Natural Language Processing Unit 1 Evaluation

Date: 29th Aug 2015

Time: 9:00 am to 1:30 pm IST

Problem # 2: Create an n-tweets summary of the given set of tweets

You are provided with a dataset that has a list of tweets in the raw form. A given tweet may consist of one or more sentences and sometimes the tweet may be just an empty line due to noise in the system. You are required to clean the tweets to bring it to a reasonable English text and then use techniques taught in the class to derive most interesting n-tweets from this corpus, output this in HTMLized form.

NOTE: The objective of this algorithm is NOT to build a product grade document summarizer but is an exam exercise that helps you understand how to use cosine similarity for different purposes. The method suggested here is not the best or ideal way to create document summaries.

Specifically, you are required to perform the following:

1. You will be given a corpus of tweets. The steps that are described below should take the number of tweets as input and process them. Since the execution time may take several minutes you need to get the algorithm working with dataset sizes of 10, 50, 100, 300, 500 etc. You may choose the n tweets out of total corpus at random, where you can choose n to be 50, 100, 300, 500
2. Tokenize and clean the tweets in order to remove twitter ids, hashtags, URLs etc. You may transform any emoticon such as ☺ or slang words in to a suitable word sequence that may best fit the context of the English text you are producing. You are required to replace urls with a word: “details”.

   NOTE: At a later step you will be HTMLizing this word by allowing navigation to the web page referred by the original URL that you replaced. You can use Ply and use your previous assignment for this purpose or use NLTK word and sentence tokenizers.

3. You are required to replace words such as: “aweeeeeeesome” that have repetitions of certain letter(s) with the right word “awesome”. You are required to use Wordnet or any other Thesaurus for some of the cleaning tasks.
4. You are required to do the necessary preprocessing that may include:
   a. Case conversions
   b. Stop-word removal
   c. Stemming

NOTE: You can use NLTK only up to this point. For all the remaining steps you are required to write your code with respect to any NLP algorithm.

5. You are required to compute the Cosine similarity matrix. You may treat each tweet as a document. If the number of tweets in the corpus given to you is n, then you need to build a n x n matrix of cosine similarity values, where each value x is: 0 <= x <= 1. Call this matrix Sr. The value 0 indicates dissimilar documents and x = 1 indicates that the documents are similar (not necessarily identical).

   NOTE: To implement similarity function you may refer the class notes for the definition of cosine similarity.

6. You need to create 2 other matrices from the matrix obtained in previous steps:
   a. Discretize the similarity matrix of real numbers Sr in to a matrix of positive integers, with values: 1 <= x <= 10, x is a positive integer. Call this matrix Si.
   b. Apply a threshold of 3 for each cell on Si to make this in to a Boolean Matrix of 0, 1. That is, compare the value of each cell if Si with 3, return 1 if the cell value <=3, 0 otherwise. Call this matrix Sb
7. Write the outputs Si and Sb as CSV formatted files
8. From Sb identify the top distinguishing tweets as follows:
   a. Compute the sum of each row of the matrix
   b. Pick the row that has the maximum sum
   c. Initialize the output list to di, where i is the row number we selected in the previous step: output = [di]
   d. For each coloumn j in this row i (that corresponds to the similarity of di, dj) add dj to the output list if the element boolean_similarity(i, j) = 1
9. Produce a summary created out of the documents in the output list generated above that conforms to reasonably good English. This summary should not have hashtags or screen names or RT words or URLs or emoticons or slang abbreviations (such as LOL).
10. Wherever you replaced a URL with the word “details” as required in an earlier step, you need to HTMLize the word by placing it between the <a> tag with the href attribute set to the URL from the raw tweet that you replaced.
11. Your output summary should be in valid HTML form, viewable on a Web browser. You may use your imagination to choose the visual formatting such as fonts, colors etc.

Deliverables:

1. Source code of your program
2. Si and Sb as CSV formatted files (viewable with MS Excel)
3. Output summary
4. A brief write up on the performance of your program: Did it produce a good summary? What should be done to improve the quality? – This should be a FaceBook post to be submitted by noon Sunday 30th Aug 2015
5. Any readme file or documentation that describes your work (optional)

Your submissions should be made to NLP course mail id ☺

Best of luck!