Robert Gordon University

CMM706 – Text Analytics 2021

Module leader	Dr. Ruvan Weerasinghe
Unit	Coursework
Weighting:	70% for Report on assignment 10% for In-class presentation 20% for Comparative study report
Learning Outcomes Covered in this Assignment:	LO1 Critically appraise extraction and search models in information retrieval and Natural Language Processing in relation to big data case studies. LO2 Critically evaluate current research and advanced scholarship in IR and NLP, their role and alternative directions for big data projects. LO3 Combine methods from NLP, topic modelling and text mining tool−kits to develop new extraction processes for real−world tasks. LO4 Plan a comparative study to evaluate and interpret results from designing and developing information retrieval and extraction systems for big data.
Handed Out:	21st June 2021
Due Date	18th July 2021, midnight. Each individual will be scheduled a 15 minute time slot to demonstrate their solution on a date(s) to be agreed in class (potentially in the week starting 9th August 2021).
Expected deliverables	One compressed electronic file containing the reports and code specified below.
Method of Submission:	Online via Moodle (see below).
Type of Feedback and Due Date:	Written feedback and marks – within 10 working days after the conclusion of the presentations.

Coursework Description

Organizations are eager to gather the views of their main stakeholders (referred to as customers hereinafter) via online forums. One of the common ways to get a 'birds eye' view of this is to use sentiment analysis on the feedback they receive. More importantly, they are particularly interested in obtaining any suggestions that their customers have on how to improve their offerings. The task below is to provide such organizations a way to do this and to do it so that suggestions on the different aspects of their service offerings are categorized in order to give appropriate priority to the various suggestions received.

You are required to produce Python 3 code in a Jupyter Notebook to do the following.

(a) Use the *hotel reviews* subset of the suggestion mining dataset provided at https://github.com/sapna13/Suggestion-Mining-Datasets. Read the csv data file into a Pandas *dataframe* and clean the data by (i) removing any remaining irrelevant content such as non-alphanumeric characters, escape sequences and possible extraneous html tags, (ii) tokenizing your text by separating them into individual words and (iii) converting the case of the tokens to obtain unique words. Print the number of posts which are suggestions and the number that are non-suggestions (opinions) in order to gauge the dimensions of the dataset. Also print the number of total words in this dataset and the number of unique words in it.

(b) Other ways of reducing 'noise' in the dataset are to (i) remove the so called *stopwords* and (ii) to *stem* or *lemmatize* the rest of the words. Print the number of total words and the number of unique words after each of the above two steps for this dataset. Also print the maximum and minimum document sizes (in words) of the posts after each step. Visualize the data using a histogram of the data for different sentence lengths (Hint: use matplotlib).

(c) Create *bag-of-words* and *TF-IDF* representations of the posts in the dataset above[1] and use two relevant *supervised learning* algorithms to classify future posts as suggestions or non-suggestions. Print the *confusion matrices* of the four (04) resulting combinations for a held-out (test) dataset.

(d) Suggest any strategies you may use to *improve* the performance of the above classifier (apart from using deep learning). Implement your suggestions as improvements to the above models and print the confusion matrix of the best representation and model you get.

(e) In general, suggestions can be categorized into particular *aspects* of the service provided by the hotel. Propose an *unsupervised* approach to categorize the suggestions (ignoring non-suggestions) into the different aspects of the service offered. How could an *optimal* set of categories be determined? Implement your solution and print the optimal number of categories your model finds. Visualize your categories by displaying word clouds or topics based on your approach.

[1] Using the *CountVectorizer* and *TFIDFVectorizer* in *scikit-learn*.

(f) If you were told that the hotel is interested in *eight (08) specific* aspects pertaining to value, location, service, food, facility, room, quality and staff, would your modeling in part (e) change? If so, *implement* this new model and visualize the categories using word clouds or topics. If not, why not?

(g) Finally, assume that the hotel gives you the attached set of *manually annotated* suggestions pertaining to the eight (08) aspects given in part (f) above, and evaluate the performance of your best model with respect to this *ground truth* data. Print appropriate metrics to evaluate your model.

(h) Discuss (without implementing) any issues with the data and models used, and any improvements you would suggest to make the overall model more useful and/or perform better. A reflection on the coursework task as a whole is expected.

The Assignment Report

You need to formulate solutions for each of parts (a) through (h) above, clearly explaining your Python code and specifying the outputs produced by the code for the dataset given in a *Jupyter Notebook* named *Solution_IDNumber.ipynb* based on the template given[2]. For each such part, a descriptive summary with an interpretation should be given for the output obtained after each executable *cell*.

Presentation

You need to explain your code and the output it produces using the Jupyter notebook for each part (a) through (h) to demonstrate your understanding.

Comparative Study Report

Many NLP tasks can benefit by the use of deep learning. Describe the literature on the *deep learning models* that are used for (i) text representation/feature extraction and (ii) text classification stating the reported accuracies of the latest state-of-the-art models. Your report should be no longer than **1000 words** and should list the references used at the end.

Assuming that the best cluster assignment that you found for the task above as the *ground truth label* for each customer review (i.e. in the absence of the data given in part (g)), fit a *deep learning* model that generalizes a predictive model for suggestion aspect classification and evaluate its performance[3]. Your implementation should be included as part (i) in the above Jupyter Notebook.

[2] The *IDNumber* part of the filename should be replaced with your *IIT ID* number.

[3] *Keras* is a simple python wrapper for the popular tensorflow library that can be used for this.

The PDF version of the report should be named, *Report_IDNumber.pdf* where your IIT ID number should replace IDNumber.

Submission

Your Jupyter Notebook and comparative study report should be submitted to Campus Moodle *as a single compressed (.zip or .rar) file* with the name *Coursework_IDNumber.zip* (or *.rar*) with the IDNumber replaced by your IIT ID number.

Coursework Marking scheme

The Coursework will be marked based on the following marking criteria:

Question	Marks	Marks provided	Comments
(a)	07		
(b)	08		
(c)	10		
(d)	08		
(e)	15		
(f)	08		
(g)	10		
(h)	04		
Presentation	10		
Comparative Study Report	20		*10% for documentation/literature* *10% for implementation*
Total	100		