**Project 8: Sentiment Analysis of Hotel Review 2**

This project aims to investigate the sentiment and test various architectures for comprehending sentiment polarity in London hotel dataset. We shall consider a manually labelled dataset available at <https://www.kaggle.com/PromptCloudHQ/reviews-of-londonbased-hotels> . The dataset contains review of the top 10 most expensive and least expensive London based hotels with hotel locations, reviews and user’s rating. We shall focus on the ambiguity that may raise among the users.

1. Use the SentiStrength from <http://sentistrength.wlv.ac.uk/> to determine the positive, negative and overall (sum of positive and negative) sentiment score for each review. Provide a database D1, which contains these information for each review.
2. Use Pearson correlation coefficient to evaluate the correlation of the overall sentiment score of each review with the user’s rating.
3. Gather all reviews pertaining to the same hotel altogether. Calculate the first and second order statistics (mean and standard deviation) of the user’s rating. Identify the hotels with low standard deviation and cases with relatively high standard deviations. Comment whether the high variation of standard deviation occurs in expensive hotel or cheap hotels (using for instance information about the hotel stars). Construct a histogram showing for each star category, the proportion of hotels for which the standard deviation is greater than some threshold ε (that you select yourself, depending on the observed range of variation of the standard deviation). We shall refer to this new class of hotels whose standard deviation is beyond the threshold as Ambiguous Class.
4. Assume that for a given hotel whose users’ rating standard deviation is higher than the pre-selected threshold, if the majority of the reviews are assigned negative sentiment according to user’s rating, then the hotel is assumed to belong to negative class. Likewise for positive sentiment. Provide therefore the two subclasses of this ambiguous class review. Provide an attribute in the database D1 indicating whether a given hotel belongs to Ambiguous class and whether it is positive or negative subclass. Draw a histogram highlighting the proportion of positive and negative subclass in the original Ambiguous Class.
5. We would like to explore the content of each of the above subclass. Concatenate all reviews of each subclass and use wordCloud (see an example at <https://www.geeksforgeeks.org/generating-word-cloud-python/> ) that highlights the most frequent wording used. Plot WordCloud for both positive and negative subclass. Comment on the findings
6. Use LDA with five topics and three words per topic to determine the topic distribution of the positive and negative subclass. Report the result in the database 1. Compare the result of the LDA output with the WordCloud finding (check the extent of the overlapping with high frequent words for instance or whether some LDA wording are directly related to WordCloud via other semantic network.
7. Consider categories generated by Empath Client <https://github.com/Ejhfast/empath-client>. Apply Empath Client to positive and negative subclasses reporting only those categories who held non-zero values. Record these findings in database D1.
8. Calculate the ratio of overlapping over the number of Empath categories as an indication of agreement between empaths and common corpus.
9. Repeat 8) by calculating the agreement between Empath categories and LDA.
10. Consider the standard five class categories often employed in hotel review: Price, Service, Parking, Room, Location, Food. Suggest your own ontology construction that gathers words associated to each category (for instance using list of synonym, thesaurus, hypernyms,…).
11. Use your developed ontology to seek the histogram of the frequency of the occurring five categories for both positive subclass and negative subclass.
12. Use the Harvard General Inquirer available in <http://www.wjh.harvard.edu/~inquirer/inquirerbasic.xls> and identify the list of adjective /adverbs pertaining to positive sentiment and that pertaining to negative sentiment. Now we would like to track the occurrence of these words in positive and negative subclass dataset. For this purpose, for each adjective/adverb who occurs in the subclass, we would like to identify nouns that are associated to it. For this purpose, use NLTK parser-tree to identify the first noun that is associated to it. For instance “Service is bad” the adjective “bad” should be linked to Service. Alternative to parser-tree would be to take a window of two words around the underlined adjective/adverb and identify the one that corresponds to a name category (you can use standard part-of-speech tagger to identify the noun word category). Report in the database, for each of these adjective/adverb, the list of nouns for which they have been found it is associated to, for both positive and negative subclass dataset. Comment on the potential misconception this may cause and potential agreement with previous findings.

Design a simple GUI of your choice that show the execution of each of the above tasks in a way to ease the task of the assessor or external end-user