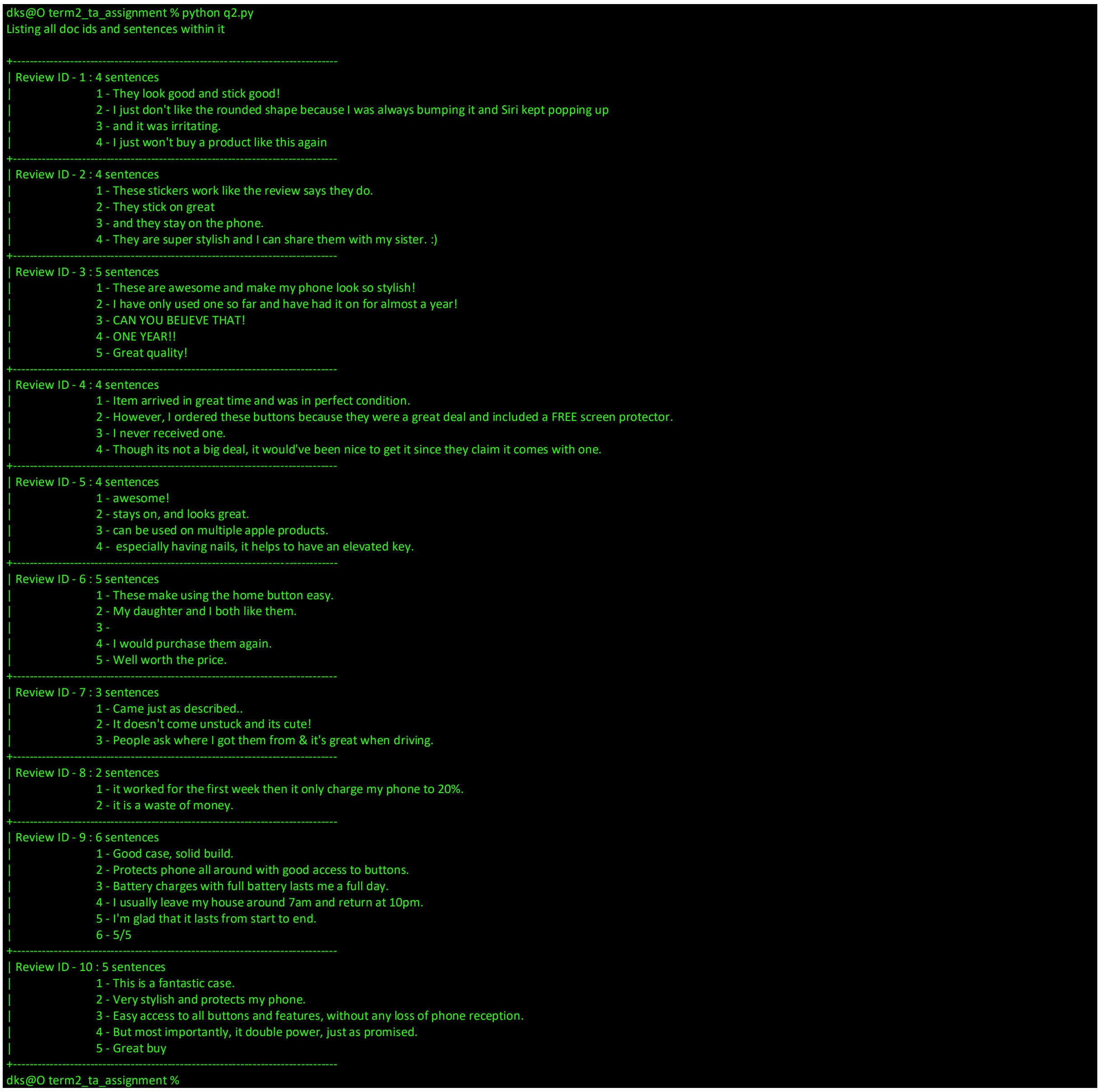
Code notes

Spacy library is used for sentence extraction from paragraph, for this to work we must preload language pack for “English” - *en_core_web_sm.*

Code Snippet

```
import spacy
from q0 import ReviewDataLoader

# Loading review data and getting only field 'Review Text' which gets stored in local variable,
# the index of Series will serve as reviewID,
q2_data = ReviewDataLoader().get_review_data()['reviewText'].head(10)
# we observe the reviewText field is in standard english language, thus loading `en_core_web_sm` for
Spacy nlp
spacy_nlp = spacy.load("en_core_web_sm")
# we all converting each review text to spacy_nlp object using pandas apply
# in next apply we all obtaining LIST of sentences from sps
sents_in_top10_reviews = q2_data.apply(spacy_nlp).apply(lambda l: list(l.sents))
print("Listing all doc ids and sentences within it\n")
for reviewId, aReview in sents_in_top10_reviews.iteritems():
    print("+--------------------------------------------------------------------------------")
    print("| Review ID -", (reviewId+1), ":", len(aReview), "sentences")
    for i, aSentence in enumerate(aReview):
        print("|", "\t", (i+1), "-", aSentence)

print("+--------------------------------------------------------------------------------")
```

Execution output of q2.py



```
dks@O term2_ta_assignment % python q2.py
Listing all doc ids and sentences within it

+--------------------------------------------------------------------------------
| Review ID - 1 : 4 sentences
|                   1 - They look good and stick good!
|                   2 - I just don't like the rounded shape because I was always bumping it and Siri kept popping up
|                   3 - and it was irritating.
|                   4 - I just won't buy a product like this again
+--------------------------------------------------------------------------------
| Review ID - 2 : 4 sentences
|                   1 - These stickers work like the review says they do.
|                   2 - They stick on great
|                   3 - and they stay on the phone.
|                   4 - They are super stylish and I can share them with my sister. :)
+--------------------------------------------------------------------------------
| Review ID - 3 : 5 sentences
|                   1 - These are awesome and make my phone look so stylish!
|                   2 - I have only used one so far and have had it on for almost a year!
|                   3 - CAN YOU BELIEVE THAT!
|                   4 - ONE YEAR!!
|                   5 - Great quality!
+--------------------------------------------------------------------------------
| Review ID - 4 : 4 sentences
|                   1 - Item arrived in great time and was in perfect condition.
|                   2 - However, I ordered these buttons because they were a great deal and included a FREE screen protector.
|                   3 - I never received one.
|                   4 - Though its not a big deal, it would've been nice to get it since they claim it comes with one.
+--------------------------------------------------------------------------------
| Review ID - 5 : 4 sentences
|                   1 - awesome!
|                   2 - stays on, and looks great.
|                   3 - can be used on multiple apple products.
|                   4 -  especially having nails, it helps to have an elevated key.
+--------------------------------------------------------------------------------
| Review ID - 6 : 5 sentences
|                   1 - These make using the home button easy.
|                   2 - My daughter and I both like them.
|                   3 - 
|                   4 - I would purchase them again.
|                   5 - Well worth the price.
+--------------------------------------------------------------------------------
| Review ID - 7 : 3 sentences
|                   1 - Came just as described..
|                   2 - It doesn't come unstuck and its cute!
|                   3 - People ask where I got them from & it's great when driving.
+--------------------------------------------------------------------------------
| Review ID - 8 : 2 sentences
|                   1 - it worked for the first week then it only charge my phone to 20%.
|                   2 - it is a waste of money.
+--------------------------------------------------------------------------------
| Review ID - 9 : 6 sentences
|                   1 - Good case, solid build.
|                   2 - Protects phone all around with good access to buttons.
|                   3 - Battery charges with full battery lasts me a full day.
|                   4 - I usually leave my house around 7am and return at 10pm.
|                   5 - I'm glad that it lasts from start to end.
|                   6 - 5/5
+--------------------------------------------------------------------------------
| Review ID - 10 : 5 sentences
|                   1 - This is a fantastic case.
|                   2 - Very stylish and protects my phone.
|                   3 - Easy access to all buttons and features, without any loss of phone reception.
|                   4 - But most importantly, it double power, just as promised.
|                   5 - Great buy
+--------------------------------------------------------------------------------
dks@O term2_ta_assignment %
```

Q4.

Take the first 10 reviews texts. Perform word tokenization, lemmatization, part-of-speech tagging. Use Spacy. Each line should have review ID (i.e., line number from the file), token (i.e. word), lemma, and POS tag

Solution of Q4

Code file: q3.py

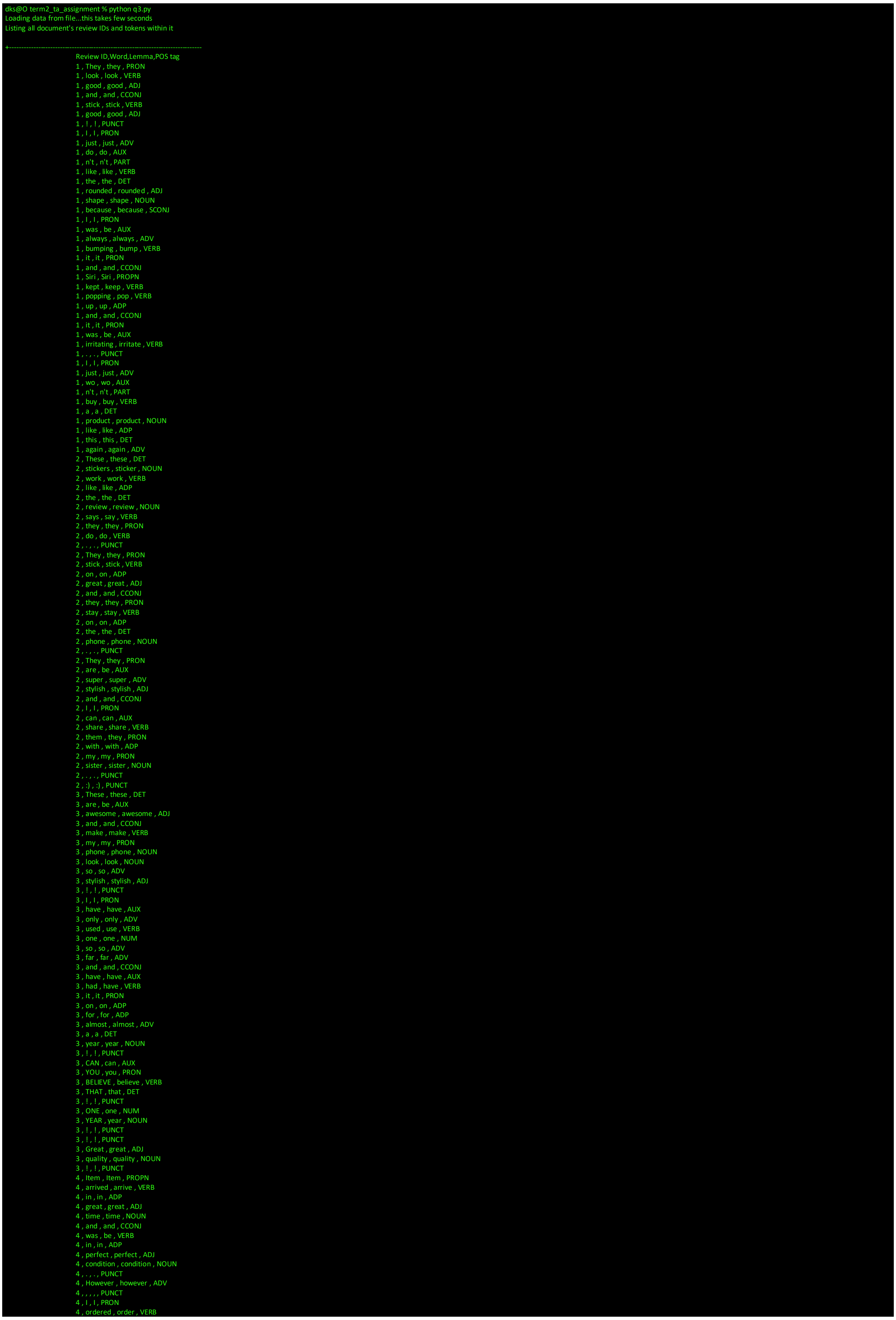
Code notes

Spacy library is used for tokenization, lemma, and POS extraction from review text, for this to work we must preload language pack for “English” - *en_core_web_sm.*

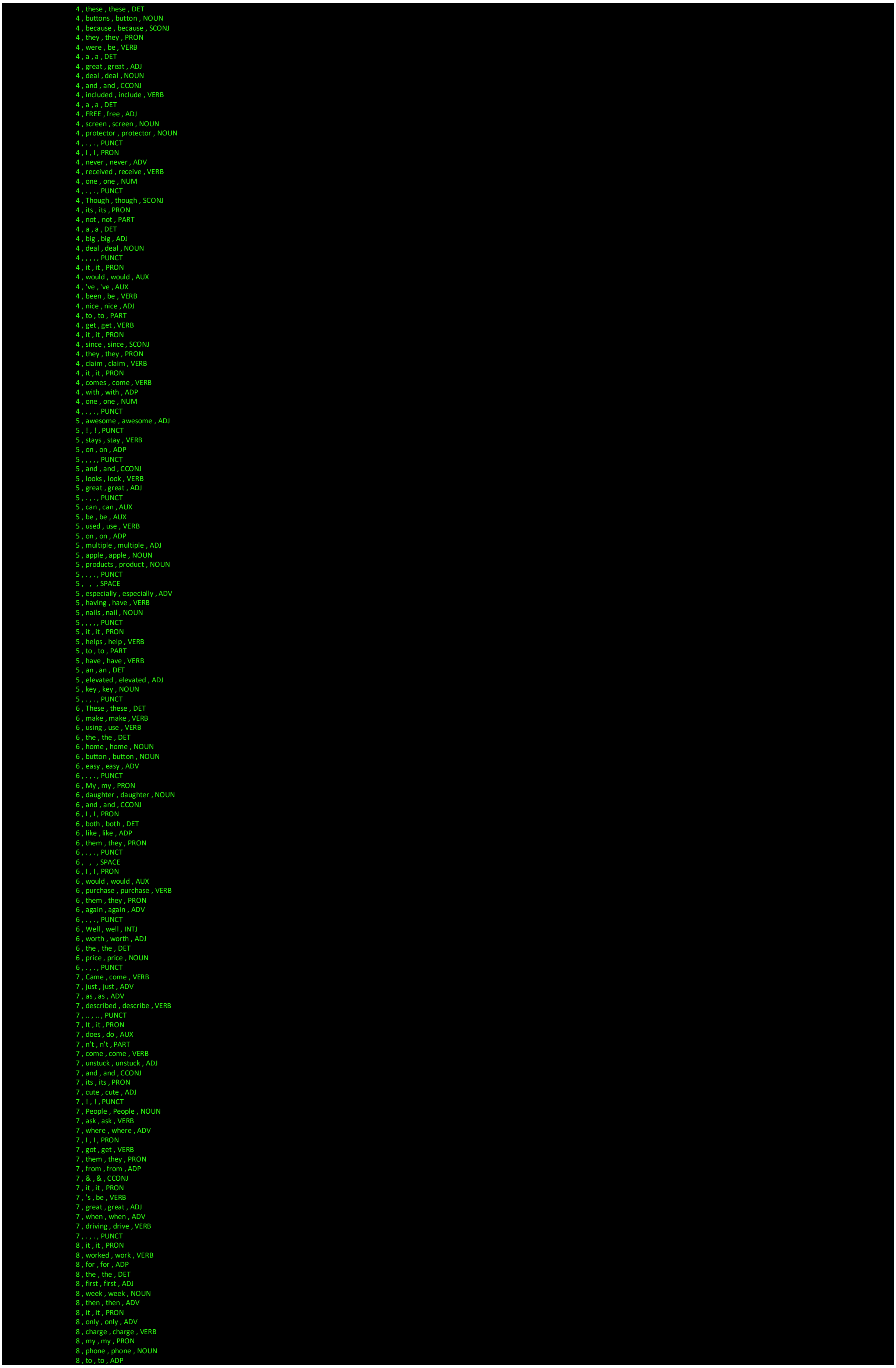
Code Snippet

```
import spacy
print("Loading data from file...this takes few seconds")
from q0 import ReviewDataLoader
# Loading review data and getting only field 'Review Text' which gets stored in local variable,
# the index of Series will serve as reviewID,
q3_data = ReviewDataLoader().get_review_data()['reviewText'].head(10)
# Loading english NLP pack from Spacy
spacy_nlp = spacy.load("en_core_web_sm")
# we are applying spacy_nlp() on each text, this converts it to nlp object, which can be used
directly to obtain tokens, lemma and pos tags
tokenized_q3_data = q3_data.apply(lambda l: spacy_nlp(l))
print("Listing all document's review IDs and tokens within it\n")
print("+------------------------------------------------------------------------------")
print("\tReview ID,Word,Lemma,POS tag")
for reviewId, review in tokenized_q3_data.iteritems():
    for token in review:
        print("\t", (reviewId + 1), ",", token.orth_, ",", token.lemma_, ",", token.pos_)
print("+------------------------------------------------------------------------------")
```

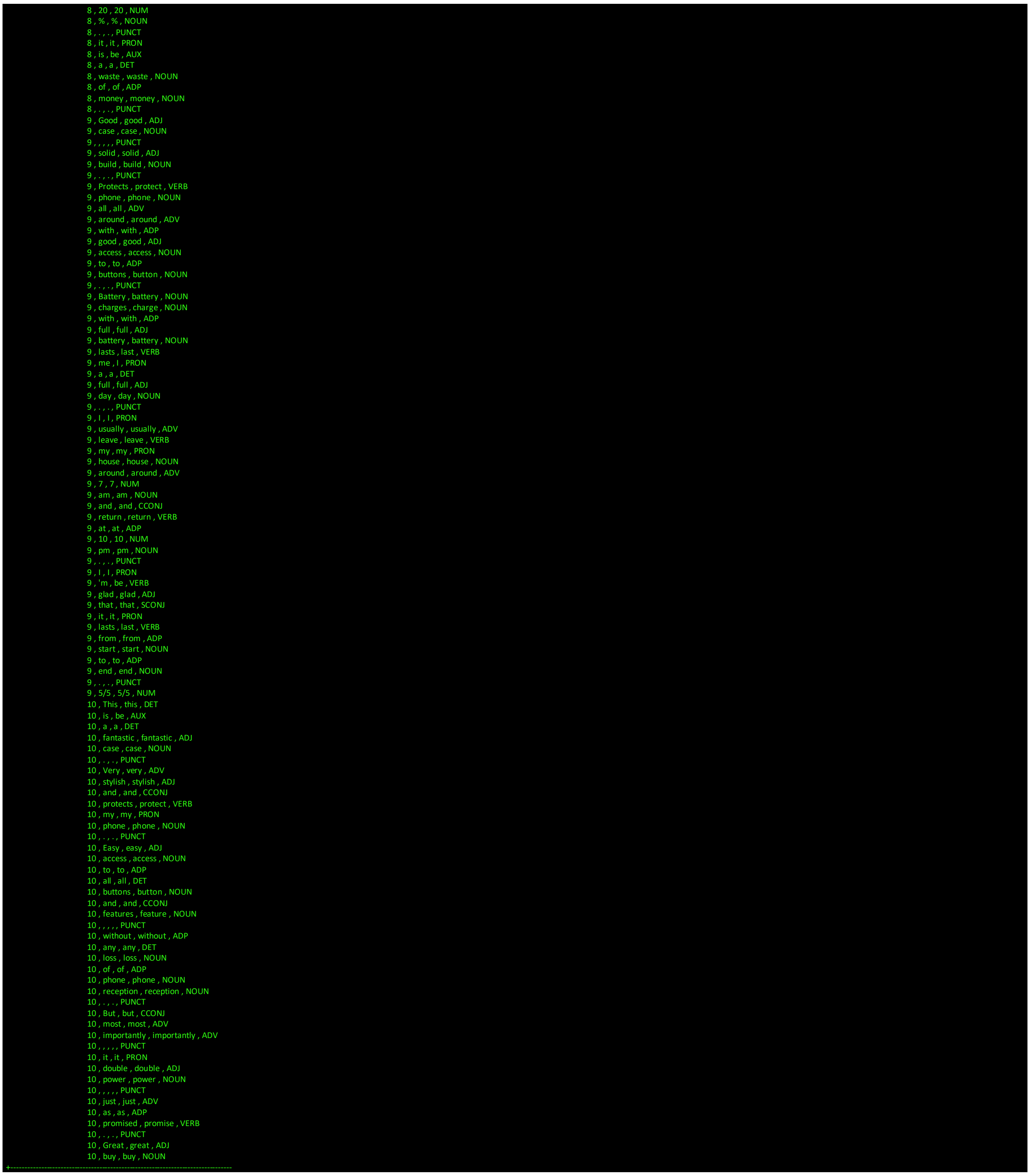
Execution output of q3.py



```
dks@O term2_ta_assignment % python q3.py
Loading data from file...this takes few seconds
Listing all document's review IDs and tokens within it

+--------------------------------------------------------------------------------
                Review ID,Word,Lemma,POS tag
                1 , They , they , PRON
                1 , look , look , VERB
                1 , good , good , ADJ
                1 , and , and , CCONJ
                1 , stick , stick , VERB
                1 , good , good , ADJ
                1 , ! , ! , PUNCT
                1 , I , I , PRON
                1 , just , just , ADV
                1 , do , do , AUX
                1 , n't , n't , PART
                1 , like , like , VERB
                1 , the , the , DET
                1 , rounded , rounded , ADJ
                1 , shape , shape , NOUN
                1 , because , because , SCONJ
                1 , I , I , PRON
                1 , was , be , AUX
                1 , always , always , ADV
                1 , bumping , bump , VERB
                1 , it , it , PRON
                1 , and , and , CCONJ
                1 , Siri , Siri , PROPN
                1 , kept , keep , VERB
                1 , popping , pop , VERB
                1 , up , up , ADP
                1 , and , and , CCONJ
                1 , it , it , PRON
                1 , was , be , AUX
                1 , irritating , irritate , VERB
                1 , . , . , PUNCT
                1 , I , I , PRON
                1 , just , just , ADV
                1 , wo , wo , AUX
                1 , n't , n't , PART
                1 , buy , buy , VERB
                1 , a , a , DET
                1 , product , product , NOUN
                1 , like , like , ADP
                1 , this , this , DET
                1 , again , again , ADV
                2 , These , these , DET
                2 , stickers , sticker , NOUN
                2 , work , work , VERB
                2 , like , like , ADP
                2 , the , the , DET
                2 , review , review , NOUN
                2 , says , say , VERB
                2 , they , they , PRON
                2 , do , do , VERB
                2 , . , . , PUNCT
                2 , They , they , PRON
                2 , stick , stick , VERB
                2 , on , on , ADP
                2 , great , great , ADJ
                2 , and , and , CCONJ
                2 , they , they , PRON
                2 , stay , stay , VERB
                2 , on , on , ADP
                2 , the , the , DET
                2 , phone , phone , NOUN
                2 , . , . , PUNCT
                2 , They , they , PRON
                2 , are , be , AUX
                2 , super , super , ADV
                2 , stylish , stylish , ADJ
                2 , and , and , CCONJ
                2 , I , I , PRON
                2 , can , can , AUX
                2 , share , share , VERB
                2 , them , they , PRON
                2 , with , with , ADP
                2 , my , my , PRON
                2 , sister , sister , NOUN
                2 , . , . , PUNCT
                2 , :) , :) , PUNCT
                3 , These , these , DET
                3 , are , be , AUX
                3 , awesome , awesome , ADJ
                3 , and , and , CCONJ
                3 , make , make , VERB
                3 , my , my , PRON
                3 , phone , phone , NOUN
                3 , look , look , NOUN
                3 , so , so , ADV
                3 , stylish , stylish , ADJ
                3 , ! , ! , PUNCT
                3 , I , I , PRON
                3 , have , have , AUX
                3 , only , only , ADV
                3 , used , use , VERB
                3 , one , one , NUM
                3 , so , so , ADV
                3 , far , far , ADV
                3 , and , and , CCONJ
                3 , have , have , AUX
                3 , had , have , VERB
                3 , it , it , PRON
                3 , on , on , ADP
                3 , for , for , ADP
                3 , almost , almost , ADV
                3 , a , a , DET
                3 , year , year , NOUN
                3 , ! , ! , PUNCT
                3 , CAN , can , AUX
                3 , YOU , you , PRON
                3 , BELIEVE , believe , VERB
                3 , THAT , that , DET
                3 , ! , ! , PUNCT
                3 , ONE , one , NUM
                3 , YEAR , year , NOUN
                3 , ! , ! , PUNCT
                3 , ! , ! , PUNCT
                3 , Great , great , ADJ
                3 , quality , quality , NOUN
                3 , ! , ! , PUNCT
                4 , Item , Item , PROPN
                4 , arrived , arrive , VERB
                4 , in , in , ADP
                4 , great , great , ADJ
                4 , time , time , NOUN
                4 , and , and , CCONJ
                4 , was , be , VERB
                4 , in , in , ADP
                4 , perfect , perfect , ADJ
                4 , condition , condition , NOUN
                4 , . , . , PUNCT
                4 , However , however , ADV
                4 , , , , , PUNCT
                4 , I , I , PRON
                4 , ordered , order , VERB
```



```
4 , these , these , DET
4 , buttons , button , NOUN
4 , because , because , SCONJ
4 , they , they , PRON
4 , were , be , VERB
4 , a , a , DET
4 , great , great , ADJ
4 , deal , deal , NOUN
4 , and , and , CCONJ
4 , included , include , VERB
4 , a , a , DET
4 , FREE , free , ADJ
4 , screen , screen , NOUN
4 , protector , protector , NOUN
4 , . , . , PUNCT
4 , I , I , PRON
4 , never , never , ADV
4 , received , receive , VERB
4 , one , one , NUM
4 , . , . , PUNCT
4 , Though , though , SCONJ
4 , its , its , PRON
4 , not , not , PART
4 , a , a , DET
4 , big , big , ADJ
4 , deal , deal , NOUN
4 , , , , , PUNCT
4 , it , it , PRON
4 , would , would , AUX
4 , 've , 've , AUX
4 , been , be , VERB
4 , nice , nice , ADJ
4 , to , to , PART
4 , get , get , VERB
4 , it , it , PRON
4 , since , since , SCONJ
4 , they , they , PRON
4 , claim , claim , VERB
4 , it , it , PRON
4 , comes , come , VERB
4 , with , with , ADP
4 , one , one , NUM
4 , . , . , PUNCT
5 , awesome , awesome , ADJ
5 , ! , ! , PUNCT
5 , stays , stay , VERB
5 , on , on , ADP
5 , , , , , PUNCT
5 , and , and , CCONJ
5 , looks , look , VERB
5 , great , great , ADJ
5 , . , . , PUNCT
5 , can , can , AUX
5 , be , be , AUX
5 , used , use , VERB
5 , on , on , ADP
5 , multiple , multiple , ADJ
5 , apple , apple , NOUN
5 , products , product , NOUN
5 , . , . , PUNCT
5 ,   ,   , SPACE
5 , especially , especially , ADV
5 , having , have , VERB
5 , nails , nail , NOUN
5 , , , , , PUNCT
5 , it , it , PRON
5 , helps , help , VERB
5 , to , to , PART
5 , have , have , VERB
5 , an , an , DET
5 , elevated , elevated , ADJ
5 , key , key , NOUN
5 , . , . , PUNCT
6 , These , these , DET
6 , make , make , VERB
6 , using , use , VERB
6 , the , the , DET
6 , home , home , NOUN
6 , button , button , NOUN
6 , easy , easy , ADV
6 , . , . , PUNCT
6 , My , my , PRON
6 , daughter , daughter , NOUN
6 , and , and , CCONJ
6 , I , I , PRON
6 , both , both , DET
6 , like , like , ADP
6 , them , they , PRON
6 , . , . , PUNCT
6 ,   ,   , SPACE
6 , I , I , PRON
6 , would , would , AUX
6 , purchase , purchase , VERB
6 , them , they , PRON
6 , again , again , ADV
6 , . , . , PUNCT
6 , Well , well , INTJ
6 , worth , worth , ADJ
6 , the , the , DET
6 , price , price , NOUN
6 , . , . , PUNCT
7 , Came , come , VERB
7 , just , just , ADV
7 , as , as , ADV
7 , described , describe , VERB
7 , .. , .. , PUNCT
7 , It , it , PRON
7 , does , do , AUX
7 , n't , n't , PART
7 , come , come , VERB
7 , unstuck , unstuck , ADJ
7 , and , and , CCONJ
7 , its , its , PRON
7 , cute , cute , ADJ
7 , ! , ! , PUNCT
7 , People , People , NOUN
7 , ask , ask , VERB
7 , where , where , ADV
7 , I , I , PRON
7 , got , get , VERB
7 , them , they , PRON
7 , from , from , ADP
7 , & , & , CCONJ
7 , it , it , PRON
7 , 's , be , VERB
7 , great , great , ADJ
7 , when , when , ADV
7 , driving , drive , VERB
7 , . , . , PUNCT
8 , it , it , PRON
8 , worked , work , VERB
8 , for , for , ADP
8 , the , the , DET
8 , first , first , ADJ
8 , week , week , NOUN
8 , then , then , ADV
8 , it , it , PRON
8 , only , only , ADV
8 , charge , charge , VERB
8 , my , my , PRON
8 , phone , phone , NOUN
8 , to , to , ADP
```



```
            8 , 20 , 20 , NUM
            8 , % , % , NOUN
            8 , . , . , PUNCT
            8 , it , it , PRON
            8 , is , be , AUX
            8 , a , a , DET
            8 , waste , waste , NOUN
            8 , of , of , ADP
            8 , money , money , NOUN
            8 , . , . , PUNCT
            9 , Good , good , ADJ
            9 , case , case , NOUN
            9 , , , , , PUNCT
            9 , solid , solid , ADJ
            9 , build , build , NOUN
            9 , . , . , PUNCT
            9 , Protects , protect , VERB
            9 , phone , phone , NOUN
            9 , all , all , ADV
            9 , around , around , ADV
            9 , with , with , ADP
            9 , good , good , ADJ
            9 , access , access , NOUN
            9 , to , to , ADP
            9 , buttons , button , NOUN
            9 , . , . , PUNCT
            9 , Battery , battery , NOUN
            9 , charges , charge , NOUN
            9 , with , with , ADP
            9 , full , full , ADJ
            9 , battery , battery , NOUN
            9 , lasts , last , VERB
            9 , me , I , PRON
            9 , a , a , DET
            9 , full , full , ADJ
            9 , day , day , NOUN
            9 , . , . , PUNCT
            9 , I , I , PRON
            9 , usually , usually , ADV
            9 , leave , leave , VERB
            9 , my , my , PRON
            9 , house , house , NOUN
            9 , around , around , ADV
            9 , 7 , 7 , NUM
            9 , am , am , NOUN
            9 , and , and , CCONJ
            9 , return , return , VERB
            9 , at , at , ADP
            9 , 10 , 10 , NUM
            9 , pm , pm , NOUN
            9 , . , . , PUNCT
            9 , I , I , PRON
            9 , 'm , be , VERB
            9 , glad , glad , ADJ
            9 , that , that , SCONJ
            9 , it , it , PRON
            9 , lasts , last , VERB
            9 , from , from , ADP
            9 , start , start , NOUN
            9 , to , to , ADP
            9 , end , end , NOUN
            9 , . , . , PUNCT
            9 , 5/5 , 5/5 , NUM
            10 , This , this , DET
            10 , is , be , AUX
            10 , a , a , DET
            10 , fantastic , fantastic , ADJ
            10 , case , case , NOUN
            10 , . , . , PUNCT
            10 , Very , very , ADV
            10 , stylish , stylish , ADJ
            10 , and , and , CCONJ
            10 , protects , protect , VERB
            10 , my , my , PRON
            10 , phone , phone , NOUN
            10 , . , . , PUNCT
            10 , Easy , easy , ADJ
            10 , access , access , NOUN
            10 , to , to , ADP
            10 , all , all , DET
            10 , buttons , button , NOUN
            10 , and , and , CCONJ
            10 , features , feature , NOUN
            10 , , , , , PUNCT
            10 , without , without , ADP
            10 , any , any , DET
            10 , loss , loss , NOUN
            10 , of , of , ADP
            10 , phone , phone , NOUN
            10 , reception , reception , NOUN
            10 , . , . , PUNCT
            10 , But , but , CCONJ
            10 , most , most , ADV
            10 , importantly , importantly , ADV
            10 , , , , , PUNCT
            10 , it , it , PRON
            10 , double , double , ADJ
            10 , power , power , NOUN
            10 , , , , , PUNCT
            10 , just , just , ADV
            10 , as , as , ADP
            10 , promised , promise , VERB
            10 , . , . , PUNCT
            10 , Great , great , ADJ
            10 , buy , buy , NOUN
+------------------------------------------------------------------------------
```

Q5.

Take the first 1000 review texts. Using gensim, create an LDA model with 10 topics. Report the top 50 words with probs for each of the ten topics. Each line has topic number, word, prob in that topic.

Solution of Q5

Code file: q4.py

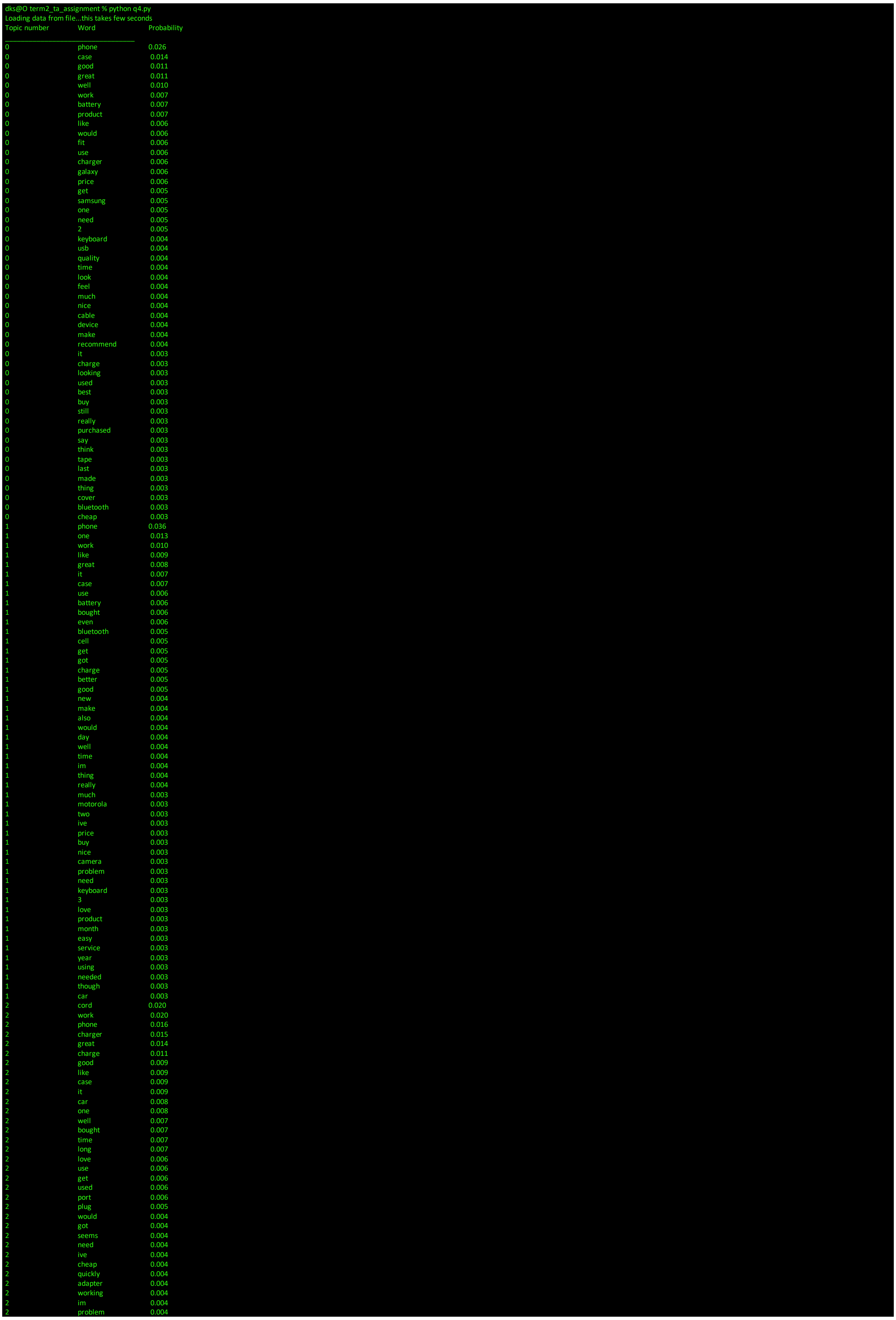
Code notes

Spacy library is used for sentence extraction from paragraph, for this to work we must preload language pack for "English" - *en_core_web_sm.*

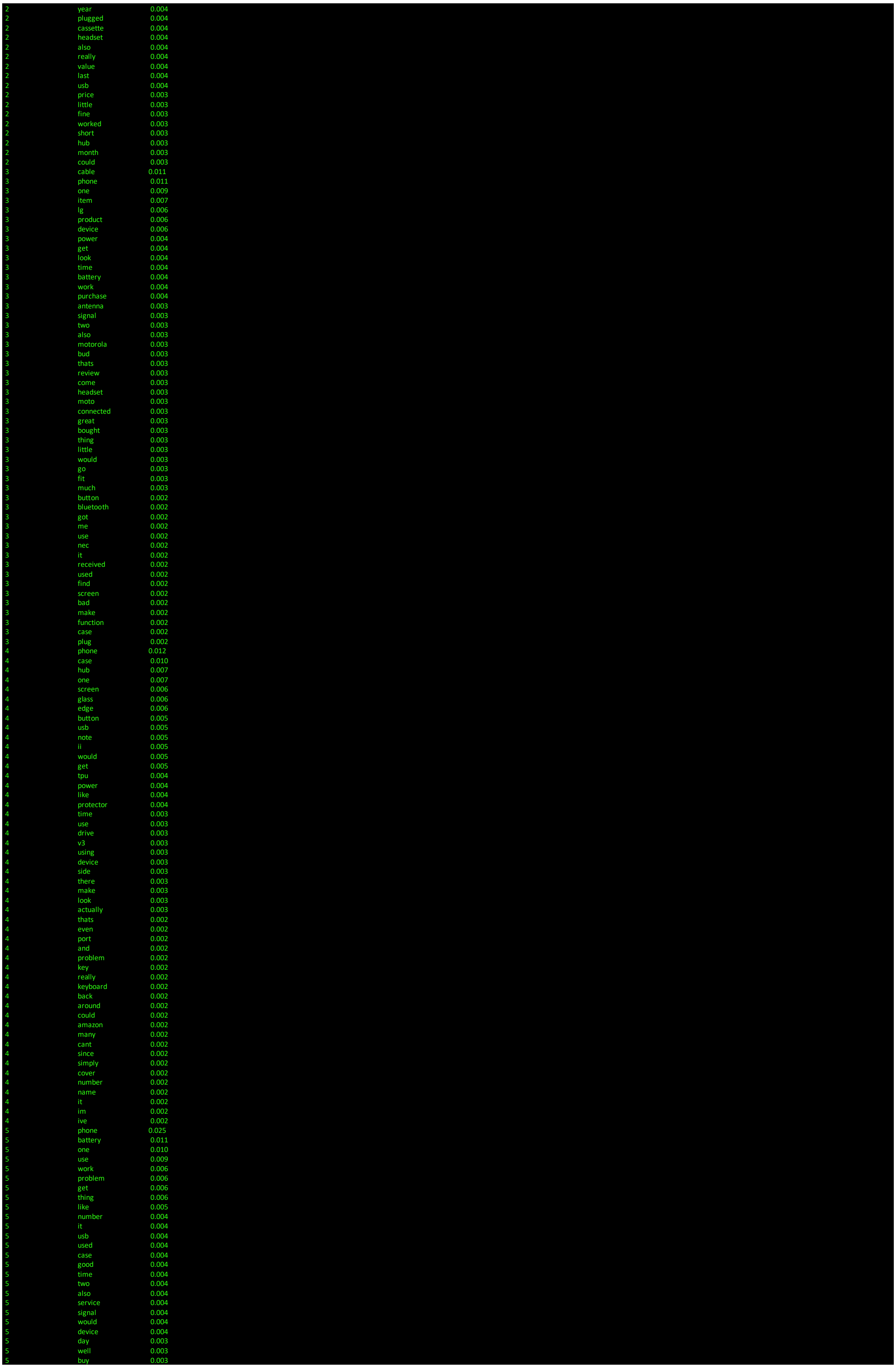
Code Snippet

```
print("Now loading the data - first 1000 review texts")
# Loading the data
q4_data = ReviewDataLoader().get_review_data()['reviewText'].head(1000)
print("Now cleaning the data - removing stopwords, punctuations and normalizing, further splitting by space and
converting it to list")
doc_clean = q4_data.apply(clean).str.split().tolist()

# Creating the term dictionary of our corpus, where every unique term is assigned an index.
dictionary = corpora.Dictionary(doc_clean)

# Converting list of documents (corpus) into Document Term Matrix using dictionary prepared above.
doc_term_matrix = [dictionary.doc2bow(doc) for doc in doc_clean]

# Running LDA Model
# Creating the object for LDA model using gensim library
Lda = gensim.models.ldamodel.LdaModel

# Running and Training LDA model on the document term matrix.
ldamodel = Lda(doc_term_matrix, num_topics=10, id2word=dictionary, passes=50)

topic_line = ldamodel.show_topics(num_topics=10, num_words=50)

print("Topic number \t Word \t Probability")
print("_________________________________")
for aLine in topic_line:
    id, word_list = aLine
    words = word_list.split("+")
    for aWord in words:
        print(id, '\t', aWord.split("*")[1].replace('"', '').strip(), '\t', aWord.split("*")[0])
print("_________________________________")
```

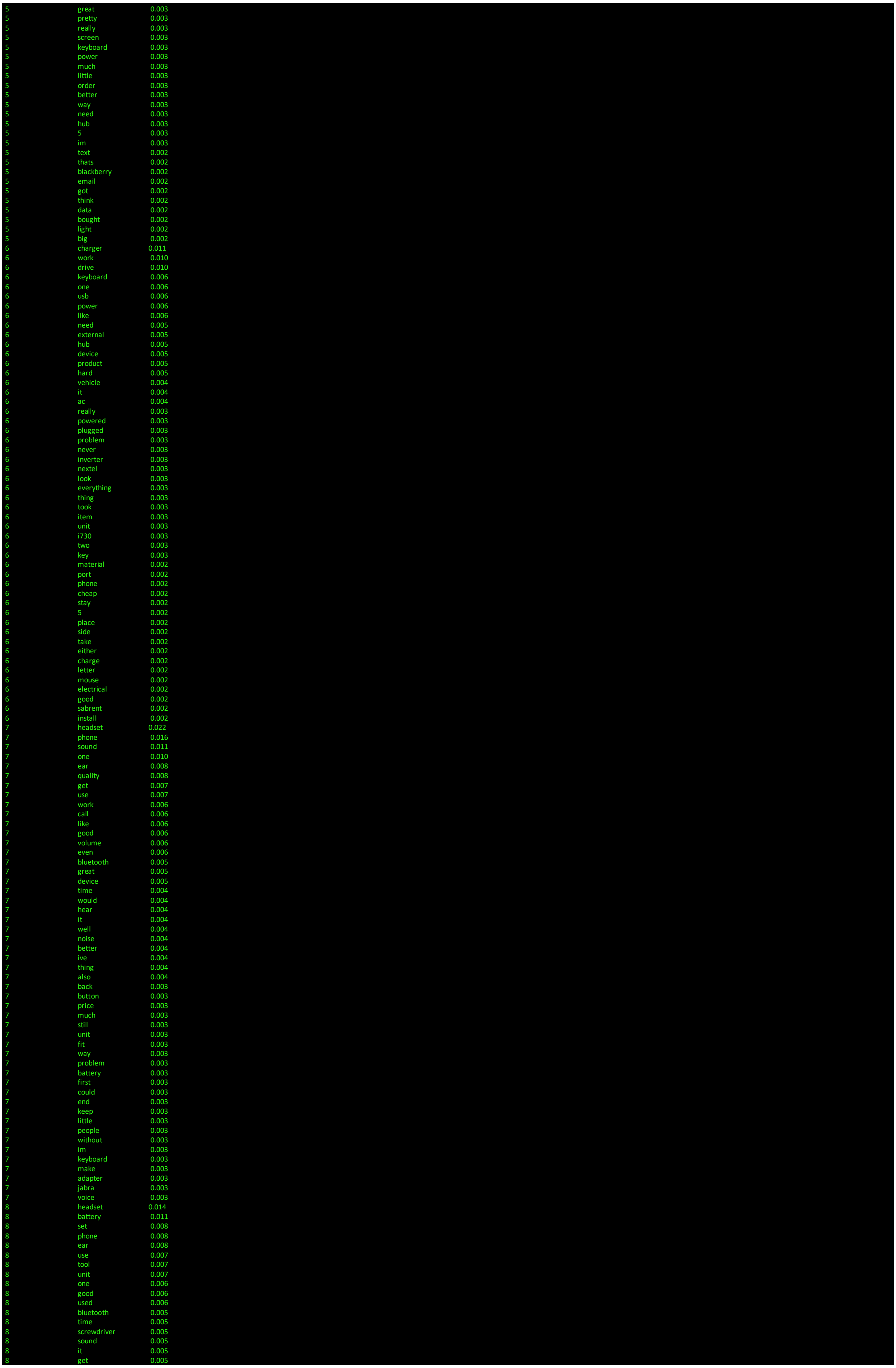
Execution output of q4.py



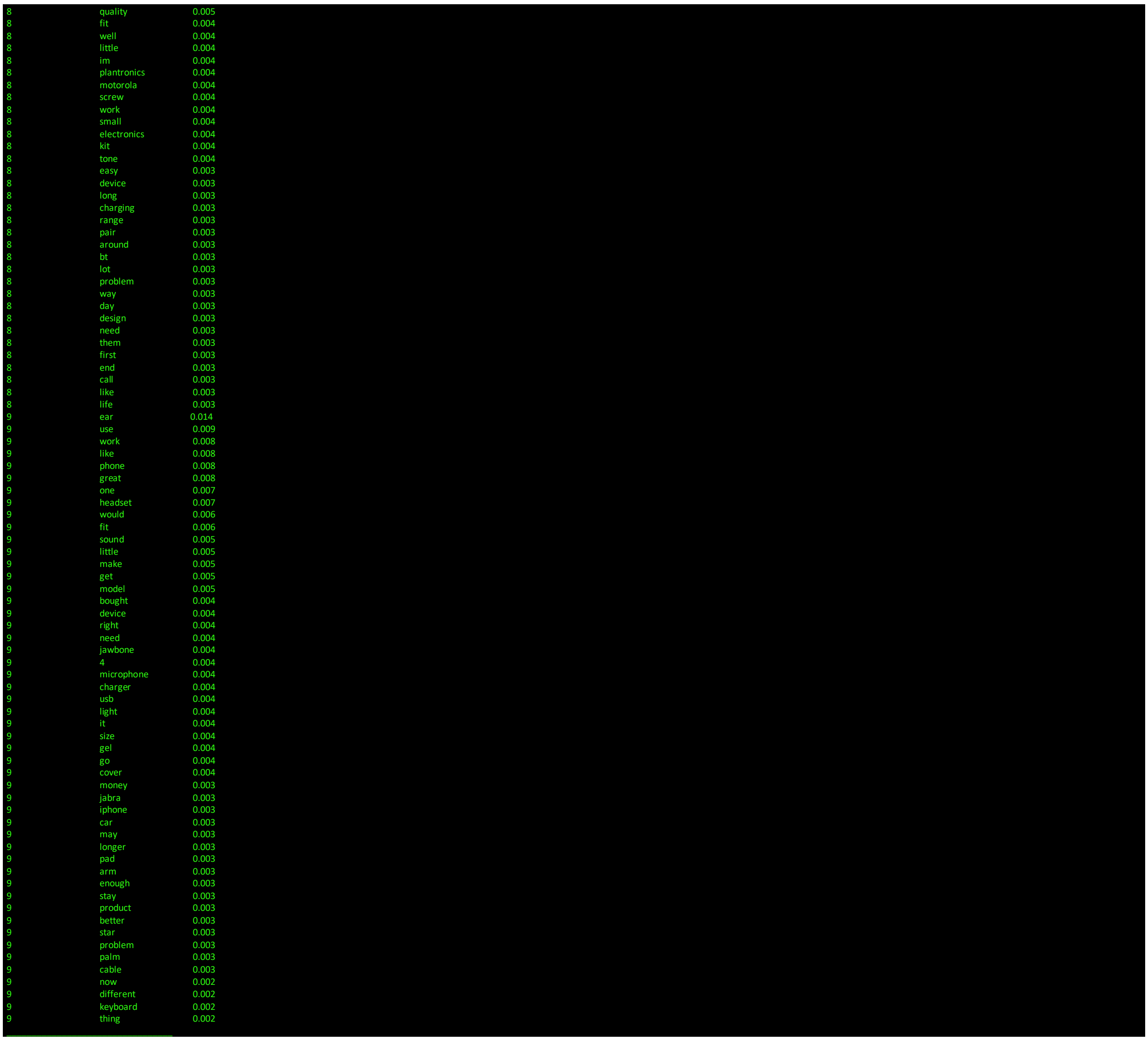
```
dks@O term2_ta_assignment % python q4.py
Loading data from file...this takes few seconds
Topic number          Word              Probability

____________________________________
0                     phone             0.026
0                     case               0.014
0                     good               0.011
0                     great              0.011
0                     well               0.010
0                     work               0.007
0                     battery            0.007
0                     product            0.007
0                     like               0.006
0                     would              0.006
0                     fit                0.006
0                     use                0.006
0                     charger            0.006
0                     galaxy             0.006
0                     price              0.006
0                     get                0.005
0                     samsung            0.005
0                     one                0.005
0                     need               0.005
0                     2                  0.005
0                     keyboard           0.004
0                     usb                0.004
0                     quality            0.004
0                     time               0.004
0                     look               0.004
0                     feel               0.004
0                     much               0.004
0                     nice               0.004
0                     cable              0.004
0                     device             0.004
0                     make               0.004
0                     recommend          0.004
0                     it                 0.003
0                     charge             0.003
0                     looking            0.003
0                     used               0.003
0                     best               0.003
0                     buy                0.003
0                     still              0.003
0                     really             0.003
0                     purchased          0.003
0                     say                0.003
0                     think              0.003
0                     tape               0.003
0                     last               0.003
0                     made               0.003
0                     thing              0.003
0                     cover              0.003
0                     bluetooth          0.003
0                     cheap              0.003
1                     phone             0.036
1                     one                0.013
1                     work               0.010
1                     like               0.009
1                     great              0.008
1                     it                 0.007
1                     case               0.007
1                     use                0.006
1                     battery            0.006
1                     bought             0.006
1                     even               0.006
1                     bluetooth          0.005
1                     cell               0.005
1                     get                0.005
1                     got                0.005
1                     charge             0.005
1                     better             0.005
1                     good               0.005
1                     new                0.004
1                     make               0.004
1                     also               0.004
1                     would              0.004
1                     day                0.004
1                     well               0.004
1                     time               0.004
1                     im                 0.004
1                     thing              0.004
1                     really             0.004
1                     much               0.003
1                     motorola           0.003
1                     two                0.003
1                     ive                0.003
1                     price              0.003
1                     buy                0.003
1                     nice               0.003
1                     camera             0.003
1                     problem            0.003
1                     need               0.003
1                     keyboard           0.003
1                     3                  0.003
1                     love               0.003
1                     product            0.003
1                     month              0.003
1                     easy               0.003
1                     service            0.003
1                     year               0.003
1                     using              0.003
1                     needed             0.003
1                     though             0.003
1                     car                0.003
2                     cord              0.020
2                     work               0.020
2                     phone              0.016
2                     charger            0.015
2                     great              0.014
2                     charge             0.011
2                     good               0.009
2                     like               0.009
2                     case               0.009
2                     it                 0.009
2                     car                0.008
2                     one                0.008
2                     well               0.007
2                     bought             0.007
2                     time               0.007
2                     long               0.007
2                     love               0.006
2                     use                0.006
2                     get                0.006
2                     used               0.006
2                     port               0.006
2                     plug               0.005
2                     would              0.004
2                     got                0.004
2                     seems              0.004
2                     need               0.004
2                     ive                0.004
2                     cheap              0.004
2                     quickly            0.004
2                     adapter            0.004
2                     working            0.004
2                     im                 0.004
2                     problem            0.004
```



2	year	0.004
2	plugged	0.004
2	cassette	0.004
2	headset	0.004
2	also	0.004
2	really	0.004
2	value	0.004
2	last	0.004
2	usb	0.004
2	price	0.003
2	little	0.003
2	fine	0.003
2	worked	0.003
2	short	0.003
2	hub	0.003
2	month	0.003
2	could	0.003
3	cable	0.011
3	phone	0.011
3	one	0.009
3	item	0.007
3	lg	0.006
3	product	0.006
3	device	0.006
3	power	0.004
3	get	0.004
3	look	0.004
3	time	0.004
3	battery	0.004
3	work	0.004
3	purchase	0.004
3	antenna	0.003
3	signal	0.003
3	two	0.003
3	also	0.003
3	motorola	0.003
3	bud	0.003
3	thats	0.003
3	review	0.003
3	come	0.003
3	headset	0.003
3	moto	0.003
3	connected	0.003
3	great	0.003
3	bought	0.003
3	thing	0.003
3	little	0.003
3	would	0.003
3	go	0.003
3	fit	0.003
3	much	0.003
3	button	0.002
3	bluetooth	0.002
3	got	0.002
3	me	0.002
3	use	0.002
3	nec	0.002
3	it	0.002
3	received	0.002
3	used	0.002
3	find	0.002
3	screen	0.002
3	bad	0.002
3	make	0.002
3	function	0.002
3	case	0.002
3	plug	0.002
4	phone	0.012
4	case	0.010
4	hub	0.007
4	one	0.007
4	screen	0.006
4	glass	0.006
4	edge	0.006
4	button	0.005
4	usb	0.005
4	note	0.005
4	ii	0.005
4	would	0.005
4	get	0.005
4	tpu	0.004
4	power	0.004
4	like	0.004
4	protector	0.004
4	time	0.003
4	use	0.003
4	drive	0.003
4	v3	0.003
4	using	0.003
4	device	0.003
4	side	0.003
4	there	0.003
4	make	0.003
4	look	0.003
4	actually	0.003
4	thats	0.002
4	even	0.002
4	port	0.002
4	and	0.002
4	problem	0.002
4	key	0.002
4	really	0.002
4	keyboard	0.002
4	back	0.002
4	around	0.002
4	could	0.002
4	amazon	0.002
4	many	0.002
4	cant	0.002
4	since	0.002
4	simply	0.002
4	cover	0.002
4	number	0.002
4	name	0.002
4	it	0.002
4	im	0.002
4	ive	0.002
5	phone	0.025
5	battery	0.011
5	one	0.010
5	use	0.009
5	work	0.006
5	problem	0.006
5	get	0.006
5	thing	0.006
5	like	0.005
5	number	0.004
5	it	0.004
5	usb	0.004
5	used	0.004
5	case	0.004
5	good	0.004
5	time	0.004
5	two	0.004
5	also	0.004
5	service	0.004
5	signal	0.004
5	would	0.004
5	device	0.004
5	day	0.003
5	well	0.003
5	buy	0.003



5	great	0.003
5	pretty	0.003
5	really	0.003
5	screen	0.003
5	keyboard	0.003
5	power	0.003
5	much	0.003
5	little	0.003
5	order	0.003
5	better	0.003
5	way	0.003
5	need	0.003
5	hub	0.003
5	5	0.003
5	im	0.003
5	text	0.002
5	thats	0.002
5	blackberry	0.002
5	email	0.002
5	got	0.002
5	think	0.002
5	data	0.002
5	bought	0.002
5	light	0.002
5	big	0.002
6	charger	0.011
6	work	0.010
6	drive	0.010
6	keyboard	0.006
6	one	0.006
6	usb	0.006
6	power	0.006
6	like	0.006
6	need	0.005
6	external	0.005
6	hub	0.005
6	device	0.005
6	product	0.005
6	hard	0.005
6	vehicle	0.004
6	it	0.004
6	ac	0.004
6	really	0.003
6	powered	0.003
6	plugged	0.003
6	problem	0.003
6	never	0.003
6	inverter	0.003
6	nextel	0.003
6	look	0.003
6	everything	0.003
6	thing	0.003
6	took	0.003
6	item	0.003
6	unit	0.003
6	i730	0.003
6	two	0.003
6	key	0.003
6	material	0.002
6	port	0.002
6	phone	0.002
6	cheap	0.002
6	stay	0.002
6	5	0.002
6	place	0.002
6	side	0.002
6	take	0.002
6	either	0.002
6	charge	0.002
6	letter	0.002
6	mouse	0.002
6	electrical	0.002
6	good	0.002
6	sabrent	0.002
6	install	0.002
7	headset	0.022
7	phone	0.016
7	sound	0.011
7	one	0.010
7	ear	0.008
7	quality	0.008
7	get	0.007
7	use	0.007
7	work	0.006
7	call	0.006
7	like	0.006
7	good	0.006
7	volume	0.006
7	even	0.006
7	bluetooth	0.005
7	great	0.005
7	device	0.005
7	time	0.004
7	would	0.004
7	hear	0.004
7	it	0.004
7	well	0.004
7	noise	0.004
7	better	0.004
7	ive	0.004
7	thing	0.004
7	also	0.004
7	back	0.003
7	button	0.003
7	price	0.003
7	much	0.003
7	still	0.003
7	unit	0.003
7	fit	0.003
7	way	0.003
7	problem	0.003
7	battery	0.003
7	first	0.003
7	could	0.003
7	end	0.003
7	keep	0.003
7	little	0.003
7	people	0.003
7	without	0.003
7	im	0.003
7	keyboard	0.003
7	make	0.003
7	adapter	0.003
7	jabra	0.003
7	voice	0.003
8	headset	0.014
8	battery	0.011
8	set	0.008
8	phone	0.008
8	ear	0.008
8	use	0.007
8	tool	0.007
8	unit	0.007
8	one	0.006
8	good	0.006
8	used	0.006
8	bluetooth	0.005
8	time	0.005
8	screwdriver	0.005
8	sound	0.005
8	it	0.005
8	get	0.005



```
8            quality        0.005
8            fit            0.004
8            well           0.004
8            little         0.004
8            im             0.004
8            plantronics    0.004
8            motorola       0.004
8            screw          0.004
8            work           0.004
8            small          0.004
8            electronics    0.004
8            kit            0.004
8            tone           0.004
8            easy           0.003
8            device         0.003
8            long           0.003
8            charging       0.003
8            range          0.003
8            pair           0.003
8            around         0.003
8            bt             0.003
8            lot            0.003
8            problem        0.003
8            way            0.003
8            day            0.003
8            design         0.003
8            need           0.003
8            them           0.003
8            first          0.003
8            end            0.003
8            call           0.003
8            like           0.003
8            life           0.003
9            ear           0.014
9            use            0.009
9            work           0.008
9            like           0.008
9            phone          0.008
9            great          0.008
9            one            0.007
9            headset        0.007
9            would          0.006
9            fit            0.006
9            sound          0.005
9            little         0.005
9            make           0.005
9            get            0.005
9            model          0.005
9            bought         0.004
9            device         0.004
9            right          0.004
9            need           0.004
9            jawbone        0.004
9            4              0.004
9            microphone     0.004
9            charger        0.004
9            usb            0.004
9            light          0.004
9            it             0.004
9            size           0.004
9            gel            0.004
9            go             0.004
9            cover          0.004
9            money          0.003
9            jabra          0.003
9            iphone         0.003
9            car            0.003
9            may            0.003
9            longer         0.003
9            pad            0.003
9            arm            0.003
9            enough         0.003
9            stay           0.003
9            product        0.003
9            better         0.003
9            star           0.003
9            problem        0.003
9            palm           0.003
9            cable          0.003
9            now            0.002
9            different      0.002
9            keyboard       0.002
9            thing          0.002
```

Q6.

Use the entire dataset. Take the first 80% dataset for train and remaining 20% for test. On the train set, obtain TFIDF features (with 50K vocabulary) and learn a multinomial Naïve Bayes model. Report the accuracy on the test set for this five-class classification problem. Accuracy should be reported as class-wise precision, recall and F1

Solution of Q6

Code file: q5.py

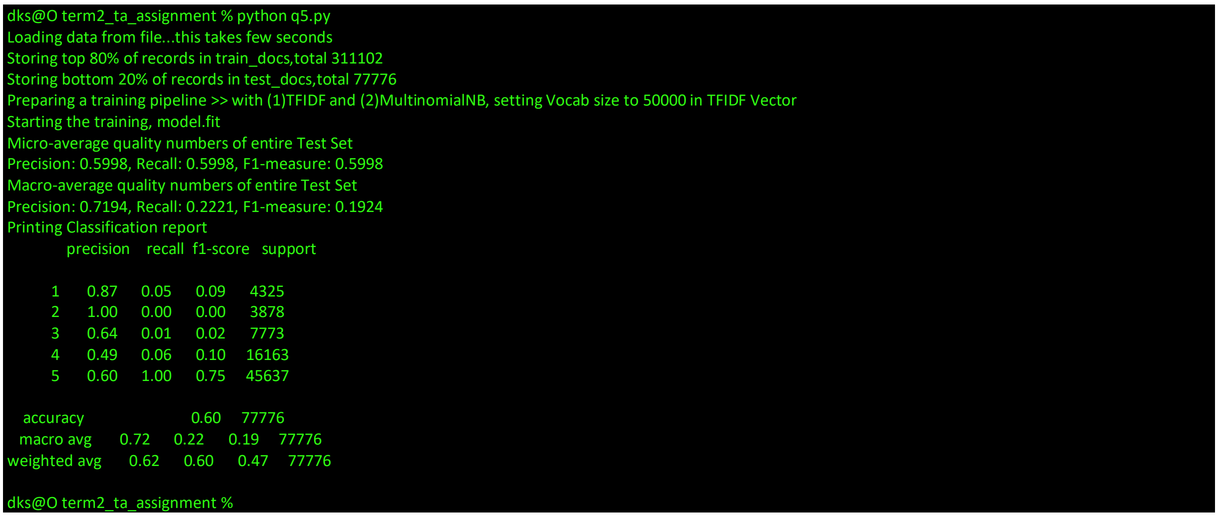
Code notes

Here nltk is used for data cleaning, pandas as primary data structure, sklearn's TFIDFVectorizer and MultinomialNB (Multinomial Naive Bayes modelling) are used to be executed in pipeline and fit the model with training data, the training data contains Review Text as data and Rating (in field Overall) as label, it trains on 80% of total data set and remaining 20% is used for testing the model, at the end it shows precision, recall and f1 score of each class for predicted labels

Code Snippet

```
def five_class_classifier(train_docs, test_docs):
    print("Preparing a training pipeline >> with (1)TFIDF and (2)MultinomialNB, "
          "setting Vocab size to 50000 in TFIDF Vector")
    model = make_pipeline(TfidfVectorizer(stop_words=stop_words, max_features=50000), MultinomialNB())
    # training the model with TFIDF and MultinomialNB pipeline
    print("Starting the training, model.fit")
    model.fit(train_docs['reviewText'], train_docs['overall'])
    worked_labels = model.predict(test_docs['reviewText'])
    # for "micro"-averaging in a multiclass setting with all labels included will produce equal precision, recall
and F,
    # while "weighted" averaging may produce an F-score that is not between precision and recall.(1)
    precision = precision_score(test_docs['overall'], worked_labels, average='micro')
    recall = recall_score(test_docs['overall'], worked_labels, average='micro')
    f1 = f1_score(test_docs['overall'], worked_labels, average='micro')
    print("Micro-average quality numbers of entire Test Set")
    print("Precision: {:.4f}, Recall: {:.4f}, F1-measure: {:.4f}".format(precision, recall, f1))
    precision = precision_score(test_docs['overall'], worked_labels, average='macro')
    recall = recall_score(test_docs['overall'], worked_labels, average='macro')
    f1 = f1_score(test_docs['overall'], worked_labels, average='macro')
    print("Macro-average quality numbers of entire Test Set")
    print("Precision: {:.4f}, Recall: {:.4f}, F1-measure: {:.4f}".format(precision, recall, f1))
    print("Printing Classification report\n", classification_report(test_docs['overall'], worked_labels))
```

Execution output of q5.py



```
dks@O term2_ta_assignment % python q5.py
Loading data from file...this takes few seconds
Storing top 80% of records in train_docs,total 311102
Storing bottom 20% of records in test_docs,total 77776
Preparing a training pipeline >> with (1)TFIDF and (2)MultinomialNB, setting Vocab size to 50000 in TFIDF Vector
Starting the training, model.fit
Micro-average quality numbers of entire Test Set
Precision: 0.5998, Recall: 0.5998, F1-measure: 0.5998
Macro-average quality numbers of entire Test Set
Precision: 0.7194, Recall: 0.2221, F1-measure: 0.1924
Printing Classification report
          precision    recall  f1-score   support

       1      0.87      0.05      0.09      4325
       2      1.00      0.00      0.00      3878
       3      0.64      0.01      0.02      7773
       4      0.49      0.06      0.10     16163
       5      0.60      1.00      0.75     45637

   accuracy                         0.60     77776
  macro avg       0.72      0.22      0.19     77776
weighted avg       0.62      0.60      0.47     77776

dks@O term2_ta_assignment %
```

Q7.

Take the first 1000 “rating-1.0” reviews. Summarize them to 1% (in terms of words) using gensim and send across your summary. Also, take the first 1000 “rating-5.0” reviews. Summarize them to approximately 300 words using gensim and send across your summary.

Solution of Q7

Code file: q6.py

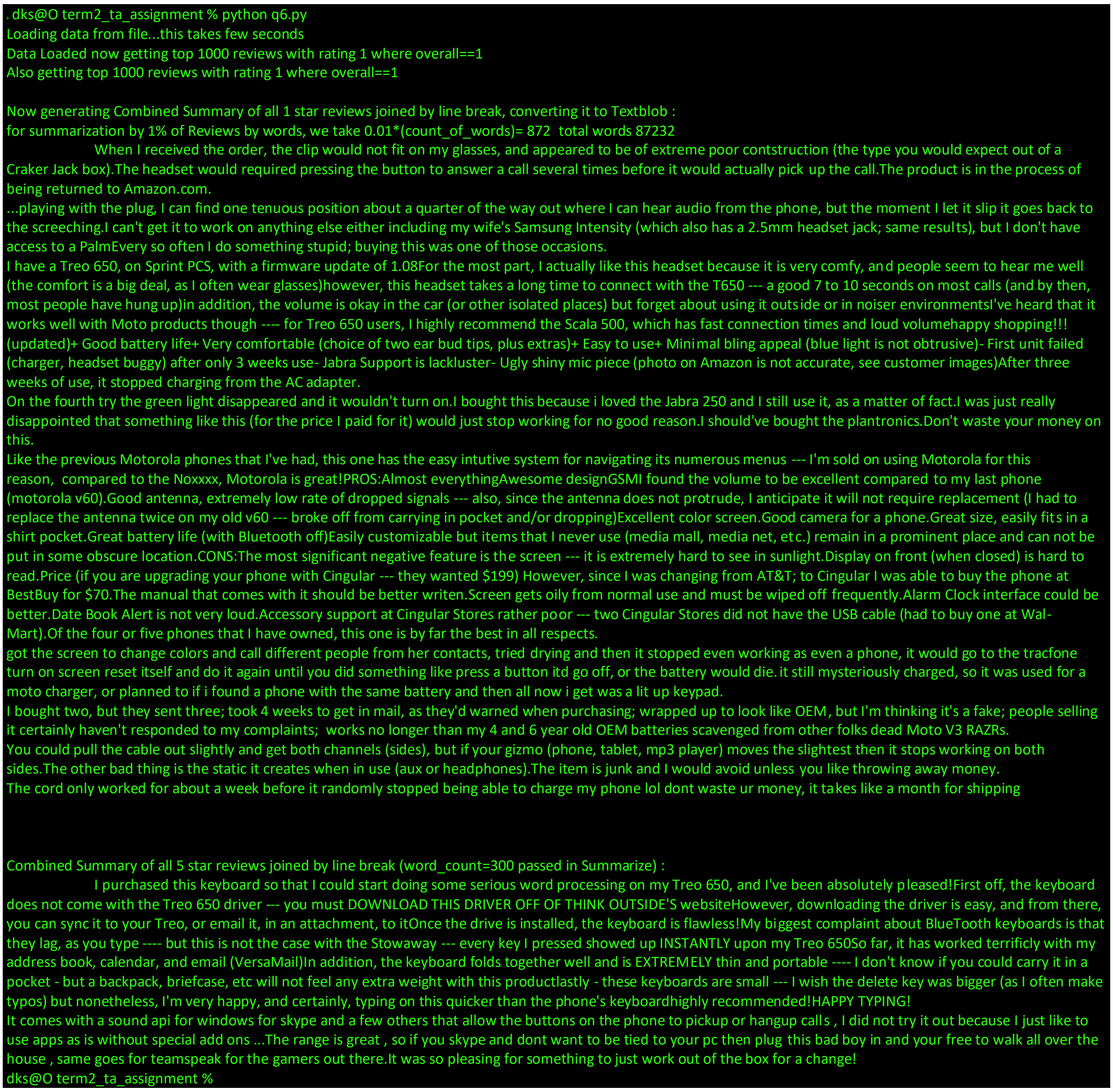
Code notes

gensim.summarization.summarizer is used for text summarization, and TextBlob just for quick word count, it loads top 1000 rating 1 and rating 5 reviews, and uses summarize method to generate summary

Code Snippet

```
from gensim.summarization.summarizer import summarize
# using textblob for word count
from textblob import TextBlob as tb

print("Loading data from file...this takes few seconds")
from q0 import ReviewDataLoader

q6_data = ReviewDataLoader().get_review_data()[['overall', 'reviewText']]

print("Data Loaded now getting top 1000 reviews with rating 1 where overall==1")

# getting top 1000 reviews with rating 1 where overall==1
rating1_texts = q6_data['reviewText'][ReviewDataLoader().get_review_data()['overall'] == 1].head(1000)

print("Also getting top 1000 reviews with rating 1 where overall==1")
# getting top 1000 reviews with rating 5 where overall==5
rating5_texts = q6_data['reviewText'][ReviewDataLoader().get_review_data()['overall'] == 5].head(1000)
print('\nNow generating Combined Summary of all 1 star reviews joined by line break, converting it to Textblob :')
rating1 = "\n".join(rating1_texts.tolist())
count_of_words = len(tb(rating1).words)
print("for summarization by 1% of Reviews by words, we take 0.01*(count_of_words)=",
      int(0.01 * count_of_words), " total words", count_of_words)

print("\t", summarize(rating1, word_count=int(len(tb(rating1).words) * 0.01)))

print('\n\n\nCombined Summary of all 5 star reviews joined by line break (word_count=300 passed in Summarize) :')
ft5 = "\n".join(rating5_texts.tolist())
print("\t", summarize(ft5, word_count=300))
```

Execution output of q6.py



```
dks@O term2_ta_assignment % python q6.py
Loading data from file...this takes few seconds
Data Loaded now getting top 1000 reviews with rating 1 where overall==1
Also getting top 1000 reviews with rating 1 where overall==1

Now generating Combined Summary of all 1 star reviews joined by line break, converting it to Textblob :
for summarization by 1% of Reviews by words, we take 0.01*(count_of_words)= 872  total words 87232
                When I received the order, the clip would not fit on my glasses, and appeared to be of extreme poor contstruction (the type you would expect out of a
Craker Jack box).The headset would required pressing the button to answer a call several times before it would actually pick up the call.The product is in the process of
being returned to Amazon.com.
...playing with the plug, I can find one tenuous position about a quarter of the way out where I can hear audio from the phone, but the moment I let it slip it goes back to
the screeching.I can't get it to work on anything else either including my wife's Samsung Intensity (which also has a 2.5mm headset jack; same results), but I don't have
access to a PalmEvery so often I do something stupid; buying this was one of those occasions.
I have a Treo 650, on Sprint PCS, with a firmware update of 1.08For the most part, I actually like this headset because it is very comfy, and people seem to hear me well
(the comfort is a big deal, as I often wear glasses)however, this headset takes a long time to connect with the T650 --- a good 7 to 10 seconds on most calls (and by then,
most people have hung up)in addition, the volume is okay in the car (or other isolated places) but forget about using it outside or in noiser environmentsI've heard that it
works well with Moto products though ---- for Treo 650 users, I highly recommend the Scala 500, which has fast connection times and loud volumehappy shopping!!!
(updated)+ Good battery life+ Very comfortable (choice of two ear bud tips, plus extras)+ Easy to use+ Minimal bling appeal (blue light is not obtrusive)- First unit failed
(charger, headset buggy) after only 3 weeks use- Jabra Support is lackluster- Ugly shiny mic piece (photo on Amazon is not accurate, see customer images)After three
weeks of use, it stopped charging from the AC adapter.
On the fourth try the green light disappeared and it wouldn't turn on.I bought this because i loved the Jabra 250 and I still use it, as a matter of fact.I was just really
disappointed that something like this (for the price I paid for it) would just stop working for no good reason.I should've bought the plantronics.Don't waste your money on
this.
Like the previous Motorola phones that I've had, this one has the easy intutive system for navigating its numerous menus --- I'm sold on using Motorola for this
reason,  compared to the Noxxxx, Motorola is great!PROS:Almost everythingAwesome designGSMI found the volume to be excellent compared to my last phone
(motorola v60).Good antenna, extremely low rate of dropped signals --- also, since the antenna does not protrude, I anticipate it will not require replacement (I had to
replace the antenna twice on my old v60 --- broke off from carrying in pocket and/or dropping)Excellent color screen.Good camera for a phone.Great size, easily fits in a
shirt pocket.Great battery life (with Bluetooth off)Easily customizable but items that I never use (media mall, media net, etc.) remain in a prominent place and can not be
put in some obscure location.CONS:The most significant negative feature is the screen --- it is extremely hard to see in sunlight.Display on front (when closed) is hard to
read.Price (if you are upgrading your phone with Cingular --- they wanted $199) However, since I was changing from AT&T; to Cingular I was able to buy the phone at
BestBuy for $70.The manual that comes with it should be better writen.Screen gets oily from normal use and must be wiped off frequently.Alarm Clock interface could be
better.Date Book Alert is not very loud.Accessory support at Cingular Stores rather poor --- two Cingular Stores did not have the USB cable (had to buy one at Wal-
Mart).Of the four or five phones that I have owned, this one is by far the best in all respects.
got the screen to change colors and call different people from her contacts, tried drying and then it stopped even working as even a phone, it would go to the tracfone
turn on screen reset itself and do it again until you did something like press a button itd go off, or the battery would die. it still mysteriously charged, so it was used for a
moto charger, or planned to if i found a phone with the same battery and then all now i get was a lit up keypad.
I bought two, but they sent three; took 4 weeks to get in mail, as they'd warned when purchasing; wrapped up to look like OEM, but I'm thinking it's a fake; people selling
it certainly haven't responded to my complaints;  works no longer than my 4 and 6 year old OEM batteries scavenged from other folks dead Moto V3 RAZRs.
You could pull the cable out slightly and get both channels (sides), but if your gizmo (phone, tablet, mp3 player) moves the slightest then it stops working on both
sides.The other bad thing is the static it creates when in use (aux or headphones).The item is junk and I would avoid unless you like throwing away money.
The cord only worked for about a week before it randomly stopped being able to charge my phone lol dont waste ur money, it takes like a month for shipping



Combined Summary of all 5 star reviews joined by line break (word_count=300 passed in Summarize) :
                I purchased this keyboard so that I could start doing some serious word processing on my Treo 650, and I've been absolutely pleased!First off, the keyboard
does not come with the Treo 650 driver --- you must DOWNLOAD THIS DRIVER OFF OF THINK OUTSIDE'S websiteHowever, downloading the driver is easy, and from there,
you can sync it to your Treo, or email it, in an attachment, to itOnce the drive is installed, the keyboard is flawless!My biggest complaint about BlueTooth keyboards is that
they lag, as you type ---- but this is not the case with the Stowaway --- every key I pressed showed up INSTANTLY upon my Treo 650So far, it has worked terrificly with my
address book, calendar, and email (VersaMail)In addition, the keyboard folds together well and is EXTREMELY thin and portable ---- I don't know if you could carry it in a
pocket - but a backpack, briefcase, etc will not feel any extra weight with this productlastly - these keyboards are small --- I wish the delete key was bigger (as I often make
typos) but nonetheless, I'm very happy, and certainly, typing on this quicker than the phone's keyboardhighly recommended!HAPPY TYPING!
It comes with a sound api for windows for skype and a few others that allow the buttons on the phone to pickup or hangup calls , I did not try it out because I just like to
use apps as is without special add ons ...The range is great , so if you skype and dont want to be tied to your pc then plug this bad boy in and your free to walk all over the
house , same goes for teamspeak for the gamers out there.It was so pleasing for something to just work out of the box for a change!
dks@O term2_ta_assignment %
```