Text Mining and Natural Language Processing – RSM317

Group Assignment 2 Due date: see Quercus

Deliverables (to be submitted on Quercus):
All Python source code / answers in a Jupyter Notebook (*.ipynb).
Any additional documents (Word, PDF, CSV, TXT, HTML, etc.) as required.
Team<X>predictions.txt (please see below).

Sentiment Analysis on Product Quality

This case assignment requires an analysis of consumer sentiment to understand perceptions of product quality (see data set ReviewsTraining.csv).
The primary objective is to utilize sentiment analysis techniques on the posted Reviews data set to extract and analyze consumer sentiments (variable "Score"). This analysis will provide actionable insights into how quality perceptions can influence consumer acceptance and preference, directly tying into the strategic considerations of companies selling such products.
The "Score" should be predicted based on text appearing in the columns "Summary" and "Text" only.

All the remaining steps have to be performed using Python!
The code should be submitted in form of a single Jupyter Notebook (*.ipynb).

Carefully preprocess the text. Describe the steps that you are taking to preprocess the text.
As we are performing sentiment analysis, bigrams such as "not good" or "very good" *could* be useful. Therefore, we would like to expand contractions such as "isn't" or "haven't" to "is not" and "have not", respectively.
You can find a list of contractions and their respective expansions in the file contractions.py that you might want to use to perform this task.
Use regular expressions or other ideas to expand all contractions in the reviews.

Hint: (from the textbook page 288)
Expanding contractions: In the English language, contractions are basically shortened versions of words or syllables. These shortened versions of existing words or phrases are created by removing specific letters and sounds. More often than not, vowels are removed from the words. Examples include "do not" to "don't" and "I would" to "I'd".
Contractions pose a problem in text normalization because we have to deal with special characters like apostrophes and we also have to convert each contraction to its expanded, original form.

As mentioned above, bigrams or trigrams etc. *could* be useful. In fact, you should be comparing at least using single tokens only to using bigrams. Discuss what challenges you encounter here!

One challenge is that depending on your tokenization of the text, two consecutive words might be considered as a bigram although they are actually separated by punctuation (or similar) in the original text.

As an example of this, also the Google search engine has problems handling this "correctly":
Search term: "family long weekend".
Some results:
Correct:
I always look at family long weekend as the kickoff to Spring.
Suggestions for things to do on family long weekend for a person …
Totally 80s Family Long Weekend.
Possibly related:
Our family's long weekend in Niagara Falls (Ontario, Canada) with a day trip to Toronto had something for every member of our family: an ...
Questionable:
Family, Long Weekend, Niagara Falls, …
El Family: Long Weekend Event Feat. Tone Of Arc.
Fun for the whole family! Long weekend calls for some ...
Completely incorrect:
font-family: 'Long Weekend', sans-serif; ...

After preprocessing the text, we are now going to build a Keras Artificial Neural Network (ANN) that is using the pre-processed text to predict the "Score".
You can see that there are five different scores. To simplify the prediction task, scores 1-3 should be regarded as negative reviews and 4-5 are positive reviews (or as you see fit).

Challenge/Bonus Question (2 extra bonus points) – not required: Can you build and train a Keras ANN that makes multi-class predictions, i.e., scores 1 – 5?

With this assignment, we begin to graduate to the big leagues of text mining modelling. In addition to the usual material you submit, we will grade you on how accurate your predictions are! (Hey, you are getting to be pretty sophisticated data analysts. 😊)

The last section of your code submission should present your best model. This part of your code should write your 500 predictions of "Score" for the test data set into an ASCII file named Team<X>predictions.txt (where <X> stands for your team number).
The ASCII file should contain only one variable, zeroes and ones (0's indicate a prediction of negative sentiment; 1's indicate a prediction of positive sentiment). Please also submit this file. Your overall percentage of correct predictions compared to the best submission will then count out of 1.5 points toward the assignment grade.

Business Insights

Using the above results, we can now derive basic business insights.
Answer the following questions:
1. What are the most frequent words associated with negative reviews? What are the main areas of underlying problems that are reflected in these words?
2. Are there recurring phrases that indicate product defects or poor customer service?
3. Based on the recurring negative-review patterns, what specific improvements or strategic changes would you recommend to address these concerns?