CHAD WYNGAARD

STUDENT NUM: 216122740

DOCUMENTATION

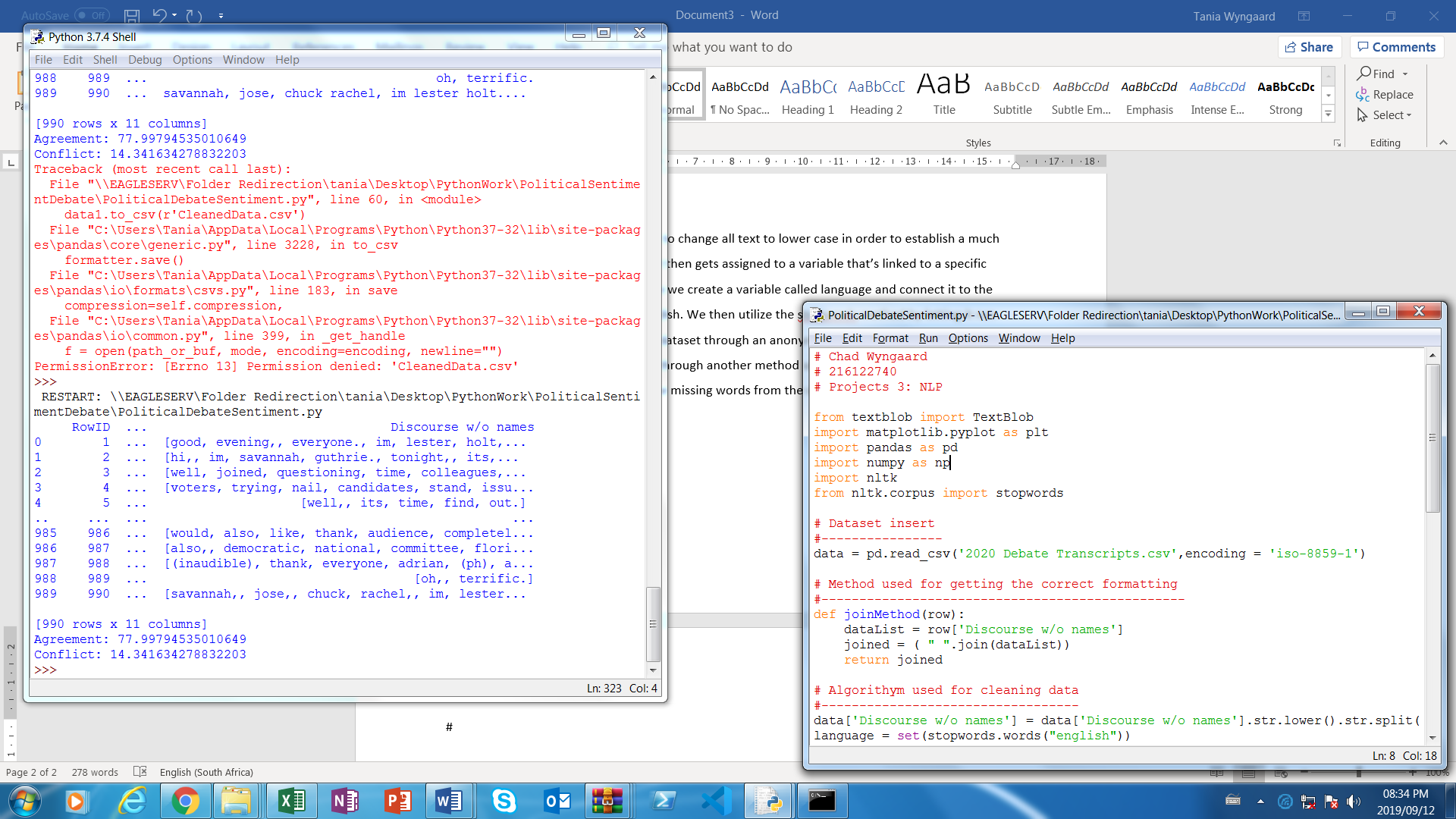
NLP ASSIGNMENT

**# Description**

This AI reads in structured csv files and cleans the data in order to perform sentiment analysis on each row in the different columns specified. The AI then produces a model based on the polarity of the negativity and positivity that took place within each speech. The AI creates a separate dataset in the form of a csv file that then contains the data after it has been cleaned.

**# Pip installation libraries**

* Pip install pandas
* Pip install matplotlib
* Pip install numpy
* Pip install nltk
* Pip install textblob

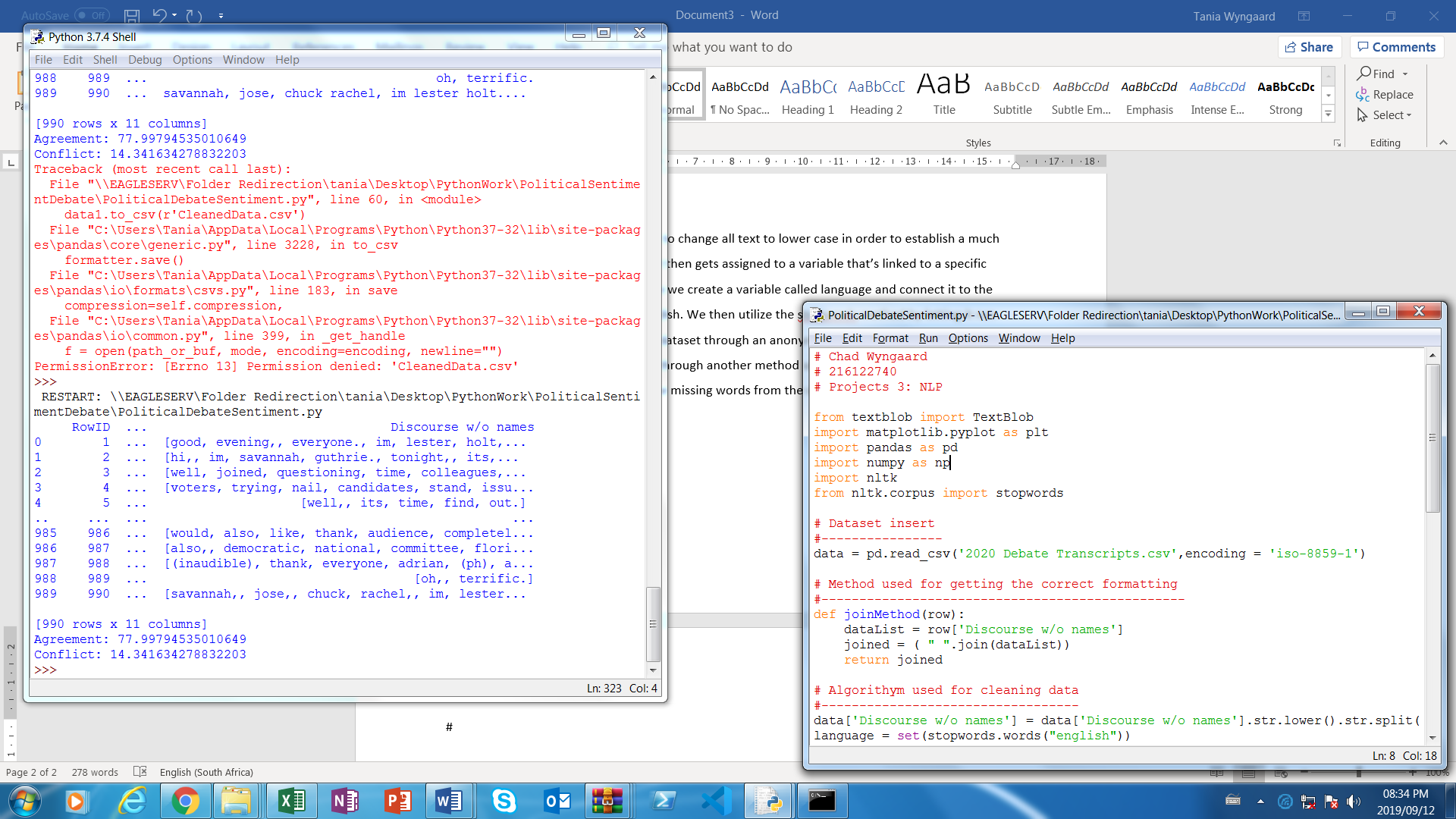


**# Imports**

The program requires certain libraries to be imported.

Required imports:

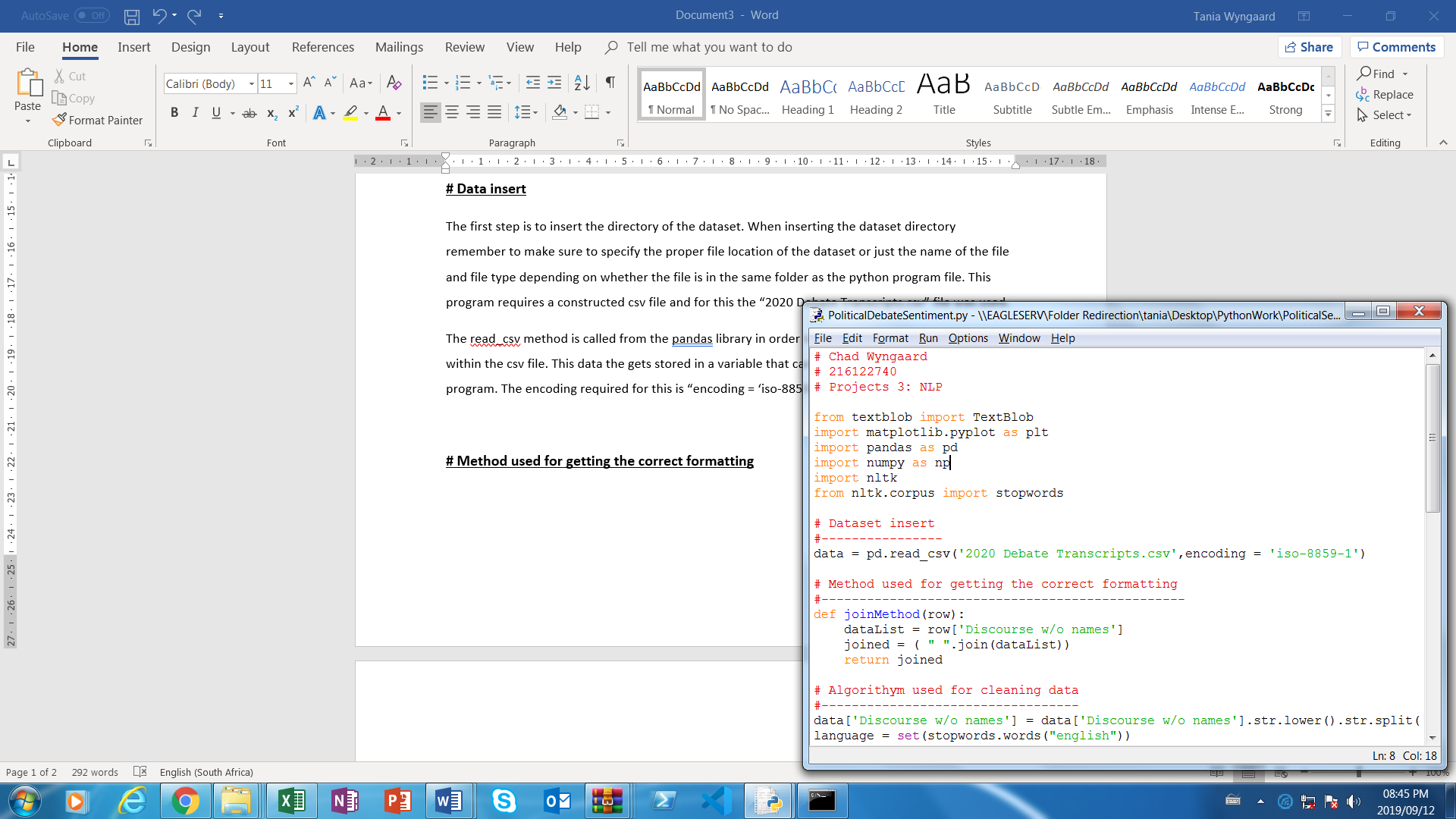
* The TextBlob directory in the textblob library
* The pyplot directory in the matplotlib library
* The pandas library
* The numpy library
* The stopwords directory in the nltk library



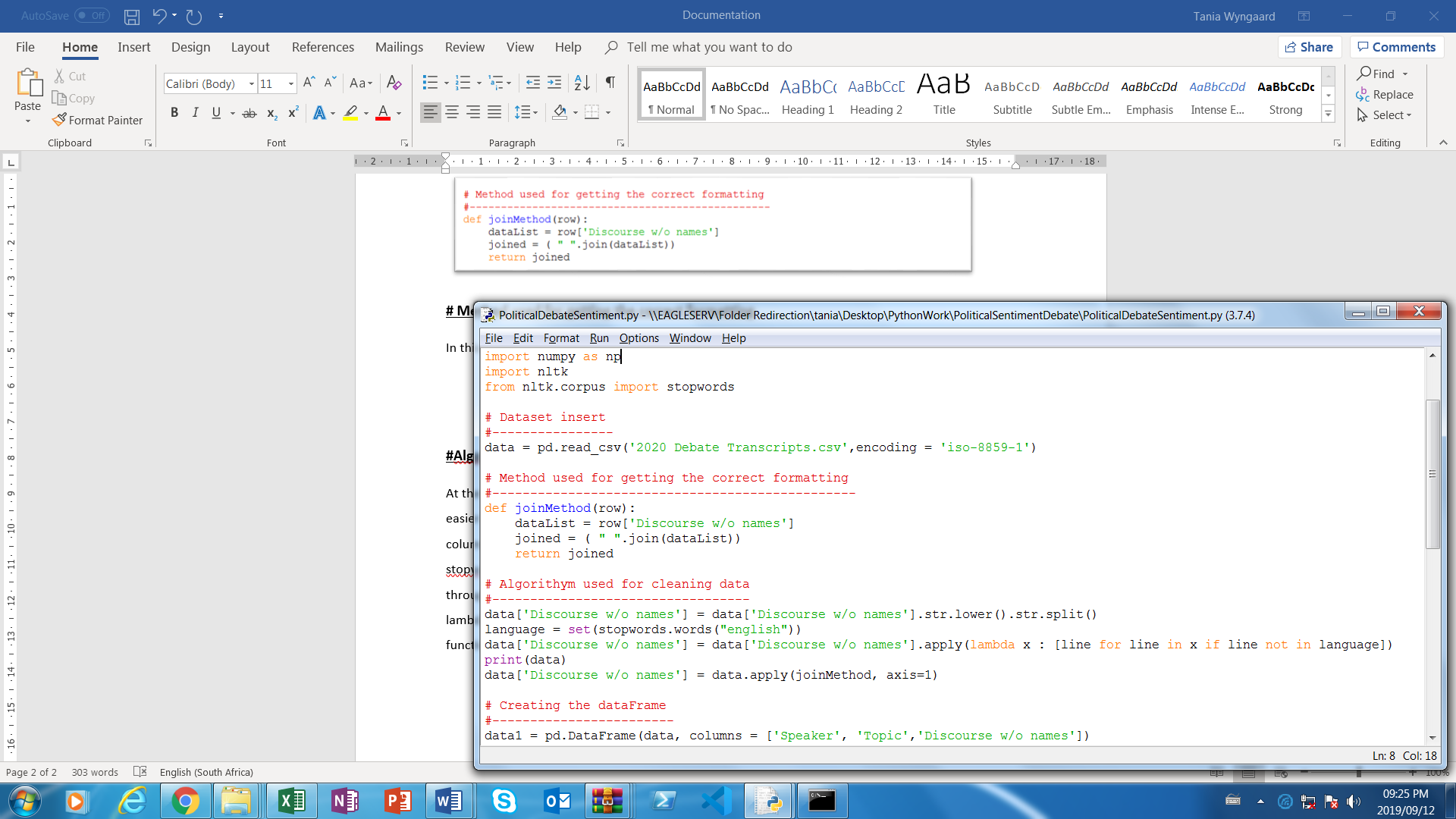
**# Data insert**

The first step is to insert the directory of the dataset. When inserting the dataset directory remember to make sure to specify the proper file location of the dataset or just the name of the file and file type depending on whether the file is in the same folder as the python program file. This program requires a constructed csv file and for this the “2020 Debate Transcripts.csv” file was used. The csv file being used only takes in the transcripts for days 1 and 2 of the 2020 Presidential Debate so the agreements might be much higher than the conflict at this point.

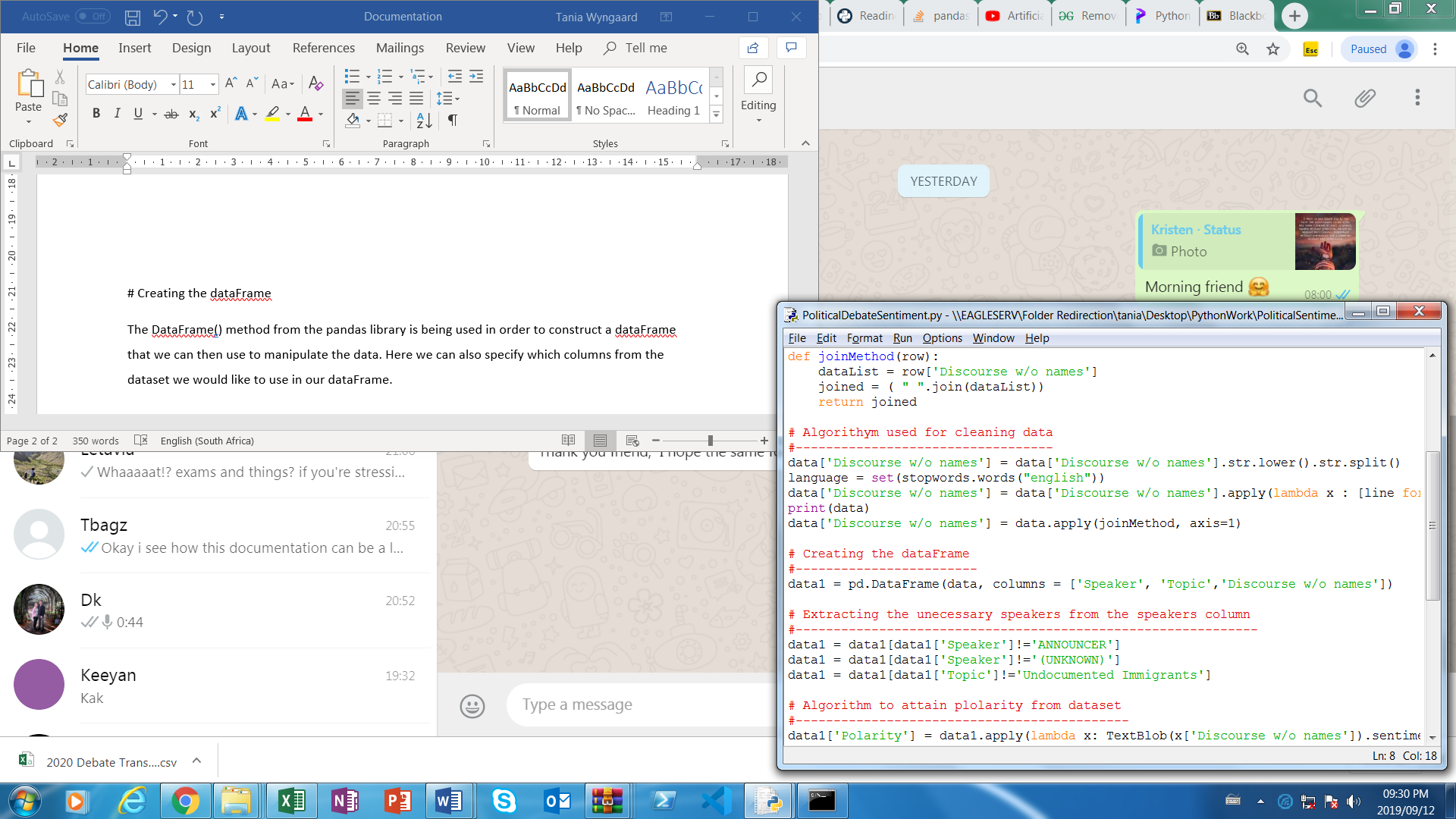
The read\_csv method is called from the pandas library in order to open and read in the data stored within the csv file. This data then gets stored in a variable that can later be reference throughout the program. The encoding required for this is “encoding = ‘iso-8859-1’ ”.



**# Method used for getting the correct formatting**

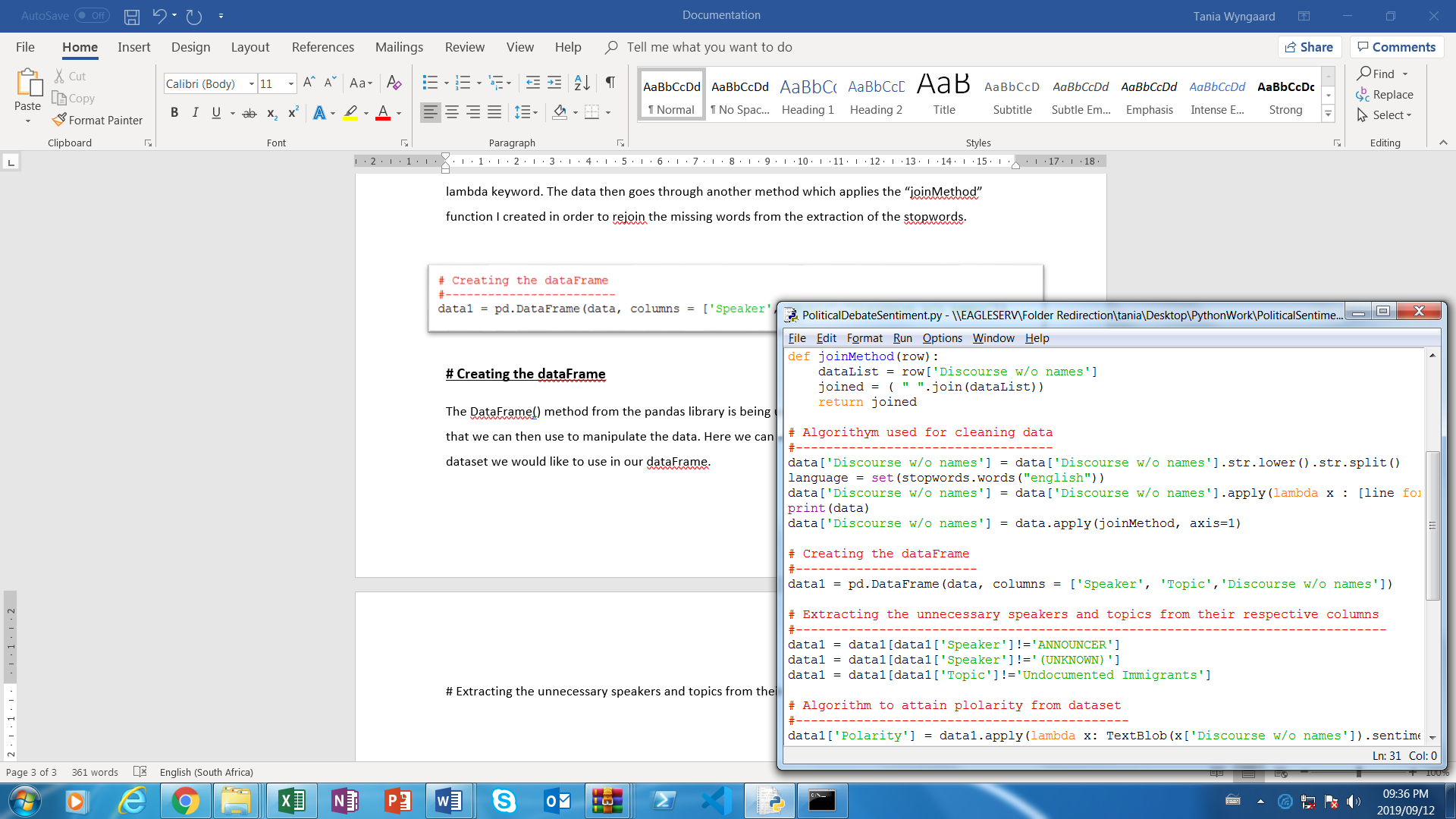
In this function the program is taking in the dataset then concatenating it after the lines have been split. The join( ) method will be used in order to concatenate the lines so that it would be converted back to a consistent state.

**# Algorithm used for cleaning data**

At the beginning the program uses the .str.lower( ) method to change all text to lower case in order to establish a much easier way of working with the data. It then gets assigned to a variable that’s linked to a specific column within the dataset. From there we create a variable called language and connect it to the stopwords method which is set to English. We then utilize the stopwords method by letting it run through every line of code within the dataset through an anonymous function defined by the lambda keyword. The data then goes through another method which applies the “joinMethod” function I created in order to rejoin the missing words from the extraction of the stopwords.

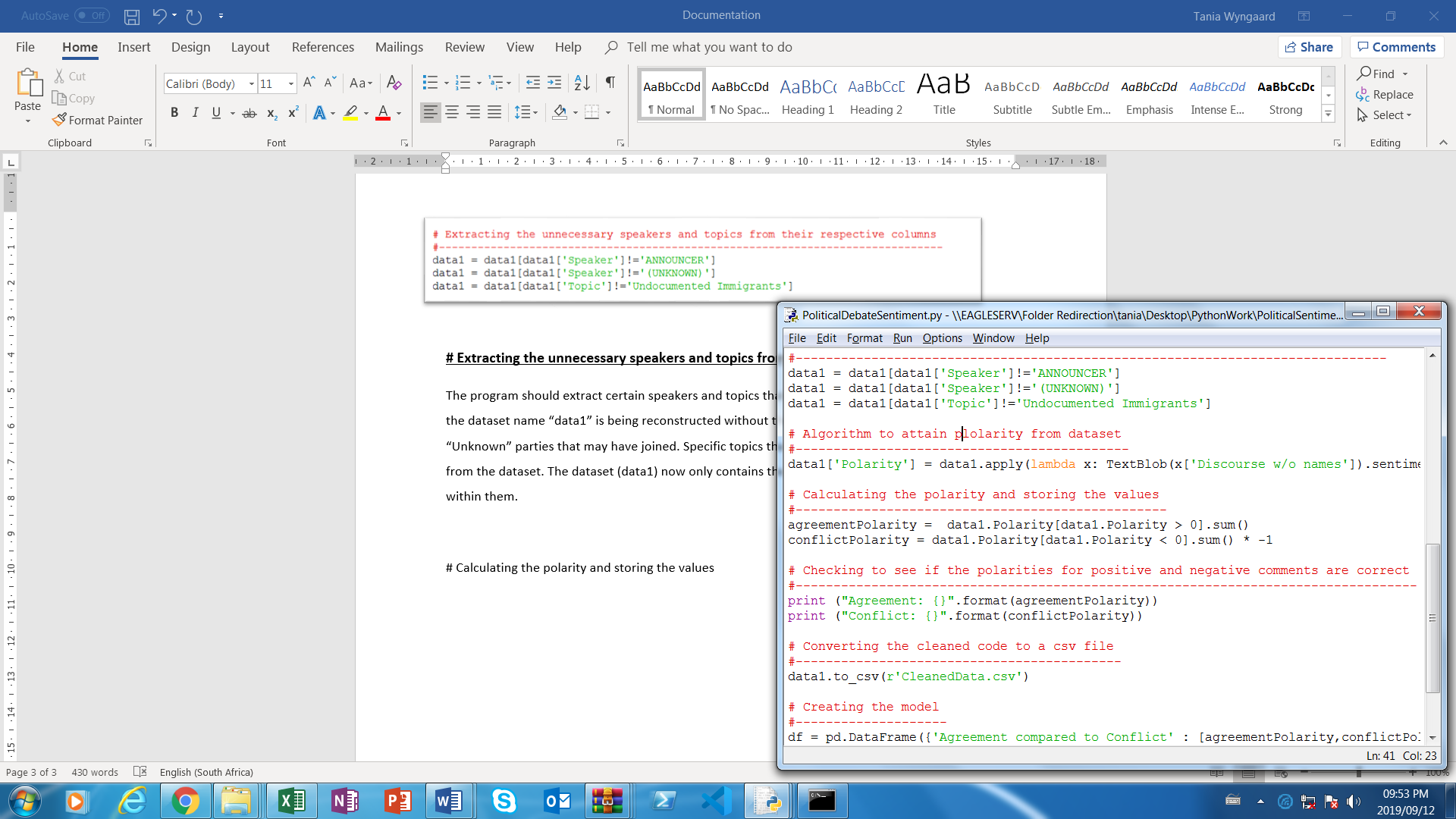
**# Creating the dataFrame**

The DataFrame() method from the pandas library is being used in order to construct a dataFrame that we can then use to manipulate the data. Here we can also specify which columns from the dataset we would like to use in our dataFrame.

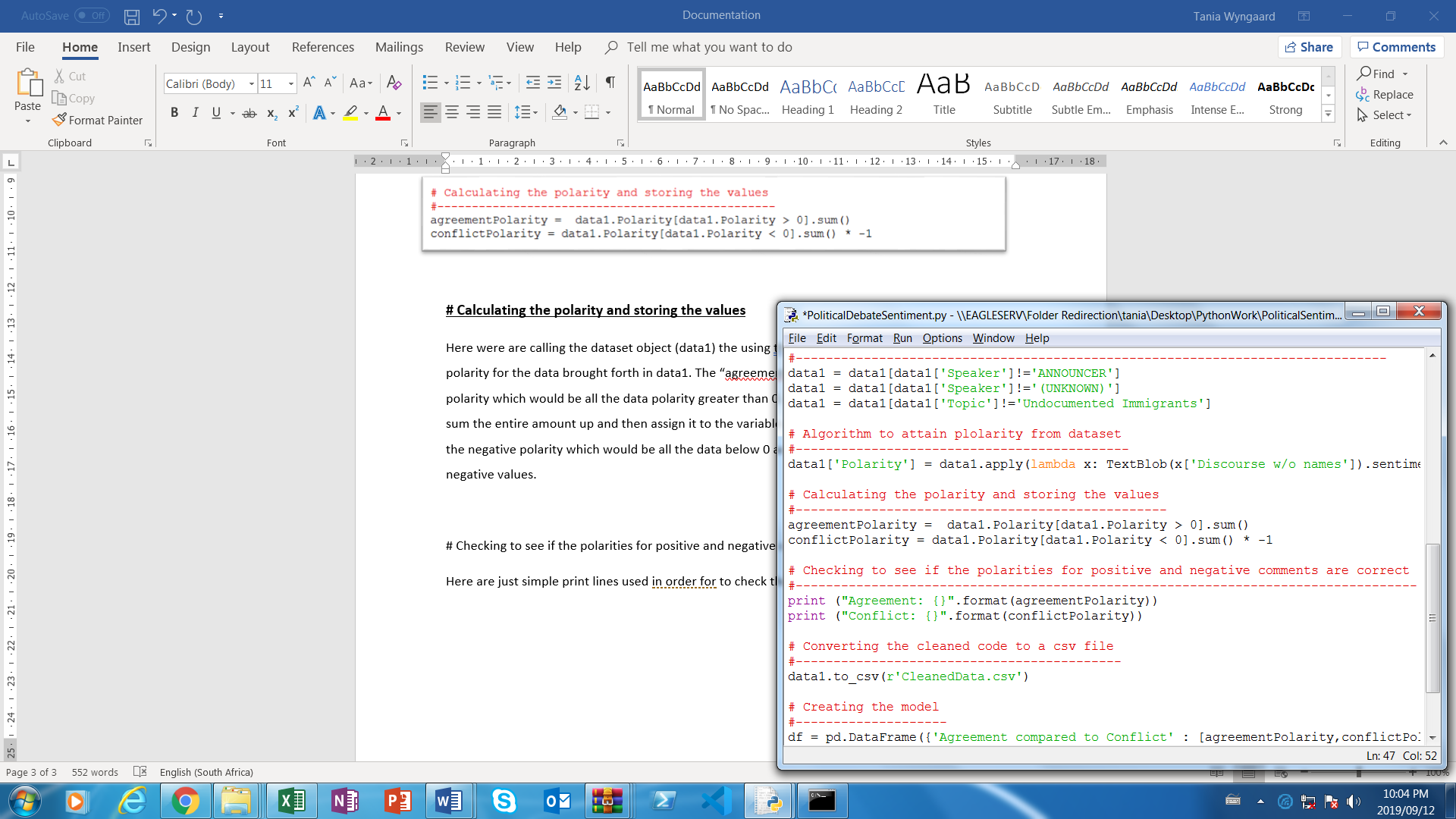


**# Extracting the unnecessary speakers and topics from their respective columns**

The program should extract certain speakers and topics that aren’t relevant to the document. Here the dataset name “data1” is being reconstructed without the “Announcers” as well as any “Unknown” parties that may have joined. Specific topics that were not document was also excluded from the dataset. The dataset (data1) now only contains the columns without these specified data within them.

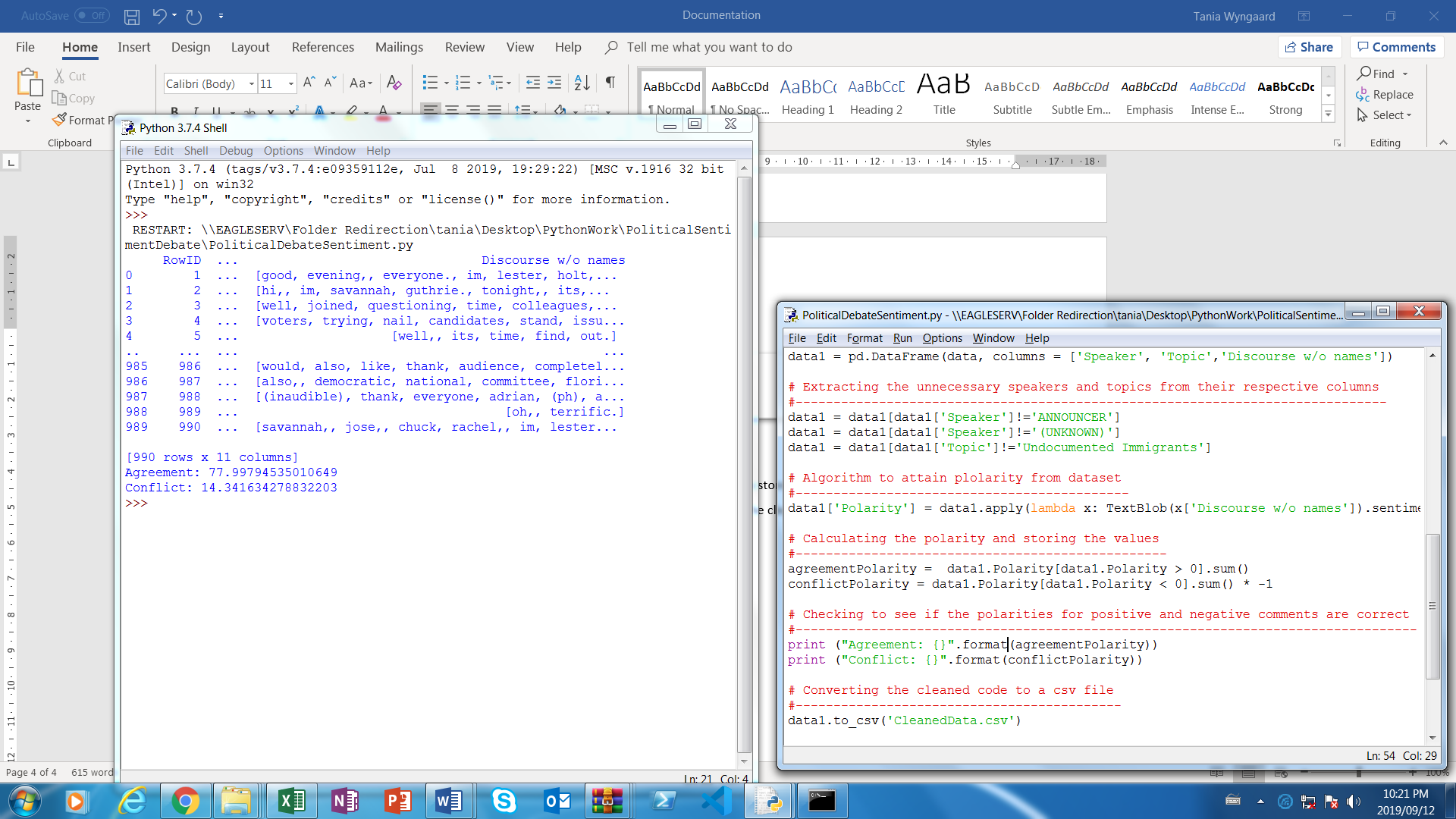


**# Calculating the polarity and storing the values**

Here the program is calling the dataset object (data1) then using the .Polarity method in order to get the polarity for the data brought forth in data1. The “agreementPolarity” variable takes the positive polarity which would be all the data polarity greater than 0. I think use the sum() method in order to sum the entire amount up and then assign it to the variable. The “conflictPolarity” variable takes in the negative polarity which would be all the data below 0 as well as adding an algorithm to get negative values.

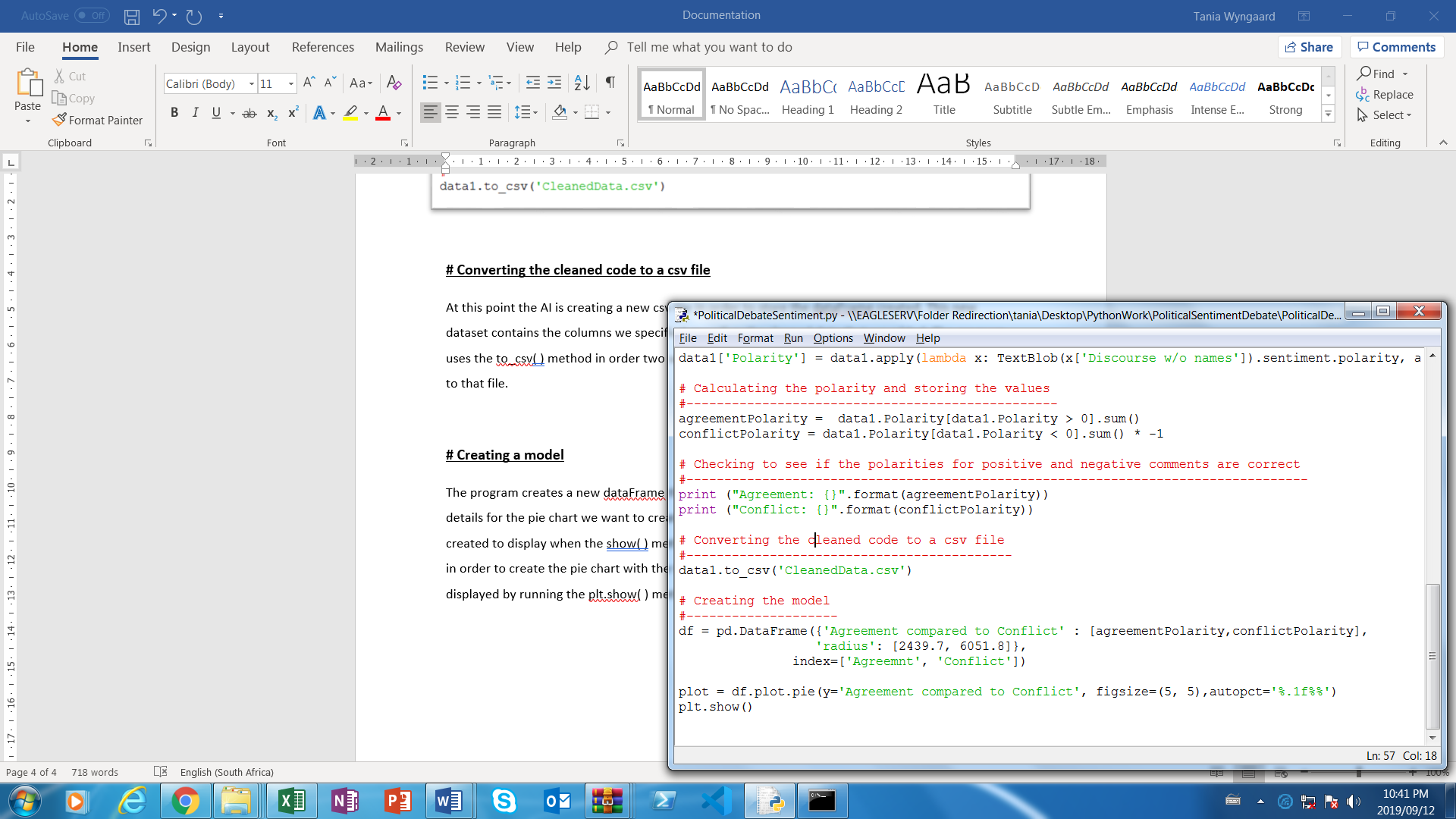
**# Checking to see if the polarities for positive and negative comments are correct**

Here are just simple print lines used in order to check the polarities that are coming through. We can use this to compare to the model we generate after.



**# Converting the cleaned code to a csv file**

At this point the AI is creating a new csv file in order to store the dataFrame created. This new dataset contains the columns we specified as well as the cleaned data that we added. The program uses the to\_csv( ) method in order two write the data to a new csv file or create a new one and write to that file.



**# Creating a model**

The program creates a new dataFrame through the pandas library which basically gives all the details for the pie chart we want to create. In the dataFrame the indexs for the chart are also created to display when the show( ) method is called. The program the uses the .plot.pie() method in order to create the pie chart with the title and specified parameters. The chart can then be displayed by running the plt.show( ) method.