1. **Run the main program. You should end up with a file with the sentiment outputs. How many positive sentiment lines do you have and how many negative sentiment lines?**  
   Positive Sentiments Lines: 306

Negative Sentiment Lines: 82

1. **Run the tests. Do the tests fail or pass?**

6 out of 10 tests failed

1. **Find out and list all the tests that fail**

for j in range(Nsample):

assert get\_sentiment(pos\_words[j] + ' ' + neg\_words[j])[0] == 'Neutral'

assert POS\_LEX & NEG\_LEX is None

assert get\_sentiment('abcdefghijklmnop')[0] == 'Neutral'

assert get\_sentiment('')[0] == 'Neutral'

assert get\_sentiment('good ' \* 100 + ' bad')[0] == 'Positive'

assert get\_sentiment('bad ' \* 100 + ' good')[0] == 'Negative'

1. **Give at least one example for each of the following and mention why it happens:**
2. **Text with positive sentiment is correctly reported as positive**

**Ans: Because of presence of positive lexicons**

**Diff>0**

excellent phone , excellent service .

['excellent']

[]

Positive

1. **Text with negative sentiment is incorrectly reported as positive**

**Ans: Because the number of positive lexicons is greater than the number of negative lexicons**

**Diff>0**

the headset that comes with the phone has good sound volume but it hurts the ears like you cannot imagine !

['good', 'like']

['hurts']

Positive

the phone comes with okay ringtones , some decent backgrounds / screensavers , but the phone has very little memory ( mine had 230kb as it arrived from amazon , so you do n't have too many options on what you can put on there ) .

['decent']

[]

Positive

1. **Text with positive sentiment is incorrectly reported as neutral**

**Ans: Because there are no positive and negative lexicons detected from positive.txt and negative.txt**

**Diff=0**

the speaker phone is very functional and i use it in the car , very audible even with freeway noise .

['audible']

['noise']

Netural

1. **Note: use space before punctuations for the test sentences**

**"This is good." does not get picked up because "good." is not a word here.**

*assert get\_sentiment('This is good.')[0]=='Positive'*

*File "../Sentiment Analysis.py", line 85, in <module>*

*run\_tests()*

*File ".. /Sentiment Analysis.py", line 73, in run\_tests*

*assert get\_sentiment('This is good.')[0]=='Positive'*

*AssertionError*

**"This is good ." will be fine for tests (but will get you bad grades in any English or any other class)**

*assert get\_sentiment('This is good .')[0]=='Positive'*

1. **There is a major bug in the code. Can you find it by running some test cases (hint: we discussed it in the class)?**
2. **Run the tests and submit information on how the old one gives wrong answer and new one gives correct answer**

Assert get\_sentiment(pos\_words[j] + ' ' + neg\_words[j])[0] == 'Neutral'

Assert POS\_LEX & NEG\_LEX is None

assert get\_sentiment('abcdefghijklmnop')[0] == 'Neutral'

assert get\_sentiment('')[0] == 'Neutral'

assert get\_sentiment('good ' \* 100 + ' bad')[0] == 'Positive'

assert get\_sentiment('bad ' \* 100 + ' good')[0] == 'Negative'

Some of the sentiments were incorrectly reported as ‘Positive’ are correctly reported as ‘Neutral’ so the new results are more accurate related to the older results

1. **Run the new code on the whole input file. How many positive and negative sentiment lines do you have now?**

New Code gives answers as

Positive: 283 words

Negative: 77 words