IST 664: Natural Language Processing

Homework-1

Name: Vijet Muley

E-mail Address: [vmuley@syr.edu](mailto:vmuley@syr.edu)

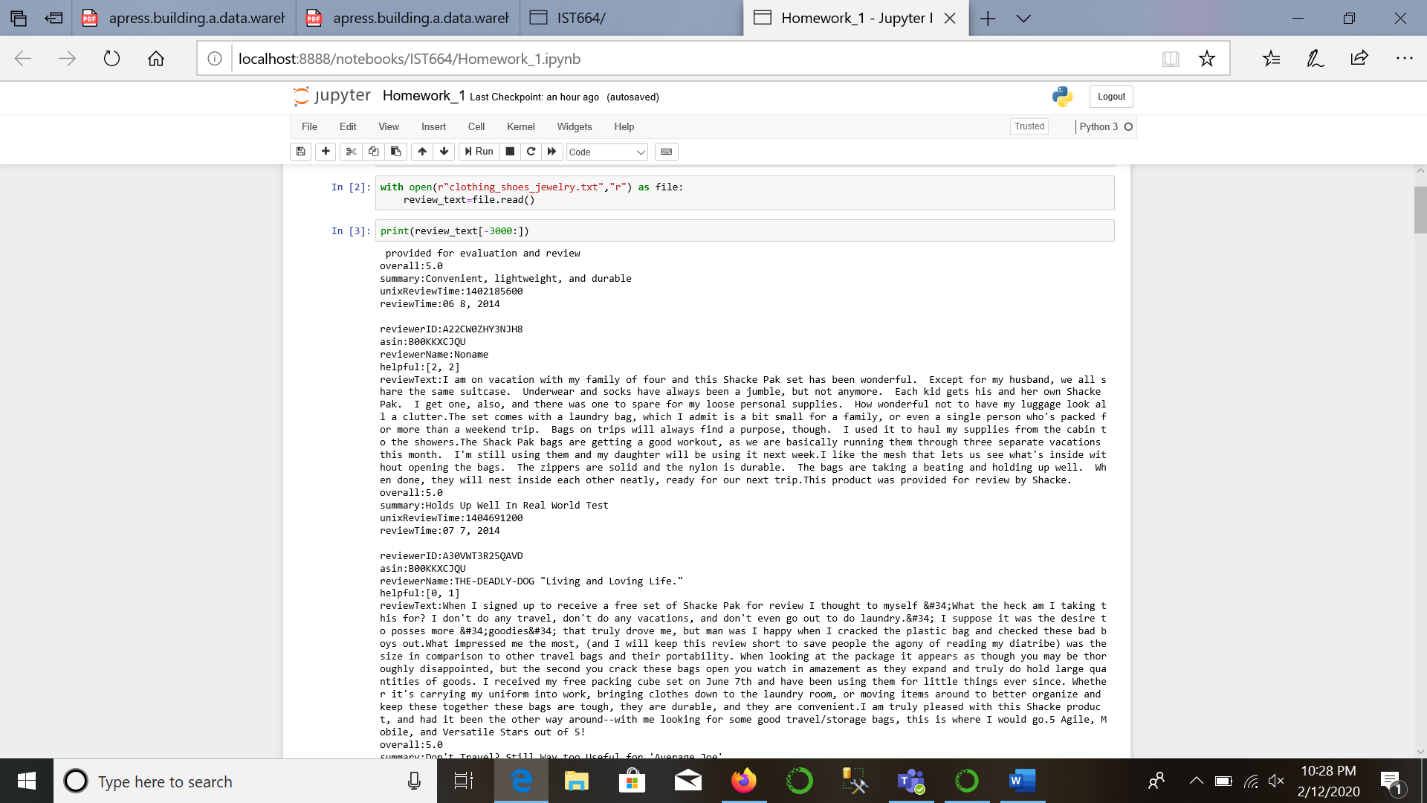
Data Pre-processing:

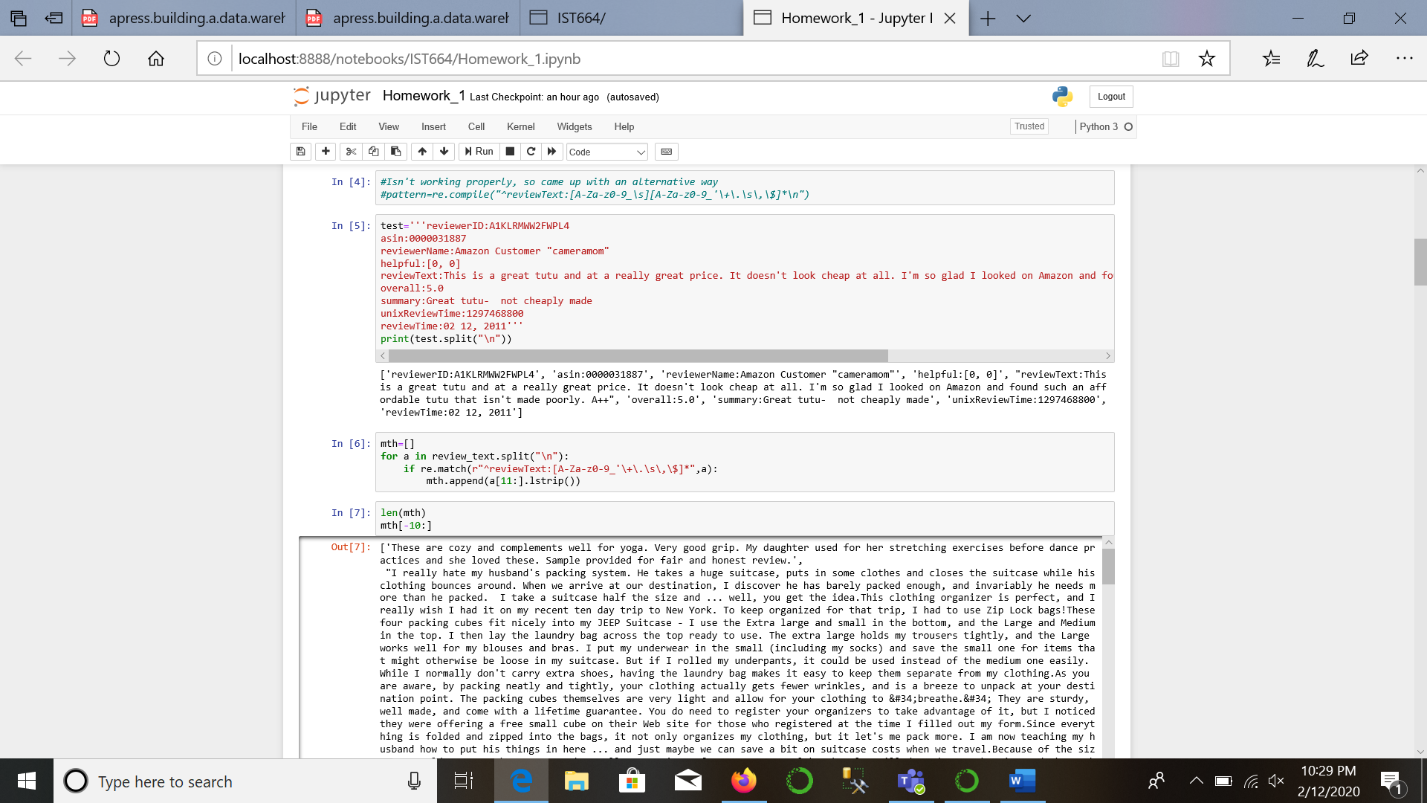
Ans: To start off, I read the text file comprising of the reviews and then looked at the last 3000 characters in the file (to know about the last reviews). I then had an idea to use an easy regular expression in a serialized way and for that I decided to test how splitting one review chunk will work. So, I made a test string with a review block and used the ‘split’ function on it. It thus gave me a list of strings that were split by ‘\n’ (newline character).

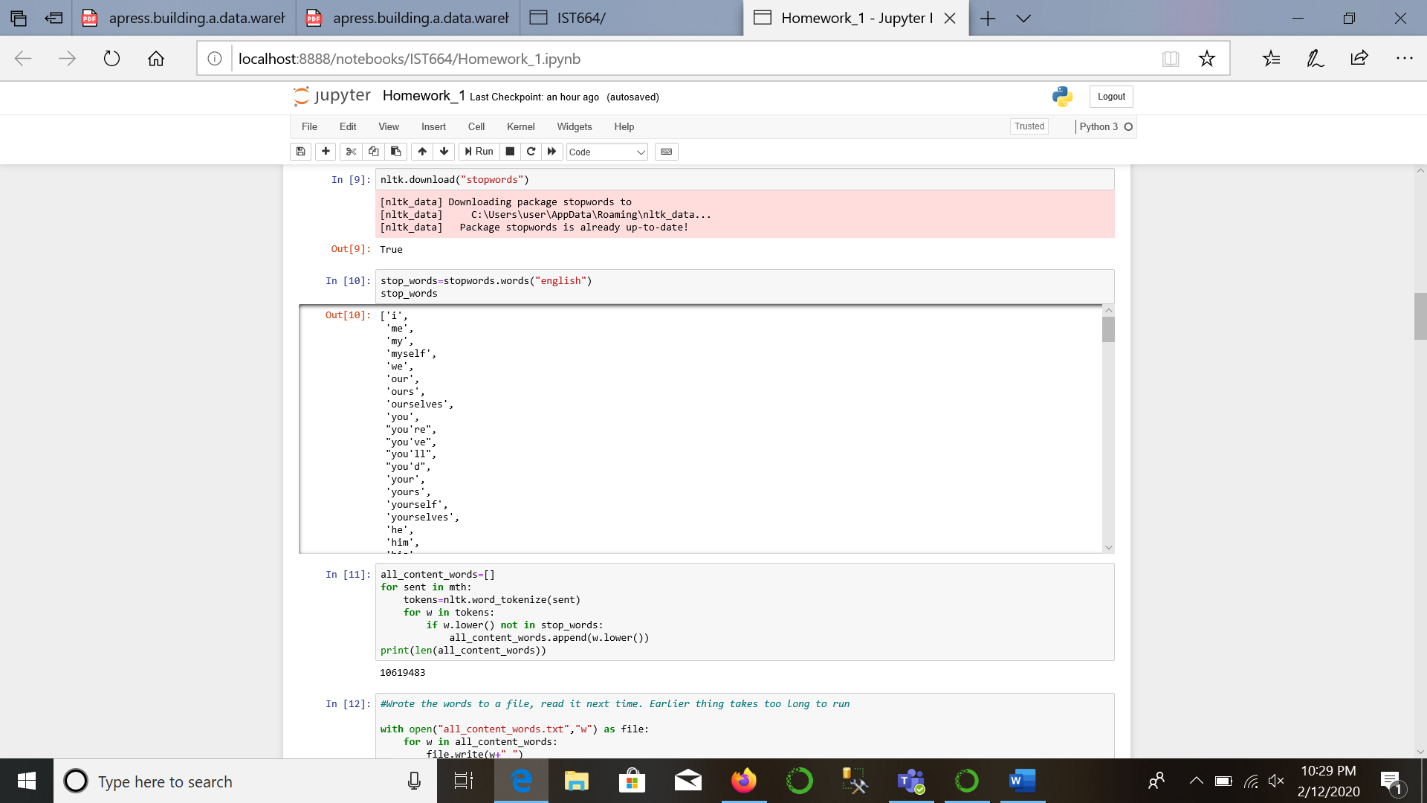
After that, to extract only the review texts from all the review chunks over the text file, I decided to combine the above technique and regular expressions. I split the whole text document using ‘split’ function by newline character and checked if every single line fit my regular expression and if it did, appended it to a list. At the end, I had a list of all the review texts that were ready to be used.

For cleaning, I first off imported the English stop words list from nltk and went through individual words in all the review texts and kept only the words that were not in the stop words list, because I wanted to focus on content words. Then, I used the ‘string’ package which gives a string comprising of commonly used punctuations and I followed the same approach as before to remove all the punctuations from my text. The reason is that I wanted to focus on content words and if I were to follow frequencies and pmi scores, they’d yield punctuations in abundance rather than words.

Following are screenshots that support my approach:









Data Analysis:

My approach to analyze comprised of 2 sub-parts that I thought would give me better insight in what the reviews are:

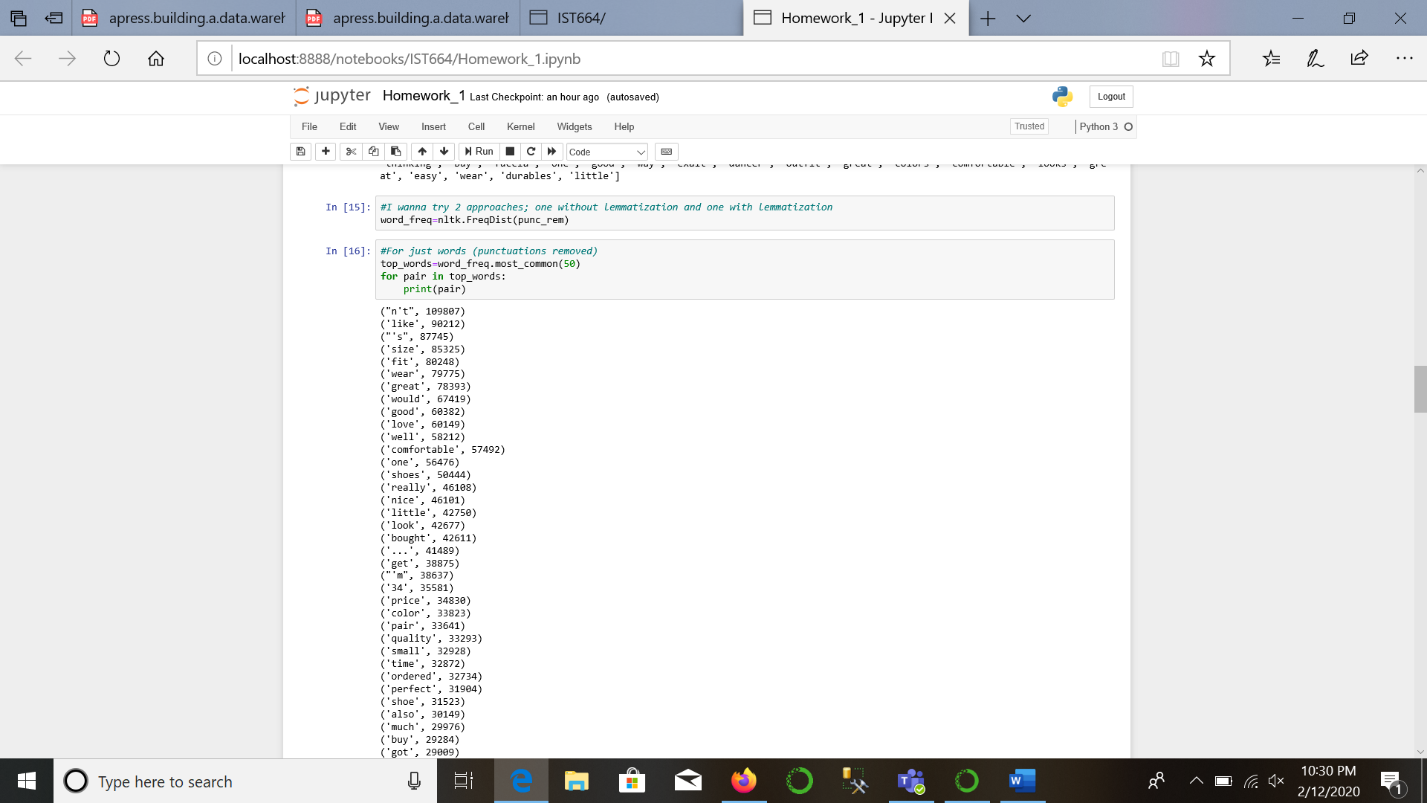
a). Data analysis on corpus that has stop words and punctuations removed but has different forms of words.

b). Data analysis on corpus that has lemmatization done on it and thus have roots of words based on the context.

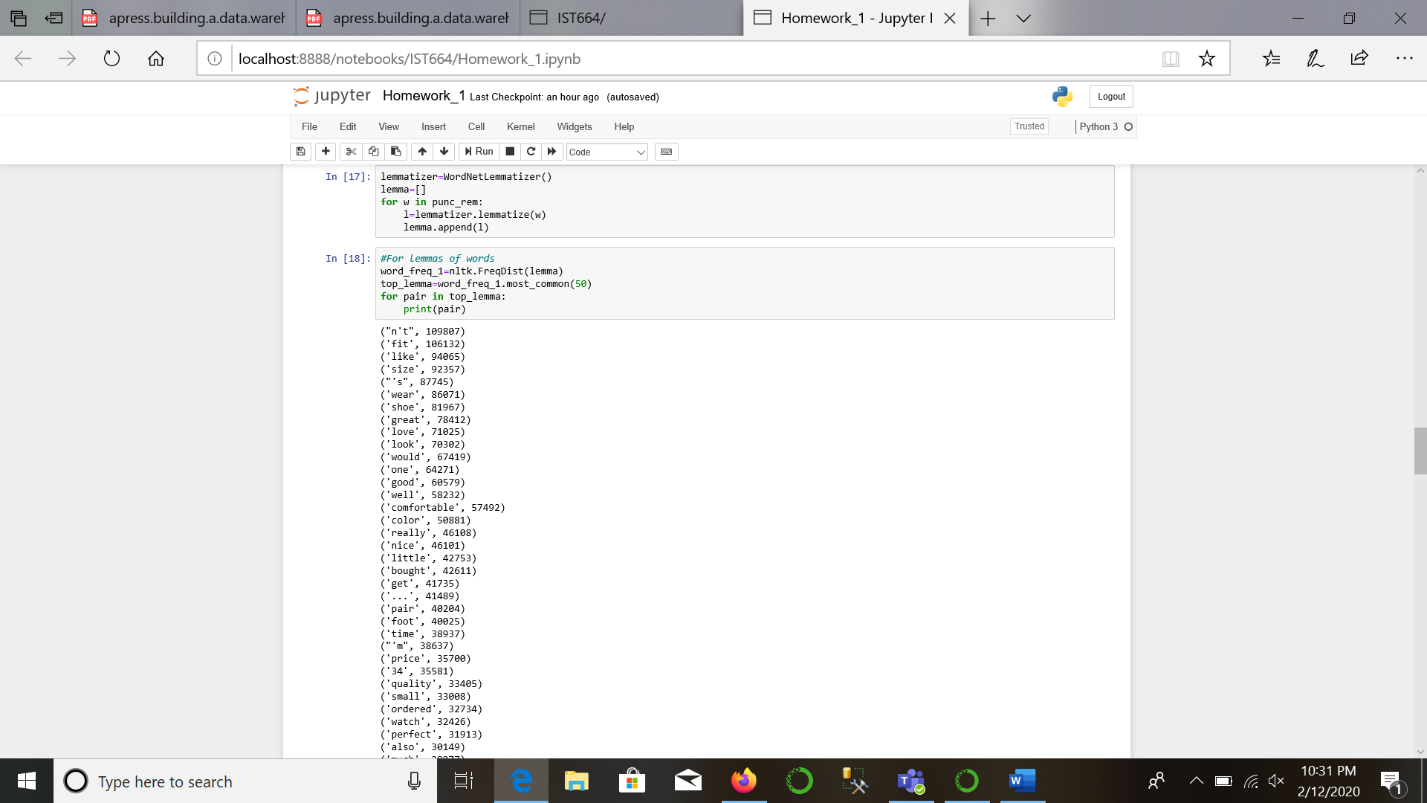
I was expecting the latter part to do better than the earlier one.

1). Top 50 words by frequency:

For all the words:

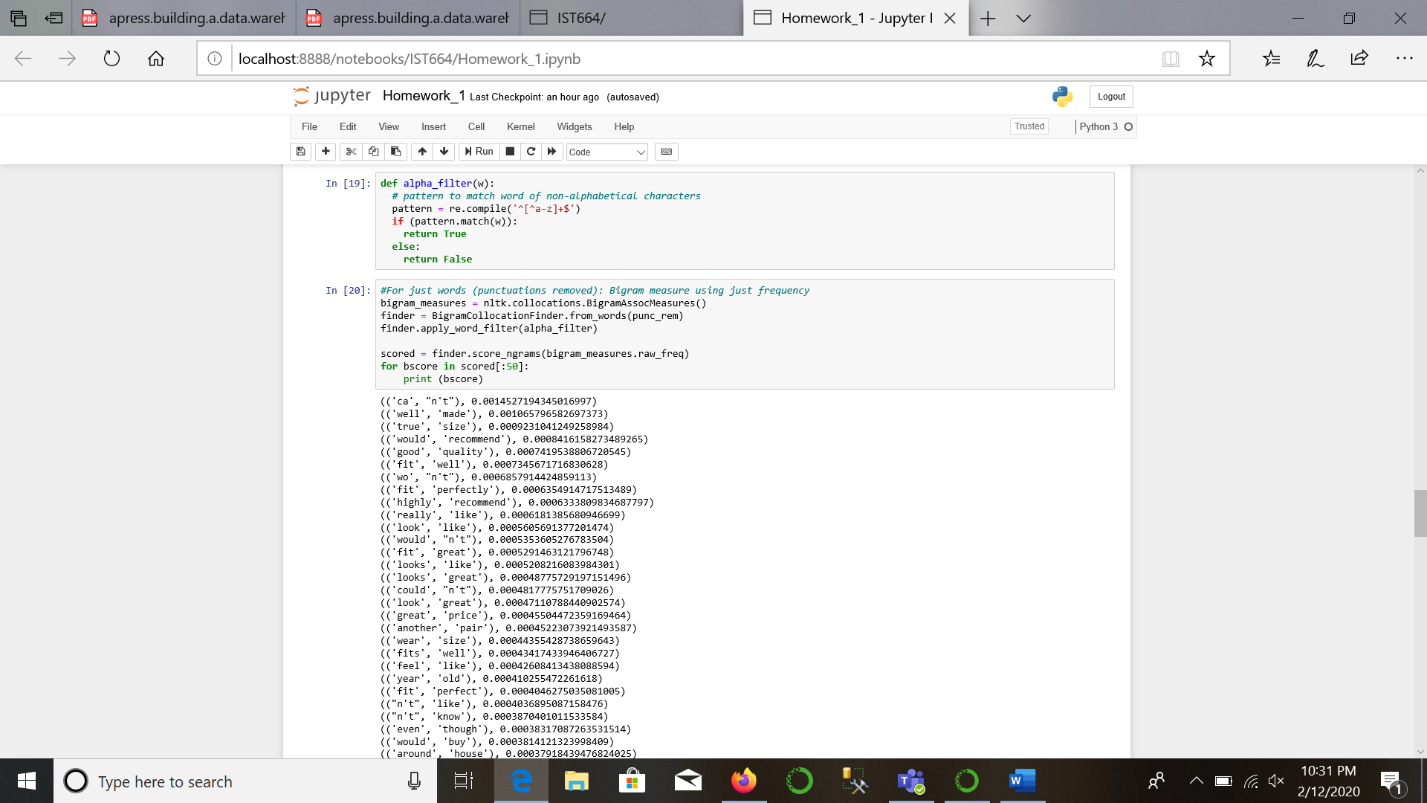


For lemmatized corpus:

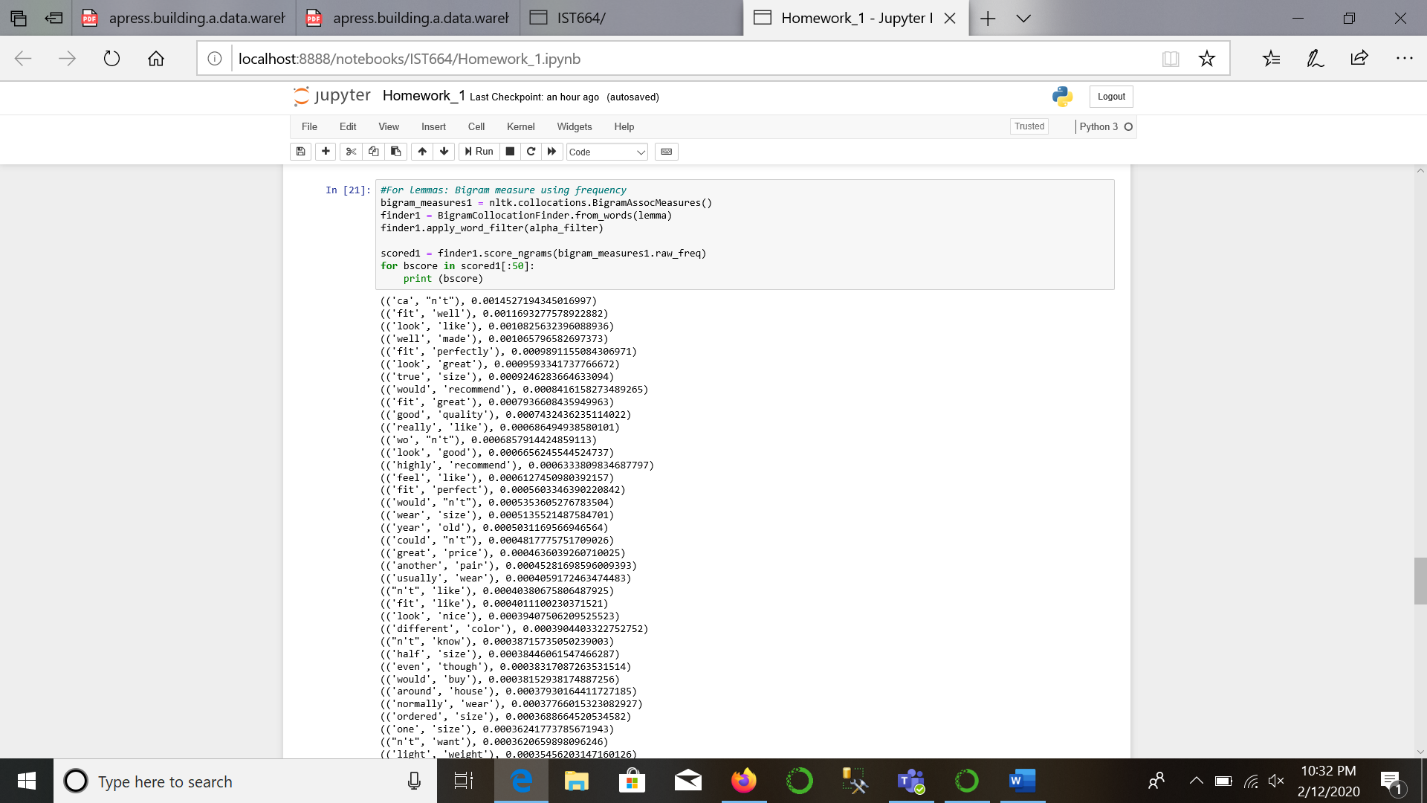


b). Top 50 bigrams by frequency:

For just word corpus:

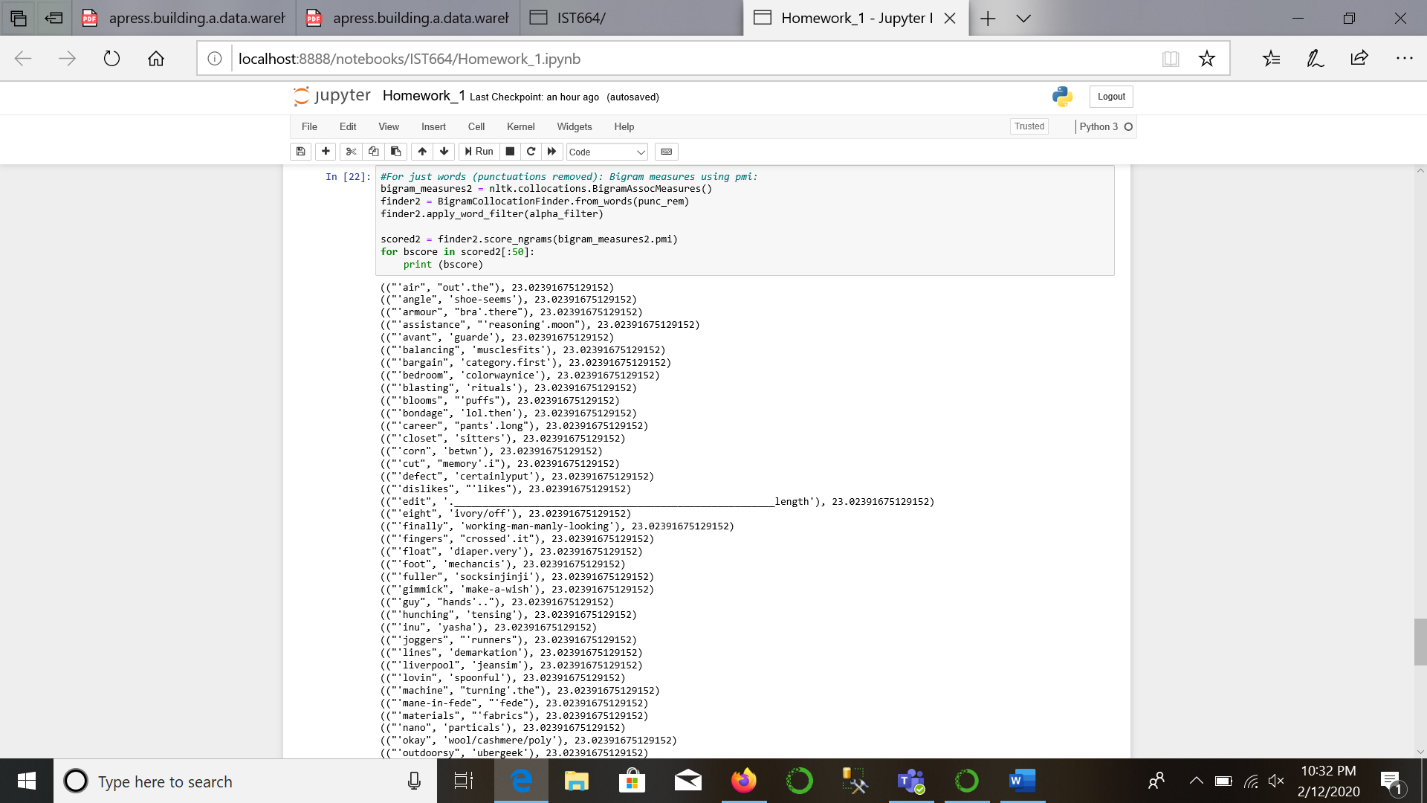


For lemmatized corpus:

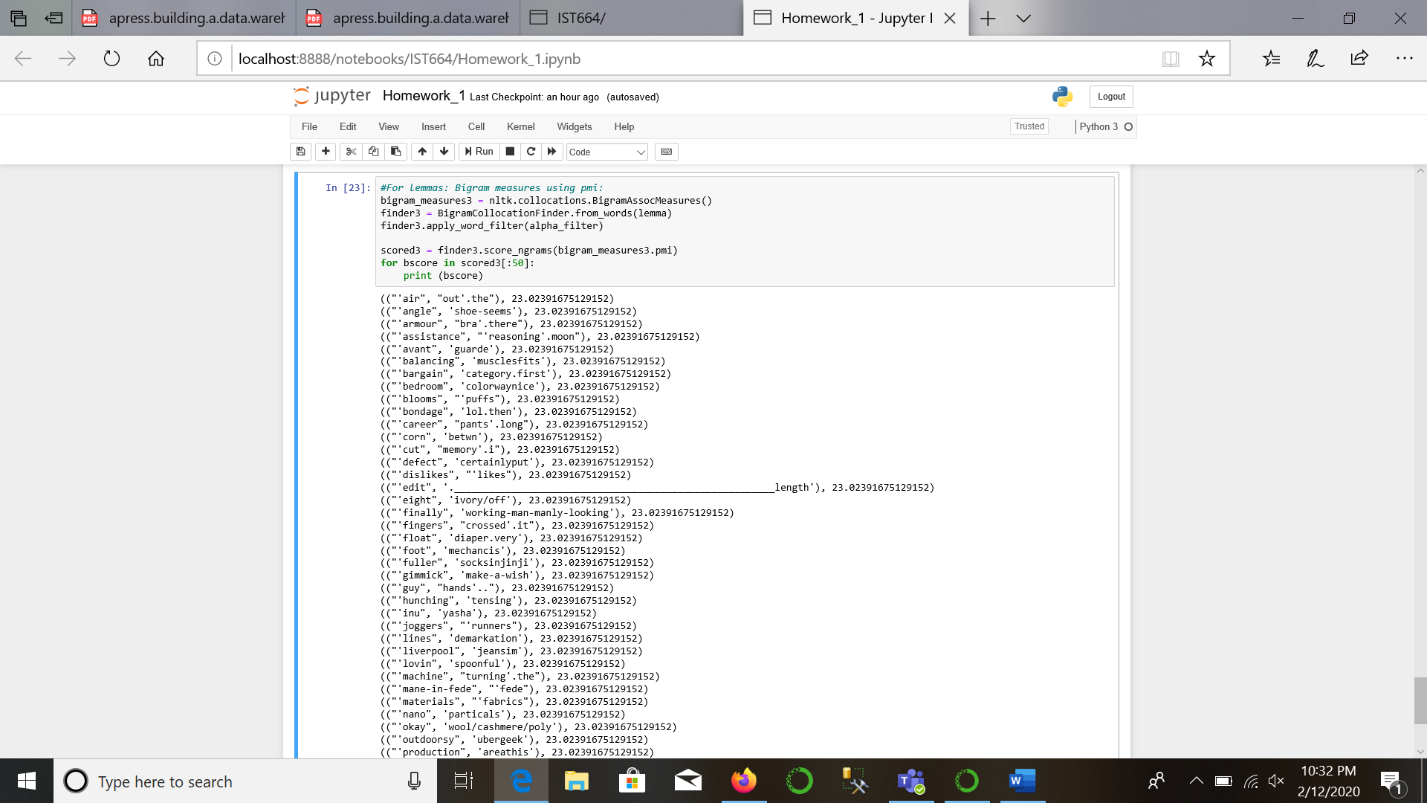


c). Top 50 bigrams based on mutual information scores:

For just the corpus:



For lemmatized corpus:



Interpretation of results:

When I started the assignment, my baseline goal was to get bigram pairs that’d help me understand the point behind reviews. Bigrams give some insight about it. I believe, my choice to do it both ways; with a normal corpus and with a lemmatized corpus did not yield drastically different results, but I believe when this assignment is converted to a project, lemmatized corpus will yield better results. The reason is that the words are being stripped down to their roots, but not just by removal of prefixes and suffixes, but based on the context they are used in. Lemmatization has a bigger time complexity. My interpretation, currently, is that getting bigrams from the lemmatized corpus based on frequency yields pair of words that describe public opinion on the fit, look and quality (mainly) of the products that are reviewed.

Additional analysis of this dataset, in my opinion, can use annotation of a review being ‘in favor’ ‘or against’ a product. The products can have common feature measures and the reviews can be tied to those products. TF-IDF will help us determine similarity between reviews and help cultivate what the overall public opinion of a product is. Not only that, but based on the product attributes and the reviews, a general idea of public review can be generated for a similar, unseen product with attributes.